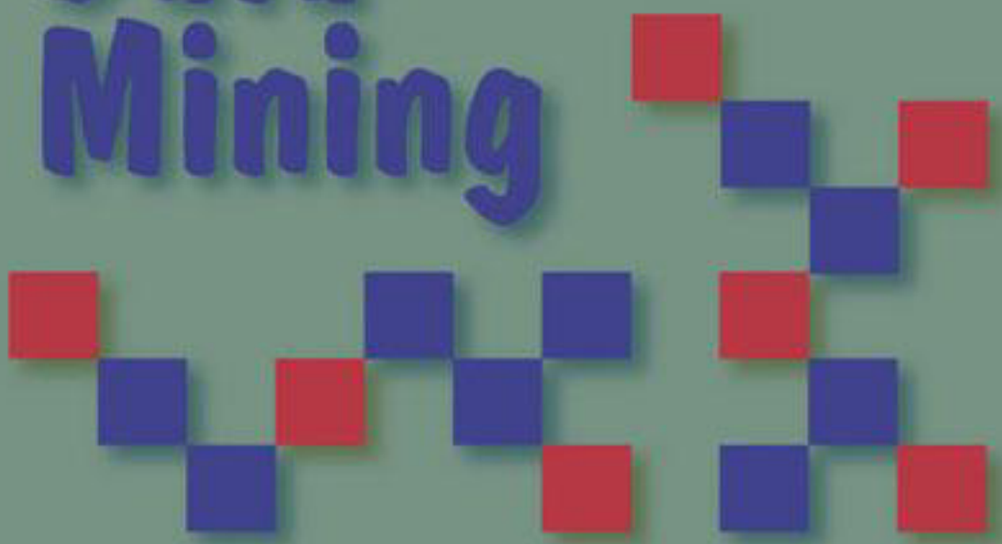
Foundations of Data Mining

T.Y. Lin, S. Smale, T. Poggio, and C.J. Liau

Fourth IEEE International Conference on

Data Mining



Brighton, United Kingdom
1-4 November 2004

Sponsored by
IEEE Computer Society Technical Committee on Computational Intelligence (TCCI)
IEEE Computer Society Technical Committee on Pattern Analysis and Machine Intelligence (TCPAMI)

IEEE COMPUTER SOCIETY

IEEE

Workshop Proceedings
Foundations of Data Mining

T.Y. Lin, S. Smale, T. Poggio, and C.J. Liau

The Fourth IEEE International Conference on Data Mining

Sponsored by the IEEE Computer Society

Brighton, UK

November 01 - 04, 2004

FDM 2004: Foundations of Data Mining
Brighton, UK, November 01, 2004

In conjunction with the Third IEEE International Conference on Data Mining

T.Y.Lin S. Smale T. Poggio C.J. Liau

Opening Remarks:

Data Mining has been developed, though vigorously, under rather ad hoc and vague concepts. For further development, a close examination on its foundations seems necessary. The central goal in this workshop is to explore various fundamental issues of data mining. The scope of the workshop includes:

1. Theory of Data Mining and Discovery
2. Similarity and Dissimilarity of Learning and Discovery
3. Logical Foundations
4. Modeling for Data Mining
5. Sampling and Complexity Reduction
6. Uncertainty in Data Mining and Discovery
7. Other New and Novel Approaches: The examination of foundation may lead to new directions

The proceedings contain 2 invited papers and 25 contributed papers to be presented at the workshop. Each paper was carefully peer-reviewed. We would like to thanks all the authors, invited speakers, and attendees for contributing to the success of the workshop. Special thanks are due to the program committee for help in reviewing the submissions.

This workshop follows the previous highly successful workshops: FDM 2002, held in Maebashi City, Japan and FDM 2003 in Melbourne, Florida, USA. We expect FDM 2004 to be equally successful.

Organizing Committee

T. Y. Lin (Chair) Department of Computer Science, San Jose State University, San Jose, California 95192, USA Tel: 408-924-5121, Fax: 408-924-9050 e-mail: tylin@cs.sjsu.edu	S. Smale (Honorary Chair), a fields medallist (Mathematical Nobel prize) Toyota Technological Institute, Chicago, IL 60637 UC-Berkeley, California University of Chicago.	Tomaso Poggio, Eugene McDermott Professor, Brain Sciences Department, M.I.T., Cambridge, MA 02142

Coordinators: Professor I-Jen Chiang, Taipei Medical University, Taipei, Taiwan; ijchiang@tmu.edu.tw

Dr. Churn-Jung Liau, Institute of Information Science, Academia Sinica, Taipei, 115, Taiwan; liaucj@iis.sinica.edu.tw

Program Committee

Michael Berthold (University of Konstanz, Germany)
Arbee L.P. Chen (National Taiwan (University, Taiwan)
Ming-Syan Chen (National Taiwan (University, Taiwan)
Zhengxin Chen (University of Nebraska, USA)
Robert Grossman (University of Illinois at Chicago)
Gia-Loi Gruenwald (University of Oklahoma, USA)
Jerzy Grzymala-Busse (University Of Kansas, USA)
Mike Hadjimichael (Naval Research Laboratory-Monterey, USA)
Larry Hall (University of South Florida, USA)
Jianchao (Jack) Han (California State University, Dominguez Hills)
Tzung-Pei Hong (National University of Kaoshiung, USA)
Xiaohua Hu (Drexall University, USA)
Larry Kerschberg (George Mason University, USA)
Marzena Kryszkiewicz (Warsaw University of Technology, Poland)
Churn-Jung Liau (Academia, Sinica, Taiwan)
Huan Liu Arizona State (University, USA)
Qing Liu (Nan-Chang (University, China)
Larry Mazlak (UC-Berkeley and (University of Cincinnati, USA)
Ernestina Menasalvas (Universidad Politécnica de Madrid , Spain)
Hiroshi Motoda (Osaka University, Japan)
Tetsuya Murai (Hokaido University, Japan)
Shojiro Nishio (Osaka University, Japan)
Masayuki Numao (Tokyo Institute of Technology, Japan)
Setsuo Ohsuga Tokyo University (Honorary Faculty), Japan)
Witold, Pedrycz (University of Alberta, Canada)
James Peters (University of Manitoba, Canada)
Fred Petry (Tulane University, USA)
Vijay Raghavan (University of Louisiana, USA)
Zbigniew W. Ras (University of North Carolina, USA)
Jan Rauch (University of Economics, Prague, Czech Republic)
M. C. Shan (Hewlett-Packard Labs, USA)
Andrzej Skoworn (University of Warsaw, Poland)
Einoshin Suzuki (Yokohama University, Japan)
Bhavani Thurasingham (National Science Foundation, USA)
Hiroshi Tsukimoto (Tokyo Denki University, Japan)
Shusaku Tsumoto (Shimane Medical University, Japan)
Guoyin Wang (Chongqing University, China)
Anita Wasilewska, (SUNY at Stony Brook, USA)
Xindong Wu, (University of Vermont, USA)
Jingtao Yao (University of Regina, Canada)
Y. Y. Yao (University of Regina, Canada)
Jongpil Yoon (University of Louisiana, USA)
Philip S Yu (IBM T.J. Watson Research Center)
Ning Zhong (Maebashi Institute of Technology, Japan)
Wojciech Ziarko (University of Regina, Canada)

Table of Contents

Invited Papers

A Theory of Parameter Free Data Mining

Ming Li
Canada Research Chair in Bioinformatics
University of Waterloo

Given a collection of genomes, can we derive their evolutionary history? What about a collection of languages? Or a collection of music scores? Or a collection of student programming assignments? Or a collection of chain letters? More generally, given a collection of sequences, can we cluster them properly? Is there an application-independent information measure which applies to all such applications?

In this talk, we will present a universal information distance and a general method to discover similarities between sequences, any type of sequences. We then apply the theory to infer the evolutionary histories of mammals, languages, programs (plagiarism detection), and chain letters.

A popular version of this talk can be found in the June 2003 issue (pp. 76-81) of Scientific American, "Chain Letters and Evolutionary Histories", by Charles H. Bennett, Ming Li and Bin Ma.

* The word of "Parameter-Free Data Mining" was coined by Keogh-Lonardi-Ratanamahatana

BI-directional BYY Learning for Mining Structures with Projected Polyhedra and Topological Map

Lei Xu *
Department of Computer Science and Engineering, Chinese University of Hong Kong
Shatin, NT, Hong Kong, P.R. China, Email: lxu@cse.cuhk.edu.hk

Abstract

Two types of learning structures are investigated from the perspective of Bayesian Ying Yang (BYY) harmony learning with a bi-directional architecture. First, the Kohonen map type of topology is revisited with a new insight and a new variant. Next, we explain how the multi-sets modelling for object detection can be reformed into a topological map of multi-set-mixture. Third, we show that independent binary factor analysis can be used to learn a type of Gaussian mixture with 2^m Gaussian densities located on vertices of a projected hyper polyhedra structure that are represented via only m real vectors such that the number of free parameters has been significantly reduced, thus with a much better generalization ability. Also, an adaptive algorithm is provided for learning not only all the parameters in this structure but also determining an appropriate m automatically during learning. Moreover, another topological type is introduced into this binary factor analysis in a sense that similar objects are encoded by inner binary codes that are close to each other in term of smallest error bits.

1. Introduction

Given data from a world of multiple objects in term of a set of samples, where each sample x_t comes from one of objects, one widely encountered task is to determine which object that each sample x_t comes from. Using a label $\ell \in L$ to denote one object, the task is to assign a correct label ℓ_t to each sample x_t that is observed with its label missing, which is usually said either that x_t is encoded by ℓ_t or that x_t is recognized as coming from the ℓ-th pattern.

Provided that each object is simply described by a vector m_ℓ that is observed via each sample x after disturbed by a noise e from a Gaussian $G(e|0, \sigma_\ell^2 I)$, or equivalently x can be regarded as coming from $G(x|m_\ell, \sigma_\ell^2 I)$. The task of estimating every m_ℓ from a given set of samples $\{x_t\}$ and the task of assigning a label ℓ to each object represented by one m_ℓ are closely coupled together, which have been widely studied either under the name of minimum Mean Square Error (MSE) clustering analysis in the pattern recognition literature [10] or under the name of Vector Quantization (VQ) in the literature of image encoding [18]. Both MSE clustering and VQ are usually implemented by the well known k-mean algorithm, which has been also widely used for various data mining problems in recent years [11].

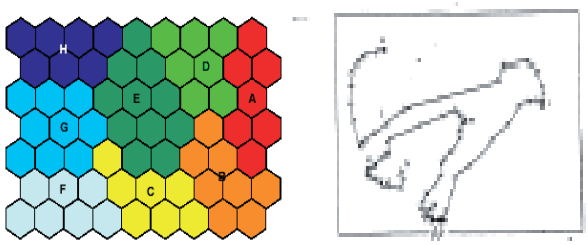


Figure 1. Topological structure

Usually, multiple objects are not isolated from each other but linked with various relations. Among them, an important type, that comes from concepts such as 'similar', 'near', etc, can be displayed by spatial relationships among objects located in the Euclidean space. Considering a regular d-dimensional lattice topology, we attempt to locate each object ℓ on one node of the lattice such that objects locating topologically near should be similar to each other, as shown in Fig.1. If we can learn from data such a topological structure, we will be able to retrieve similar objects or pattern classes simply from neighbors, which takes an important roles in tasks of content based retrieval, missing pattern recovering, and tracing temporal patterns as encountered in bio-informatics, financial engineering.

Intuitively, to build such a topological structure we need a similarity measure to judge whether two objects are similar. Even so, a direct placement of all the objects on a lattice

* The work described in this paper was fully supported by a grant from the Research Grant Council of the Hong Kong SAR (Project No: CUHK4225/04E).

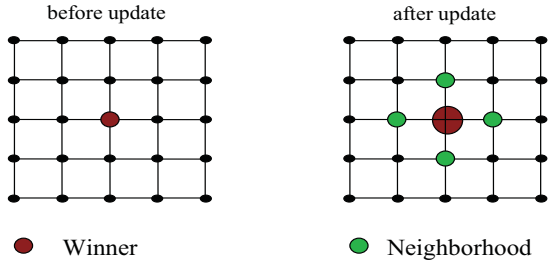


Figure 2. One member wins, a family gains

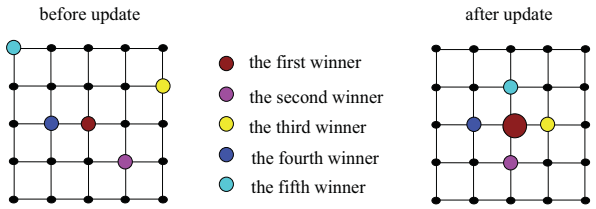


Figure 3. Strongers gain and then teaming together

under a given similarity measure is computationally a hard combinatorial problem. Interestingly, this problem has been implemented approximately in help of a biological brain dynamics of self-organization [17], featured by a Mexican hat type interaction, namely, neurons in near neighborhood excite each other with learning, while neurons far away inhibit each other with de-learning. Computationally, such a dynamic process can be further simplified by certain heuristic strategies.

One widely used is the well known Kohonen self-organizing map [14] that implements a strategy of *one member wins, a family gains* That is, as long as one member wins in the winner-take-all competition, all the members of a family will gain regardless whether other members are strong or not. As shown in Fig.2, with each node on the lattice associated with a mean vector m_ℓ that represents an object or class, a winner-take-all competition is made per sample x_t to get the winner

$$\ell^* = arg\ \min \|x_t - m_\ell\|^2. \tag{1}$$

Then considering a small neighborhood N_ℓ of ℓ^* that usually consists of 2^d knots directly connected to ℓ^*, we update

$$m_\ell^{new} = m_\ell^{old} + \eta_t(x_t - m_\ell^{old}), \forall \ell \in N_\ell. \tag{2}$$

As long as an appropriate size N_ℓ is specified, this learning will finally result in a map on which nodes located near each other have their corresponding mean vectors being close to each other too. In the literature, a great number of studies have been made on applying and extending the Kohonen map.

In [30], we also get an alternative strategy of *strongers gain and then teaming together*. That is, a number of strongers in competition will be picked as winners who not only gain learning but also are teamed together to become neighbors. As experimentally demonstrated in [6], this strategy can speed up self-organization, especially at the early stage of learning. Also, we can combine it with the Kohonen map strategy by using it at an early stage and subsequently switching to the Kohonen map.

In many applications, it is not enough to represent each object simply by a vector m_ℓ or even $G(x|m_\ell, \sigma_\ell^2 I)$ after taking noise in consideration. It is further considered that each object is described by a parametric distribution $q(x|\theta_\ell, \ell)$ plus

$$q(\ell) = \sum_{j=1}^{k} \alpha_j \delta(\ell - j),$$
$$\text{with the constraint } \alpha_\ell \geq 0, \sum_{\ell=1}^{k} \alpha_\ell = 1, \tag{3}$$

where α_ℓ denotes a priori probability that x comes from the ℓ-th object. As a result, the MSE clustering task has been extended to estimate α_ℓ and θ_ℓ which is equivalent to learning the dependence structures in the format of

$$q(x) = \sum_{\ell} \alpha_\ell q(x|\theta_\ell, \ell). \tag{4}$$

It is usually called finite mixture and learning can be made in help of the EM algorithm [8, 19, 16].

In [30], the strategies given in Fig.2 and Fig.3 have been further extended to get topology between objects with $\alpha_\ell q(x|\theta_\ell, \ell)$ as a similarity measure. Specifically, eq.(1) and eq.(2) are extended as follows:

$$\begin{aligned} \ell^* &= arg\ \max[q(x_t|\theta_\ell, \ell)\alpha_\ell], \\ \alpha_\ell^{new} &= \frac{\alpha_\ell^{old} + \eta_t}{1 + \eta_t \# N_\ell}, \ \forall \ell \in N_\ell, \\ \theta_\ell^{new} &= \theta_\ell^{old} + \eta_t \nabla_{\theta_\ell} \ln q(x_t|\theta_\ell, \ell), \ \forall \ell \in N_\ell, \end{aligned} \tag{5}$$

where $\#S$ denotes the number of elements in S. In this way, topological maps of various models can be obtained for applications in complicated situations [30]. When $q(x|\theta_\ell, \ell) = G(x|m_\ell, \Sigma_\ell)$, m_ℓ can be updated by eq.(2) and Σ_ℓ is updated as follows:

$$\Sigma_\ell^{new} = (1 - \eta_t)\Sigma_\ell^{old} + \eta_t(x_t - m_\ell^{old})(x_t - m_\ell^{old})^T, \tag{6}$$

for $\ell \in N_\ell$.

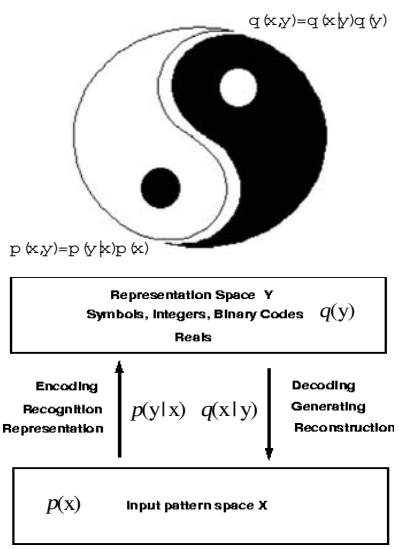


Figure 4. Bayesian Ying-Yang System

In this paper, from the perspective of Bayesian Ying Yang (BYY) harmony learning with a bi-directional architecture, we start at revisiting the above topological learning with a new variant and also an insight on learning regularization. Then, we explain how the multi-sets modelling learning, firstly proposed in 1994 [38, 37] for modelling objects in typical shapes such as lines, circles, and ellipses, as well as pre-specified templates [38] and [37] in the fields of computer vision and image recognition, can be formed into a topological map of multi-set-mixture. Moreover, we show that independent binary factor analysis can be used to learn a type of Gaussian mixture with 2^m Gaussian densities located on vertices of a projected hyper polyhedra structure that are represented via only m real vectors such that the number of free parameters has been significantly reduced in comparison with an ordinary Gaussian mixture and thus a much better generalization ability is obtained. Furthermore, an adaptive algorithm is provided for learning not only all the parameters in this structure but also determining an appropriate m automatically during learning. Also, another topological type is introduced into this binary factor analysis in a sense that similar objects are encoded by inner binary codes that are close to each other in term of smallest error bits.

2. Bayesian Ying-Yang Harmony Learning

As shown in Fig.4, a BYY system considers coordinately learning two complement representations of the joint distribution $p(x, y)$:

$$\begin{aligned} p(u) &= p(x, y) = p(y|x)p(x), \\ q(u) &= q(x, y) = q(x|y)q(y), \end{aligned} \tag{7}$$

basing on $p(x)$ that is estimated from a set of samples $\{x_t\}_{t=1}^N$, while $p(y|x)$, $q(x|y)$ and $q(y)$ are unknowns but subject to certain pre-specified structural constraints. In a compliment to the famous Chinese ancient Ying-Yang philosophy, the decomposition of $p(x, y)$ coincides the Yang concept with the visible domain by $p(x)$ regarded as a Yang space and the forward pathway by $p(y|x)$ as a Yang pathway. Thus, $p(x, y)$ is called Yang machine. Similarly, $q(x, y)$ is called Ying machine with the invisible domain by $q(y)$ regarded as a Ying space and the backward pathway by $q(x|y)$ as a Ying path.

On one hand, we can interpret that each x is generated from an invisible inner representation y via a backward path distribution $q(x|y)$ or called a generative model

$$q(x) = \int q(x|y)q(y)\mu(dy) \tag{8}$$

that maps from an inner distribution $q(y)$. In this case, $p(y|x)$ is not explicitly specified or said being free to be specified, while two pre-specified parametric models $q(x|y)$ and $q(y)$ form a backward path to fit the observations of x. We say that the Ying-Yang system in this case has a backward architecture (shortly B-architecture).

On the other hand, we can interpret that each x is represented as being mapped into an invisible inner representation y via a forward path distribution $p(y|x)$ or called a representative model

$$p(y) = \int p(y|x)p(x)\mu(dx) \tag{9}$$

that matches the inner density $q(y)$. In this case, $q(x|y)$ is not explicitly specified or said being free to be specified. Forming a forward path, $p(x)$ is estimated from a given set of samples and then is mapped via pre-specified parametric model $p(y|x)$ into $p(y)$ by eq.(9) to match a pre-specified parametric model $q(y)$. We say that the Ying-Yang system in this case has a forward architecture (shortly F-architecture).

Moreover, the above two architectures can be combined with $p(y|x)$, $q(x|y)$ and $q(y)$ are all pre-specified parametric models. In this case, we say that the Ying-Yang system in this case has a Bi-directional architecture (shortly BI-architecture).

The name of BYY system not just came from the above direct analogy between eq.(7) and the Ying-Yang concept, but also is closely related to that the principle of making learning on eq.(7) is motivated from the well known harmony principle of the Ying-Yang philosophy, which is different from making $p(x)$ by eq.(8) fit a set of samples $\{x_t\}_{t=1}^N$ under the ML principle [21] or its approximation [13] as well as simply the least mean square error criterion [40], and also different from making $q(y)$ by eq.(10) satisfy certain pre-specified properties such as maximum entropy [4] or matching the following independent density [3]:

$$q(y) = \prod_{j=1}^m q(y^{(j)}). \tag{10}$$

Under this harmony principle, the Ying-Yang pair by eq.(7) is learned coordinately such that the pair is matched in a compact way as the Ying-Yang sign shown in Fig.4. In other words, the learning is made in a twofold sense that

- The difference between the two Bayesian representations in eq.(7) should be minimized.
- The resulted entire BYY system should be of the least complexity.

Mathematically, this principle can be implemented by [36, 31, 30]

$$\max_{\theta,m} H(\theta, m), \qquad (11)$$
$$H(\theta, m) = H(p||q) = \int p(y|x)p(x) \ln [q(x|y)q(y)]\mu(dx)\mu(dy) - \ln z_q,$$

where θ consists of all the unknown parameters in $p(y|x)$, $q(x|y)$, and $q(y)$ as well as $p(x)$ (if any), while m is the scale parameter of the inner representation y. The task of determining θ is called *parameter learning*, and the task of selecting m is called *model selection* since a collection of specific BYY systems by eq.(7) with different scale values corresponds to a family of specific models that share a same system configuration but in different scales. Furthermore, the term $Z_q = -\ln z_q$ imposes regularization on learning [28, 30, 32], via two types of implementation. One is called data smoothing that provides a new solution to the hyper-parameter for a Tikinov-like regularization [22], and the other is called normalization that causes a new conscience de-learning mechanism similar to that of the rival penalized competitive learning (RPCL) [39, 32, 30].

Usually $p(x)$ is fixed at a non-parametric Parzen window density estimate [9]:

$$p_h(x) = \frac{1}{N} \sum_{t=1}^{N} G(x|x_t, h^2 I), \qquad (12)$$

where $h > 0$ is a given *smoothing parameter*, $p_0(x) = p_h(x)_{h=0}$ is simply empirical density. While $p(y|x)$ is either free in a B-architecture or a parametric form in a BI-architecture and thus will be pushed into its least complexity form. E.g., $p(y|x)$ in a B-architecture will be determined by $\max_{p(y|x)} H(p||q)$, resulting in the following least complexity form:

$$p(y|x) = \delta(y - y(x)), \;\; y(x) = arg \max_{y} [q(x|y)q(y)]. \qquad (13)$$

On the other hand, the matching nature of harmony learning will further push $q(x|y)$ and $q(y)$ towards their corresponding least complexity forms, which makes model selection possible, e.g., m is appropriately determined.

Referring details in [30], this least complexity nature introduces a new mechanism that makes model selection implemented either automatically during the following parameter learning with m initialized large enough:

$$\max_{\theta} H(\theta), \quad H(\theta) = H(\theta, m), \qquad (14)$$

which makes θ take a specific value such that m is effectively reduced to an appropriate one, as shown in Fig.5(b). This feature is not shared by the existing approaches in the literature. By the conventional approaches, parameter learning and model selection are made in a two-phase style. First, parameter learning is made usually under the maximum likelihood principle. Then, model selection is made by a different criterion, e.g., AIC [1], MDL [20], etc. These model selection criteria are usually not good for parameter learning, while the maximum likelihood criterion is not good for model selection, especially on a small size of training samples.

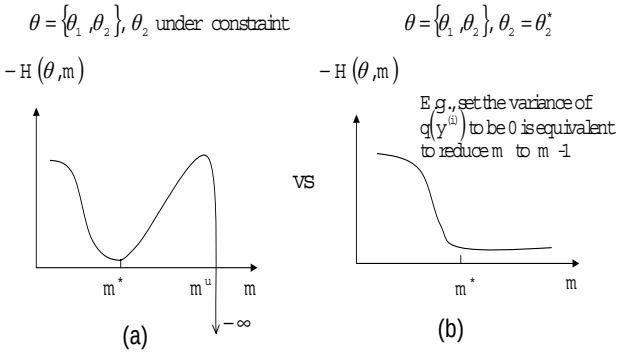


Figure 5. (a) Model selection made after parameter learning on every m in a given interval $[m_d, m_u]$, (b) Automatic model selection with parameter learning on a value m of large enough.

In certain circumstances. E.g., to compare with the existing model selection criteria such as AIC, MDL, the BYY harmony learning by eq.(11) still needs to be implemented in a two-phase style to make studies comparable. Specifically, the first phase implementation of eq.(11) is made with m enumerated from small values incrementally. At each specific m, the inner representation y is pre-specified to be uniform [30] such that automatic model selection will not happen during learning by eq.(14) in the first phase, we need to implement the second phase by the following type of model selection criteria obtained from this mechanism:

$$\min_{m} J(m), \;\; J(m) = -H(\theta^*, m), \qquad (15)$$

as shown in Fig.5(a).

Moreover, in such a two phase style, parameter learning for getting θ^* can be implemented with eq.(14) replaced by the following Kullback divergence based parameter learning:

$$\min_{\theta} KL(\theta) = \int p(y|x)p(x) \ln \frac{p(y|x)p(x)}{q(x|y)q(y)} \mu(dx)\mu(dy). \qquad (16)$$

Particularly, on a B-architecture, the minimization of the above $KL(\theta)$ with respect to a free $p(y|x)$ will result in

$$p(y|x) = \frac{q(x|y)q(y)}{q(x)},\ q(x) = \int q(x|y)q(y)\mu(dy),$$
$$KL(\theta) = \int p(x) \ln \frac{p(x)}{q(x)} \mu(dx), \tag{17}$$

which becomes equivalent to ML learning on $q(x)$ when $p(x) = p_0(x)$ is given by eq.(12) [36]. In this case, we actually implement ML learning in the first phase and then model selection by eq.(15) in the second phase.

Moreover, the implementation of both eq.(14) and eq.(16) can be made by alternatively performing the following two steps:

$$\begin{array}{ll} \text{Ying-step:} & \text{fix } p(x,y), \text{update unknowns in } q(x,y), \\ \text{Yang-step:} & \text{fix } q(x,y), \text{update unknowns in } p(x,y), \end{array} \tag{18}$$

which is called the Ying-Yang alternative procedure. It is guaranteed that either of $-H(\theta)$ and $KL(\theta)$ gradually decreases until becomes converged. The details are referred to [30].

3. Model Selection, Learning Regularization, and Topological Preservation

The harmony learning by eq.(14) attempts to compress the representation space via the least complexity nature that is demonstrated with a winner-take-all (WTA) competition by eq.(13). This type of parameter learning aims at a compact inner representation with an automatic model selection by discarding extra representation space during parameter learning. However, there is no free lunch. The WTA operation by eq.(13) locally per sample will make learning become sensitive to the initialization of parameters and the manner that samples are presented, which usually leads to a local maximum solution for eq.(14).

With a soft competition by $p(y|x)$ in eq.(17) to replace the WTA competition by eq.(13), the ML learning, or equivalently the KL learning by eq.(16) with a B-architecture and an empirical density by eq.(12), is regularized with a more spread inner representation that improves the local maximum problem. However, there is no free lunch too. It makes the model selection ability considerably weaken, especially on a small size of samples. Thus, making model selection by eq.(15) is needed after parameter learning. However, as discussed previously in the introduction section, the two phase style implementation costs computation extensively. Instead of the two phase style, regularization to the WTA by eq.(13) may also be imposed to the harmony learning by eq.(14) such that automatic model selection still occurs via either some external help or certain internal mechanism.

Externally, we can combine the KL learning by eq.(16) with the harmony learning by eq.(14), by which we get a spectrum of learning models. The details are referred to Sec. 23.4.2 in [26]. Another spectrum, that also varies between model selection ability and regularization ability, can be obtained via internally replacing $\ln(r)$ by a family of convex functions for divergence measuring. Also, two different forms of the term $Z_q = -\ln z_q$ introduce two types of regularization on learning under the name z-regularization. The details are referred to Sec.22.6.3 in [25].

Internally, regularization to the WTA by eq.(13) can be imposed during the harmony learning by eq.(14) via a constrained $p(y|x)$ in a BI-architecture. Instead of letting $p(y|x)$ free to be decided by eq.(13), we consider a BI-architecture with $p(y|x)$ designed in a structure that will not lead to the WTA by eq.(13). Specifically, different structures of $p(y|x)$ will lead to regularization with different features, which are shortly summarized under the name BI-regularization.

Typical examples are discussed as follows:

(a) *A collection of ordered winners* We consider that a collection of winners shares-the-all (STA) instead of only one winner-take-all (WTA), such that the local optimal problem can be alleviated. In the cases that y takes discrete values, we consider

$$p(y|x_t) = \sum_{y' \in A_t} \eta_t(y') \delta(y - y'), \tag{19}$$

where A_t consists a collection of values that y' may take, with each value denoting a unknown winner, and the number of winners is decided by an awarding scheme, e.g., the number is $\#A_t = 7$ for a scheme of one 1st prize, two 2nd prizes, four 3rd prizes. Correspondingly, $\eta_t(y')$ represents the prizes to be presented to the winners, e.g., $\eta_t(y')$ takes the value a_1 for the 1st price only at one in A_t, the value a_2 for the 2nd price at two in A_t, and the value a_3 for the 3rd price at four in A_t, where we have $a_1 > a_2 > a_3$ and $\sum_{y \in A_t} \eta_t(y') = 1$. Specifically, which ones in A_t get what prizes are determined by $\max_{p(y|x)} H(p\|q)$ that lead to A_t consisting of the first 7 largest values of $q(x_t|y)q(y)$, with the first one for the 1st prize, the next two for the 2nd prizes, and the rest for the 3rd prizes. Then, the parameters θ of $q(x_t|y)q(y)$ are updated to increase the following $L_{x|y}(\theta)$

$$L_{x|y}(\theta) = \sum_{y' \in A_t} \eta_t(y') \ln [q(x|y')q(y')]. \tag{20}$$

In implementation, it can be made via gradient ascending of either this $L_{x|y}(\theta)$ or $\eta_t(y') \ln [q(x|y')q(y')]$ per $y' \in A_t$. For the latter, those updating rules for the

case with eq.(13) on y_t can be directly adopted on every $y' \in A_t$ simply with the learning step size η_t replaced by $\eta_t \eta_t(y')$.

(b) *A winning team* Instead of considering a collection of winners, we can also consider that competition is made among teams with each team consisting of individuals with similar qualifications. Thus, the winner-take-all is replaced by "all the individuals of the winning team share the all". In implementation, we still consider eq.(19) but with a different A_t that consists of one y_t plus a set of values of y' that are close to y_t. Moreover, $\eta_t(y')$ takes a_1 at y_t and smaller values for other y' according to its closeness to y_t. E.g., in the case that y is a binary vector, A_t consists of those of y' that differ from y_t with only one bit. In the case that y is real, we consider

$$p(y|x) = G(y|y_t, h_y^2 I), \tag{21}$$

with a given $h_y^2 > 0$ that can be determined in cooperation with a z_q-regularization.

(c) *Competitive experts* Considering to approximate the deterministic mapping function that has to be obtained by eq.(13) via optimization, we consider

$$p(y|x) = \sum_{j=1}^{n} \beta_j(x)\delta(y - f_j(x, \theta_{y|x,j})),$$
$$\sum_{j=1}^{n} \beta_j(x) = 1,\ \beta_j(x) = 0,\ or\ 1,$$

from which eq.(13) is simplified into

$$\begin{aligned} &p(y|x) = \delta(y - y(x)), \\ &y(x) = f_{j^*(x)}(x|\theta_{y|x,j^*(x)}), \\ &j^*(x) = arg\max_j [q(x|y)q(y)]_{y=f_j(x|\theta_{y|x,j})}. \end{aligned} \tag{22}$$

That is, there are n experts that competes to perform the mapping $x \to y$. In this case, its regularization role can be observed from the perspective that the number of local maximums in eq.(13) considerably reduces to simply n possibilities. However, though the obtained $y(x)$ is a global solution of eq.(13) under the constraint by eq.(22), it can be far away from the global solution of eq.(13) with no constraint on $p(y|x)$. This can affect the performance of the learned BYY system too. One solution is let n to be large enough.

(d) *$p(y|x)$ in specific structure* In certain situations, we know or approximately know the structure of $p(y|x)$ from considering the optimal inverse structure of $q(x|y)q(y)$. One example is encountered when both $q(x|y)$ and $q(y)$ are Gaussian. In this case, it follows from eq.(13) that

$$y(x) = W x + m. \tag{23}$$

Another example is using $p(y|x)$ in eq.(17), especially a Gaussian mixture when $q(x|y) = G(x|\mu_y, \Sigma_y)$, which was firstly proposed in [32] and has been further shown in [15] that this type of regularization actually performs a RPCL-like learning mechanism.

It also deserves to note that the a joint consideration on the structure of $p(y|x)$ and the form of the term $Z_q = -\ln z_q$ may further lead to an improvement. One typical case is the structure given by either eq.(23) or eq.(22) where we know an analytic expression of $y(x)$ that is usually differentiable with respect to x. In this case, with $p(x) = p_h(x)$ by eq.(12) put into eq.(11), we get

$$\begin{aligned} &H(\theta, m) = \tfrac{1}{N}\sum_{t=1}^{N}\int G(x|x_t, h^2 I)\times \\ &\ln[q(x|y(x))q(y(x))]dx - \ln z_q(h) \\ &= \tfrac{1}{N}\sum_{t=1}^{N} \ln[q(x_t|y_t)q(y_t)] - \ln z_q(h) \\ &+h^2 Tr[\frac{\partial^2 \ln Q(x)}{\partial x \partial x^T}]_{x=x_t},\ y_t = y(x_t), \\ &Q(x) = q(x|y(x))q(y(x)). \end{aligned} \tag{24}$$

In this case, $Z_q = Z_q(h)$, data smoothing regularization acts in the domain of x directly via h and in the domain of y indirectly via h and $Q(x)$. Ignoring the part of $q(y(x))$, the above equations returns to the case of Eqn.(30) in [24].

However, the above eq.(24) is not applicable when $y(x)$ by eq.(13) does not give analytic expression or the obtained expression is too complicated to compute its second order derivatives. Instead, we consider eq.(21) and $Z_q = Z_q(h_x, h_y)$, with data smoothing regularization acting in the domain of x directly via h_x and in the domain of y directly via h_y. The details are referred to Sec.2(B) in [31] and Sec.2.2.2 in [30].

3.1. Kohonen Learning, Multi-set Mixture, and Topological Map

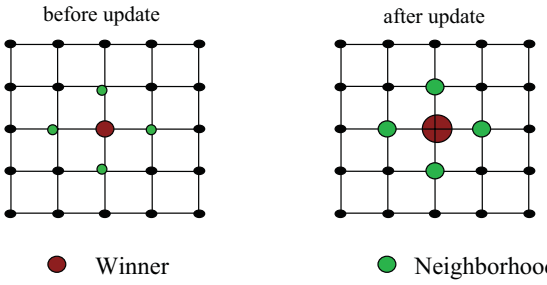


Figure 6. A team wins, a team gains

We further consider eq.(19) but with A_t being a winning team and that $y = \ell$ takes a set of discrete labels with each

discrete label denoting a location index on a lattice topology as shown Fig.2. It follows from eq.(19) with y replaced by ℓ that we have

$$p(\ell|x_t) = \sum_{\ell' \in A_t} \eta_t(\ell')\delta(\ell - \ell'). \tag{25}$$

In this case, A_t can be a winning region that consists of a set of ℓ' that are located in the neighborhood of ℓ_t. To get a better understanding, we focus at the following special case

$$\ell^* = arg\max_{\ell} \sum_{j \in N_\ell} \eta_t(j) \ln [q(x_t|\theta_j, j)\alpha_j],$$
$$\eta_t(j) = \begin{cases} 1-\gamma, & j = \ell, \\ \frac{\gamma}{m}, & j \neq \ell \in N_\ell, \end{cases} \tag{26}$$

where $\eta_t(j)$ puts a heavy weight at ℓ and much lowered weight $0 < \gamma < 1$ at its neighbors in N_ℓ. As shown in Fig.6, instead of each individual participating the competition, each node together with its neighbors joints in the competition. Then, the updating is made as in eq.(5) and eq.(6). That is, we get a strategy that *a team wins, a team gains*. At the beginning, γ can be set at a very small value and thus the situation is similar to the Kohonen map. As learning goes, γ gradually increases such that neighbors take their roles in the competition. This can avoid that an already organized part of map is disturbed by an isolated abnormal winner. Roughly, the Kohonen learning can be regarded as a rough approximation of this third strategy.

Firstly proposed in 1994 [38, 37], the multi-set modelling learning is proposed for modelling objects in typical shapes such as lines, circles, and ellipses, as well as pre-specified templates [38, 37] in the fields of computer vision and image recognition. Main results and certain historic remarks have been recently summarized in [28] under the name of multi-set-mixture.

Though topological learning in eq.(5) and eq.(6) applies to objects in any distribution $q(x|\theta_\ell)$, an efficient implementation can be made only when each $q(x|\theta_\ell)$ is Gaussian, i.e., eq.(4) is a Gaussian mixture. However, Gaussian mixture and multi-set-mixture become conceptually equivalent and exchangeable only on tasks of modelling lines, planes, and subspaces. For the tasks of modelling circles, ellipses, and other pre-specified shapes, multi-set-mixture goes far beyond Gaussian mixture, for which we need a new technique to implement its learning.

As shown in Fig.7(a), samples from each object include one deterministic part plus random noise. The deterministic part is described by $S(\theta_\ell)$, a set of finite points or a continuous set of real points in R^d, subject to a parametric set θ of a finite number of unknown parameters to be determined. Each $S(\theta_\ell)$ represents a shape such as line, curve, and ellipsis, as well as a pre-specified shape. Subject to such a set $S(\theta_\ell)$, a sample x is represented by

$$\hat{x}_\ell = arg \min_{y \in S(\theta_\ell)} \varepsilon(x, y),$$

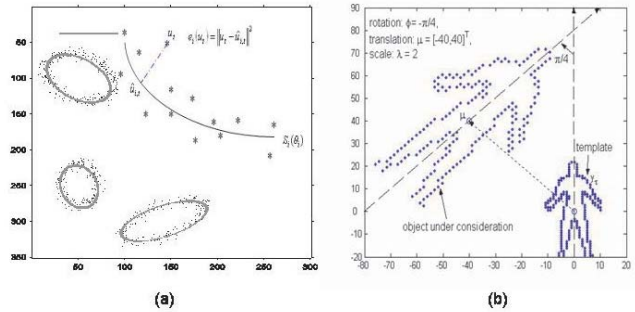


Figure 7. Multi-sets mixture

$$\varepsilon(x, y) = \mathcal{C}(e(x, y)),\ e(x, y) = x - y, \tag{27}$$

where $\hat{x}_\ell$ is called the best reconstruction of x by $S(\theta_\ell)$, and $e(x, \theta_\ell) = e(x, \hat{x}_\ell)$ is called the reconstruction error of $S(\theta_\ell)$ per sample x. Moreover, $\varepsilon(x, y)$ is a given cost measure for the discrepancy $e(x, y)$ such that $\varepsilon(x, y) \geq 0$ and $\varepsilon(x, y) = 0$ if and only if $e(x, y) = 0$ or $x = y$. The most widely used $\varepsilon(x, y)$ is

$$\varepsilon(x, y) = e(x, y)^T \Sigma_\ell^{-1} e(x, y), \tag{28}$$

which is called the Mahalanobis distance with Σ_ℓ being positively defined, e.g., it can be obtained from the Riemannian metric [2] on $S(\theta_\ell)$. This $\varepsilon(x, y)$ degenerates to the square distance between x, y when $\Sigma_\ell = I$. In this case, we call $\hat{x}_\ell$ the least square reconstruction of x by $S(\theta_\ell)$.

The best modelling of $S(\theta_\ell)$ to a given set of samples is made by determining θ_ℓ such that

$$\min_{\theta_\ell} \sum_{t=1}^{N} \varepsilon(x, \theta_\ell),\ \varepsilon(x, \theta_\ell) = \mathcal{C}(e(x, \theta_\ell)) = \min_{y \in S(\theta_\ell)} \varepsilon(x, y), \tag{29}$$

where $\varepsilon(x, \theta_\ell)$ denotes the cost between the discrepancy between x and its best reconstruction $\hat{x}_\ell$ via $S(\theta_\ell)$. It can get an explicit expression when the parameter set represents a line, a plane, a subspace, and a circle [28]. Generally it is obtained via a minimizing procedure via searching $\hat{x}_\ell$. While parameters $\{\theta_\ell\}$ can be learned via RPCL learning [34]:

$$\theta_\ell^{new} = \theta_{old}^{new} - h_{t,\ell}\eta_t \nabla_{\theta_\ell} \varepsilon(x, \theta_\ell),$$
$$h_{t,\ell} = \begin{cases} \gamma_c, & \text{if } \ell = \ell^* = arg\min_j \varepsilon(x_t, \theta_j), \\ -\gamma_r, & \ell = arg\max_{j \neq \ell^*} \varepsilon(x_t, \theta_j), \\ 0, & \text{otherwise.} \end{cases} \tag{30}$$

More generally, given a set of samples $\mathcal{Y}_\ell$ that represents a contour of a specific shape, we have $y + a_\ell, \forall y \in \mathcal{Y}_\ell$ for a shape resulted from a displacement a_ℓ, and

$$S(\theta_\ell) = \{\lambda_\ell R(\phi_\ell)(y + a_\ell) : \forall y \in \mathcal{Y}_\ell\}, \tag{31}$$

where $R(\phi_\ell)$ is a rotation matrix and $\theta_\ell = \{a_\ell, \phi_\ell, \lambda_\ell\}$. It represents a shape resulted from a displacement a_ℓ, a rotation of an angle ϕ_ℓ and a scaling by λ_ℓ, as shown in Fig. 7(b). Correspondingly, fitting the shape by eq.(29) becomes

$$\min_{\theta_\ell} \sum_{t=1}^{N} \min_{y \in \mathcal{Y}_\ell} \|x_t - \lambda_\ell R(\phi_\ell)(y + a_\ell)\|^2. \quad (32)$$

Conceptually, a finite mixture can be applied to describe the above multi-set modelling. However, directly considering x leads to a mixture of non-Gaussian densities is not easy to implement [36, 35] since we do not know nonGaussian distributions of x. Instead, we can regard the reconstruction error $e(x_t, \theta_\ell)$ coming from a Gaussian $G(e|0, \Sigma_\ell)$ and as a whole e comes from a Gaussian mixture $q(e|\theta) = \sum_{\ell=1}^{k} \alpha_\ell G(e|0, \Sigma_\ell)$. Considering the BYY harmony learning by eq.(11), with $p(e, x, \ell) = p(e|x, \ell)p(\ell|x)p(x)$ as $p(y|x)p(x)$ and $q(e, x, \ell) = q(x|e, \ell)q(e|\ell)q(\ell)$ as $q(x|y)q(y)$, given $p(x) = p_0(x)$ and $p(e|x_t, \ell) = G(e|e(x_t, \theta_\ell), h^2 I)$ we have

$$\begin{aligned}
&H = \int p(e|x, \ell)p(\ell|x)p_0(x) \times \\
&\ln [q(x|e, \ell)G(e|0, \Sigma_\ell)\alpha_\ell] dx de d\ell = H_h(\theta) + c_t, \\
&H_h(\theta) = \frac{1}{N} \sum_{t=1}^{N} \sum_{\ell=1}^{k} p(\ell|x_t) \int G(e|e(x_t, \theta_\ell), h^2 I) \\
&\ln [G(e|0, \Sigma_\ell)\alpha_\ell] de \\
&= \frac{1}{N} \sum_{t=1}^{N} \sum_{\ell=1}^{k} p(\ell|x_t) \ln [\alpha_\ell G(e(x_t, \theta_\ell)|0, \Sigma_\ell)] \\
&-0.5h^2 \sum_{\ell=1}^{k} \alpha_\ell Tr[\Sigma_\ell^{-1}], \\
&c_t = \frac{1}{N} \sum_{t=1}^{N} \sum_{\ell=1}^{k} p(\ell|x_t) \int G(e|e(x_t, \theta_\ell), h^2 I) \times \\
&\ln q(x_t|e, \ell) de = \frac{1}{N} \sum_{t=1}^{N} \ln q(x_t),
\end{aligned} \quad (33)$$

where c_t is irrelevant to learning, since knowing x_t already means that it does not relate to any other thing.

When $h = 0$, from the above $H_h(\theta)$ we can implement topological learning on a multi-set mixture by eq.(5) or plus eq.(26) with $q(x|\theta_\ell)$ replaced by $G(e|0, \Sigma_\ell)$. Moreover, Σ_ℓ is still updated by eq.(6) and updating on θ_ℓ takes the following specific form

$$\theta_\ell^{new} = \theta_\ell^{old} - \eta_t \frac{\partial e^T(x_t, \theta_\ell)}{\partial \theta_\ell} \Sigma_\ell^{-1} e(x_t, \theta_\ell), \ \forall \ell \in N_\ell. \quad (34)$$

When $h \neq 0$, a regularization under the name of data smoothing takes its action via updating Σ_ℓ modified as follows:

$$\begin{aligned}
&\Sigma_\ell^{new} = (1 - \eta_t)\Sigma_\ell^{old} + \\
&\eta_t [h^2 I + (x_t - m_\ell^{old})(x_t - m_\ell^{old})^T], \forall \ell \in N_\ell.
\end{aligned} \quad (35)$$

Moreover, the value of h is determined via

$$\begin{aligned}
&\max_h \ H(\theta), \ H(\theta) = H_h(\theta) - \ln z_q(h), \\
&z_q(h) = \sum_{\tau=1}^{N} \sum_{j=1}^{k} p(e(x_\tau, \theta_j)), \\
&p(e) = \frac{1}{N} \sum_{t=1}^{N} \sum_{\ell=1}^{k} p(\ell|x_t) G(e|e(x_t, \theta_\ell), h^2 I),
\end{aligned} \quad (36)$$

where h^2 can be learned via a gradient ascending on $h^{new} = h^{old} + \eta_t \frac{dH(\theta)}{dh}$. Details about data smoothing regularization are referred to [26].

In special case that N_ℓ has only one element, with

$$p(\ell|x) = \frac{\alpha_\ell G(e(x_t, \theta_\ell)|0, \Sigma_\ell)}{\sum_{\ell=1}^{k} \alpha_\ell G(e(x_t, \theta_\ell)|0, \Sigma_\ell)}, \quad (37)$$

and eq.(5) and eq.(6) implemented with $\eta_t = \eta_0 p(\ell|x_t)$, we return to an adaptive EM algorithm for $\max H_0(\theta)$ on a multi-set mixture. After learning, we can also select the number k of objects via $\min_k J(k)$ with

$$\begin{aligned}
&J(k) = 0.5 \sum_{\ell=1}^{k} \alpha_\ell (|\Sigma_\ell| + h^2 Tr[\Sigma_\ell^{-1}] - \ln \alpha_\ell) \\
&+ \ln z_q(h).
\end{aligned} \quad (38)$$

It should be noted that regarding the reconstruction error $e(x_t, \theta_\ell)$ coming from a Gaussian $G(e|0, \Sigma_\ell)$ works well in the cases of representing a line, a plane, and a subspace but only acts as a kind of rough approximation in the cases of representing of a circle and an ellipse.

Given any point x, its best reconstruction $\hat{x}_\ell$ by a circle is given by the intersecting point of the circle and the straight line from x to the center of the circle. The error $e(x_t, \theta_\ell) = \|x - \hat{x}_\ell\|^2$ can range from 0 to ∞ when x is outside the circle but only from 0 to R^2 when x is inside the circle, where R is the radius of the circle. Thus, instead of $G(e|0, \Sigma_\ell)$, a better representation for $q(e|\ell)$ is given as follows:

$$q(e|\ell) = \begin{cases} G(e|0, \Sigma_\ell), & \text{for } x \text{ outside the circle,} \\ Q(\|x - \hat{x}_\ell\|^2), & \text{for } x \text{ inside the circle,} \end{cases} \quad (39)$$

where $Q(r) \geq 0$ is monotonically decreasing from $Q(0) = G(0|0, \Sigma_\ell)$ to $Q(R^2) = 0$ subject to $\int_0^{R^2} Q(r)dr = 0.5$. For examples, $Q(r)$ can be monotonically decreasing linearly or quadratically.

Moreover, an ellipse can be parameterized as $(x - a)^T \phi^T \Lambda \phi (x - a) = R^2$ with $\Lambda = diag[1, \lambda_2, \cdots, \lambda_d]$ being a positive diagonal matrix and ϕ is a rotation matrix. Via a transformation $z = \Lambda^{0.5}\phi(x - a)$, the ellipse in the x domain is mapped into a circle $z^T z = R^2$ in the z domain. Thus, one way is to turn every sample x_t into $z_t = \Lambda^{0.5}\phi(x_t - a)$ and find $\hat{z}_t$ as the intersecting point of the circle $z^T z = R^2$ and the straight line from

$z_t = \Lambda^{0.5}\phi(x_t - a)$ to the origin 0. Then, we make learning in the space of z to fit a circle $z^T z = R^2$ by considering $\|z_t - \hat{z}_t\|^2 = \|\Lambda^{0.5}\phi(x_t - a) - \hat{z}_t\|^2$ to determine all the unknown parameters, which is equivalent to fitting an ellipse in the original space of x.

This transformation technique applies directly to a set of samples from only one ellipse. When the samples come from k ellipses, an ellipse can be parameterized as $(x - a_\ell)^T \phi_\ell^T \Lambda_\ell \phi(x - a_\ell) = R_\ell^2$ with $\Lambda_\ell = diag[1, \lambda_{\ell,2}, \cdots, \lambda_{\ell,d}]$ being a positive diagonal matrix and ϕ_ℓ is a rotation matrix. Via a transformation $z_\ell = \Lambda_\ell^{0.5}\phi_\ell(x - a_\ell)$, the ellipse is mapped into a circle $z_\ell^T z_\ell = R_\ell^2$. If we already know x_t from the ellipse ℓ, we can turn it into $z_{\ell,t} = \Lambda_\ell^{0.5}\phi(x_t - a_\ell)$ and then fit it as above discussed. However, when the information about each sample from which ellipse is missing, each sample can be mapped into $z_{\ell,t}, \ell = 1, \cdots, k$ possible values. A solution is selecting one with $\ell^* = arg\max_\ell p(\ell|x_t)$. Each time after parameters are updated the space of z, $p(\ell|x_t)$ is also updated by $p(\ell|x_t) = p(\ell|z_{\ell^*,t})$ that can be calculated in the space of z.

4. Independent Subspace and Binary Factor Analysis

4.1. Independent Subspace

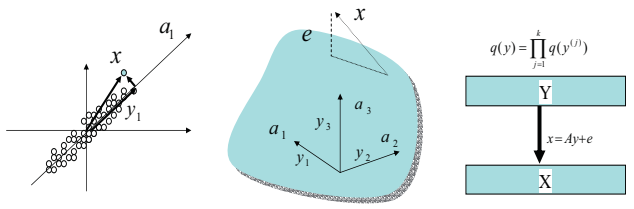


Figure 8. Subspace structures spanned by linear independent base vectors and featured by probabilistic independent coordinate variables

As shown in Fig.8, a typical dependence structure, that can provide an overall insight on a set of samples in a high dimensional space $x \in R^d$ with $x = [x^{(1)}, \cdots, x^{(d)}]^T$, is an appropriate linear subspace that covers most of samples. Such a subspace is featured by the following three ingredients:

- **a set of bases vectors** $a_j \in R^d, j = 1, \cdots, m$ that span the subspace, which provides the support on which data samples can have different specific distributions. To avoid redundance, it is naturally that all the $a_j \in R^d, j = 1, \cdots, m$ are mutually linear independent. That is,

$$det[A^T A] \neq 0, \quad A = [a_1, \cdots, a_m]. \tag{40}$$

- **coordinates** $y = [y^{(1)}, \cdots, y^{(m)}]^T$, with each $y^{(j)}$ denoting the corresponding coordinate on the basis a_j for representing each x, that is, we have

$$\hat{x} = \textstyle\sum_{j=1}^m y^{(j)} a_j, \tag{41}$$

  for representing $x \in R^d$. A set of samples from a specific distribution $p(x)$ is mapped into a inner representation by a specific distribution $q(y)$ supported on a subspace with a much lower dimension m. To maximize the representative capacity, redundance between any pair $y^{(i)}$ and $y^{(j)}$ should be removed. In a probabilistic sense, it means that eq.(10) is satisfied.

- **the residual** $x - \hat{x} = e$ that indicates how well a set of samples is described by this subspace. $e = 0$ means that x is located within the subspace and x is completely represented by $\hat{x}$, and $e \neq 0$ means that x is located outside the subspace and $\hat{x}$ is a projection of x on the subspace, with an error e for using $\hat{x}$ to represent x. If the best subspace is found to represent the samples, we have either $e = 0$ for every sample when the samples of x have not been polluted by noise or otherwise $e \neq 0$ describes the noise. This e should be independent from y and often regarded as from Gaussian $G(e|0, \Sigma)$ with Σ being usually isotopic $\Sigma = \sigma^2 I$ or sometime diagonal in a complicated situation.

The above discussion is made on assuming that both the origins of x and y coordinates are located at zero. Generally, we can get $x - \mu$ or $y - \nu$ located at zero. It follows from $x - \hat{x} = e$ and eq.(41) that we have

$$x = \begin{cases} Ay + e, & \text{(a) } Ex = 0, Ey = 0, \\ Ay + \mu + e & \text{(b) } \mu = Ex - AEy, \\ A(y - \nu) + \mu + e, & \text{(c) } \nu = Ey, \mu = Ex, \end{cases}$$
$$A = [a_1, \cdots, a_m], \tag{42}$$

which, especially the case (a), is widely referred under different names in the literatures. One is called *linear generative model* since it describes how x is generated via a linear model. The other is called *latent or hidden model* since y is not directly visible from observation. It is also called factor analysis (FA) model, regarding the components of y as the underlying factors.

Several typical examples of eq.(42) have been investigated in the past decades, mainly featured by $q(y^{(j)})$ in different distributions.

When $y^{(j)}$ is a real variable from a Gaussian distribution $G(y^{(j)}|0, \lambda^{(j)\,2})$, eq.(42) in its case (a) has been widely studied in the literature of statistics under the name of factor

analysis. Here, we call it Gaussian factor analysis and leave the name of factor analysis for the general case of eq.(42) with eq.(10). Particularly, we are lead to principal component analysis (PCA) when $\Sigma = \sigma^2 I$. Further studies have also been made in recent years on Eq.(42) in its case (b) with $y^{(j)}$ being a real variable from a nonGaussian distribution. One particular example of such $q(y^{(j)})$ is the following Gaussian mixture:

$$\begin{aligned} & y = \nu + \varepsilon,\ Ey = \nu, \\ & q(\varepsilon^{(j)}) = \sum_i \beta_{ji} G(y^{(j)}|m_{ji}, \lambda_{ji}^2), \\ & \sum_i \beta_{ji} m_{ji} = 0,\ \sum_i \beta_{ji} = 1,\ 0 \leq \beta_{ji} \leq 1. \end{aligned} \tag{43}$$

Moreover, when $\lambda_\ell^{(j)\ 2} \to 0, \ell = 1, \cdots, k$, we are lead to the case that $y^{(j)}$ is constrained to take only finite isolated points $r_\ell, \ell = 1, \cdots, k$ (shortly ℓ is used to denote the choice $y^{(j)} = r_\ell$). The details of studies are referred to [29].

4.2. Binary Factor Analysis and Adaptive Algorithm

When $y_t^{(j)}$ is a binary number that comes from a Bernoulli distribution:

$$q(y^{(j)}) = (\nu^{(j)})^{y^{(j)}} (1 - \nu^{(j)})^{1 - y^{(j)}}, \tag{44}$$

eq.(42) of case (b) with $e \neq 0$ has been studied under the name of Binary Factor Analysis (BFA) [33, 31, 30] or multiple cause model [21, 7].

It follows from eq.(18) that we can obtain an adaptive algorithm that implements parameter learning as follows:

The Yang Step

$$\begin{aligned} & y_t = f(x_t, \theta_f) = arg \max_y D(y, x_t), \\ & D(y, x_t) = \ln[G(x_t|A\varepsilon + \mu, \Sigma) \times \\ & \prod_j (\nu^{(j)})^{y^{(j)}} (1 - \nu^{(j)})^{1 - y^{(j)}}]_{\varepsilon = y - \nu}. \end{aligned}$$

The Ying Step

$$\begin{aligned} (a)\quad & \varepsilon_t = y_t - \nu^{old},\ e_t = x_t - \mu^{old} - A^{old}\varepsilon_t, \\ & \mu^{new} = \mu^{old} + \eta_t e_t,\ \nu^{new} = \nu^{old} + \eta_t \varepsilon_t, \\ & A^{new} = A^{old} + \eta_t \delta A, \\ & \text{where } \delta A \text{ is a step of moving along the ascent} \\ & \text{direction of } \ln G(x_t|A\varepsilon + \mu, \Sigma) \text{ subject to eq.(40),} \\ (b)\quad & \Sigma^{new} = (1 - \eta_t)\Sigma^{old} + \eta_t \delta\Sigma,\ E_t = e_t e_t^T, \\ & \delta\Sigma = \begin{cases} E_t, & \Sigma \text{ is general,} \\ diag[E_t], & \Sigma \text{ is diagonal,} \\ \frac{1}{d} Tr[E_t] I, & \Sigma = \sigma^2 I, \end{cases} \\ (c)\quad & \text{updating } q(y^{(j)}) \text{ by } \zeta_j^{new} = \zeta_j^{old} + \eta_t \varepsilon_t^{(j)}, \\ & \nu^{(j)\ new} = \frac{1}{1 + e^{-\zeta_j^{new}}}, \\ & \text{If } \nu^{(j)\ new}(1 - \nu^{(j)\ new}) \to 0 \text{ constantly,} \\ & \text{discard the component } y_t^{(j)}. \end{aligned} \tag{45}$$

where $\eta_t > 0$ is a learning step size, it can be different for updating different parameters, we simply use the same notation η_t for simplicity, and $diag[B]$ means the diagonal part of the matrix B.

One example for δA in (a) is

$$\delta A = e_t \varepsilon_t^T, \tag{46}$$

which makes the computing on updating A very simple. However, to keep eq.(40) satisfied, we need to compute $det[A^T A]$ per updating or frequently. If we find $det[A^T A] = 0$ constantly, we need to reduce the dimension of y from m to $m - 1$. After such a reduction, the previous learning result may be disturbed. If still $det[A^T A] = 0$ constantly, we need to further reduce the dimension of y. Such a process will repeat until eq.(40) becomes always satisfied.

Another example for δA in (a) is same as that introduced in Sec.5.3(3) in [29] or more clearly in Sec.IV(B) [23]. That is, we make the following singular value decomposition

$$\begin{aligned} & A = UDV^T = \sum_{j=1}^m d_j u_j v_j^T, \\ & U = [u_1, \cdots, u_m],\ V = [v_1, \cdots, v_m], \\ & U^T U = I,\ VV^T = I, \end{aligned}$$

where u_j is a d-dimension vector and v_j is a m-dimension vector. It can be observed that $det[A^T A] = 0$ if any one of $d_j, j = 1, \cdots, m$ is zero. Thus, we remove the corresponding $u_j, v_j, y^{(j)}$ if $d_j = 0$ constantly. This type of dimension reduction of y makes the previous learning result disturbed in a minimum extent. To save the computation of making the SVD decomposition $A = UDV^T$, updating A is replaced by updating U, V, D as follows:

$$\begin{aligned} U^{new} &= U^{old} + \eta_0 (g_U^T - U^{old} g_U U^{old}), \\ g_U &= \frac{\partial \ln G(x_t|UDV^T\varepsilon + \mu, \Sigma)}{\partial U} = \Sigma^{old\ -1} e_t \varepsilon_t^T V D, \\ V^{new} &= V^{old} + \eta (g_V - V^{old} g_V^T V^{old}), \\ g_V &= \frac{\partial \ln G(x_t|UDV^T\varepsilon + \mu, \Sigma)}{\partial V} = \varepsilon_t e_t^T \Sigma^{old\ -1} U D, \\ D^{new} &= D^{old} + \eta_0 diag[U\Sigma^{old\ -1} e_t \varepsilon_t^T V], \end{aligned} \tag{47}$$

where the updating on V and U is made under the constraint $V^T V = I$ and $UU^T = I$.

During learning, maximizing $\sum_{j=1}^m [\nu^{(j)} \ln \nu^{(j)} + (1 - \nu^{(j)}) \ln (1 - \nu^{(j)})]$ will also push $\nu^{(j)\ new}(1 - \nu^{(j)\ new}) \to 0$ constantly if the dimension $y_t^{(j)}$ is extra [29]. When this happens, we can remove the corresponding $u_j, v_j, y^{(j)}$.

If needed, we may also alternatively implement BFA in a two stage style. At the first stage, parameter learning is made either by the ML learning [5, 12, 33] or by the above learning with every $\nu^{(j)} = 0.5$ and δA given by eq.(46). In this case, there is no need to keep eq.(40) satisfied. At the second stage we select a best number m by

$$\min_m J(m),\ J(m) = 0.5 d \ln \sigma^2 + J_y(m) \tag{48}$$

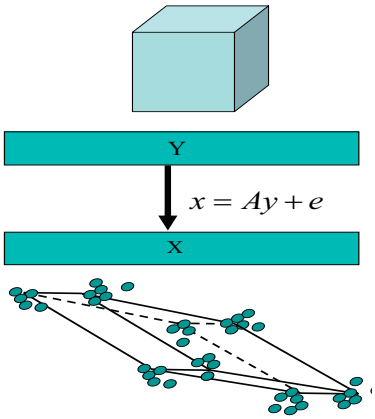


Figure 9. A mixture of 8 Gaussian clusters with their means located on vertices of a polyhedra that is obtained by an affine transformation from a cubic.

$$J_y(m) = \begin{cases} m \ln 2, & \nu^{(j)} = 0.5, \\ -\sum_{j=1}^{m} [\nu^{(j)} \ln \nu^{(j)} & \\ +(1-\nu^{(j)}) \ln (1-\nu^{(j)})], & \text{ML learning.} \end{cases}$$

5. Gaussian Densities on Projected Polyhedra

We can get further insights on BFA from the following two aspects:

- In a special case with $\Sigma = \sigma^2 I$ and $\nu = 0$, it implements $\min_A E\|x - \mu - Ay\|^2$ subject to y coming from eq.(10) with eq.(44). That is, it minimizes the residual $e = x - \hat{x}$ that x is represented by its projection $\hat{x} = Ay + \mu$ on the subspace spanned by the column vectors of A while the projection is constrained to take only 0 or 1 according to eq.(44). In other words, $\hat{x}$ can only be the sum of a subset of the column vectors of A.
- $q(x)$ that corresponds to eq.(42) is a mixture of 2^m Gaussians with each Gaussian having a same covariance matrix Σ but its mean locating at one vertex of a polyhedra, which, as shown in Fig.9, is obtained by an arbitrary affine transformation from a m dimensional hypercubic. In this way, 2^m mean vectors are obtained from the m column vectors of A only. Moreover, the proportion of each Gaussian is simply $q(y)$ with m free parameters instead of $2^m - 1$ free parameters. Since the number of free parameters has been reduced significantly, this constrained Gaussian mixture gets a better generalization ability.

Following the BYY harmony learning with a BI-directional regularization at the special case given by eq.(19), we consider

$$p(y|x_t, \vartheta) = \sum_{y' \in N_\vartheta} \eta_t(y') \delta(y - y'),$$
$$\eta_t(y') = \begin{cases} 1-\gamma, & \text{for } y' = \vartheta, \\ \frac{\gamma}{\#N_\vartheta}, & \text{for } y' \in N_\vartheta \text{ but } y' \neq \vartheta, \end{cases} \quad (49)$$

where N_ϑ consists of ϑ and a set of values that is different from ϑ by only one bit. The idea is that a vertex ϑ and its neighboring vertices should describe samples that are similar in certain extent. Thus, the mapping to a vertex is shared with a small fraction γ by its neighboring vertices. Due to this constraint, we get

$$\tilde{\vartheta} = arg \max_{\vartheta} D(\vartheta, x_t),$$
$$D(\vartheta, x_t) = \sum_{y \in N_\vartheta} \eta_t(y) D(y, x_t). \quad (50)$$

Correspondingly, learning by eq.(45) is implemented with the following modifications:

$$\begin{array}{ll} Yang - step & \text{Instead of getting only one } y_t, \text{ a set } N_{\bar{\vartheta}} \\ & \text{of samples of } y \text{ is obtained,} \\ Ying - step & \text{each updating is repeated for every} \\ & y_t \in N_{\bar{\vartheta}} \text{ with } \eta_t \text{ replaced by } \eta_t \eta_t(y_t). \end{array} \quad (51)$$

Moreover, γ can be a very small value at the beginning and then gradually increases as learning goes.

With the above learning, similar patterns will be mapped onto vertices that are nearby each other. That is, the mapping may reserve the topological relation among patterns as well. As a result, not only we may use a sample with certain information missing to reconstruct the corresponding pattern but also we may recall out a number of patterns that are similar to a particular one.

6. BFA Variants

A binary factor based subspace is also useful to another important class of applications that each component $x^{(i)}$ only takes 1 or 0. That is, both y and x are binary vectors. In such cases, the BFA is no longer applicable directly. If using the probabilities $P(y|x)$ to describe how y is generated from x, the number of free parameters will be an order of $2^m \times 2^d$, which needs a large size of samples to learn. Many efforts have been made in the current literature to handle this type of problems. One example is called multiple cause mixture [21]. It models each bit $\hat{x}^{(j)} = 1 - \prod_i (1 - y_i a_{ij})$ via binary a_{ij} and then matches the observed bit $x^{(j)}$ with a heuristic cost function. In this setting, the number of free parameters reduces significantly to $md + m$. Also, the ML learning is proposed on this model [7], with each binary code x interpreted as Bernoulli via defining the probability that $x^{(j)} = 1$ in help of a generating model $1 - \prod_i (1 - y_i a_{ij})$.

However, the process of learning the values of $a_{i,j}$ is a combinatorial optimization that needs to search 2^{md} choices.

We consider that A is a real matrix via the following structure [27, 29]:

$$
\begin{aligned}
& q(x|y) = \prod_{i=1}^{d} (\mu^{(i)})^{x^{(i)}} (1-\mu^{(i)})^{1-x^{(i)}}, \\
& \mu^{(i)} = \frac{1}{1+e^{-\hat{x}^{(i)}}}, \\
& \hat{x} = A(y-\nu)+\mu, \\
& f(x_t, \theta_f) = arg\max_y D(y, x_t), \\
& D(y, x_t) = \ln[\prod_{i=1}^{d} (\mu^{(i)})^{x_t^{(i)}} (1-\mu^{(i)})^{1-x_t^{(i)}} \times \\
& \prod_{j=1}^{m} (\nu^{(j)})^{y^{(j)}} (1-\nu^{(j)})^{1-y^{(j)}}], \\
& D(y, x_t) = \sum_{i=1}^{d} [x_t^{(i)} \ln \mu^{(i)} + (1-x_t^{(i)}) \ln (1-\mu^{(i)})] \\
& + \sum_{j=1}^{m} [y^{(j)} \ln \nu^{(j)} + (1-y^{(j)}) \ln (1-\nu^{(j)})], \qquad (52)
\end{aligned}
$$

with the updating on A still made by eq.(45) with $e_t = x_t - \mu$.

Though being able to turn a combinatorial enumeration into gradient based local search, the representation in the form $\prod_{i=1}^{d} (\mu^{(i)})^{x^{(i)}} (1-\mu^{(i)})^{1-x^{(i)}}$ can not cover mutual information among the components of x. To improve this shortcoming, we here propose a generalized BFA that is able to handle the case that both y and x are binary vectors, still in help of the model eq.(42). We consider an observation space with noise e. Its dimension n may be different from the dimensions of both x, y. In this space, x is not directly observable but observed via a set of linear bases vectors $[w_1, \cdots, w_d] = W$ with the coordinates $[x^{(1)}, \cdots, x^{(d)}]^T = x$, respectively, i.e., $Wx = x'$ is observed.

In implementation, we get y_t by the Yang step in eq.(45) with x_t replaced by $x'_t = Wx_t$. Then we update Σ and $q(y^{(j)})$ by the Ying step (b) in eq.(45). Moreover, we make other updating as follows:

$$
\begin{aligned}
& e_t = Wx_t - A\varepsilon_t - \mu, \\
& A^{new} = A^{old} + \eta_t e_t \varepsilon_t^T, \\
& W^{new} = W^{old} - \eta_t e_t x_t^T. \qquad (53)
\end{aligned}
$$

After learning, we set up a mapping $x_t \to y_t$ via $Wx_t = x'_t$ inserted into the Yang step in eq.(45) in place of x_t. Also, we set up a mapping $y_t \to x_t$ via $x_t = arg\max_x \|Wx - A\varepsilon_t - \mu\|^2$. This bi-directional binary relation can be applied to rule based inferences. Also, this generalized BFA can be directly modified to cover the cases that each $x^{(i)}$ takes a number of discrete values.

In certain applications, we encounter the so called nonnegative factor analysis problem with both x, y only taking negative values. Actually, the above BFA and extensions can be regarded as special examples of this type. Other examples come from the cases that the components of x are real positive numbers, which can also be handled by the BFA via a slight modification

$$
A = [\alpha_{ij}^2],\ \alpha_{ij}^{new} = \alpha_{ij}^{old} + \eta_t e_t^{(i)} \varepsilon_t^{(j)} \alpha_{ij}^{old}. \qquad (54)
$$

Alternatively, we can also turn each component of $\hat{x} = Ay + \mu$ to a nonnegative value via a simple nonlinear transform, e.g., $\eta = \xi^2$ [29] with the following modification

$$
e_t^{(i)} = x_t^{(i)} - (\hat{x}_t^{(i)})^2,\ a_{ij}^{new} = a_{ij}^{old} + \eta_t e_t^{(i)} \hat{x}_t^{(i)} \varepsilon_t^{(j)}. \qquad (55)
$$

In addition, the generalized BFA by eq.(53) can also be directly applied to the cases that the components of x are real positive numbers.

Again, all the above learning algorithms can be extended to the case of eq.(51) with $D(\vartheta, x_t)$ in eq.(50).

Still, in the updating by eq.(53), eq.(54), and eq.(55), we need to compute $det[A^T A]$ per updating or frequently. If we find $det[A^T A] = 0$ constantly, we need to reduce the dimension of y until eq.(40) become always satisfied.

7. Concluding Remarks

From the perspective of BYY harmony learning with a bi-directional architecture, the Kohonen map type of topological structure is revisited with a new insight and a new variant. Then, it has been further extended to a multi-set -mixture based topological map for object detection. Moreover, an adaptive BFA algorithm is provided for learning a type of Gaussian mixture with 2^m Gaussian densities located on vertices of a projected hyper polyhedra structure that are represented via only m real vectors, with an appropriate m determined automatically during learning. Also, another topological type is introduced into this projected hyper polyhedra structure.

References

[1] Akaike, H (1974), "A new look at the statistical model identification", *IEEE Tr. Automatic Control*, **19**, 714-723.

[2] Amari, S. & Nagaoka, H., (2000), *Methods of Information Geometry*, AMS Translations of Mathematical Monographs, Vol. 191, Oxford University Press.

[3] Amari, SI, Cichocki, A, & Yang, HH (1996), "A new learning algorithm for blind separation of sources", in DS Touretzky, et al, eds, *Advances in Neural Information Processing 8*, MIT Press, 757-763.

[4] Bell, A & Sejnowski, T (1995), "An information maximization approach to blind separation and blind deconvolution", *Neural Computation, 17*, 1129-1159.

[5] Belouchrani, A., & Cardoso, J.-F. (1995), "Maximum likelihood source separation by the expectation-maximization technique: deterministic and stochastic implementation", *Proc. NOLTA95*, 49-53.

[6] Chan, KY, Chu, WS, & Xu, L (2003), "Experimental Comparison between two computational strategies for topological self-organization", *Proc. of IDEAL03*, Lecture Notes in Computer Science, LNCS 2690, Springer-Verlag, pp410-414.

[7] Dayan, P. & Zemel, R. S., (1995), "Competition and multiple cause models", *Neural Computation 7*, 565-579.

[8] Dempster, AP, Laird, NM, & Rubin, DB, (1977), "Maximum- likelihood from incomplete data via the EM algorithm", *J. of Royal Statistical Society*, **B39**, 1-38.

[9] Devijver, PA, & Kittler, J (1982), *Pattern Recognition: A Statistical Approach*, Prentice-Hall.

[10] Duda, RO, & Hart, PE (1973), *Pattern classification and Scene analysis*, Wiley.

[11] Fayyad, UM , et al (1996), *Advances in Knowledge Discovery and Data Mining*, AAAI/MIT Press.

[12] Grellier, O. & Comon, P. (1998), "Blind Separation of Discrete Sources," *IEEE Signal Processing Letters, 5(8)*, 212-214. .

[13] Hinton, GE, Dayan, P, Frey, BJ, & Neal, RN (1995), "The wake-sleep algorithm for unsupervised learning neural networks", *Science 268*, 1158-1160.

[14] Kohonen, T (1995), *Self-Organizing Maps* , Springer-Verlag, Berlin.

[15] Ma, J, Wang, T, & Xu, L (2004), "A gradient BYY harmony learning rule on Gaussian mixture with automated model selection", *Neurocomputing 56*, 481-487.

[16] McLachlan, GJ & Krishnan, T (1997) *The EM Algorithm and Extensions*, John Wiley & Son, INC.

[17] von der Malsburg, Ch (1973), "Self-organization of orientation sensitive cells in the striate cortex", Kybernetik **14**, 85-100.

[18] Nasrabadi, N & King, RA (1988), "Image coding using vector quantization: a review," *IEEE Trans. Communnication*, **36**, 957-971 .

[19] Redner, RA & Walker, HF (1984), "Mixture densities, maximum likelihood, and the EM algorithm", *SIAM Review*, **26**, 195-239.

[20] Rissanen, J (1999), "Hypothesis selection and testing by the MDL principle", *Computer Journal*, **42** (4), 260-269.

[21] Saund, E, (1995), "A multiple cause mixture model for unsupervised learning", *Neural Computation*, Vol.7, pp51-71.

[22] Tikhonov, AN & Arsenin, VY (1977), *Solutions of Ill-posed Problems*, Winston and Sons.

[23] Xu, L (2004a), "Advances on BYY harmony learning: information theoretic perspective, generalized projection geometry, and independent factor auto-determination", *IEEE Trans on Neural Networks, Vol. 15, No. 4*, pp885-902.

[24] Xu, L. (2004b), "Advances on BYY Harmony Learning: Information Theoretic Perspective, Generalized Projection Geometry, and Independent Factor Auto-determination", *IEEE Trans on Neural Networks, Vol. 15, No. 5*, pp885-902.

[25] Xu, L. (2004), "Bayesian Ying Yang Learning: (I) A Unified Perspective for Statistical Modeling", *Intelligent Technologies for Information Analysis*, N. Zhong and J. Liu (eds), Springer, pp613-658.

[26] Xu, L. (2004), "Bayesian Ying Yang Learning (II): A New Mechanism for Model Selection and Regularization", *Intelligent Technologies for Information Analysis*, N. Zhong and J. Liu (eds), Springer, pp661-706.

[27] Xu, L. (2003), "BYY Learning, Regularized Implementation, and Model Selection on Modular Networks with One Hidden Layer of Binary Units", *Neurocomputing*, Vol.51, p227-301.

[28] Xu, L. (2003), "Data smoothing regularization, multi-sets-learning, and problem solving strategies", *Neural Networks, Vol. 15, Nos. 5-6*, 817-825.

[29] Xu, L. (2003), "Independent Component Analysis and Extensions with Noise and Time: A Bayesian Ying-Yang Learning Perspective", *Neural Information Processing Letters and Reviews*, Vol.1, No.1, 1-52.

[30] Xu, L (2002), "BYY Harmony Learning, Structural RPCL, and Topological Self-Organizing on Mixture Models ", *Neural Networks, Vol. 15, Nos. 8-9*, 1125-1151.

[31] Xu, L (2001), "BYY Harmony Learning, Independent State Space and Generalized APT Financial Analyses ", *IEEE Tr. Neural Networks*, **12** (4), 822-849.

[32] Xu, L (2001), "Best Harmony, Unified RPCL and Automated Model Selection for Unsupervised and Supervised Learning on Gaussian Mixtures, Three-Layer Nets and ME-RBF-SVM Models", *Intl J of Neural Systems* **11** (1), 43-69.

[33] Xu, L (1998), "Bayesian Kullback Ying-Yang Dependence Reduction Theory", *Neurocomputing 22 (1-3)*, 81-112, 1998.

[34] Xu, L, (1998), Rival Penalized Competitive Learning, Finite Mixture, and Multisets Clustering, *Proc. of IJCNN98*, Anchorage, Alaska, Vol.II, 2525-2530.

[35] Xu, L, (1996), "Bayesian-Kullback YING-YANG learning scheme: reviews and new results", *Proc. Intl. Conf. on Neural Information Processing (ICONIP96)*, Vol.1, Sept. 24-27, 1996, Hong Kong, pp59-67.

[36] Xu, L, (1995), "Bayesian-Kullback Coupled YING-YANG Machines: Unified Learnings and New Results on Vector Quantization", *Proc. Intl. Conf. on Neural Information Processing (ICONIP95)*, Beijing, China, 977-988.

[37] Xu, L (1995), "A unified learning framework: multisets modeling learning," *Proceedings of 1995 World Congress on Neural Networks*, vol.1, pp35-42.

[38] Xu, L, (1994), "Multisets Modeling Learning: An Unified Theory for Supervised and Unsupervised Learning", Invited Talk, *Proc. IEEE ICNN94*, June 26-July 2, 1994, Orlando, Florida, Vol.I, pp.315-320.

[39] Xu, L, Krzyzak, A & Oja, E (1993), "Rival Penalized Competitive Learning for Clustering Analysis, RBF net and Curve Detection", *IEEE Tr. on Neural Networks 4*, 636-649.

[40] Xu, L (1991&93) "Least mean square error reconstruction for self-organizing neural-nets", *Neural Networks 6*, 627-648, 1993. Its early version on *Proc. IJCNN91'Singapore*, 2363-2373, 1991.

Contributed Papers

Compact Representations of Sequential Classification Rules

Elena Baralis, Silvia Chiusano, and Luigi Mantellini

Politecnico di Torino
Dipartimento di Automatica ed Informatica
Corso Duca degli Abruzzi, 24
10129 Torino, Italy
{elena.baralis,silvia.chiusano,luigi.mantellini}@polito.it

Summary. Mining frequent sequential patterns is a relevant data mining task, which finds applications such as web mining, bioinformatic data analysis, and text mining. A further recent step is the exploitation of sequential information for classification purposes.

In this paper we address the problem of mining sequential classification rules. Unfortunately, while high support thresholds may yield an excessively small rule set, the solution set becomes rapidly huge for decreasing support thresholds. In this case, the extraction process becomes time consuming (or is unfeasible), and the generated model is too complex for human analysis.

We propose two compact forms to encode the knowledge available in a sequential classification rule set. These forms are based on the abstractions of general rule, specialistic rule, and complete compact rule. The forms are obtained by extending the concept of closed itemset and generator itemset to the the context of sequential rules. Experimental results show that a significant compression ratio is achieved by means of both proposed forms.

1 Introduction

Association rules [2] describe the co-occurence among data items in a large amount of collected data. They have been profitably exploited for classification purposes [10, 14, 5]. In this case, rules are called classification rules and their consequent contains the class label. Classification rule mining is the discovery of a rule set in the training dataset to form a model of data, also called classifier. The classifier is then used to classify new data for which the class label is unknown.

Data items in an association rule are unordered. However, in many application domains (e.g., web log mining, DNA and proteome analysis) the order among items is an important feature. Sequential patterns have been first introduced in [3] as a sequential generalization of the itemset concept. In [20, 27] efficient algorithms to extract sequences from datasets are proposed. The algorithms are based on lattice theory and prefix-projection. In this paper, we propose classification rules based on

sequential patterns. We define as *sequential classification rule* an implication where the antecedent is a sequence and the consequent is a class label. This definition is a classification specialization of the notion of sequential association rule proposed in [24] for web logging applications.

In large or highly correlated datasets, rule extraction algorithms have to deal with the combinatorial explosion of the solution space. This causes (i) the rule extraction process to be frequently unfeasible, and (ii) the solution set to be hardly understandable by a human being. To cope with this problem, pruning of the generated rule set based on some quality indexes (e.g., χ^2, confidence and support) is usually performed. In this way rules which are redundant from a functional point of view [10, 14] are discarded. A different approach consists in generating equivalent representations [4] that are more compact and without information loss. The compact form in [4] is an extension of the concepts of closure and generator itemset [19, 16, 17, 18, 15, 6, 25] to the associative classification domain.

In this paper we propose two compact forms to represent sets of sequential classification rules. These forms are based on the concepts of closed sequence and generator sequence, and use them to summarize a large rule set with a small number of compact rules. The first compact form is based on the concept of generator sequence, which is an extension to sequential patterns of the concept of generator itemsets [18]. Generator sequences code the minimal and non redundant information with respect to all sequences coded into a closed sequence. Based on generator sequences, we define general sequential rules. The collection of all general sequential rules extracted from a dataset represents a sequential classification rule cover. A rule cover encodes all useful classification information in a sequential rule set (i.e., is equivalent to it for classification purposes), but does not allow the regeneration of the complete rule set.

While the notion of generator sequence, to our knowledge, is new, closed sequences have been introduced in [23, 21]. A closed sequence is the maximal sequence representing all sequences coded in a closure. Based on closed sequences, we define closed sequential rules. A closed sequential rule is the most specialistic (i.e., characterized by the longest sequence) rule into a set (closure) of equivalent rules. Unfortunately, closed sequences, differently from generator sequences, do not yield a classification rule cover. The second proposed compact form exploits jointly closed sequences and their associated generator sequences. In particular, to allow regeneration of the complete rule set, to each closed sequential rule is associated the complete set of its generator sequences.

We also introduce a specialized type of sequence, the contiguous sequence. A sequence is contiguous when the items appearing in it are always adjacent (i.e., no other items are interleaved). Contiguous sequences are interesting in many biological contexts like DNA and proteome analyis, where the domain of items is characterized by very low cardinality. All theoretical results presented in this paper hold for both the general and contiguous sequence domains.

The paper is organized as follows. Section 2 introduces the problem statement and basic definitions. Sections 3 and 4 describe the compact forms for sequences and for sequential rules, respectively. Section 5 reports preliminary experimental result on the compression effectiveness of the proposed techniques. Finally, Section 6 draws conclusions and outlines future work.

2 Problem statement

In this section we introduce notation and fundamental definitions for sequential data mining.

Definition 1 (Sequence). *Let $\mathcal{I}$ be a set of items. A sequence S on $\mathcal{I}$ is an ordered list of events, denoted $S = (e_1, e_2, \ldots, e_n)$, where each event $e_i \in S$ is an item in $\mathcal{I}$.*

In a sequence, each item can appear multiple times, in different events. The overall number of items in S is the length of S, denoted $|S|$. A sequence of length n is called n-sequence.

In this paper we focus on single item sequences. In these sequences, each event contains a single item. Our definition of sequence is a restriction of the definition of sequence proposed in [3, 27], where each event contains more items. Single item sequences seem more adequate for specific application domains where each element of the sequence is a single symbol (e.g., a word or an aminoacid).

A dataset $\mathcal{D}$ for sequence mining consists of a set of sequences. Each sequence in $\mathcal{D}$ is characterized by a unique identifier, named Sequence Identifier (SID). When dataset $\mathcal{D}$ is used for classification purposes, each sequence in $\mathcal{D}$ is labeled by means of a class label c. Hence, dataset $\mathcal{D}$ is a set of tuples (SID, S, c), where S is a sequence identified by the value SID and c is a class label belonging to the set $\mathcal{C}$ of class labels in $\mathcal{D}$. Table 1 reports a very simple sequence dataset, used as a running example in this paper.

SID	Sequence	Class
0	$ADCA$	c_1
1	$ADCBA$	c_2
2	ABE	c_1
3	$FGHFJ$	c_1
4	$FGIFJ$	c_1

Table 1. Example sequence dataset $\mathcal{D}$

In the following, we introduce the concept of subsequence with constraints.

Definition 2 (Matching function). *Let $X = (x_1, x_2, \ldots, x_l)$ and $Y = (y_1, y_2, \ldots, y_m)$ be two arbitrary sequences. A function $\psi : \{1, \ldots, m\} \longrightarrow \{1, \ldots, l\}$ is a matching function from Y to X if ψ is strictly monotonically increasing and $\forall j \in \{1, \ldots, m\}$ it is $y_j = x_{\psi(j)}$.*

Definition 3 (Constrained Subsequence). *Let Ψ be a set of matching functions between two arbitrary sequences, and $X = (x_1, x_2, \ldots, x_l)$ and $Y = (y_1, y_2, \ldots, y_m)$ two arbitrary sequences. Y is a subsequence of X with respect to Ψ, written as $Y \sqsubseteq_\Psi X$, iff $\exists \psi \in \Psi$ matching Y to X.*

When Ψ is the universe of all possible matching functions, we omit it for the sake of readability and we say simply that sequence Y is a subsequence of sequence X.

A particular type of subsequence relation is the *contiguous subsequence relation*, where the elements of sequence Y match with elements of sequence X without gap, i.e., no other element is allowed to be interleaved. In this case, the matching function can be characterized as $\psi(j) = \text{offset} + j$. When the first (last) $|Y|$ elements of X are equal to the elements of Y in the same order, Y is a prefix (suffix) subsequence of X.

Consider the example dataset in Table 1. DA is a non-contiguous subsequence of both sequences $ADCA$ and $ADCBA$. Sequence DC is a contiguous subsequence of $ADCA$, where the matching function is $\psi(j) = 1 + j$. Sequence CD is not a subsequence of any sequence in the example dataset because it is not possible to build a matching function with respect to Definition 2.

The contiguity constraint is particularly interesting in the biological application domain. In DNA or proteome, which are long sequences of symbols, there is high correlation between contiguous elements, but correlation rapidly decreases with distance. With this rationale, we exploit the contiguity constraint to reduce the problem complexity and the number of extracted sequences with a low loss of representativeness.

The support of a sequence X [3] in a dataset $\mathcal{D}$ is the number of sequences in $\mathcal{D}$ that contain X. Formally, $\sup(X) = \text{Card}(\{(SID, S) \in \mathcal{D} : X \sqsubseteq_\Psi S\})$. A sequence X is frequent with respect to a given support threshold *minsup* when $\sup(X) \geq minsup$.

A sequential rule [3] is an implication in the form $X \rightarrow Y$, where X and Y are sequences in $\mathcal{D}$. X and Y are respectively the antecedent and the consequent of the rule. In this paper we derive from sequential rules the classification rules to be used for classification purposes.

Definition 4 (Sequential Classification Rule). *A sequential classification rule in $\mathcal{D}$ is an implication $r : X \rightarrow c$, where X is a sequence in $\mathcal{D}$, and c is a class label in $\mathcal{C}$.*

Differently from general sequential rules, the consequent of a sequential classification rule belongs to set $\mathcal{C}$, which is disjoint from $\mathcal{I}$. We say that a rule $r : X \rightarrow c$ covers (or classifies) a data object d if $X \sqsubseteq_\Psi d$. In this case, r classifies d by assigning to it class label c. Obviously, the contiguity constraint in the rule antecedent yields *contiguous sequential classification rules*.

Similarly to associative classification, we measure the quality of a sequential classification rule $r : X \rightarrow c$ by means of two quality indexes [10, 14], named rule support and rule confidence. The indexes measure the estimated accuracy of r in predicting the correct class for a data object d. The rule support is the number of sequences in $\mathcal{D}$ which contain X and are labeled by c, $\sup(r) = \text{Card}(\{(SID, S, c) \in \mathcal{D} : X \sqsubseteq_\Psi S \wedge c = c_i\})$. The rule confidence is given by the ratio $\text{conf}(r) = \sup(r)/\sup(X)$. A sequential rule is said to be frequent with respect to a given support threshold *minsup* if $\sup(r) \geq minsup$.

3 Compact Sequence Representations

To tackle with the generation of a large number of association rules, several alternative forms have been proposed for the compact representation of frequent item-

sets. Among them, maximal itemsets [7], closed itemsets [15, 26], free sets [11], disjunction-free generators [12], and deduction rules [13].

Recently, in [21, 23] the concept of closed itemset has been extended to represent frequent sequences.

Definition 5 (Closed sequence). *An arbitrary sequence X in $\mathcal{D}$ is a closed sequence with respect to a matching function set Ψ iff $\nexists Y$ in $\mathcal{D}$ such that (i) $X \sqsubset_\Psi Y$ and (ii)* $\sup(X) = \sup(Y)$.

In [21, 23] the definition of closed sequence was proposed in the case of uncostrained matching. In this paper, we address the case of *contiguous closed sequence*, when the sequence contains adjacent elements, and *non-contiguous closed sequence* when matching is unconstrained.

Intuitively, a closed sequence is the maximal subsequence common to a set of sequences in $\mathcal{D}$. A closed sequence X is a compact representation of all the subsequences Y that are (i) subsets of it, and (ii) included in the same sequences in $\mathcal{D}$. The closed sequence X which encodes Y is called the *sequential closure* of Y. A sequence database $\mathcal{D}$ can be encoded by means of the whole set of its closed sequences.

In the example dataset, ADA is a non-contiguous closed sequence, which represents the sequences ADA, AD, DA, AA, and D contained in sequences 0 and 1. Instead, ADC is a contiguous closed sequence, also contained in sequences 0 and 1. ADC, AD, DC, C and D are the sequences represented in it.

To completely characterize closed sequences, in this paper we also extend the concept of generator itemset [17, 18] to the domain of sequences. A generator sequence is the shortest sequence among those represented in a closed sequence.

Definition 6 (Generator sequence). *An arbitrary sequence X in $\mathcal{D}$ is a generator sequence with respect to a matching function set Ψ iff $\nexists Y$ in $\mathcal{D}$ such that (i) $Y \sqsubset_\Psi X$ and (ii)* $\sup(X) = \sup(Y)$.

Analogously to closed sequences, the contiguity constraint yields *contiguous generators*, while the absence of constraint gives *non-contiguous generators*. In the example dataset, D and AA are non-contiguous generators for the non-contiguous closed sequence ADA. C and D are contiguous generators for the contiguous closed sequence ADC.

Based on Definition 6, all the sequences represented in a closed sequence X can be generated starting from every generator sequence in X and "extending" it within X. In other words if Z is a sequence represented by a closed X and an associated generator $Y \sqsubseteq_\Psi X$, then Z is contained in X, and Y is contained in Z.

In the context of association rules, the closure for an arbitrary itemset is unique. The property of uniqueness is lost in the context of sequences for both contiguous and non-contiguous sequences. Hence, an arbitrary sequence X can be encoded by a set of closed sequences. We call this set, the *closure sequence set* of X, denoted $\mathcal{CS}(X)$. From this property it follows that a given generator sequence can generate different closed sequences.

For example, consider the contiguous closed sequences FG and FJ in the example dataset. The set of generators for FG is $\{F, G\}$, and for FJ is $\{F, J\}$. Hence, generator sequence F is associated to both closed sequences. Instead, G is a generator only for FG while J only for FJ.

4 Compact Representations of Sequential Classification Rules

In this section we propose two compact representations to encode the knowledge available in a sequential classification rule set. These representations are based on the concepts of closed and generator sequence introduced in the previous section.

The next theorem exploits the concept of sequential closure to characterize a set of sequential classification rules having the same values for both rule support and confidence.

Theorem 1. *Let $r_i : M \rightarrow c_i$ be an arbitrary sequential classification rule in $\mathcal{D}$, where M is a closed sequence in $\mathcal{D}$. Then, $\forall r_j : X \rightarrow c_j$ in $\mathcal{D}$, with $c_i = c_j$ and $M \in \mathcal{CS}(X)$, is (i) $\sup(r_i) = \sup(r_j)$, and (ii) $\mathrm{conf}(r_i) = \mathrm{conf}(r_j)$.*

Proof. By hypothesis, $M \in \mathcal{CS}(X)$. Hence, for the properties of the sequential closure presented in Section 3, M and X are contained in the same sequences in $\mathcal{D}$. Hence, $\sup(M) = \sup(X)$. Furthermore, rules r_i and r_j are contained in the same subset of sequences in $\mathcal{D}$, labeled by class $c_i = c_j$, and thus $\sup(r_i) = \sup(r_j)$. It trivially follows that $\mathrm{conf}(r_i) = \mathrm{conf}(r_j)$.

By theorem above, rules with the same consequent, and whose antecedents have the same sequential closure, are characterized by the same values of support and confidence. For example, consider the two contiguous rules $AD \rightarrow c_1$ and $DC \rightarrow c_1$ in the example dataset. These rules have both equal support and confidence since the contiguous closed sequence ADC belongs to the sequential closure set of both AD and DC. Analogously, non-contiguous rules $DA \rightarrow c_1$ and $AA \rightarrow c_1$ have the same values of support and confidence, since both sequences DA and AA are encoded in the non-contiguous closed sequence ADA. We note that the theorem above states a sufficient but not necessary condition.

In the next section we exploit the theorem above to introduce the concepts of general and specialistic classification rule. These rules characterize the more general (shorter) and more specific (longer) classification rules in a given classification rule set. We then exploit the concepts of general and specialistic rule to define the two compact forms presented in Section 4.2 and 4.3, respectively.

4.1 General and Specialistic Rule

In associative classification [10, 14, 22], a shorter rule (i.e., a rule with less elements in the antecedent) is often preferred to longer rules with lower confidence and/or support with the intent of both avoiding the risk of overfitting, and reducing the size of the classifier. However, in some applications (e.g., modeling surfing paths in web log analysis [24]), longer sequences may be more accurate since they contain more signature information about the user-access patterns. In these cases, longest-matching rules may be preferred to shorter ones.

To characterize both kind of rules, we propose the definition of specialization of a sequential classification rule.

Definition 7 (Classification Rule Specialization). *Let $r_i : X \rightarrow c_i$ and $r_j : Y \rightarrow c_j$ be two arbitrary sequential classification rules in $\mathcal{D}$. r_j is a specialization of r_i iff (i) $X \sqsubset_{\Psi} Y$, (ii) $c_i = c_j$, (iii) $\sup(r_i) = \sup(r_j)$, and (iv) $\mathrm{conf}(r_i) = \mathrm{conf}(r_j)$.*

Based on Definition 7, a classification rule r_j is a specialization of a rule r_i if r_i is more general than r_j, i.e., r_i has fewer conditions than r_j in the antecedent. Hence, any data object covered by r_j can be also covered by r_i, while the vice versa is not true. r_j and r_i both assign the same class label and have equal support and confidence.

Definition 7 is based on a similar definition proposed in the context of associative classification rules [4]. With respect to the definition of specialistic rule proposed in [10, 14, 22], the definition in [4] is more restrictive. In fact the two rules are required to have the same confidence, support and class label.

Based on Definition 7, we now introduce the concept of general rule. This is the rule with the shortest antecedent, among all rules having same class label, rule support and confidence.

Definition 8 (General Rule). *Let $\mathcal{R}$ be the set of frequent sequential classification rules in $\mathcal{D}$, and $r_i \in \mathcal{R}$ an arbitrary rule. r_i is a general rule in $\mathcal{R}$ iff $\nexists r_j \in \mathcal{R}$, such that r_i is a specialization of r_j.*

In the example dataset, $D \rightarrow c_1$ is a contiguous general rule with respect to the rules $DC \rightarrow c_1$ and $ADC \rightarrow c_1$. Instead, $AA \rightarrow c_1$ is a non-contiguous general rule for the non-contiguous rule $ADA \rightarrow c_1$.

The next lemma formalizes the concept of general rule by means of the concept of generator sequence. The lemma follows from Definitions 6 and 8.

Lemma 1 (General Rule). *Let $\mathcal{R}$ be the set of frequent sequential classification rules in $\mathcal{D}$, and $r \in \mathcal{R}$ an arbitrary rule. r is a general rule in $\mathcal{R}$ iff X is a generator sequence in $\mathcal{D}$.*

Based on Definition 7, we define the concept of specialistic rule.

Definition 9 (Specialistic Rule). *Let $\mathcal{R}$ be an arbitrary set of frequent sequential classification rules in $\mathcal{D}$, and $r_i \in \mathcal{R}$ an arbitrary rule. r_i is a specialistic rule in $\mathcal{R}$ iff $\nexists r_j \in \mathcal{R}$ such that r_j is a specialization of r_i.*

Based on the definition above, for a specialistic rule $r \in \mathcal{R}$, there is no rule in $\mathcal{R}$ such that r is included in it, and the two rules have both equal support and confidence. For example, $ADC \rightarrow c_1$ is a contiguous specialistic rule in the example dataset, with support 20% and confidence 50%. The contiguous rules $ADCA \rightarrow c_1$ and $ADCBA \rightarrow c_1$ which include it have support equal to 20% and confidence 100%.

The next lemma formalizes the concept of specialistic rule by means of the concept of closed sequence. The lemma follows from Definitions 6 and 9.

Lemma 2 (Specialistic Rule). *Let $\mathcal{R}$ be the set of frequent sequential classification rules in $\mathcal{D}$, and $r \in \mathcal{R}$ an arbitrary rule. r is a specialistic rule in $\mathcal{R}$ iff X is a closed sequence in $\mathcal{D}$.*

4.2 Sequential Classification Rule Cover

In this section we present a compact form which is based on the general rules in a given set $\mathcal{R}$. This form allows the classification of unlabeled data without information loss with respect to the complete rule set $\mathcal{R}$. Hence, it is equivalent to $\mathcal{R}$ for classification purposes.

Intuitively, we say that two rule sets are equivalent if they contain the same knowledge. When referring to a classification rule set, its knowledge is represented by its capability in classifying an arbitrary data object d. Note that d can be matched by different rules in $\mathcal{R}$. Each rule r labels d with a class c. The estimated accuracy of r in predicting the correct class is usually given by r's support and confidence.

The equivalence between two rule sets can be formalized in terms of rule cover.

Definition 10 (Sequential Classification Rule Cover). *Let $\mathcal{R}_1$ and $\mathcal{R}_2 \subseteq \mathcal{R}_1$ be two arbitrary sequential classification rule sets extracted from $\mathcal{D}$. $\mathcal{R}_2$ is a sequential classification rule cover of $\mathcal{R}_1$ if, (i) $\mathcal{R}_2$ is minimal, and $\forall r_i \in \mathcal{R}_1$ and $r_i : X \rightarrow c_i$, $\exists r_j \in \mathcal{R}_2$ and $r_j : Y \rightarrow c_j$, such that (ii) $Y \sqsubseteq_{\Psi} X$, (iii) $c_i = c_j$, (iv)* $\sup(r_i) = \sup(r_j)$*, and (v)* $\text{conf}(r_i) = \text{conf}(r_j)$.

When $\mathcal{R}_2 \subseteq \mathcal{R}_1$ is a classification cover of $\mathcal{R}_1$, the two sets classify in the same way an arbitrary data object d. If a rule $r_i \in \mathcal{R}_1$ labels d with class c, then in $\mathcal{R}_2$ there is a rule r_j, not necessarily identical to r_i, which labels d with the same class. r_i and r_j have both same support and same confidence. It follows that $\mathcal{R}_1$ and $\mathcal{R}_2$ are equivalent for classification purposes.

For a given rule set $\mathcal{R}$, the subset of its general rules is a general classification rule cover of $\mathcal{R}$. The next theorem proves this property. From the theorem it follows that this compact representation of $\mathcal{R}$ is equivalent to it for classification purposes.

Theorem 2 (Sequential Classification Rule Cover). *Let $\mathcal{R}$ be the set of frequent sequential rules in $\mathcal{D}$, and $\mathcal{G}$ the set of frequent generator sequences in $\mathcal{D}$. The subset of rules in $\mathcal{R}$ having as antecedent the elements of $\mathcal{G}$, is a sequential classification rule cover of $\mathcal{R}$*

$$CRC = \{r : G \rightarrow c | G \in \mathcal{G} \wedge r \in \mathcal{R}\} \tag{1}$$

Theorem 2 can be proved based on the characteristics of the generator sequences. Consider an arbitrary rule $r_i : X \rightarrow c$ in $\mathcal{R}$. Two options are possible. (i) X is a generator sequence. Hence, r_i belongs to CRC. (ii) X is not a generator sequence. In this case, there must be at least a rule $r_j : Y \rightarrow c$ in $\mathcal{R}$ such that Y is a generator sequence and r_i is a specialization of r_j based on Definition 7. Hence, r_j belongs to CRC. From (i) and (ii) it follows that CRC is a sequential classification rule cover of $\mathcal{R}$ according to Definition 10.

Figure 1 reports the classification rule cover for the example dataset, when rules are extracted by considering $minsup = 1$ and enforcing the contiguity constraint. We note that the sequential classification rule cover set does not allow the regeneration of the complete rule set.

4.3 Complete Compact Classification Rule Set

In this section we present a compact form to encode a classification rule set, which, differently from the classification rule cover presented in the previous section, allows the regeneration of the original rule set $\mathcal{R}$. Hence, it is named *complete*. The proposed representation relies on the notions of closed and generator sequences.

In the compact form, both general and specialistic rules are explicitly represented. All the remaining rules are summarized by means of an appropriate encoding. The compact form consists of a set of elements named *compact rules*. Each

Rule	Sup [%]	Conf [%]
$E \rightarrow c_1$	40	100
$F \rightarrow c_1$	40	100
$I \rightarrow c_1$	40	100
$J \rightarrow c_1$	40	100
$G \rightarrow c_1$	20	100
$H \rightarrow c_1$	20	100
$AB \rightarrow c_1$	20	100
$BA \rightarrow c_2$	20	100
$CA \rightarrow c_1$	20	100
$CB \rightarrow c_2$	20	100
$A \rightarrow c_1$	40	66
$B \rightarrow c_1$	20	50
$B \rightarrow c_2$	20	50
$C \rightarrow c_1$	20	50
$C \rightarrow c_2$	20	50
$D \rightarrow c_1$	20	50
$D \rightarrow c_2$	20	50
$A \rightarrow c_2$	20	33

Fig. 1. Sequential classification rule cover with contiguity constraint for the example dataset ($minsup = 1$)

compact rule includes a specialistic rule, a set of general rules, and encodes a set of rules that are specializations of them.

Definition 11 (Compact Rule). *Let M be an arbitrary closed sequence in $\mathcal{D}$, and $\mathcal{G}$ the set of its generator sequences. Let $c \in \mathcal{C}$ be an arbitrary class label. Then, $\mathcal{F} : (\mathcal{G}, M) \rightarrow c$ is a compact rule in $\mathcal{D}$.*

An arbitrary compact rule $\mathcal{F} : (\mathcal{G}, M) \rightarrow c$ represents all the rules $r : Y \rightarrow c$ in $\mathcal{D}$ with the following characteristics: r is labeled with the same class as $\mathcal{F}$, and M belongs to the sequential closure set of Y, i.e., $M \in \mathcal{CS}(Y)$. Hence, the rule set represented in $\mathcal{F}$ includes: (i) the rule $r : M \rightarrow c$, which is a specialistic rule since M is a closed sequence; (ii) the set of rules $r : G \rightarrow c, G \in \mathcal{G}$, that are general rules since G is a generator sequence; (iii) a set of rules $r : Y \rightarrow c$ that are a specialization of rules in (ii). For these rules, the antecedent Y is a subsequence of M (i.e., $Y \sqsubseteq_{\Psi} M$), and it completely includes at least one of the generator sequences in $\mathcal{G}$ (i.e., $\exists G \in \mathcal{G} | G \sqsubseteq_{\Psi} Y$).

Based on the selected matching functions in Ψ, a compact rule can represent a set of contiguous or non-contiguous sequential classification rules. In the example dataset, the contiguous classification rules $C \rightarrow c_1$, $D \rightarrow c_1$, $AD \rightarrow c_1$, $DC \rightarrow c_1$, and $ADC \rightarrow c_1$ are represented in the compact rule $(\{C, D\}, ADC) \rightarrow c_1$. Instead, the non-contiguous classification rules $E \rightarrow c_1$, $AE \rightarrow c_1$, $BE \rightarrow c_1$, and $ABE \rightarrow c_1$ are encoded in the compact rule $((\{E\}, ABE) \rightarrow c_1)$.

As stated in the next lemma, the rules represented in a compact rule are characterized by the same values of rule support and confidence. The lemma directly follows from Definition 11 and Theorem 1.

Lemma 3. *Let $\mathcal{F} : (\mathcal{G}, M) \rightarrow c$ be an arbitrary compact rule in $\mathcal{D}$. For each rule r represented in $\mathcal{F}$ is (i)* $\sup(r) = \sup(M \rightarrow c)$*, and (ii)* $\text{conf}(r) = \text{conf}(M \rightarrow c)$.

We use the concept of compact rule to encode the set $\mathcal{R}$ of frequent sequential classification rules. The next theorem proves that the compact rule set representing $\mathcal{R}$ is minimal and complete, since it represents all the rules in $\mathcal{R}$.

Theorem 3 (Compact classification rule set). *Let $\mathcal{R}$ be the set of frequent sequential classification rules in $\mathcal{D}$. Let $\mathcal{M}$ be the set of frequent closed sequences, and $\mathcal{G}$ the set of frequent generator sequences in $\mathcal{D}$. The compact rule set*

$$CCRS = \{\mathcal{F} : (\mathcal{G}, M) \rightarrow c\}, \tag{2}$$

is a minimal, complete representation of $\mathcal{R}$ iff $\forall r : X \rightarrow c$ in $\mathcal{R}$ such that $X \in \mathcal{M}$, then $\exists \mathcal{F} : (\mathcal{G}, M) \rightarrow c$ in CCRS with (i) $M = X$ and (ii) $\mathcal{G}$ includes all generator sequences for X.

The theorem above can be proved based on the characteristics of the closed and generator sequences. The rules in $\mathcal{R}$ having as antecedent either a generator or a closed sequence are explicitly represented in the set $CCRS$. Hence, the set $\mathcal{R}$ can be generated from the compact rules in $CCRS$. It follows that the set $CCRS$ is a complete representation of $\mathcal{R}$. Furthermore, let remove an arbitrary compact rule from $CCRS$. Hence, the rules encoded in the compact rule and having as antecedent either a generator or a closed sequence can not be generated from the set $CCRS$. It follows that the set $CCRS$ is a minimal representation of $\mathcal{R}$.

Figure 2 shows the compact classification rule set for the example dataset when enforcing the contiguity constraint. When $minsup = 1$, the sequential classification rule set includes 53 contiguous rules. The corresponding compact rule set includes 14 compact rules. Hence, the compression factor achieved in this case is 26.4%.

5 Experimental results

Preliminary experimental results have been run to evaluate the compression achievable by means of the proposed compact representations. Experiments have been run by considering the four datasets in Figure 3, where the number of items, sequences, and class labels for each dataset are reported. The Reuters-21578 news dataset [9] includes textual data. The other three are biological datasets: DNA and Promoters [9], including collections of DNA sequences, and the Escherichia Coli's protein sequences from RCSB Protein Data Bank [8].

We developed an algorithm to extract the compact classification rule set from a sequential dataset. The sequential classification rule cover representation can be easily derived from it. Currently, the algorithm focuses on the extraction of the compact forms with contiguity constraint. However, it can be easily extended to support the extraction of the compact forms without constraint. The algorithm is based on a levelwise search [1], and computes the set of frequent closed sequences in

Compact rule	Represented rules	Sup [%]	Conf [%]
$(\{G\}, G) \rightarrow c_1$	$G \rightarrow c_1$	40	100
$(\{F, G\}, FG) \rightarrow c_1$	$F \rightarrow c_1$, $G \rightarrow c_1$, $FG \rightarrow c_1$	40	100
$(\{F, J\}, FJ) \rightarrow c_1$	$F \rightarrow c_1$, $J \rightarrow c_1$, $FG \rightarrow c_1$	40	100
$(\{E, AB\}, ABE) \rightarrow c_1$	$E \rightarrow c_1$, $AB \rightarrow c_1$, $BE \rightarrow c_1$, $ABE \rightarrow c_1$	20	100
$(\{CA\}, ADCA) \rightarrow c_1$	$CA \rightarrow c_1$, $DCA \rightarrow c_1$, $ADCA \rightarrow c_1$	20	100
$(\{BA, CB\}, ADCBA) \rightarrow c_2$	$BA \rightarrow c_2$, $CB \rightarrow c_2$, $CBA \rightarrow c_2$, $DCB \rightarrow c_2$, $ADCB \rightarrow c_2$, $DCBA \rightarrow c_2$, $ADCBA \rightarrow c_2$	20	100
$(\{H\}, FGHFJ) \rightarrow c_1$	$H \rightarrow c_1$, $GH \rightarrow c_1$, $HF \rightarrow c_1$, $FGH \rightarrow c_1$, $GHF \rightarrow c_1$, $HFJ \rightarrow c_1$, $FGHF \rightarrow c_1$, $GHFJ \rightarrow c_1$, $FGHFJ \rightarrow c_1$	20	100
$(\{I\}, FGIFJ) \rightarrow c_1$	$I \rightarrow c_1$, $GI \rightarrow c_1$, $IF \rightarrow c_1$, $FGI \rightarrow c_1$, $GIF \rightarrow c_1$, $IFJ \rightarrow c_1$, $FGIF \rightarrow c_1$, $GIFJ \rightarrow c_1$, $FGIFJ \rightarrow c_1$	20	100
$(\{A\}, A) \rightarrow c_1$	$A \rightarrow c_1$	40	66
$(\{B\}, B) \rightarrow c_1$	$B \rightarrow c_1$	20	50
$(\{B\}, B) \rightarrow c_2$	$B \rightarrow c_2$	20	50
$(\{C, D\}, ADC) \rightarrow c_1$	$C \rightarrow c_1$, $D \rightarrow c_1$, $AD \rightarrow c_1$, $DC \rightarrow c_1$, $ADC \rightarrow c_1$	20	50
$(\{C, D\}, ADC) \rightarrow c_2$	$C \rightarrow c_2$, $D \rightarrow c_2$, $AD \rightarrow c_2$, $DC \rightarrow c_2$, $ADC \rightarrow c_2$	20	50
$(\{A\}, A) \rightarrow c_2$	$A \rightarrow c_2$	20	33

Fig. 2. Compact classification rule set with contiguity constraint for the example dataset ($minsup = 1$).

increasing length. At the k^{th} iteration, the algorithm generates the set of frequent closed sequences of length k. Each closed sequence is provided of the necessary

Dataset	Sequences #	Items #	Classes #
DNA	2000	4	3
Promoters	107	4	2
E. Coli	1186	20	8
Reuters-21578	6490	28982	10

Fig. 3. Datasets.

information to compute the compact classification rules encoded by it. The algorithm was coded in standard ANSI C. Experiments were run on an Intel Pentium 4, with 1.5GHz CPU clock rate and 1GByte RAM.

We performed rule extraction for decreasing support thresholds. For each dataset, in Figure 4 we report the number of rules in the frequent sequential classification rule set ($\mathcal{R}$), in the classification rule cover (CRC), and in the compact classification rule set ($CCRS$). Figure 4 also shows the compression rate ($\mathcal{CF}\%$) achieved by means of the two compact representations. This index measure the ratio between the number of rules in the compact form, and in the set $\mathcal{R}$.

Results show that the proposed compact representations yield significant benefits for low support thresholds. In this case, set $\mathcal{R}$ contains a large number of rules, while both compact forms have a significantly smaller size. For example, with support 0.05%, DNA dataset yields over 4 million rules, but only 110884 compact rules (with $\mathcal{CF}$ about 2.45%, i.e., about 100 times smaller) and 167455 general rules (with $\mathcal{CF}$ about 3.70%). When increasing support, the compact forms get close to the whole rule set $\mathcal{R}$.

Higher compression rates are achieved in the datasets where the information is more correlated. In these datasets, especially when considering low support thresholds, a set of subsequences can appear repeatedly in the training dataset. The two proposed compact representations allow modelling this regularity. Examples are the collections of DNA sequences (DNA and Promoters datasets), and textual data (Reuters dataset). A different behaviour characterizes the dataset representing proteins (E. Coli dataset), where the compression rate is lower. This effect is probably due to the fact that proteome contains less redundant information with respect to DNA.

We also performed preliminary experiments on classification accuracy by exploiting the compact forms proposed in this paper. We used a modified version of L^3 algorithm [5], which yielded encouraging accuracy results.

6 Conclusions and future work

In this paper we have introduced two compact representations to encode the knowledge available in a sequential classification rule set. The sequential classification rule cover is defined by means of the concept of generator sequence and yields a simple rule set, which is equivalent to the complete rule set. Compact rules are characterized by a more complex structure, based on closed sequences and their associated generator sequences. The complete compact rule set, while providing a similar compression ratio, allows us to regenerate the entire set of frequent sequential classification rules from the compact form.

Preliminary experiments on textual and biological datasets show that the compression ratio is significant for low support thresholds and correlated datasets. In this case, traditional techniques would generate a huge amount of classification rules.

As future work, we plan to exploit our compact representations to design an effective classifier. A promising direction is the integration of both sequential and associative classification rules, to exploit both the specific characterization provided by sequential rules and the general representation given by associative classification rules.

a) Reuters-21578

sup [%]	sup [abs]	$\mathcal{R}$	$CCRS$		CRC	
			#	$\mathcal{CF}$ [%]	#	$\mathcal{CF}$ [%]
0.05	2	530639	48336	9.11	58793	11.08
0.1	7	10307	9345	90.67	9400	91.20
0.5	33	1401	1401	100.00	1401	100.00
1.0	65	835	835	100.00	835	100.00

b) DNA

sup [%]	sup [abs]	$\mathcal{R}$	$CCRS$		CRC	
			#	$\mathcal{CF}$ [%]	#	$\mathcal{CF}$ [%]
0.05	1	4527168	110884	2.45	167455	3.70
0.10	2	416657	109044	26.17	117647	28.24
0.20	4	90551	65914	72.79	66666	73.62
0.50	10	27006	26754	99.07	26765	99.11
1.00	20	12966	12963	99.98	12963	99.98

c) Promoters

sup [%]	sup [abs]	$\mathcal{R}$	$CCRS$		CRC	
			#	$\mathcal{CF}$ [%]	#	$\mathcal{CF}$ [%]
1	2	147462	2910	1.97	4753	3.22
2	3	31345	2834	9.04	3369	10.75
4	5	2509	1884	75.09	1935	77.12
8	9	1013	985	97.24	991	97.83

d) E. Coli

sup [%]	sup [abs]	$\mathcal{R}$	$CCRS$		CRC	
			#	$\mathcal{CF}$ [%]	#	$\mathcal{CF}$ [%]
0.10	2	825204	359823	43.60	368645	44.67
0.20	4	261848	253243	96.71	253887	96.96
0.50	9	126813	126746	99.96	126748	99.95
1.00	17	67134	67128	99.99	67128	99.99
2.00	34	43100	43100	100.00	43100	100.00

Fig. 4. Frequent classification rule set ($\mathcal{R}$), sequential classification rule cover (CRC), and compact classification rule set ($CCRS$).

References

1. R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, pages 207–216. ACM Press, 1993.
2. R. Agrawal and R. Srikant. Fast algorithm for mining association rules. *In VLDB'94, Santiago, Chile*, September 1994.

3. R. Agrawal and R. Srikant. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering*, pages 3–14. IEEE Computer Society, 1995.
4. E. Baralis and S. Chiusano. Minimal non redundant classification rule sets. In *IEEE ICDM Workshop on Foundations of Data Mining and Discovery*. IEEE, 2002.
5. E. Baralis and P. Garza. A lazy approach to pruning classification rules. In *International Conference on Data Mining, Proceedings. 2002*.
6. Y. Bastide, R. Taouil, N. Pasquier, and L. Lakhal. Discovering frequent closed itemsets for association rules. *In ICDE'99*, January 1999.
7. R. J. Bayardo. Efficiently mining long patterns from databases. *In ACM SIGMOD'99*, 1999.
8. H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, H. Weissig, I. Shindyalov, and P. Bourne. The protein data bank. *Nucleic Acids Research*, 28:235–242, 2000.
9. C. Blake and C. Merz. UCI repository of machine learning databases, 1998.
10. B.Liu, W. Hsu, and Y. Ma. Integrating classification and association rule mining. *In KDD'98, New York, NY*, August 1998.
11. J.-F. Boulicaut, A. Bykowski, and C. Rigotti. Free-sets: A condensed representation of boolean data for the approximation of frequency queries. *In Data Mining and Knowledge Discovery journal, 7(1), pp. 5-22, Kluwer Academics Publishers*.
12. A. Bykowski and C. Rigotti. A condensed representation to find frequent patterns. *In PODS'01*, 2001.
13. T. Calders and B. Goethals. Mining all non-derivable frequent itemsets. *In Proc. of the 6th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'02), pp. 74-85, Springer*, 2002.
14. W. Li, J. Han, and J. Pei. CMAR: Accurate and efficient classification based on multiple class-association rules. *In ICDM'01, San Jose, CA*, November 2001.
15. N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Closed set based discovery of small covers for association rules. In *Proc. 15emes Journees Bases de Donnees Avancees, BDA*, pages 361–381, 1999.
16. N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient mining of association rules using closed itemsets lattice. *In Information Systems*, 1999.
17. N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Closed itemsets discovery of small covers for association rules. *In Networking and Information Systems*, June 2001.
18. N. Pasquier, Y. Bastide, R. Taouil, G. Stumme, and L. Lakhal. Mining minimal non-redundant association rules using frequent closed itemsets. *In DOOD'00*, 2000.
19. J. Pei, J. Han, and R. Mao. CLOSET: An efficient algorithm for mining frequent closed itemsets. *In SIGMOD Int'l Workshop on Data Mining and Knowledge Discovery*, May 2000.
20. J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu. PrefixSpan mining sequential patterns efficiently by prefix projected pattern growth, ICDE 2001,. pages 215–226.
21. J. Wang and J. Han. Bide: Efficient mining of frequent closed sequences. In *Proceedings of the 20th International Conference on Data Engineering*, page 79. IEEE Computer Society, 2004.

22. K. Wang, S. Zhou, and Y. He. Growing decision trees on support-less association rules. *In KDD'00, Boston, MA*, August 2000.
23. X. Yan, J. Han, and R. Afshar. Clospan: Mining closed sequential patterns in large datasets.
24. Q. Yang, T. Li, and K. Wang. Building association rule based sequential classifiers for web document prediction. *Journal of Data Mining and Knowledge Discovery*, 8(3):253–273, 2004.
25. M. Zaki. Generating non-redundant association rules. *In KDD'00*, 2000.
26. M. Zaki and C.-J. Hsiao. Charm: An efficient algorithm for closed itemset mining. *In SIAM'02*, 2002.
27. M. J. Zaki. Spade: An efficient algorithm for mining frequent sequences. *Mach. Learn.*, 42(1-2):31–60, 2001.

Extracting Rules From Support Vector Machine Classifiers

Xiuju Fu[*], *GihGuang Hung*[*]
Liping Goh[*] and Tsau Young Lin [†]
[*]Institute of High Performance Computing
[*]Singapore 117528
[†] San Jose State University
Email: fuxj@ihpc.a-star.edu.sg, tylin@cs.sjsu.edu

In recent years, support vector machine (SVM) [2][3] has attracted lots of interest for its capability in solving classification and regression problems. Successful applications of SVM have been reported in various areas, including but not limited to areas in communication [4], time series prediction [5], and bioinformatics [1]. In many applications, it is desirable to know not only the classification decisions but also what leads to the decisions. However, SVMs offer little insight into the reasons why SVM has made its final results. It is desirable to develop a rule extraction algorithm to reveal knowledge embedded in trained SVMs and represent the classification decisions based on SVM classification results by linguistic rules.

Rule extraction from SVM can facilitate data mining clients in many aspects:

- Increase perceptibility from SVM decisions
- Refine initial domain knowledge, for example, remove irrelevant attributes which do not play a role in rule decision making
- Explain data concepts by linguistic rules to clients
- Find active attributes in decision making

This paper exploits the fact that the decisions from a non-linear SVM classifier could be decoded into linguistic rules based on the information provided by support vectors and decision function. Given a support vector of a certain class, cross points between each line, which is extended from the support vector along each axis, and SVM decision hyper-curve are searched first. A hyper-rectangular rule is derived from these cross points. The hyper-rectangle is tuned by a tuning phase in order to exclude those out-class data points. Finally, redundant rules are merged to produce a compact rule set. Simultaneously, important attributes could be highlighted in the extracted rules. Rule extraction results from our proposed method could follow SVM classifier decisions very well. Comparisons between our method and other rule extraction methods are also carried out on several benchmark data

sets. Higher rule accuracy is obtained in our method with fewer number of premises in each rule.

References

[1] M. P. S. Brown, W. N. Grundy, D. Lin, N. Critianini, C. Sungnet, T. S. Furey, M. Ares, D. Haussler (2000), "Knowledge-based Analysis of Microarray Gene Expression Data Using Support Vector Machines", *Proceedings of National Academy of Sciences*, Vol. 97, No. 1, pp. 262-267.

[2] C. J. C. Burges (1996), "Simplified Support Vector Decision Rules", *13th International Conference on Machine Learning*.

[3] C. J. C. Burges (1998), "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, 2(2),pp. 955-974.

[4] X. H. Gong, A. Kuh (1999), "Support vector machine for multiuser detection in CDMA communications", *Conference Record of the Thirty-Third Asilomar Conference on Signals, Systems, and Computers*, Vol. 1, pp. 680-684.

[5] F. Girosi, S. Mukherjee, E. Osuna (1997), "Nonlinear prediction of chaotic time series using support vector machines", *Proceedings of the 1997 IEEE Workshop on Neural Networks for Signal Processing*, pp. 511 -520.

Towards a Methodology for Data mining Project Development : The Importance of Abstraction

P. González-Aranda[(1)], E. Menasalvas[(1)], S. Millán[(2)], F. Segovia[(1)*]
[(1)] School of Computer Science, Technical University of Madrid, Madrid, Spain.
{pgonzalez, emenasalvas, fsegovia}@fi.upm.es
[(2)] Universidad del Valle, Cali , Colombia.
millan@eisc.univalle.edu.co

Abstract

Despite the existence of data mining standards such as Crisp-DM, SEMMA, PMML, up to date, data mining projects are being developed more as an art than as a science. The process depends completely on the expertise of the data miner since no method is available to make the process systematic and automatic. This is due to a lack of data mining problem conceptualization. In this sense, a deep understanding of both of the data to be analyzed and the application domain of the results as well as of the data mining functions is needed. Knowing the meaning of the data to be analyzed: facts they represent, constraints and context under which they were captured and the constrains underneath the data mining functions to be applied, will make it possible to find out whether the business goals to achieve are feasible. However, up to date, there is no formal method to describe this elements in such a way that the quality of results quality can be assure. In this paper, we present the basis for an abstract model to conceptualize these elements. This setting is a step towards a methodology for data mining project development that will be in itself the main basis for automatizing the process.

1 Introduction

Data is the key element of every data mining project. Data must represent the part of the real world domain that it is going to be analyzed. Moreover, data must be understood so that its correctness and adequacy to the problem to be solved can be evaluated. The process of knowledge creation and enhancement comes from information which is nothing else than data that have been collected, accessed, formatted and analyzed [21]. Consequently, the success of a data mining project needs to be secured far before the modelling step. More precisely, success depends firstly on the Business Understanding, Data Understanding and Data Preprocessing steps which are currently being developed as an art.

According to [21], one of the essential elements of effective mining is the availability of domain relevant data: " your analysis is only as good as the data you use . The author also establishes, among the common pitfalls of data mining implementation, the following:

- Not being able to efficiently communicate mining results within an organization.
- Not having the right data to conduct effective analysis.
- Not using existing data correctly.

The question that arises is whether the adequateness of a set of data for a problem can be established when preparing the project plan and how this set of data can be used to produce the expected results. Up to date, there is no formal methodology to help with this task. In order to do so, a conceptualization of the data mining problem together with the data will be needed. This conceptual model will facilitate communication among the human resources involved: the data analyst, the data engineer, the domain expert and the data miner analytical personnel [18].

Such conceptualization would be the key to establishing which business objectives have a chance to be achieved and under which circumstances, and it would be a first step towards the automatization of the data mining process.

Only the experience of expert data miners can help in solving this task. In order to automatize the process, the set of factors that the expert takes into account when deciding which business objectives are feasible or not it would have to be analyzed. Going deeper into the process it is clear that the adequateness of the data is analyzed taking into account goals to fulfil. Finally, this can be translated into analyzing whether the data, together with the knowledge extracted

*This work has been partially supported by Ministerio de Educacion y Ciencia (Spain)under project TIN2004-05873

from the experts, can be transformed so that just by being the input of a certain data mining algorithm will produce the required patterns.
Thus, quality of the data, in this context, is not only related to the technical quality, let us say, proper model, percentage of null values, ... but it also has to do with the meaning of the attributes, precedence of each piece of data, relationship among data, and finally how the data fulfil the requirements of the data mining functions.
In this paper, we present a first approach towards a systematic way to develop data mining project by means of the conceptualization of each factor involved in the proper development: standard representation of goals to fulfil, techniques to be used, and any information to be analyzed before the project plan is developed. Thus, independently of who the person to develop the problem may be, the tasks to be performed together with the inputs, outputs, and risks will be settled in a standard way.
The rest of the paper has been organized as follows. Section 2 presents the related work in which it will be clear that although some efforts have been made towards a data mining methodology, no such methodology already exists mainly because the conceptualization of the problem is missing. In section 3 a deep analysis of problem is done so to discover the elements to conceptualize: business domain on the one hand, data mining functions on the other. Section 4.2 presents a deeper analysis of the data sources as they are the main source of information in a data mining problem. To end with in section 5 discussion and conclusions of the research so far as well as future lines of study are outlined.

2 Related Work

In the information age when data generated and stored by modern organizations increase in an extraordinary way, data mining tasks [35] become a necessary and fundamental technology. A lot of data mining research has been focusing on the development of algorithms for performing different tasks, i.e. clustering, association and classification. [27] [36] [22] [34] [29] [30] [17] [4] [25] [5] [24] [14], and on their applications to diverse domains. Though one major challenge in data mining according to [11]is getting researchers to agree on a common standard for preprocessing tasks, standards related to applying the data mining process to operational processes, and systems. In this sense, the Predictive Model Markup Language(PMML) [12] provides several components (Data Dictionary, Mining Schema, Transformation Dictionary, Models) useful for producing data mining models. The Data Dictionary includes only information about type of data and range of values. Semantic information is not taken into account.
Several proposals have been developed in order to offer a guide to implementing data mining projects [13][32][7]. The Common Warehouse Model for Data Mining (CWM DM) [13] proposed by the Object Management Group, introduces a CWM Data Mining metamodel integrated by the following conceptual areas: A core Mining metamodel and metamodels representing the data mining subdomains of Clustering, Association Rules, Supervised, Classification, Approximation, and Attribute Importance.
The Cross-Industry Standard Process for Data Mining (CRISP-DM), was proposed in 1997 [7] to establish the standard data mining process. CRISP-DM steps include several phases: business understanding, data understanding, data preparation, modelling , evaluation and deployment. AT 1999 SAS Institute proposed the SEMMA methodology integrated by five phases: Sample, Explore, Modify, Model and Assess. The data mining process starts by taking a representative sample of the target population to which a confidence level is associated. Then, this sample is explored and analyzed using visualization and statistical tools in order to obtain a set of significant variables that will become the input for a selected model.The selected model is analyzed. The goal of this step is to determine relationships among variables. In this phase, both statistical methods (e.g. discriminant analysis, clustering, and regression analysis)and data-oriented methods (e.g. neural networks, decision trees, association rules) can be used. In this process the final phase consists of evaluating the model and comparing it with different statistical methods and samples.
All of the above models depend heavily on the analysts (business, domain experts, data miners) knowledge. There seems to exist a need for an intermediate level of conceptualization which can provide an interface between the experts and the clients.
According to Grossman et al. [11] " although efforts have been done to homogenize terminology and concepts among standards more work is required ". A framework to develop a unified model for data mining is proposed in [19]. The goal of the model is to provide a uniform data structure for all data mining patterns and operators to manipulate them. The model is designed under a three-view architecture (Process view, model view and data view) that includes a process model and data views. The model view contains a set of mining models with information about mining results. All these approaches and standards do not take the semantics of the data into account.
On the other hand, new techniques to add semantics to data mining tasks have been proposed. However, the semantics of the data in the data mining process has been strongly related to human involvement in the process itself. Data mining projects involve qualified personnel [18], i.e. business analysts, data analysts, data engineers, domain experts, data miners, knowledge engineers, strategy managers, project managers [20]. A significant contribution to semantic data

has been done by the semantic web community [23][2] [8][31] to enrich web resources with metadata. In the web domain, both semantics and mining are combined [2] to improve web mining results with ontologies and metadata. In order to add semantics to web documents, Berendt et al. [2] proposed several approaches to extract semantics from the web to help knowledge engineers. Several studies have been developed for using ontologies to improve web content mining. An interesting proposal, called M^4, dealing with a metadata structure is presented in [1]. Based on case-based reasoning, an approach that enables automatization of preprocessing and reusability of defined preprocessing cases for data mining applications, is proposed. In this proposal, a case is defined in terms of the specification of a data mining task, the data to be mined and the set of preprocessing operators to be applied to the data. M^4, the Mining Mart MetaModel, is a metamodel designed for a metadata-driven software package to perform preprocessing for data mining.

3 Analysis of the Problem

Business Intelligence [6] is: " a fairly new term that incorporates a broad variety of processes and technologies to harvest and analyze specific information to help a business make sound decisions ".

In this paper we use the term business to refer to any activity developed in a company in the most general sense, no matter the nature and aim of such activity (commercial, governmental, education, ...). Data mining is one of the technologies that make Business Intelligence solutions be implemented. In fact, in any business intelligence solution should include a data mining project to extract the intelligence of the business that will be accordingly deployed. However, data mining projects are being developed more as an art than as an engineering process.

The only approach to develop a task in a systematic way is described in Crisp-DM. However, complete tasks development is dependant on the data miner expert.

Data mining experts have made the process of translating business goals into data mining goals, automatic. When doing so, the expert does not only take into account data mining techniques but also their constraints, inputs, outputs, the order in which algorithms will be applied and dependencies between inputs and outputs. As part of the process, the expert automatically evaluates different choices depending on the intermediate results and/or inputs. The quality of the overall process will finally determine the quality of the obtained results.

The first question that arises is: Which is the methodology to be followed to translate business objectives into data mining objectives? Unluckily, there is no such methodology but if we think on how to obtain it new questions will arise: How a business objective is expressed? Do we have any standard to express business objectives in a uniform way? What is a data mining goal? How are data mining goals achieved? Which are the requirements of data mining functions? Do we have a standard to establish data mining goals?.

The main goal is then, to make this process explicit: to generate a method to perform the required tasks in a systematic way. This method will guarantee the automatic generation of feasibility plans for each business goal being translated into data mining goals no matter who the person in charge of the process may be. A first step towards this method will be the definition of certain mechanisms of abstraction to obtain a model of the objectives of the project.

The goal of this abstraction is to provide data mining analyst with a method to systematically describe the goals of the project. Deeply analyzing any activity of the organization (even external to it) that generate data that will be potentially used as input in a data mining project as well as the data themselves and the data mining functions will highlight important concepts that are common in any data mining project independently of the domain. These aspects will set the basis for a definition of elements that will make it possible to represent (to abstract) the business domain that is the target of the data mining solution.

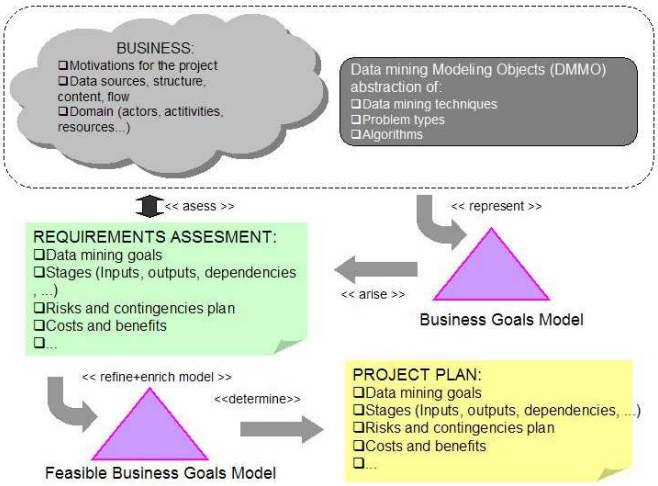


Figure 1. Project Plan Definition Process

Figure 1 depicts the basic steps, tools and intermediate results that underlay the establishment of a systematic method to define data mining goals.

Previous to the definition of such a model there is a need to find a standard way to represent all the elements identified as relevant in the business domain to be analyzed. Mandatory elements that compose this information are: objectives

and motivations underlying the project, scope of application of the expected results and structure, content and flow of the data to be analyzed. Besides, technical elements related to the very nature of the data mining project will have to be incorporated to the previous information. The blending and abstraction of these two pieces of information, will result not only in a model of shared understanding for client and data miner but will also be a tool to determine factible data mining goals and consequently the basis to establish milestones to achieve along the project development. In figure 1, DMMO (Data Mining Modelling Objects) denotes the set of all the compounding elements of the Modelling Language. Elements of the business domain will be abstracted using DMMO, generating the Business Objective Model. Together with this intermediate model, a document that we have called Requirements Assesment will be produced. This is a first approach to project goals in which special attention is given to critical factors in the development (risks, constraints, information required, ...). Critical factors will depend both on the goals themselves and on the tools and techniques used to achieve them.
The requirement assessment document will be analyzed jointly with the client to enrich and refine the previous business goals model. The resulting model from this analysis is what has been called Feasible Business goals Model in the figure. In this model, goals previously identified but analyzed as not feasible have been removed. From the refined model and making used once again of DMMO the project plan will be produced almost in a automatic way as the model will represent relevant aspects both of the domain and of the tools themselves. Due to the abstraction capability of the elements of DMMO we propose, the plan will contain detail information about: techniques, tools, kind of data mining to be solved, inputs, outputs, flor of data and dependencies. Thus, risk and contingencies, cost, milestones, ... will be identified.

3.1 Setting/Abstracting business objectives

A data mining project arises when a given organization needs to solve a set of problems that can be addressed by means of data mining techniques. In a data mining project some critical success factors can be identified. However, the most important factor is related to the clear understanding of the business goals. Moreover, once the goals are understood, they must be translated into data mining goals and then, into data mining problem types. A data mining expert who knows what type of problems can be solved, and which are the most suitable techniques, algorithms and tools to be applied, is required.
However, not only identifying the data mining problems to be solved is enough. We should also be able to find out if the available data to be analyzed fulfil a set of general requirements or conditions. As it was said in 4.1, every problem type will require different kinds of data. In the following, we will describe the different existing problems as well as their requirements.
Eliciting, analyzing and graphically depicting concepts is no easy task [10]. The bottom line to business success is to increase the knowledge of decision makers at every level of an organization. The process of knowledge creation and enhancement comes from information which is nothing else than data that have been collected, accessed, formatted and analyzed [21].
For data mining to be successful, a good understanding of the business objectives that finally establish the requirements of the system to be developed is needed. Independently of how good the data mining techniques can be, a system whose requirements are poorly specified will end up with a disappointed end user [28]. Both the client and the data miner play an important role when establishing business goals. The client has to formulate his problem while the data miner tries to understand it in order to be able to translate it into data mining functions. During this task, it is relevant to keep in mind the following aspects (inspired from software engineering [28]: " Business domain as well as functional domain of the problem has to be represented and understood ".

3.1.1 Business domain

Every data mining project can collectively be described as data analysis and knowledge extraction to obtain the intelligence of the business. This definition contains the key to understand business objectives in a data mining project: Data. Data do not only represent the activities or business processes that have generated them. Activities are influenced by relevant factors of the business that are also hidden in the data themselves. Consequently, data implicitly carry important knowledge about the business that should be extracted to be able to correctly capture the information hidden in data. Along this section, we will try to highlight which are these determinant factors so to take them into account in the abstraction mechanism.
Though talking about intelligence, data mining does not involve deductive processes. On the contrary, it is an inductive process that analyzes the data to extract knowledge: it accepts data from different sources, manipulates them and obtains an output and patterns of knowledge, that if of good quality, will be deployed. This is the general setting of the process no matter the domain or organization we are dealing with.
In the process of data analysis and knowledge extraction, different perspectives of the data being analyzed are taken into account: data sources, information content of the data (knowledge to explain the data), data structure and data

flow. To fully understand the process, all of them must be considered.
Related to the content, data have to be enriched so that goals can be fulfilled. Data that are the source of a data mining project were never designed, captured and stored thinking they would become the input of the data mining process. Consequently, an effort is needed to transform them so that knowledge can be extracted. However, the aim of any data mining project is to help the decision maker do a better job. Thus, any element that can be determinant when making a decision should be analyzed. In this sense, the mayor problem is establishing these elements as it is as equally fatal to leave one element out as it is to introduce erroneous elements. In any case, elements that can be decisive when making decisions have to do with the operations developed in the company, the internal organization of the company as well as business rules, and finally the external conditions related to the business (competitors) and general (political, social, . . . ,) events. In order for the process to be data miner and client independent, this is to say, to be able to obtain the same goals no matter who the experts leading and developing the process are, a systematic abstract way to express this content information is needed. The first naïve approach is to conceptualize this information to discover the concepts and properties the available data represent with respect to the business. Only this way, we should be able to establish if data comply with the requirements of each function within the data mining process. Data content is deeply dealt with in section 4.

3.2 Setting/Abstracting the data mining functions domain

Lots of classifications of data mining problems can be found in the literature. In [7] authors describe six kinds of problems: data description and summarization, segmentation, concept description, classification, prediction and dependency analysis. Usually the data mining project involves different problem types that together will achieve the goals of the project. In [16] and [3] the various types of data mining algorithms such as memory-based reasoning, link analysis, decision trees, neural networks, . . . are explained. Data mining common tasks are identified: classification, estimation, prediction, affinity grouping, clustering and description. Moreover, they explain which data mining techniques are more appropriate for every type of problem.
Although in [3] it is stated which data-mining techniques are best for what types of business applications, it starts with data mining objectives already identified. A further description of the problem, so that a mapping could be done between business objectives and data mining problems, is missing. This mapping will help on the one hand to see if certain business objectives are feasible or not, and, on the other, it would provide means to interpret the patterns to be obtained.
In order to do a mapping between business goals and data mining goals, a conceptualization of the data mining functions definition needs to be established for each kind of problem: constraints, required inputs for each input, relationship with the business, expected outputs, techniques that are appropriate, ... This conceptualization could be used in the business understanding step to establish requirements to fulfil certain goals.
Related to data mining conceptualization, not only each function has to be dealt with. Data are transformed along the data mining process in order to obtain the proper results. Settled this way, the input data will be transformed generating intermediate data that will sometimes become final results and after more transformations, will produce the output. Along the process, more data can enter from internal or external sources. Transformations applied over the data will define the process to be developed.
The data going through the set of transforming steps would have to comply with the requirements of each transforming function. These functions requirements can be divided into two categories depending on its relation to data semantics. Requirements that are data semantics independent are the ones related to the function itself regardless of the domain. Thus, certain algorithms require data to have a special format or auxiliary structures to run (hierarchies, ...).
Besides, there are requirements made by the function to the input data depending on the semantics of the data. Consider, for example, a clustering function. If demographic clusters are required with such an such support, then the data to enter will have to be related to demographics. This is why we say that requirements will be related to data structure as well as to content and both are important when establishing the goals together with the data flow.
On the other hand, data mining patterns cannot be interpreted depending only on the function or/and technique used to obtain it. Thus after applying any function, lets take clustering as an example, a set of patterns is obtained but to evaluate their quality and consequently the success of the process, not only measures related to the patterns, clusters in this case, (number of elements, cohesion, . . .) are needed but also some values to measure the results according to user expectations. The latter are a mixture of understanding the meaning of each pattern, cluster, together with the business requirements.
The conceptualization of the data mining problems will also provide a basis for understanding the meaning of the obtained patterns, analyzing the features of the instance of the problem that has been performed.
However, data mining problems cannot be analyzed to abstract common features on their own. Data mining problems impose certain requirements to the input data. These

requirements (content, structural, ...) have to be complied with by the input data to obtain the appropriate result. Consequently, there is a need before deeply analyzing data mining problems to further analyze data from different perspectives, including technical, structure and content.

4 A first approach to Data Conceptualization

In [26] the key to a data mining successful project is outlined: think about the data that you need to gather from the perspective of the information you want to deliver.
Discovering behaviors, patterns or trends is only possible if we have data about the domain we want to analyze. However, having data does not mean that the discovery process is going to be successful. Data must fulfil a set of critical requirements in such a way that by analyzing them a particular problem can be solved.
In this context, good quality data means adequateness of the data to fulfil a goal. Thus Adequateness can only be analyzed studying the requirements that goal fulfilment impose on data.

4.1 Critical Requirements of Data

There are some critical requirements related to data that will lead a data mining project to be successful.

Quality Data mining is the process of analyzing a huge amount of data intended to find useful information for decision making. However, if there is no useful information hidden in the data, it will obviously be impossible to obtain interesting results. As in [7], it is possible to figure it out at the very beginning of the project. During the Data Understanding phase it is possible to guess first findings or initial hypothesis and their impact on the remainder of the project. Besides, the analyst should examine some aspects of the data that may have altered the results of the analysis or could have even made achieving the goals of the project impossible. Some common aspects to check include [33]: missing or null values; whether all possible values are represented; the plausibility and the spelling of values; attributes providing the same information but in different formats.

Interpretation The interpretation of the findings extracted from the data depends on how well we understand the data. This understanding is at the same time related to the context or environment as well as to the domain they stand for. Data interpretation is closely related to their semantics. Data by themselves do not mean much. Making use of meta data information will facilitate data understanding and would make the interpretation of the results of a data mining project a lot easier.

Usefulness Data analysis experts should be familiar with the types of problems they are able to solve as well as with the algorithms, techniques and tools to be used. Each problem type requires data of a particular nature. It calls for a team task between a data engineer and a data analyst [18] to identify the types of data required for every data mining problem to be solved for fulfilling a project goals. For instance, if a fraud detection model is to be obtained (a classification model), transactional information as to where frauds occurred as well as information related to the people involved, will then be needed. Consequently, the potentiality of the data for each problem type should be analyzed. It may happen that a particular attribute considered essential in some cases it might just be considered obvious in others.

Hence, when facing a data mining project apart from identifying goals, types of problems to be solved and techniques to be applied, understanding the available data will be needed to measure their quality, their degree of interpretability and their utility.

4.2 Towards Data typification

It is necessary to make use of meta data information about the data to be analyzed wherever we can. Meta data should include information not only about the source of the data but also about the concepts they stand for.
As in [26], there are three types of data depending on the source which should be used in every data mining project. Generally, we can find data sources within and outside a business organization.

- Transactional data. This is very relevant in a data mining project because, as in the case of a business organization, the data contain information about the activities the company is involved on. Typically, internal data is considered more valuable data, because they reveal true insights into the business and its products [9]. Therefore, they will represent the customer's past behaviour. And, as it is well known, analyzing past behavior is the best way to predict future ones.

- Collected data. Transactional data about the activities performed by the business organization is often spread all over the different databases of the company. Collecting these data is and effort that is worth trying. It increases the possibility of enriching transactional data with more information that will, for sure, improve the quality of the analysis to be carried on.

Both transactional and collected data can be considered internal data, as they are coming from the internal databases of the company.

- External data. Typically, the internal data of the company stand for just a subset of the total amount of information that could be useful for analysis. Therefore, these data must often be enriched or complemented with external data sources such as surveys, panels, micro-marketing tools, ...

Taking into account the aspects of the domain to be analyzed and whose data we are talking about we distinguish three different types of data.[15].

- Content data. It consists of data related to individual events, for instance, interactions with the users. It is fact oriented since it records the details or facts of customer encounters. It reflects an activity that has occurred.

- Contextual data. These type of data refers to the conditions or environment under which whatever individual event occurs. It provides one additional level of information complementary to content data, giving a more comprehensive view of those factors that could have influenced the customers behavior. This kind of information is changing across time so it is important to record not only present conditions but those that happened in the past. In the case of a company, this set of data could include the context of the company (suppliers, competitors, marketing campaigns, ...), context of the customers (demographics, economics, psychology, ...) and general context (politics, laws, economics, market, ...).

- Analytical data. The integration of data coming from individual events and from the context will be the input for analytical processes. The analysis will evaluate the relationship between the occurred events under different circumstances identifying patterns, trends and behaviors. The results of this analysis will be part of analytical information to be incorporated to every intelligence process within a company, for instance.

The above described classifications are not independent of each other. On the contrary, they are complementary and different since the criteria to classify the data is also different. The effective integration of these data types will lead the data mining project to be successful.
The principles towards the first steps to data abstraction lay underneath these typifications. This is just a first approach to data abstraction but the important point to be highlighted is that any of the classifications presented above do include information about the gathered data related not only to the data themselves but to the organization and/or activities generating them. Enriching the data this way provide analysts with a tool to establish whether relevant data for data mining functions are available so that feasible data mining goals can be stated and consequently, so can be business objectives.

5 Discussion and Conclusion

The main reason for data mining to be developed more as and art than as a science can be found in the lack of an abstract description of the elements involved: data and the domain they represent on the one hand and data mining functions on the other. Such a conceptualization of the problem would make it possible to automatize or at least to help developers to decide about the feasibility of the goals to be achieved. In this paper, we have presented a first approach to such abstraction. Our approach is towards an abstract description of the data involved, domain independent and goals oriented. Goal oriented means that the abstraction main aim is to help analyzing goals that will be feasible. For this purpose the abstraction will have to collect information related to the main factors in the process: the business goals themselves on the one hand, and the data mining functions on the other.
Thus, data abstraction has to gather all relevant information that would be important for the business goals to be achieved, this is to say, not only the data itself related to the activity generating it; factors involved, relationship with other activities, external factors influencing the values, ..., but also capturing and abstracting information related to content, transaction and context. On the other hand, data mining functions abstraction has to include not only the information related to the function itself: technique, kind of patterns it generates, but most importantly, requirements and constraints the data has to comply in order to generate the proper set of patterns. Both abstractions will provide the analyst with enough information to study the adequateness of the data for a given business problem and at the end, the feasibility of each goal. Not only feasibility will be established and consequently the project plan, but the model will help preparing the risk plan since the set of requirements for each task would be analyzed.
The paper has presented a deep analysis of the approach and a first data typification has been presented. We are currently working on the data mining function global abstraction. Once the concepts to abstract will be clear, our next goal for the data mining model to be obtained is the representation of the elements in a standard way.

6 Acknowledgments

The research has been partially supported by Ministerio de Educacion y Ciencia (project TIN2004-05873) and by

Universidad Politécnica de Madrid under Project WEB-RT.

References

[1] V. A., K. J., and Z. R. M4 -a metanodel for data preprocessing. 2001.

[2] B. Berendt, A. Hotho, and G. Stumme. Towards semantic web mining. *ISWC 2002, LNCS 2342*, pages 264–278, 2002.

[3] M. Berry and G. Linoff. *Data Mining Techniques for Marketing, Sales and Customer Support*. John Wiley and Sons, 1998.

[4] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur. Dynamic item set counting and implication rules for market basket data. 1997.

[5] A. C. and W. S. Data mining with decision trees and decision rules. 1997.

[6] Commerce-Database.com. Business intelligence definition.

[7] CRISP-DM Consortium:. *CRISP-DM 1.0. Step-by-step data mining guide*, 1.0 edition, August 2000.

[8] T. B. Despeyroux T. Web sites and semantics. 2001.

[9] W. G. Dirk Arndt. Data management in analytical customer relationship management. 2000.

[10] K. M. Graw and K. Harbison-Briggs. *Knowledge acquisition: Principles and guidelines*. Mc-Graw-Hill, 1986.

[11] M. G. Grossman R., Hornick M. Data mining standards initiatives. 2002.

[12] D. M. Group. The predictive model markup language pmml. http:// www.dmg.org.

[13] O. M. Group. Common warehouse metamodel - data mining, March 2002.

[14] S. K. Guha S., Rastogi R. Cure: An efficient clustering algorithm for large databases. 1998.

[15] J. Hall. Business intelligence: The missing link in your crm strategy. *DM Review Magazine, June 2004 Issue*, 2004.

[16] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*, volume 550 pages. Morgan Kaufmann Publishers, August 2000.

[17] J. Han, J. Pei, , and Y. Yin. Mining frequent patterns without candidate generation. 2000.

[18] M. Hofmann and B. Tierney. The involvement of human resources in large scale data mining projects. In *Proceedings of the 1st international symposium on Information and communication technologies*, pages 103–109. Trinity College Dublin, 2003.

[19] G. I. Framework for data mining and kdd. 2002.

[20] H. Karim. Exploring data mining implementation. 2001.

[21] S. Kudyba. Data mining efforts increase business productivity and efficiency. *Interview with Stephan Kudyba - President of Null Sigma Inc.*, 2001.

[22] T. Y. Lin and E. Louie. Data mining using granular computing: fast algorithms for finding association rules. pages 23–45, 2002.

[23] H. M., N. B., V. I., and V. M. Thesus:organizing web document collections based on link semantics. 2003.

[24] K. M., W. L., G. W., C. S., and H. J. Generalization and decision tree induction: Efficient classification in data mining. 1997.

[25] M. M., A. R., and R. J. Sliq: A fast scalable classifier for data mining. 1996.

[26] J. Noonan. Data mining strategies. *DM Review Magazine, July 2000 Issue*, 2000.

[27] Z. Pawlak. Information systems: theoretical foundations. *Information Systems*, 6(3):205–218, 1981.

[28] R. Pressman. *Software engineering: A practioner's approach*. Mc-Graw-Hill, 1997.

[29] A. R., M. H., S. R., T. H., and V. A. Fast discovery of association rules. 1996.

[30] S. R. and A. R. Fast algorithms for mining association rules. 1994.

[31] D. S., E. N., G. D., G. D., G. R., J. A., K. T., R. S., T. A., T. J., and Z. J. Semtag and seeker: Bootstrapping the semantic web via automated semantic annotation. 2003.

[32] SAS. Semma – sample, explore, modify, model, assess.

[33] C. Shearer. The crisp-dm model: The new blueprint for data mining. *Journal of Data Warehousing. Volume 5 Number 4 Fall 2000. Pages 13-22*, 2000.

[34] D. Slezak, J. Wroblewski, and M. S. Szczuka. Constructing extensions of bayesian classifiers with use of normalizing neural networks. In *Foundations of Intelligent Systems, 14th International Symposium,ISMIS 2003, Maebashi City, Japan, October 28-31, 2003, Proceedings*, volume 2871 of *Lecture Notes in Computer Science*, pages 408–416. Springer, 2003.

[35] F. U., P.-S. G, and S. P. From data mining to knowledge discovery: An overview. 1996.

[36] W. Ziarko. Variable precision rough set model. *J. Comput. Syst. Sci.*, 46(1):39–59, 1993.

Three Approaches to Missing Attribute Values—A Rough Set Perspective

Jerzy W. Grzymala-Busse
Department of Electrical Engineering and Computer Science
University of Kansas, Lawrence, KS 66045, USA
Jerzy@ku.edu
and
Institute of Computer Science
Polish Academy of Sciences, 01-237 Warsaw, Poland

Abstract.

A new approach to missing attribute values, based on the idea of an attribute-concept value, is studied in the paper. This approach, together with two other approaches to missing attribute values, based on "do not care" conditions and lost values are discussed using rough set methodology, including attribute-value pair blocks, characteristic sets, and characteristic relations. Characteristic sets are generalization of elementary sets while characteristic relations are generalization of the indiscernibility relation. Additionally, three definitions of lower and upper approximations are discussed and used for induction of certain and possible rules.

1. Introduction

In this paper data sets are presented in the form of decision tables, where columns are labeled by variables and rows by case (or example) names. Variables are categorized into independent variables, also called attributes, and dependent variables, also called decisions. Usually decision tables have only one decision. The set of all cases that correspond to the same decision value is called a concept (or a class).

In most papers on rough set theory it is assumed that values, for all variables and all cases, are specified. For such tables the indiscernibility relation, one of the most fundamental ideas of rough set theory, describes cases that can be distinguished from each other.

However, in many real-life applications, data sets have missing attribute values, or, in different words, the corresponding decision tables are incompletely specified. For simplicity, incompletely specified decision tables will be called incomplete decision tables.

In data mining two main strategies are used to deal with missing attribute values. The former strategy is based on conversion of incomplete data sets (i.e., data sets with missing attribute values) into complete data sets and then acquiring knowledge, e.g., by rule induction or tree generation from complete data sets. In this strategy conversion of incomplete data sets to complete data sets is a preprocessing to the main process of data mining. In the later strategy, knowledge is acquired from incomplete data sets taking into account that some attribute values are missing. The original data sets are not converted into complete data sets.

Typical examples of the former strategy include [4, 11]:

- replacing missing attribute values by the most common (most frequent) value of the attribute,
- replacing missing attribute values restricted to the concept. For each concept missing attribute values are replaced by the most common attribute value restricted to that concept,
- for numerical attributes, missing attribute value may be replaced by the attribute average value,
- for numerical attributes, missing attribute value may be replaced by the attribute average value restricted to the concept,
- assigning all possible values of the attribute. A case with a missing attribute value is replaced by a set of new examples, in which the missing attribute value is replaced by all possible values of the attribute,
- assigning all possible values of the attribute restricted to the concept,
- ignoring cases with missing attribute values. An original data set, with missing attribute values, is replaced by a new data set with deleted cases containing missing attribute values,
- considering missing attribute values as special values.

The later strategy is exemplified by the C4.5 approach to missing attribute values [18] or by a modified LEM2 algorithm [10, 13]. In both algorithms original data sets with missing attribute values are not preprocessed, i.e., data sets are not preliminarily converted into complete data sets.

Note that from the view point of rough set theory, in the former strategy the conventional indiscernibility relation may be applied to describe the process of data mining since, after preprocessing, the data set is complete (has no missing attribute values). Furthermore, lower and upper approximations, other basic ideas of rough set theory, are also conventional.

In this paper we will concentrate on the later strategy used for rule induction, i.e., we will assume that the rule sets are induced from the original data sets, with missing attribute values, not preprocessed as in the former strategy.

We will assume that there are three reasons for decision tables to be incomplete. The first reason is that an attribute value, for a specific case, is lost. For example, originally the attribute value was known, however, due to a variety of reasons, currently the value is not available. Maybe it was recorded but later it was erased. The second possibility is that an attribute value was not relevant—the case was decided to be a member of some concept, i.e., was classified, or diagnosed, in spite of the fact that some attribute values were not known. For example, it was feasible to diagnose a patient in spite of the fact that some test results were not taken (here attributes correspond to tests, so attribute values are test results). Since such missing attribute values do not matter for the final outcome, we will call them "do not care" conditions. The third possibility is a partial "do not care" condition: we assume that the missing attribute value belongs to the set of typical attribute values for all cases from the same concept. Such a missing attribute value will be called an attribute-concept value. Calling it *concept "do not care" condition* would be perhaps better, but this name is too long.

The main objective of this paper is to study incomplete decision tables, assuming that in the same decision table some attribute values may be lost, some may be "do not care" conditions, and some may be attribute-concept values. Decision tables with lost values and "do not care" conditions were studied in [7–9, 12].

For such incomplete decision tables there are three special cases: in the first case all missing attribute values are lost, in the second case all missing attribute values are "do not care" conditions, and in the third case all missing attribute vales are attribute-concept values. Incomplete decision tables in which all attribute values are lost, from the viewpoint of rough set theory, were studied for the first time in [13], where two algorithms for rule induction, modified to handle lost attribute values, were presented. This approach was studied later in [20–22], where the indiscernibility relation was generalized to describe such incomplete decision tables.

On the other hand, incomplete decision tables in which all missing attribute values are "do not care" conditions, again from the view point of rough set theory, were studied for the first time in [4], where a method for rule induction was introduced in which each missing attribute value was replaced by all values from the domain of the attribute. Originally such values were replaced by all values from the entire domain of the attribute, later by attribute values restricted to the same concept to which a case with a missing attribute value belongs. Such incomplete decision tables, with all missing attribute values being "do not care conditions", were extensively studied in [14, 15], including extending the idea of the indiscernibility relation to describe such incomplete decision tables.

Rough set methodology for incomplete decision tables with missing attribute values of the type attribute-concept values is presented in this paper for the first time, though it was briefly mentioned in [10].

In general, incomplete decision tables are described by characteristic relations, in a similar way as complete decision tables are described by indiscernibility relations [7].

For complete decision tables, once the indiscernibility relation is fixed and the concept (a set of cases) is given, the lower and upper approximations are unique.

For incomplete decision tables, for a given characteristic relation and the concept, there are three different possible ways to define lower and upper approximations, called singleton, subset, and concept approximations [7]. The singleton lower and upper approximations were studied in [14, 15, 20–22]. Similar ideas were studied in [2, 19, 23–25]. In this paper we further discuss applications to data mining of all three kinds of approximations: singleton, subset and concept. As it was observed in [7], singleton lower and upper approximations are not applicable in data mining.

The next topic of this paper is demonstrating how certain and possible rules may be computed from incomplete decision tables. An extension of the well-known LEM2 (Learning from Examples Module, version 2) rule induction algorithm [1, 5], called MLEM2, was introduced in [6]. LEM2 is a component of the LERS (Learning from Examples based on Rough Sets) data mining system. Originally, MLEM2 induced certain rules from incomplete decision tables with numerical attributes and with missing attribute values interpreted as lost.

Using the idea of lower and upper approximations for incomplete decision tables, MLEM2 was further extended to induce both certain and possible rules from a decision table with some numerical attributes and with some attribute values being lost and some attribute values being "do not care" conditions.

2. Complete data: elementary sets and indiscernibility relation

An example of a decision table, taken from [10], is presented in Table 1.

Table 2. An example of a complete decision table

	Attributes			Decision
	Temperature	Headache	Nausea	Flu
1	high	yes	no	yes
2	very_high	yes	yes	yes
3	high	no	no	no
4	high	yes	yes	yes
5	high	yes	yes	no
6	normal	yes	no	no
7	normal	no	yes	no
8	normal	yes	no	yes

Rows of the decision table represent *cases*, while columns are labeled by *variables*. The set of all cases will be denoted by U. In Table 1, $U = \{1, 2, ..., 8\}$. Independent variables are called *attributes* and a dependent variable is called a *decision* and is denoted by d. The set of all attributes will be denoted by A. In Table 1, $A = \{Temperature, Headache, Nausea\}$. Any decision table defines a function ρ that maps the direct product of U and A into the set of all values. For example, in Table 1, $\rho(1, Temperature) = high$. Function ρ describing Table 1 is completely specified (total). A decision table with completely specified function ρ will be called *completely specified*, or, for the sake of simplicity, *complete*.

Rough set theory [16, 17] is based on the idea of an indiscernibility relation, defined for complete decision tables. Let B be a nonempty subset of the set A of all attributes. The indiscernibility relation IND(B) is a relation on U defined for $x, y \in U$ as follows

$$(x, y) \in \text{IND}(B) \text{ if and only if } \rho(x, a) = \rho(y, a) \text{ for all } a \in B.$$

The indiscernibility relation IND(B) is an equivalence relation. Equivalence classes of IND(B) are called *elementary sets* of B and are denoted by $[x]_B$. For example, for Table 1, elementary sets of IND(A) are {1}, {2}, {3}, {4, 5}, {6, 8}, {7}. The indiscernibility relation IND(B) may be computed using the idea of blocks of attribute-value pairs. Let a be an attribute, i.e., $a \in A$ and let v be a value of a for some case. For complete decision tables if $t = (a, v)$ is an attribute-value pair then a *block* of t, denoted $[t]$, is a set of all cases from U that for attribute a have value v. For Table 1,

[(Temperature, high)] = {1, 3, 4, 5},

[(Temperature, very_high)] = {2},

[(Temperature, normal)] = {6, 7, 8},

[(Headache, yes)] = {1, 2, 4, 5, 6, 8},

[(Headache, no)] = {3, 7},

[(Nausea, no)] = {1, 3, 6},

[(Nausea, yes)] = {2, 4, 5, 7}.

The indiscernibility relation IND(B) is known when known are all elementary blocks of IND(B). Such elementary blocks of B are intersections of the corresponding attribute-value pairs, i.e., for any case $x \in U$,

$$[x]_B = \bigcap \{[(a, \rho(a, v))] \mid a \in B\}$$

We will illustrate the idea how to compute elementary sets of B for Table 1 and $B = A$.

$[1]_A$ = [(Temperature, high)] $\cap$ [(Headache, yes)] $\cap$ [(Nausea, no)] = {1},

$[2]_A$ = [(Temperature, very_high)] $\cap$ [(Headache, yes)] $\cap$ [(Nausea, yes)] = {2},

$[3]_A$ = [(Temperature, high)] $\cap$ [(Headache, no)] $\cap$ [(Nausea, no)] = {3},

$[4]_A = [5]_A$ = [(Temperature, high)] $\cap$ [(Headache, yes)] $\cap$ [(Nausea, yes)] = {4, 5},

$[6]_A = [8]_A$ = [(Temperature, normal)] $\cap$ [(Headache, yes)] $\cap$ [(Nausea, no] = {6, 8},

$[7]_A$ = [(Temperature, normal)] $\cap$ [(Headache, no] $\cap$ [(Nausea, yes)] = {7}.

3. Incomplete data: characteristic sets and characteristic relations

For data sets with missing attribute values, the corresponding function ρ is incompletely specified (partial). A decision table with incompletely specified function ρ will be called *incompletely specified*, or *incomplete*.

In the sequel we will assume that all decision values are specified, i.e., they are not missing. Also, we will assume that all missing attribute values are denoted by "?", by "*" or by "–", lost values will be denoted by "?", "do not care" conditions will be denoted by "*", and attribute-concept values by "–". Additionally, we will assume that for each case at least one attribute value is specified.

Incomplete decision tables are described by characteristic relations instead of indiscernibility relations. Also, elementary sets are replaced by characteristic sets. An example of an incomplete table is presented in Table 2.

Table 1. An example of an incomplete decision table

	Attributes			Decision
	Temperature	Headache	Nausea	Flu
1	high	–	no	yes
2	very_high	yes	yes	yes
3	?	no	no	no
4	high	yes	yes	yes
5	high	?	yes	no
6	normal	yes	no	no
7	normal	no	yes	no
8	–	yes	*	yes

For incomplete decision tables the definition of a block of an attribute-value pair must be modified.

- If an attribute a there exists a case x such that $\rho(x, a) =$?, i.e., the corresponding value is lost, then the case x should not be included in any block $[(a, v)]$ for all values v of attribute a.
- If for an attribute a there exists a case x such that the corresponding value is a "do not care" condition, i.e., $\rho(x, a) =$ *, then the corresponding case x should be included in blocks $[(a, v)]$ for all specified values v of attribute a.
- If for an attribute a there exists a case x such that the corresponding value is a attribute-concept value, i.e., $\rho(x, a) =$ –, then the corresponding case x should be included in blocks $[(a, v)]$ for all specified values v of attribute a that are members of the set $V(x, a)$, where

$$V(x, a) = \{\rho(y, a) \mid y \in U, \rho(y, d) = \rho(x, d)\},$$

and d is the decision.

These modifications of the definition of the block of attribute-value pair are consistent with the interpretation of missing attribute values, lost, "do not care" conditions, and attribute-concept values. Also, note that the attribute-concept value is the most universal, since if $V(x, a) = \emptyset$, the definition of the attribute-concept value is reduced to the lost value, and if $V(x, a)$ is the set of all values of an attribute a, the attribute-concept value becomes a "do not care" condition.

For Table 2, for case 1, $\rho(1,$ Headache$) =$ –, and $V(1,$ Headache$) =$ {yes}, so we add the case 1 to [(Headache, yes)]. For case 3, $\rho(1,$ Temperature$) =$?, hence case 3 is not included in either of the following sets: [(Temperature, high)], [(Temperature, very_high)], and [(Temperature, normal)]. Similarly, $\rho(5,$ Headache$) =$?, so the case 5 is not included in [(Headache, yes)] and [(Headache, no)]. Also, $\rho(8,$ Temperature$) =$ –, and $V(8,$ Temperature$) =$ {high, very_high}, so the case 8 is a member of both [(Temperature, high)] and [(Temperature, very_high)]. Finally, $\rho(8,$ Nausea$) =$ *, so the case 8 is included in both [(Nausea, no)] and [(Nausea, yes)]. Thus,

[(Temperature, high)] = {1, 4, 5, 8},

[(Temperature, very_high)] = {2, 8},

[(Temperature, normal)] = {6, 7},

[(Headache, yes)] = {1, 2, 4, 6, 8},

[(Headache, no)] = {3, 7},

[(Nausea, no)] = {1, 3, 6, 8},

[(Nausea, yes)] = {2, 4, 5, 7, 8}.

For a case $x \in U$, the characteristic set $K_B(x)$ is defined as the intersection of the sets $K(x, a)$, for all $a \in B$. If $\rho(x, a)$ is specified, then $K(x, a)$ is the block $[(a, \rho(x, a)]$ of attribute a and its value $\rho(x, a)$. If $\rho(x, a) =$ * or $\rho(x, a) =$? then the set $K(x, a) = U$. If $\rho(x, a) =$ –, then the corresponding set $K(x, a)$ is equal to the union of all blocks of attribute-value pairs (a, v), where $v \in V(x, a)$. The way of computing characteristic sets needs a comment. For both "do not care" conditions and lost values the corresponding set $K(x, a)$ is equal to U because the corresponding attribute a does not restrict the set $K_B(x)$: if $\rho(x, a) =$ *, the value of the attribute a is irrelevant; if $\rho(x, a) =$?, only existing values need to be checked. However,

the case when $\rho(x, a) = -$ is different, since the attribute a restricts the set $K_B(x)$. Furthermore, the description of $K_B(x)$ should be consistent with other (but similar) possible approaches to missing attribute values, e.g., an approach in which each missing attribute value is replaced by the most common attribute value restricted to a concept. Here the set $V(x, a)$ contains a single element and the characteristic relation is an equivalence relation. Our definition is consistent with this special case in the sense that if we compute a characteristic relation for such a decision table using our definition or if we compute the indiscernibility relation as for complete decision tables using definitions from Section 2, the result will be the same. For Table 2 and $B = A$,

$K_A(1) = \{1, 4, 5, 8\} \cap \{1, 2, 4, 6, 8\} \cap \{1, 3, 6, 8\} = \{1, 8\}$,

$K_A(2) = \{2, 8\} \cap \{1, 2, 4, 6, 8\} \cap \{2, 4, 5, 7, 8\} = \{2, 8\}$,

$K_A(3) = U \cap \{3, 7\} \cap \{1, 3, 6, 8\} = \{3\}$,

$K_A(4) = \{1, 4, 5, 8\} \cap \{1, 2, 4, 6, 8\} \cap \{2, 4, 5, 7, 8\} = \{4, 8\}$,

$K_A(5) = \{1, 4, 5, 8\} \cap U \cap \{2, 4, 5, 7, 8\} = \{4, 5, 8\}$,

$K_A(6) = \{6, 7\} \cap \{1, 2, 4, 6, 8\} \cap \{1, 3, 6, 8\} = \{6\}$,

$K_A(7) = \{6, 7\} \cap \{3, 7\} \cap \{2, 4, 5, 7, 8\} = \{7\}$, and

$K_A(8) = (\{1, 4, 5, 8\} \cap \{2, 8\}) \cap \{1, 2, 4, 6, 8\} \cap U = \{1, 2, 4, 8\}$.

The characteristic set $K_B(x)$ may be interpreted as the smallest set of cases that are indistinguishable from x using all attributes from B, and using given interpretation of missing attribute values. Thus, $K_A(x)$ is the set of all cases that cannot be distinguished from x using all attributes. Also, note that the previous definition is an extension of a definition of $K_B(x)$ from [7–9, 12]: for decision tables with only lost values and "do not care" conditions, both definitions are identical.

The characteristic relation $R(B)$ is a relation on U defined for $x, y \in U$ as follows

$$(x, y) \in R(B) \text{ if and only if } y \in K_B(x).$$

The characteristic relation $R(B)$ is reflexive but—in general—does not need to be symmetric or transitive. Also, the characteristic relation $R(B)$ is known if we know characteristic sets $K(x)$ for all $x \in U$. In our example, $R(A) = \{(1, 1), (1, 8), (2, 2), (2, 8), (3, 3), (4, 4), (4, 8), (5, 4), (5, 5), (5, 8), (6, 6), (7, 7), (8, 1), (8, 2), (8, 4), (8, 8)\}$.

For decision tables, in which all missing attribute values are lost, a special characteristic relation LV(B) was defined by J. Stefanowski and A. Tsoukias in [21], see also [20, 22]. Characteristic relation LV(B) is reflexive, but—in general—does not need to be symmetric or transitive.

For decision tables where all missing attribute values are "do not care" conditions a special characteristic relation DCC(B) was defined by M. Kryszkiewicz in [14], see also, e.g., [15]. Relation DCC(B) is reflexive and symmetric but—in general—not transitive.

Obviously, characteristic relations LV(B) and DCC(B) are special cases of the characteristic relation $R(B)$. For a completely specified decision table, the characteristic relation $R(B)$ is reduced to IND(B).

4. Lower and upper approximations

For completely specified decision tables lower and upper approximations are defined using the indiscernibility relation. Any finite union of elementary sets of B is called a *B-definable set*. Let X be any subset of the set U of all cases. The set X is called *concept* and is usually defined as the set of all cases defined by a specific value of the decision. In general, X is not a B-definable set. However, set X may be approximated by two B-definable sets, the first one is called a *B-lower approximation of X*, denoted by $\underline{B}X$ and defined as follows

$$\{x \in U \mid [x]_B \subseteq X\}.$$

The second set is called an *B-upper approximation of X*, denoted by $\overline{B}X$ and defined as follows

$$\{x \in U \mid [x]_B \cap X \neq \emptyset\}.$$

The above way of computing lower and upper approximations, by constructing them from singletons x, will be called the first method. The B-lower approximation of X is the greatest B-definable set, contained in X. The B-upper approximation of X is the least B-definable set containing X.

As it was observed in [16], for complete decision tables we may use a second method to define the B-lower approximation of X, by the following formula

$$\cup \{[x]_B \mid x \in U, [x]_B \subseteq X\}$$

and the B-upper approximation of x may de defined, using the second method, by

$$\cup \{[x]_B \mid x \in U, [x]_B \cap X \neq \emptyset).$$

For Table 1 and $B = A$, A-lower and A-upper approximations are:

$\underline{A}\{1, 2, 4, 8\} = \{1, 2\}$,
$\underline{A}\{3, 5, 6, 7\} = \{3, 7\}$,
$\overline{A}\{1, 2, 4, 8\} = \{1, 2, 4, 5, 6, 8\}$,
$\overline{A}\{3, 5, 6, 7\} = \{3, 4, 5, 6, 7, 8\}$.

For incompletely specified decision tables lower and upper approximations may be defined in a few different ways. To begin with, the definition of definability should be modified. Any finite union of characteristic sets of B is called a *B-definable* set. Following [7], we suggest three different definitions of approximations. Again, let X be a concept, let B be a subset of the set A of all attributes, and let $R(B)$ be the characteristic relation of the incomplete decision table with characteristic sets $K(x)$, where $x \in U$. Our first definition uses a similar idea as in the previous articles on incompletely specified decision tables [14, 15, 20–22], i.e., lower and upper approximations are sets of singletons from the universe U satisfying some properties. Thus we are defining lower and upper approximations by analogy with the above first method, by constructing both sets from singletons. We will call these definitions *singleton*. A singleton B-lower approximation of X is defined as follows:

$$\underline{B}X = \{x \in U \mid K_B(x) \subseteq X\}.$$

A singleton B-upper approximation of X is

$$\overline{B}X = \{x \in U \mid K_B(x) \cap X \neq \emptyset\}.$$

In our example presented in Table 2 let us say that $B = A$. Then the singleton A-lower and A-upper approximations of the two concepts: {1, 2, 4, 8} and {3, 5, 6, 7} are:

$\underline{A}\{1, 2, 4, 8\} = \{1, 2, 4, 8\}$,
$\underline{A}\{3, 5, 6, 7\} = \{3, 6, 7\}$,
$\overline{A}\{1, 2, 4, 8\} = \{1, 2, 4, 5, 8\}$,
$\overline{A}\{3, 5, 6, 7\} = \{3, 5, 6, 7\}$.

Note that $\overline{A}\{3, 5, 6, 7\} = \{3, 5, 6, 7\}$. However, the set {3, 5, 6 7} is not A-definable, so a set of rules, induced from {3, 5, 6, 7}, cannot cover precisely this set. In general, singleton approximations should not be used for data mining.

The second method of defining lower and upper approximations for complete decision tables uses another idea: lower and upper approximations are unions of elementary sets, subsets of U. Therefore we may define lower and upper approximations for incomplete decision tables by analogy with the second method, using characteristic sets instead of elementary sets. There are two ways to do this. Using the first way, a *subset* B-lower approximation of X is defined as follows:

$$\underline{B}X = \cup \{K_B(x) \mid x \in U, K_B(x) \subseteq X\}.$$

A *subset* B-upper approximation of X is

$$\overline{B}X = \cup \{K_B(x) \mid x \in U, K_B(x) \cap X \neq \emptyset\}.$$

Since any characteristic relation $R(B)$ is reflexive, for any concept X, singleton B-lower and B-upper approximations of X are subsets of subset B-lower and B-upper approximations of X, respectively. For the same the decision presented in Table 2, the subset A-lower and A-upper approximations are:

$\underline{A}\{1, 2, 4, 8\} = \{1, 2, 4, 8\}$,
$\underline{A}\{3, 5, 6, 7\} = \{3, 6, 7\}$,
$\overline{A}\{1, 2, 4, 8\} = \{1, 2, 4, 5, 8\}$,
$\overline{A}\{3, 5, 6, 7\} = \{3, 4, 5, 6, 7, 8\}$.

The second possibility is to modify the subset definition of lower and upper approximation by replacing the universe U from the subset definition by a concept X. A *concept* B-lower approximation of the concept X is defined as follows:

$$\underline{B}X = \cup \{K_B(x) \mid x \in X, K_B(x) \subseteq X\}.$$

Obviously, the subset B-lower approximation of X is the same set as the concept B-lower approximation of X. A *concept* B-upper approximation of the concept X is defined as follows:

$$\overline{B}X = \cup \{K_B(x) \mid x \in X, K_B(x) \cap X \neq \emptyset\} = \cup \{K_B(x) \mid x \in X\}.$$

The concept B-upper approximation of X are subsets of the subset B-upper approximations of X. For the decision presented in Table 2, the concept A-lower and A-upper approximations are:

$\underline{A}\{1, 2, 4, 8\} = \{1, 2, 4, 8\}$,
$\underline{A}\{3, 5, 6, 7\} = \{3, 6, 7\}$,
$\overline{A}\{1, 2, 4, 8\} = \{1, 2, 4, 8\}$,
$\overline{A}\{3, 5, 6, 7\} = \{3, 4, 5, 6, 7, 8\}$.

For complete decision tables, all three definitions of lower approximations, singleton, subset and concept, coalesce to the same definition. Also, for complete

decision tables, all three definitions of upper approximations coalesce to the same definition. This is not true for incomplete decision tables, as our example shows.

5. Rule induction

The same idea of blocks of attribute-value pairs is used in the rule induction algorithm LEM2. LEM2 explores the search space of attribute-value pairs. Its input data file is a lower or upper approximation of a concept, so its input data file is always consistent. Rules induced from the lower approximation of the concept *certainly* describe the concept, so they are called *certain*. On the other hand, rules induced from the upper approximation of the concept describe the concept only *possibly* (or *plausibly*), so they are called *possible* [3].

Rules in LERS format (every rule is equipped with three numbers, the total number of attribute-value pairs on the left-hand side of the rule, the total number of examples correctly classified by the rule during training, and the total number of training cases matching the left-hand side of the rule) induced from Table 2 using concept approximations are:

the certain rule set:

2, 3, 3
(Temperature, high) & (Headache, yes) -> (Flu, yes)

1, 2, 2
(Temperature, very_high) -> (Flu, yes)

1, 2, 2
(Temperature, normal) -> (Flu, no)

1, 2, 2
(Headache, no) -> (Flu, no)

and the possible rule set:

2, 3, 3
(Temperature, high) & (Headache, yes) -> (Flu, yes)

1, 2, 2
(Temperature, very_high) -> (Flu, yes)

2, 1, 3
(Temperature, high) & (Nausea, no) -> (Flu, no)

1, 2, 2
(Temperature, normal) -> (Flu, no)

1, 2, 2
(Headache, no) -> (Flu, no)

6. Conclusions

Three approaches to missing attribute values are presented in a unified way. The main applied tool is a characteristic relation, a generalization of the indiscernibility relation. It is shown that all three approaches to missing attribute values may be described using the same idea of attribute-value blocks. Moreover, attribute-value blocks are useful not only for computing characteristic sets but also for computing characteristic relations, lower and upper approximations, and, finally for rule induction. Additionally, using attribute-value blocks, it is quite easy to combine a few strategies to handle missing attribute values within the same data set. Thus, the entire data mining process, starting from computing characteristic relations and ending with rule induction, may be implemented using the same simple tool: attribute-value blocks.

References

[1] C.-C. Chan and J. W. Grzymala-Busse. On the attribute redundancy and the learning programs ID3, PRISM, and LEM2. Department of Computer Science, University of Kansas, TR-91-14, December 1991, 20 pp.

[2] S. Greco, B. Matarazzo, and R. Slowinski. Dealing with missing data in rough set analysis of multi-attribute and multi-criteria decision problems. In *Decision Making: Recent developments and Worldwide Applications*, ed. by S. H. Zanakis, G. Doukidis and Z. Zopounidis, Kluwer Academic Publishers, Dordrecht, Boston, London, 2000, 295–316.

[3] J. W. Grzymala-Busse. Knowledge acquisition under uncertainty—A rough set approach. *Journal of Intelligent & Robotic Systems* 1, 1 (1988), 3–16.

[4] J. W. Grzymala-Busse. On the unknown attribute values in learning from examples. Proc. of the ISMIS-91, 6th International Symposium on Methodologies for Intelligent Systems, Charlotte, North Carolina, October 16–19, 1991, 368–377, *Lecture Notes in Artificial Intelligence*, vol. 542, Springer-Verlag, Berlin, Heidelberg, New York, 1991.

[5] J. W. Grzymala-Busse. LERS—A system for learning from examples based on rough sets. In *Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory*, ed. by R. Slowinski, Kluwer Academic Publishers, Dordrecht, Boston, London, 1992, 3–18.

[6] J. W. Grzymala-Busse. MLEM2: A new algorithm for rule induction from imperfect data. Proceedings of the 9th

International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU 2002, July 1–5, Annecy, France, 243–250.

[7] J. W. Grzymala-Busse. Rough set strategies to data with missing attribute values. Proceedings of the Workshop on Foundations and New Directions in Data Mining, associated with the third IEEE International Conference on Data Mining, November 19–22, 2003, Melbourne, FL, 56–63.

[8] J. W. Grzymala-Busse. Characteristic relations for incomplete data: A generalization of the indiscernibility relation. Proceedings of the RSCTC'2004, the Fourth International Conference on Rough Sets and Current Trends in Computing, Uppsala, Sweden, June 1–5, 2004. Lecture Notes in Artificial Intelligence 3066, Springer-Verlag 2004, 244–253.

[9] J. W. Grzymala-Busse. Rough set approach to incomplete data. Proceedings of the ICAISC'2004, the Seventh International Conference on Artificial Intelligence and Soft Computing, Zakopane, Poland, June 7–11, 2004. Lecture Notes in Artificial Intelligence 3070, Springer-Verlag 2004, 50–55.

[10] J. W. Grzymala-Busse. Data with missing attribute values: Generalization of idiscernibility relation and rule induction. *Transactions on Rough Sets*, Lecture Notes in Computer Science Journal Subline, Springer-Verlag, vol. 1 (2004) 78–95.

[11] J. W. Grzymala-Busse and M. Hu. A comparison of several approaches to missing attribute values in data mining. Proceedings of the Second International Conference on Rough Sets and Current Trends in Computing RSCTC'2000, October 16–19, 2000, Banff, Canada, 340–347.

[12] J. W. Grzymala-Busse and S. Siddhaye. Rough set approaches to rule induction from incomplete data. Proceedings of the IPMU'2004, the 10th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Perugia, Italy, July 4–9, 2004, vol. 2, 923–930.

[13] J. W. Grzymala-Busse and A. Y. Wang. Modified algorithms LEM1 and LEM2 for rule induction from data with missing attribute values. Proc. of the Fifth International Workshop on Rough Sets and Soft Computing (RSSC'97) at the Third Joint Conference on Information Sciences (JCIS'97), Research Triangle Park, NC, March 2–5, 1997, 69–72.

[14] M. Kryszkiewicz. Rough set approach to incomplete information systems. Proceedings of the Second Annual Joint Conference on Information Sciences, September 28–October 1, 1995, Wrightsville Beach, NC, 194–197.

[15] M. Kryszkiewicz. Rules in incomplete information systems. *Information Sciences* 113 (1999) 271–292.

[16] Z. Pawlak. Rough Sets. *International Journal of Computer and Information Sciences*, 11 (1982) 341–356.

[17] Z. Pawlak. *Rough Sets. Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, Dordrecht, Boston, London, 1991.

[18] J. R. Quinlan. *C*4.5: *Programs for Machine Learning*. Morgan Kaufmann Publishers, 1993.

[19] R. Slowinski and D. Vanderpooten. A generalized definition of rough approximations based on similarity. *IEEE Transactions on Knowledge and Data Engineering* 12 (2000) 331–336.

[20] J. Stefanowski. *Algorithms of Decision Rule Induction in Data Mining*. Poznan University of Technology Press, Poznan, Poland, 2001.

[21] J. Stefanowski and A. Tsoukias. On the extension of rough sets under incomplete information. Proceedings of the 7th International Workshop on New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, RSFDGrC'1999, Yamaguchi, Japan, 73–81.

[22] J. Stefanowski and A. Tsoukias. Incomplete information tables and rough classification. *Computational Intelligence* 17 (2001) 545–566.

[23] Y. Y. Yao. Two views of the theory of rough sets in finite universes. *International J. of Approximate Reasoning* 15 (1996) 291–317.

[24] Y. Y. Yao. Relational interpretations of neighborhood operators and rough set approximation operators. *Information Sciences* 111 (1998) 239–259.

[25] Y. Y. Yao. On the generalizing rough set theory. Proc. of the 9th Int. Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing (RSFDGrC'2003), Chongqing, China, Oct. 19–22, 2003, 44–51.

Fining Active Membership Functions in Fuzzy Data Mining

[1]Tzung-Pei Hong, [2]Chun-Hao Chen, [3]Yu-Lung Wu, [2]Vincent S.M. Tseng
[1]*Department of Electrical Engineering, National University of Kaohsiung*
[2]*Department of Information Engineering, National Cheng-Kung University*
[3]*Department of Information Management, I-Shou University*
tphong@nuk.edu.tw, chchen6814@gmail.com, wuyulung@isu.edu.tw, tsengsm@mail.ncku.edu.tw

Abstract

This paper proposes a fuzzy data-mining algorithm for extracting both association rules and membership functions from quantitative transactions. The number of membership functions for each item is not predefined, but can be dynamically adjusted. A GA-based framework for finding membership functions suitable for mining problems is proposed. The encoding of each individual is divided into two parts. The control genes are encoded into bit strings and used to determine whether membership functions are active or not. The parametric genes are encoded into real-number strings to represent membership functions of linguistic terms. The fitness of each set of membership functions is evaluated using the fuzzy-supports of the linguistic terms in the large 1-itemsets and the suitability of the derived membership functions. The suitability of membership functions considers overlap, coverage and usage factors.

1. Introduction

Data mining is most commonly used in attempts to induce association rules from transaction data. Transaction data in real-world applications, however, usually consist of quantitative values. Designing a sophisticated data-mining algorithm able to deal with various types of data presents a challenge to workers in this research field.

Recently, fuzzy set theory has been used more and more frequently in intelligent systems because of its simplicity and similarity to human reasoning. In [4], we proposed a mining approach that integrated fuzzy-set concepts with the *apriori* mining algorithm [1] to find interesting itemsets and fuzzy association rules in transaction data with quantitative values. In that paper, the membership functions were assumed to be known in advance. The given membership functions may, however, have a critical influence on the final mining results. This paper thus modifies the previous algorithm and proposes a new fuzzy data-mining algorithm for extracting both association rules and membership functions from quantitative transactions.

In the past, Srikant and Agrawal proposed a mining method [7] to handle quantitative transactions by partitioning the possible values of each attribute. Hong *et al*. proposed a fuzzy mining algorithm to mine fuzzy rules from quantitative data [4]. They transformed each quantitative item into a fuzzy set and used fuzzy operations to find fuzzy rules. Wang *et al.* used GAs to tune membership functions for intrusion detection systems based on similarity of association rules [11]. Kaya *et al.* [6] proposed a GA-based clustering method to derive a predefined number of membership functions for getting a maximum profit within an interval of user specified minimum support values. In this paper, we will try to derive an unknown number of membership functions from quantitative transactions by using a divide-and-conquer genetic strategy.

2. A GA-Based Mining Framework

In this section, the fuzzy and GA concepts are used to discover both useful association rules and suitable membership functions from quantitative values. A GA-based framework for achieving this purpose is proposed in Figure 1.

The proposed framework is divided into two phases: mining membership functions and mining fuzzy association rules. Assume the number of items is m. In the phase of mining membership functions, it maintains m populations of membership functions, with each population for an item I_j ($1 \leq j \leq m$). Each chromosome in a population represents a possible set of membership functions for that item. Next, in the phase of mining fuzzy association rules, the sets of membership function for all the items are gathered together and used to mine the interesting rules from the given quantitative database. Our

fuzzy mining algorithm proposed in [5] is adopted to achieve this purpose.

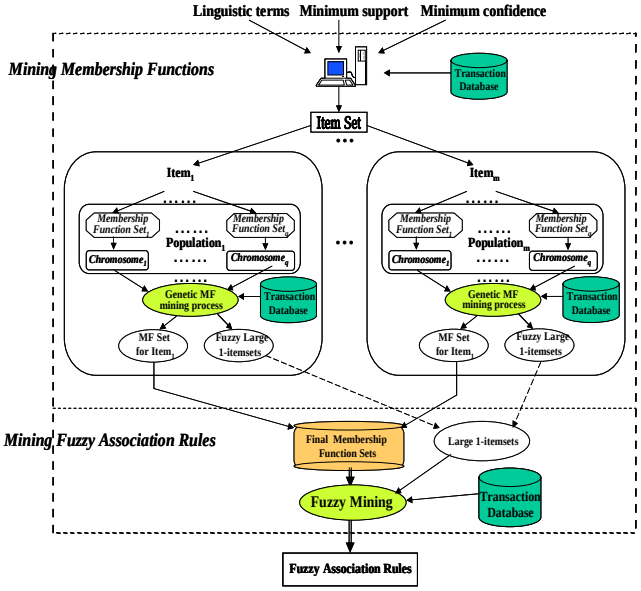


Figure 1: The proposed GA-based framework for fuzzy mining

3. Chromosome Representation

Several possible encoding approaches in GAs have been described in [2, 8, 9, 10]. In this paper, we adopt the encoding approach similar to that in [8]. Each individual is divided into two parts, control genes and parametric genes. In the first part, control genes are encoded into bit strings and used to determine whether parametric genes are active or not. In the second part, each set of membership functions for an item is encoded as parametric genes with real-number schema.

Assume the membership functions are triangular. Three parameters are thus used to represent a membership function. Each parametric gene thus consists of three real values. Figure 2 shows an example for item I_j, where R_{jk} denotes the membership function of the k-th linguistic term and r_{jkp} indicates the p-th parameter of fuzzy region R_{jk}.

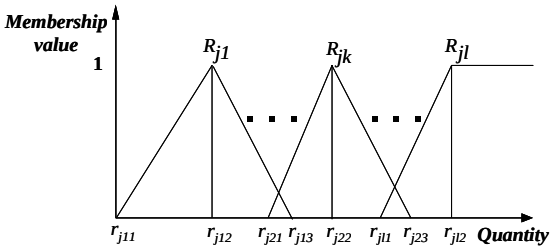


Figure 2: The set of membership functions for item I_j

The parametric genes of item I_j can be represented as a string of $r_{j11}r_{j12}r_{j13}r_{j21}r_{j22}r_{j23}$ … $r_{jl1}r_{jl2}r_{jl3}$, where $r_{jl3} = \infty$. The control genes of Item I_j can be represented as a bit string of $b_{j1}b_{j2}...b_{jT}$, where T is the maximum possible number of linguistic terms. The bit b_{ji} indicates whether the *i-th* membership function is active or not. If b_{ji}=1, the i-th membership function is active, meaning it will be used in the later fuzzy mining process. If b_{ji}=0, it is inactive. All the individuals in the same population thus have the same string length. Below, an example is given to demonstrate the process of encoding membership functions.

Example 1: Assume there are four items in a transaction database: milk, bread, cookies and beverage. Also assume a possible set of membership functions for Item *milk* is given as shown in Figure 3.

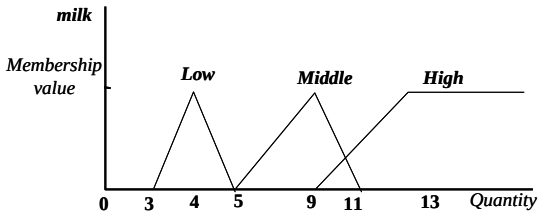


Figure 3: An example of a possible set of membership functions for Item *milk*

There are three active linguistic terms, *Low*, *Middle*, and *High*, for this item. According to the proposed encoding scheme, the individual for representing the set of membership functions in Figure 3 is encoded as shown in Figure 4.

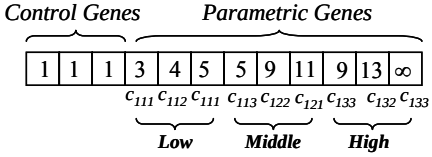


Figure 4: The chromosome representation for the set of membership functions in Figure 3

4. Mining Membership Functions and Fuzzy Association Rules

4.1 Initial Population

A genetic algorithm requires a population of feasible solutions to be initialized and updated during the evolution process. As mentioned above, each individual within the population is a set of triangular membership functions for a certain item. Each membership function corresponds to a linguistic term in the item. The initial set

of chromosomes is randomly generated with some constraints of forming feasible membership functions.

4.2 Fitness and Selection

In order to develop a good set of membership functions from an initial population, the genetic algorithm selects parent sets of membership functions with high fitness values for mating. An evaluation function is defined to qualify the derived sets of membership functions. Before the fitness of each set of membership functions is formally described, several related terms are first explained below.

The overlap ratio of two membership functions R_{jk} and R_{ji} ($k < j$) is defined as the overlap length divided by the minimum of the right span of R_{jk} and the left span of R_{ji}. That is,

$$overlap\_ratio(R_{jk}, R_{ji}) = \frac{overlap(R_{jk}, R_{ji})}{\min(c_{jk3} - c_{jk2}, c_{ji2} - c_{ji1})},$$

where $overlap(R_{jk}, R_{ji})$ is the overlap length of R_{jk} and R_{ji}.

If the overlap length is larger than the minimum of the above two half spans, then these two membership functions are thought of as a little redundant. Appropriate punishment must then be considered in this case. Thus, the overlap factor of the membership functions for an item I_j in the chromosome C_q is defined as:

$$\sum_{\substack{k \neq i \\ R_{jk}, R_{ji} \text{ are active}}} [\max((\frac{overlap(R_{jk}, R_{ji})}{\min(c_{jk3} - c_{jk2}, c_{ji2} - c_{ji1})}), 1) - 1] \cdot$$

The coverage ratio of membership functions for an item I_j is defined as the coverage range of the functions divided by the maximum quantity of that item in the transactions. The more the coverage ratio is, the better the derived membership functions are. Thus, the coverage factor of the membership functions for an item I_j in the chromosome C_q is defined as:

$$coverage\_factor(C_q) = \frac{1}{\frac{range(R_{j1}, ..., R_{jl})}{max(I_j)}},$$

where $range(R_{j1}, R_{j2}, \ldots, R_{jl})$ is the coverage range of the active membership functions, l is the number of active membership functions for I_j, and $max(I_j)$ is the maximum quantity of I_j in the transactions.

The usage ratio of membership functions for an item I_j is defined as the number of large-1 itemsets for I_j divided by the number of active linguistic terms. Note that the maximum possible number of large-1 itemsets for an item is the number of its active linguistic terms. The more the usage ratio is, the better the derived membership functions are. Thus, the usage factor of the membership functions for an item I_j in the chromosome C_q is defined as:

$$usage\_factor(C_q) = \frac{l_{C_q}}{max(|L_1^{C_q}|, 1)},$$

where l_{Cq} is the active linguistic terms of chromosome C_q, and $max(|L_1^{C_q}|, 1)$ is the maximum of the number of large-1 itemsets and 1.

The suitability of the set of membership functions in a chromosome C_q is thus defined as $k_1 * overlap\_factor(C_q) + k_2 * coverage\_factor(C_q) + k_3 * usage\_factor(C_q)$, where k_1, k_2, k_3 are weighting factors.

The fitness value of a chromosome C_q is then defined as:

$$f(C_q) = \frac{\sum_{X \in L_1^{C_q}} fuzzy\_support(X)}{suitability(C_q)},$$

where $L_1^{C_q}$ is the set of large 1-itemsets obtained by using the set of membership functions in C_q, and $fuzzy\_support(X)$ is the fuzzy support of the 1-itemset X derived from C_q in the given transaction database.

The suitability factor used in the fitness function can reduce the occurrence of the two bad kinds of membership functions shown in Figure 5, where the first one is too redundant, and the second one is too separate. It can also help generate an appropriate number of membership functions for an item.

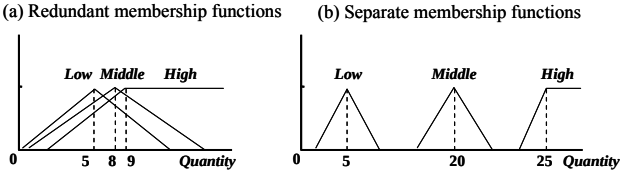


Figure 5: Two bad sets of membership functions

The overlap factor in suitable(C_q) is designed for avoiding the first bad case, and the coverage factor is for the second one.

4.3 Genetic Operators

Genetic operators are important to the success of specific GA applications. In our approach, different crossover operators are performed for control genes and parametric genes. For control genes, the single-point crossover and the binary one-point mutation operators are used. For parametric genes, the max-min-arithmetical (MMA) crossover operator proposed in [3] and the one-point mutation for real numbers are used. The max-min-arithmetical (MMA) crossover operator proceeds as follows. Assume there are two parent chromosomes with their parametric genes as

$$C_u^t = (c_1, ..., c_h, ..., c_Z)$$
$$C_w^t = (c_1^{'}, ..., c_h^{'}, ..., c_Z^{'}).$$

The max-min-arithmetical (MMA) crossover operator will generate the following four candidate chromosomes from them.

1. $C_1^{t+1} = (c_{11}^{t+1}, ..., c_{1h}^{t+1}, ..., c_{1Z}^{t+1})$, where $c_{1h}^{t+1} = dc_h + (1-d)c_h^{'}$,
2. $C_2^{t+1} = (c_{21}^{t+1}, ..., c_{2h}^{t+1}, ..., c_{2Z}^{t+1})$, where $c_{2h}^{t+1} = dc_h^{'} + (1-d)c_h$,
3. $C_3^{t+1} = (c_{31}^{t+1}, ..., c_{3h}^{t+1}, ..., c_{3Z}^{t+1})$, where $c_{3h}^{t+1} = \min\{c_h, c_h^{'}\}$,
4. $C_4^{t+1} = (c_{41}^{t+1}, ..., c_{4h}^{t+1}, ..., c_{4Z}^{t+1})$, where $c_{4h}^{t+1} = \max\{c_h, c_h^{'}\}$

where the parameter d is either a constant or a variable whose value depends on the age of the population. The best two chromosomes of the four candidates are then chosen as the offspring.

The one-point mutation operator for real numbers will create a new fuzzy membership function by adding a random value ε (may be negative) to one parameter of an existing linguistic term, say R_{jk}. Assume that r_{jkp} represents a parameter of R_{jk}. The parameter of the newly derived membership function may be changed to $r_{jkp} + \varepsilon$ by the mutation operation. Mutation at a parameter of a fuzzy membership function may, however, disrupt the order of the resulting fuzzy membership functions. These fuzzy membership functions then need rearrangement according to their values. An example is given below to demonstrate the mutation operation.

Example 2: Continuing from Example 1, assume the mutation point is set at c_{122} and the random value ε is set at 3. The mutation process is shown in Figure 6.

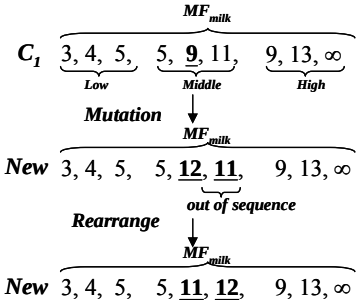


Figure 6: A mutation operation

5. The Proposed Mining Algorithm

According to the above description, the proposed algorithm for mining both membership functions and fuzzy association rules is described below.

The proposed mining algorithm:

INPUT: A body of n quantitative transaction data, a set of m items, a maximum possible number T of linguistic terms, a support threshold α, a confidence threshold λ, and a population size P.

OUTPUT: A set of fuzzy association rules with its associated set of membership functions.

STEP 1: Randomly generate m populations, each for an item; Each individual in a population represents a possible set of membership functions for that items.

STEP 2: Encode each set of membership functions into a string representation in the way mentioned above.

STEP 3: Calculate the fitness value of each chromosome in each population by the following substeps:

STEP 3.1: For each transaction datum D_i, i=1 to n, and for each item I_j, j=1 to m, transfer the quantitative value $v_j^{(i)}$ into a fuzzy set $f_j^{(i)}$ represented as:

$$\left(\frac{f_{j1}^{(i)}}{R_{j1}} + \frac{f_{j2}^{(i)}}{R_{j2}} + ... + \frac{f_{jl}^{(i)}}{R_{jl}}\right),$$

using the corresponding membership functions represented by the chromosome, where R_{jk} is the k-th fuzzy region (term) of item I_j, $f_{jl}^{(i)}$ is $v_j^{(i)}$'s fuzzy membership value in region R_{jk}, and l (= $|I_j|$) is the number of active linguistic terms for I_j.

STEP 3.2: For each item region R_{jk}, calculate its scalar cardinality on the transactions as follows:

$$count_{jk} = \sum_{i=1}^{n} f_{jk}^{(i)}.$$

STEP 3.3: For each R_{jk}, $1 \le j \le m$ and $1 \le k \le |I_j|$, check whether its $count_{jk}$ over n is larger than or equal to the minimum support threshold α. If R_{jk} satisfies the above condition, put it in the set of large 1-itemsets (L_1). That is:

$$L_1 = \{R_{jk} \mid count_{jk}/n \ge \alpha,\ 1 \le j \le m \text{ and } 1 \le k \le |I_j|\}.$$

STEP 3.4: Set the fitness value of the chromosome as the sum of the fuzzy supports (the scalar cardinalities / n) of the fuzzy regions in L_1 divided by $suitability(C_q)$. That is:

$$f(C_q) = \frac{\sum_{X \in L_1} fuzzy\_support(X)}{suitability(C_q)}.$$

STEP 4: Execute crossover operations on each population.
STEP 5: Execute mutation operations on each population.
STEP 6: Using the selection criteria to choose individuals in each population for the next generation.
STEP 7: If the termination criterion is not satisfied, go to Step 3; otherwise, do the next step.
STEP 8: Gather the sets of membership functions, each of which has the highest fitness value in its population.

The sets of the best membership functions gathered from each population are then used to mine fuzzy association rules from the given quantitative database. Our fuzzy mining algorithm proposed in [5] is then adopted to achieve this purpose. It first transforms each quantitative value into a fuzzy set of linguistic terms using the derived membership functions. It then calculates the scalar cardinality of each linguistic term on all the transaction data. The mining process based on fuzzy counts is then performed to find fuzzy association rules.

6. An Example

In this section, an example is given to illustrate the proposed mining algorithm. Assume there are four items in a transaction database: milk, bread, cookies and beverage. The data set includes the six transactions shown in Table 1.

Table 1. Six transactions in this example

TID	Items
T1	(milk, 5); (bread, 10); (cookies, 7); (beverage, 7).
T2	(milk, 7); (bread, 14); (cookies, 12).
T3	(bread, 15); (cookies, 12); (beverage, 10).
T4	(milk, 2); (bread, 5); (cookies, 5).
T5	(bread, 9).
T6	(milk, 13); (beverage, 12).

Assume the maximum possible number (T) of fuzzy regions for each item is set at 4. The actual number of membership functions of each item will be derived by the proposal mining algorithm. Four populations are randomly generated, each for one item. Assume the population size is 10 in this example. Each population then includes 10 individuals. Each individual in the first population is a set of membership functions for item *milk*. Similarly, an individual in the other populations is a set of membership functions respectively for bread, cookies, and beverage.

Each set of membership functions for an item is encoded into a chromosome according to the proposed representation. Assume the ten individuals in each of the four populations are randomly generated. The fitness value of each chromosome is then calculated. Take the chromosome C_1 in $Population_3$ as an example. The membership functions in C_1 for *cookies* are represented as (1 1 1 1, 0 3 5, 3 5 10, 6 13 16, 15 20 20). The quantitative value of each item in each transaction datum is transformed into a fuzzy set according to the active membership functions represented by that chromosome. Take the first item in transaction *T1* as an example. The contents of *T1* include (milk, 5), (bread, 10), (cookies, 7), and (beverage, 7). The amount "7" of item *cookies* is then converted into the fuzzy set:

$$(\frac{0}{cookies.Low} + \frac{0.6}{cookies.LowMiddle} + \frac{0.14}{cookies.MiddleHigh} + \frac{0}{cookies.High})$$

by using the membership functions in C_1 in $Population_3$. The fuzzy count of any fuzzy region is checked against the predefined minimum support value α. Assume in this example, α is set at 0.25. Two large 1-itemset, *cookies.LowMiddle* and *cookies.MiddleHigh*, are thus derived from the membership functions of C_1 in $Population_3$. The fuzzy support of *cookies.LowMiddle* and *cookies.MiddleHigh* are 0.266 and 0.31. The suitability of C_1 is calculated as *overlap_factor*(C_1) + *coverage_factor*(C_1) + *usage_factor*(C_1) = 3 (= (0 + 0 + 0 + 0 + 0 + 0) + 1 + 2). The fitness value of C_1 is thus (0.266 + 0.31)/3 (= 0.192). The fitness values of all the chromosomes in the four populations are calculated with their results shown in Table 2.

Table 2. The fitness values of all the chromosomes in the four initial populations

$Population_1$	***f***	**$Population_2$**	***f***
C_1	0	C_1	0.286
C_2	0.084	C_2	0.104
C_3	0	C_3	0.177
C_4	0.057	C_4	0.200
C_5	0	C_5	0
C_6	0.043	C_6	0.253
C_7	0	C_7	0.070
C_8	0	C_8	0.242
C_9	0	C_9	0.183
C_{10}	0	C_{10}	0.074
$Population_3$	***f***	**$Population_4$**	***f***
C_1	0.192	C_1	0.049
C_2	0.073	C_2	0.075
C_3	0.077	C_3	0.065
C_4	0.240	C_4	0
C_5	0.066	C_5	0.044
C_6	0.044	C_6	0.062
C_7	0	C_7	0.058
C_8	0.065	C_8	0.060
C_9	0	C_9	0.060

C_{10}	0.214	C_{10}	0.083

The crossover and mutation operators are then executed on the populations to generate possible offspring. The best ten chromosomes in each population are then selected as the next generation. The same procedure is then executed until the termination criterion is satisfied. The best chromosome (with the highest fitness value) is then output as the membership functions for deriving fuzzy association rules. After the evolutionary process terminates, the final set of membership functions for each item is shown in Figure 7.

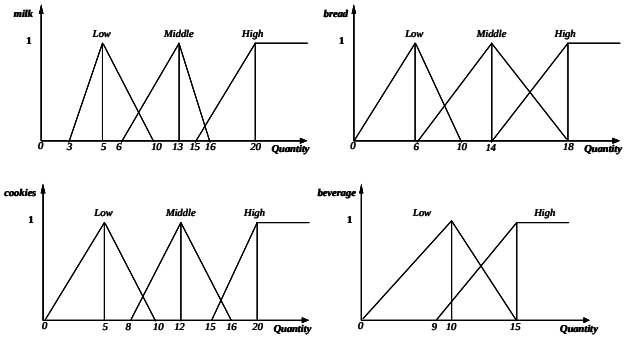


Figure 7: The final set of membership functions

After the membership functions are derived, the fuzzy mining method proposed in [5] is then used to mine fuzzy association rules from the quantitative database.

7. Experimental Results

In this section, experiments made to show the performance of the proposed approach are described. They were implemented in Java on a personal computer with Intel Pentium 4 2.00GHz and 256MB RAM. 64 items and 10000 transactions were used in the experiments. In each data set, the numbers of purchased items in transactions were first randomly generated. The purchased items and their quantities in each transaction were then generated. An item could not be generated twice in a transaction. The initial population size P is set at 50, the crossover rate p_c is set at 0.8, and the mutation rate p_m is set at 0.01. The parameter d of the crossover operator is set at 0.35 according to [3] and the minimum support α is set at 400.

After 500 generations, the final membership functions are apparently much better than the original ones. For example, the initial membership functions of some four items among the 64 items are shown in Figure 8.

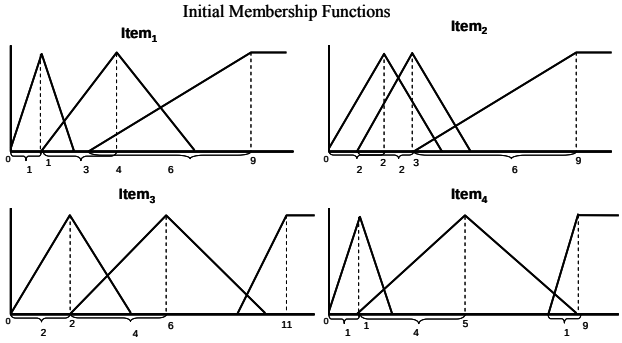


Figure 8: The initial membership functions of some four items

In Figure 8, the membership functions have the bad types of shapes that are defined in the previous section. After 500 generations, the final membership functions for the same four items are shown in Figure 9. It is easily seen that the membership functions in Figure 9 is better than those in Figure 8. The two bad kinds of membership functions don't appear in the final results. The adopted fitness function thus works.

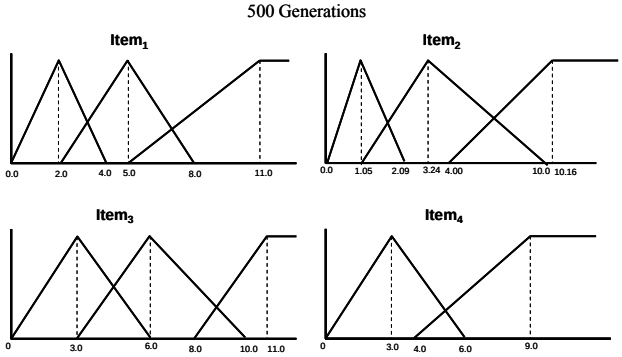


Figure 9: The final membership functions of some four items after 500 generations

The average fitness values of the chromosomes in $population_1$ along with different numbers of generations are shown in Figure 10. As expected, the curve gradually goes upward, finally converging to a certain value. The other populations have similar behavior.

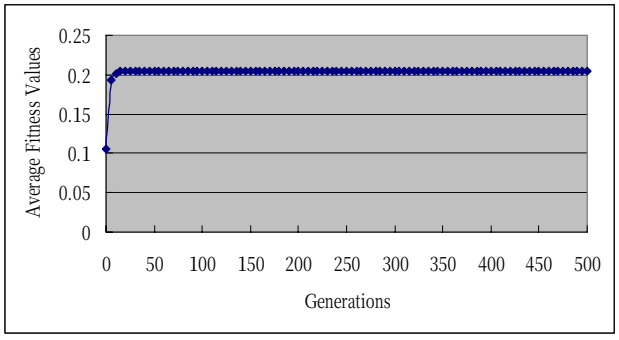


Figure 10: The average fitness values along with different numbers of generations in $population_1$

8. Conclusion and Future Works

In this paper, we have proposed a GA-based fuzzy data-mining algorithm for extracting both association rules and membership functions from quantitative transactions. The number of membership functions for each item is not predefined, but can be dynamically adjusted. Since the fitness of each set of membership functions is evaluated by the fuzzy-supports of the linguistic terms in the large 1-itemsets and the suitability of the derived membership functions, the derivation process can easily be done by the divide-and–conquer strategy. The experimental results show that the proposed fitness function works. Our approach can reduce human experts' intervention during the mining process, thus saving much acquisition time. In the future, we will continuously attempt to enhance the GA-based mining framework for more complex problems.

Acknowledgment

The authors would like to thank Mr. Chien-Shing Chen for his help in making the experiments. This research was supported by the National Science Council of the Republic of China under contract NSC93-2213–E-390-001.

References

[1] R. Agrawal and R. Srikant, “Fast algorithm for mining association rules,” *The International Conference on Very Large Databases*, 1994, pp. 487-499.

[2] O. Cordón, F. Herrera, and P. Villar, “Generating the knowledge base of a fuzzy rule-based system by the genetic learning of the data base,” *IEEE Transactions on Fuzzy Systems*, Vol. 9, No. 4, 2001.

[3] F. Herrera, M. Lozano and J. L. Verdegay, “Fuzzy connectives based crossover operators to model genetic algorithms population diversity,” *Fuzzy Sets and Systems,* Vol. 92, No. 1, pp. 21–30, 1997.

[4] T. P. Hong, C. S. Kuo and S. C. Chi, "Mining association rules from quantitative data", *Intelligent Data Analysis*, Vol. 3, No. 5, 1999, pp. 363-376.

[5] T. P. Hong, C. S. Kuo and S. C. Chi, "Trade-off between time complexity and number of rules for fuzzy mining from quantitative data," *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems,* Vol. 9, No. 5, 2001, pp. 587-604.

[6] M. Kaya, and R. Alhajj, “A clustering algorithm with genetically optimized membership functions for fuzzy association rules mining,” *The IEEE International Conference on Fuzzy Systems*, 2003, pp. 881 -886.

[7] R. Srikant and R. Agrawal, “Mining quantitative association rules in large relational tables,” *The 1996 ACM SIGMOD International Conference on Management of Data*, Monreal, Canada, June 1996, pp. 1-12.

[8] K. S. Tang, K.F. Man, A. F. Liu and S. Kwong, “Mining fuzzy memberships and rules using hierarchical genetic algorithms,” *IEEE Transactions on Industrial Electronics*, Vol. 45, No. 1, 1998, pp.162 - 169

[9] C. H. Wang, T. P. Hong and S. S. Tseng, “Integrating fuzzy knowledge by genetic algorithms,” *IEEE Transactions on Evolutionary Computation,* Vol. 2, No.4, pp. 138-149, 1998.

[10] C. H. Wang, T. P. Hong and S. S. Tseng, “Integrating membership functions and fuzzy rule sets from multiple knowledge sources,” *Fuzzy Sets and Systems,* Vol. 112, pp. 141-154, 2000.

[11] W. Wang and S. M. Bridges, “Genetic algorithm optimization of membership functions for mining fuzzy association rules,” *The International Joint Conference on Information Systems, Fuzzy Theory and Technology*, 2000, pp. 131-134.

Fuzzy Probability Approximation Space and Its Information Measures

Qinghua Hu, Daren Yu
Harbin Institute of Technology, China

Abstract

Rough set theory has attracted much attention in modeling with imprecise and incomplete information. A generalized approximation space, called fuzzy probability approximation space has been proposed by introducing probability into fuzzy approximation space. The novel definition combines three types of uncertainty into a model. Information or knowledge is considered as a partition of the universe in rough set framework. We introduce novel entropy to measure knowledge quantity implied in fuzzy probability approximation space. It's shown that the information measure for fuzzy probability approximation space is a rational extension of the Shannon's one and it will degrade to Shannon's entropy in case where attributes are nominal and objects are equality-probable. Then a uniform information measure for Pawlak's rough set model, fuzzy rough set model and fuzzy probability rough set model is formed based on Yager's entropy.

1 Introduction

Rough set methodology has been witnessed great success in modeling imprecise and incomplete information. Rough set methodology presents a novel paradigm to deal with uncertainty and has been applied to feature selection [1, 2], knowledge reduction [3], rule extraction [4,5,6], uncertainty reasoning [7,8] and granularity computing [9,10,39,43,44,45].The Pawlak's rough set model doesn't consider uncertainty induced by fuzziness and probability in applications. Some generalizations of Pawlak's model were proposed where fuzzy sets and fuzzy relations exist. Rough set theory and fuzzy set theory were put together, rough-fuzzy sets and fuzzy-rough sets were defined in [11,12]. The properties and axiomatization of fuzzy rough set theory [13-17] were analyzed in detail. And the generalized methods were applied to mining stock price [18], vocabulary for information retrieval [19] and fuzzy decision rules [20, 21].

The normal rough set models, both Pawlak's rough set model and fuzzy rough set model, implicitly take an assumption that the objects are equality-probable. However, in practice it is not necessary that the objects are uniformly distributed. A probability distribution may be defined over U. A theory on probability approximation space or a probability rough set model is desirable in this case.

Given a universe U, a probability distribution on U, and some nominal, real-valued or fuzzy attributes, it's interesting in constructing a measure to compute the discernibility power of a family of attributes or equivalence relations, which can lead to likelihood to compare the knowledge quantity generated by different attributes or relations. It will help us find the important attribute set and redundancy of information system. Shannon [22] defined an information measure of a random variable within the frame of communication theory. Forte and Kampe [23, 24] gave the axiomatic information measure, where the word "information" was associated both to measures of events and measures of partitions and suggested that the uncertainty measure is associated to a family of partitions of a given referential space. In [26, 27]a measure, suitable to operate on

domains over which fuzzy equivalence relations have been defined, was introduced, where the semantics of fuzzy events was taken into account. Uncertainty measure on fuzzy partitions generated by fuzzy equivalence relations was analyzed in documents [28, 29].

In rough set framework, attributes are called knowledge which is used to classify the elements into indiscernible clusters. Knowledge introduced by an attribute set implies in the partitions of a referential universe. More knowledge will lead to a finer partition, and then we can get a more perfect approximation of a subset in universe. Therefore knowledge decreases uncertainty in characterizing the concepts. Diminishment of uncertainty can be considered as an increase of knowledge. In this paper we will unify the representation and use the term "knowledge", instead of uncertainty. First we use Shannon's entropy to compute the knowledge quantity introduced by nominal attributes or crisp equivalence relations, then an extension information measure will be presented, which is suitable for the case where fuzzy attributes or fuzzy relations are defined. Based on the extension, the problem of measuring the information in fuzzy approximation spaces is solved.

The rest of the paper is organized as follows: we will review some definitions about fuzzy rough set model and give fuzzy probability rough set model in section 2. Section 3 introduced an extended information measure for fuzzy equivalence relation and fuzzy partition. Then we apply the proposed information measures to fuzzy probability approximation space section 4. The conclusion is given in section 5.

2 Fuzzy probability approximation space

In this section we will integrate three types of uncertainty — probability, fuzziness and roughness together, and present the definition of fuzzy probability approximation space.

Definition 1 Given a non-empty finite set X, R is a relation defined on X, denoted by a relation matrix $M(R)$:

$$M(R)=\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{pmatrix}$$

where $r_{ij} \in [0, 1]$ is the relation value of x_i and x_j.

R is a fuzzy equivalence relation, if $\forall x, y, z \in X$, $\tilde{R}$ satisfies

1) Reflectivity: $R(x, x) = 1, \ \forall x \in U$;
2) Symmetry: $R(x, y) = R(y, x), \ \forall x, y \in U$;
3) Transitivity: $R(x, Z) \geq \min_{y}\{R(x, y), R(y, z)\}$.

Given arbitrary set X, R is a fuzzy equivalence relation defined on X. $\forall x, y \in X$, some operations on relation matrices are defined as

1) $R_1 = R_2 \ \Leftrightarrow \ R_1(x, y) = R_2(x, y), \ \forall x,y \in X$;
2) $R = R_1 \bigcup R_2 \Leftrightarrow R(x,y) = \max\{R_1(x, y), \ R_2(x, y)\}$;
3) $R = R_1 \bigcap R_2 \Leftrightarrow R(x,y) = \min\{R_1(x, y), \ R_2(x, y)\}$;
4) $R_1 \subseteq R_2 \ \Leftrightarrow \ R_1(x, y) \leq R_2(x, y)$.

A crisp equivalence relation will generate a crisp partition and a fuzzy equivalence relation generates a fuzzy partition.

Definition 2 The fuzzy equivalence classes generated by a fuzzy equivalence relation R is defined as

$$U / \tilde{R} = \{[x_i]_{\tilde{R}}\}_{i=1}^{n},$$

where $[x_i]_{\tilde{R}} = \left\{\frac{r_{i1}}{x_1} + \frac{r_{i2}}{x_2} + \cdots + \frac{r_{in}}{x_n}\right\}$.

Example 1. Given an object set $X = \{x_1, x_2, x_3\}$, R_1 is fuzzy equivalence relation on X as follows:

$$R_1 = \begin{pmatrix} 1 & 0.9 & 0 \\ 0.9 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Then the fuzzy equivalence classes are

$$[x_1]_{R_1} = \left\{\frac{1}{x_1} + \frac{0.9}{x_2} + \frac{0}{x_3}\right\},$$

$$[x_2]_{R_1} = \left\{\frac{0.9}{x_1} + \frac{1}{x_2} + \frac{0}{x_3}\right\},$$

$$[x_3]_{R_1} = \left\{ \frac{0}{x_1} + \frac{0}{x_2} + \frac{1}{x_3} \right\}.$$

Theorem 1. Given arbitrary set X, R is a fuzzy equivalence relation defined on X. The fuzzy quotient set of X by relation R is denoted by X. $\forall x,y \in X$, we have

1) $R(x, y)=0 \Leftrightarrow [x]_R \cap [y]_R = 0$

2) $\underset{x \in X}{\vee} [x]_R = 1$

3) $[x]_R = [y]_R \Rightarrow R(x, y) = 1$

Definition 3 A three-tuple $< U, P, \tilde{R} >$ is a fuzzy probability approximation space (shortly, FPAS) or a fuzzy probability information system (FPIS), where U is a nonempty and finite set, called the universe, P is the probability distribute over U, R is a family of fuzzy equivalence relations defined on U .

Definition 4 Given a fuzzy probability approximation space $< U, P, \tilde{R} >$, $\tilde{X}$ is a fuzzy subset of U. The *lower approximation* and *upper approximation*, denoted by $\underline{\tilde{R}}X$ and $\overline{\tilde{R}}X$, are defined as

$$\begin{cases} m_{\underline{\tilde{R}}X}(x) = \wedge\{m_X(y) \vee (1 - \tilde{R}(x, y)) : y \in U\}, x \in U \\ m_{\underline{\tilde{R}}X}(x) = \vee\{(m_X(y) \wedge \tilde{R}(x, y) : y \in U\}, x \in U \end{cases}.$$

These definitions are the rational extension of some models. Let's derive the other model from these definitions.

Case 1 X is a crisp subset and R is a crisp equivalence relation on U:

$$\begin{aligned} m_{\underline{\tilde{R}}X}(x) = 1 &\Leftrightarrow \forall y \in U, m_X(y) \vee (1 - R(x, y)) = 1 \\ &\Leftrightarrow \forall y \in U : y \notin X \rightarrow (x, y) \notin R \\ &\Leftrightarrow \forall y \notin X \rightarrow y \notin [x]_R \\ &\Leftrightarrow [x]_R \subseteq X \end{aligned}$$

$$\begin{aligned} m_{\overline{\tilde{R}}X}(x) = 1 &\Leftrightarrow \exists y \in U : m_X(y) = 1, R(x, y) = 1 \\ &\Leftrightarrow X \cap [x]_R \neq f \end{aligned}$$

In this case these definitions are consistent with Pawlak, rough set model.

Case 2 X is a fuzzy subset of U and R is a crisp equivalence relation on U:

$$\begin{aligned} m_{\underline{R}X}(x) &= \wedge\{m_X(y) \vee (1 - R(x, y)) : y \in U\} \\ &= \wedge\{m_X(y) : R(x, y) = 1\} \\ &= \wedge\{m_X(y) : y \in [x]_R\} \end{aligned}$$

$$\begin{aligned} m_{\overline{R}X}(x) &= \vee\{m_X(y) \wedge R(x, y) : y \in U\} \\ &= \vee\{m_X(y) : R(x, y) = 1\} \\ &= \vee\{m_X(y) : y \in [x]_R\} \end{aligned}$$

In this case, the rough sets are called rough fuzzy sets.

Case 3 X is a subset of U and R is a fuzzy equivalence relation on U:

$$\begin{aligned} m_{\underline{R}X}(x) &= \min\{m_X(y) \vee (1 - R(x, y)) : y \in U\} \\ &= \min_{y \notin X}\{1 - R(x,y)\} \end{aligned}$$

$$\begin{aligned} m_{\overline{R}X}(x) &= \max\{m_X(y) \wedge (1 - R(x, y)) : y \in U\} \\ &= \max_{y \in X} R(x,y) \end{aligned}$$

From the above analysis we can conclude that the definitions of lower and upper approximations of fuzzy set in fuzzy information system are the rational generalizations of classic model. Fuzzy probability information system (FPIS) is the general case of the other rough set model. FPIS will degenerate to the normal fuzzy information system if probability distribution is uniform and fuzzy information system will degenerates to Pawlak's rough set model if equivalence relation is crisp and X is the crisp subset of U.

The membership of an object $x \in U$, belonging to the fuzzy positive region is defined as

$$m_{POS_{\tilde{B}}(d)} = \sup_{X \subseteq U/d} m_{\underline{\tilde{B}}X}(x).$$

Definition 5 Given a fuzzy probability information system <U, P, A>, B and d are two subset of attribute set A, the dependency degree of d to B is defined as

$$g_B(d) = \sum_{x \in U} p(x) m_{POS_B(d)}(x).$$

The difference between fuzzy approximation space and fuzzy probability approximation space is introducing probability distribute over U. This leads to a more general generalization of Pawlak's rough set model. In classic rough set model take the equality-probability assumption.

So $p(x_i)=1/n$, $i=1, 2, \cdots, n$. Then

$$\begin{aligned} g_B(d) &= \sum_{x\in U} p(x) m_{POS_B(d)}(x) \\ &= \frac{1}{n}\sum_{x\in U} m_{POS_B(d)}(x) \\ &= \frac{\sum_{x\in U} m_{POS_B(d)}(x)}{|U|} \end{aligned}$$

This formula is the same as that in fuzzy rough set model [30], which shows that the fuzzy probability approximation space will degrade to a fuzzy approximation space when the equality-probability assumption is taken.

Definition 6 Given a fuzzy information system <*U*, *A*, *V*, *f*>, $B\subseteq A$, $a\in B$, if $U/B=U/(B-a)$, we say knowledge *a* is *redundant* or *superfluous* in *B*. otherwise, we say knowledge *a* is *indispensable*. If any *a* belonging to *B* is *indispensable*, we say *B* is *independent*. If attribute subset $B\subseteq A$ is *independent* and *U/B=U/A*, we say *B* is a *reduct* of *A*.

Definition 7 Given a fuzzy information system <*U*, *A*, *V*, *f*>, $A=C\cup d$. *B* is a subset of *C*. $\forall a\in B$, *a* is redundant in *B* relative to *d* if $g_{B-a}(d)=g_B(d)$, otherwise *a* is indispensable. *B* is independent if $\forall a\in B$ is indispensable, otherwise *B* is dependent. *B* is a subset of *C*. *B* is a reduct of *C* if *B* satisfies:

1) $g_B(d)=g_C(d)$;
2) $\forall a\in B: g_{B\text{-}a}(d)<g_B(d)$.

Comparing the fuzzy probability approximation space with fuzzy approximation space we find that the central difference is in the function of dependency. In fuzzy approximation space, we assume the objects are uniformly distributed and $p(x_i)=1/|U|$. In the fuzzy probability approximation space the probability of x_i is $p(x_i)$. When the probability $p(x_i)=1/|U|$, the fuzzy probability approximation space degrades to a fuzzy approximation space, and if the equivalence relation and the object subset to be approximated are both crisp, we get a Pawlak's approximation space.

In applications the probability can be considered as a weight of the object. Probability is only one of the weighting methods. Weighting gives us a novel dimension to inject information out of data into processing, which can integrate the prior information with data.

3 Information on fuzzy equivalence relations

Shannon's information measure just works in the case where a crisp equivalence relation or a crisp partition is defined, which is suitable for Pawlak's rough set model. In this section we will give a novel formula to compute Shannon's entropy for crisp relation matrix representation, and then a generalization of the entropy is proposed for fuzzy relation matrices. Furthermore, we will present another generalization for probability fuzzy information systems and use the proposed entropies to measure the information in fuzzy probability approximation spaces.

3.1 Shannon's entropy measures in relation matrix form for crisp equivalence relations

Given an information system <*U*, *A*, *V*, *f*>, Arbitrary relation $R\subseteq U\times U\rightarrow\{0, 1\}$ can be denoted by a relation matrix *M*(*R*):

$$M(R)=\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{pmatrix}$$

where r_{ij} is the relation value between element x_i and x_j. If *R* is an equivalence relation we say *M*(*R*) is an equivalence relation matrix.

An equivalence relation matrix satisfies:

1) Reflectivity: $R(x, x)=1, \forall x\in U$;
2) Symmetry: $R(x, y)=R(y, x), \forall x, y\in U$;
3) Transitivity: $R(x, y)=1, R(y, z)=1 \Rightarrow R(x, z)$.

Given an arbitrary set *X*, $R\subseteq X\times X$, $\forall x, y\in X$, some operations on relation matrix are defined as

1) $R_1=R_2 \Leftrightarrow R_1(x, y)=R_2(x, y), \forall x,y\in X$;

2) $R = R_1 \bigcup R_2 \Leftrightarrow R(x,y) = \max\{R_1(x,y),\ R_2(x,y)\}$;

3) $R = R_1 \bigcap R_2 \Leftrightarrow R(x,y) = \min\{R_1(x,y),\ R_2(x,y)\}$;

4) $R_1 \subseteq R_2 \Leftrightarrow R_1(x,y) \le R_2(x,y)$.

There are some properties between crisp attribute set and relations induced by the corresponding attributes:

1) $A = B \Rightarrow R_A = R_B$;

2) $A \supseteq B \Rightarrow R_A \subseteq R_B$;

3) $C = A \bigcup B \Rightarrow R_C = R_A \bigcap R_B$.

The equivalence class contained x_i with respect to relation R is denoted by

$$[x_i]_R = \left\{ \frac{r_{i1}}{x_1} + \frac{r_{i2}}{x_2} + \cdots + \frac{r_{in}}{x_n} \right\}$$

where $r_{ij} = 0$ or 1. The cardinality of $[x_i]_R$ is defined as

$$|[x_i]_R| = \sum_{j=1}^{n} r_{ij} .$$

Definition 8 Given an information system <*U, A, V, f*>, arbitrary equivalence relation R on U, denoted by a relation matrix $M(R)$, then we define the information measure for relation R as

$$H(R) = -\frac{1}{n} \sum_{i=1}^{n} \log \lambda_i ,$$

where $\lambda_i = \frac{|[x_i]_R}{n}$.

Theorem 2 Given an information system <*U, A, V, f*>, $B \subseteq A$, R_B is an equivalence relation generated by attributes B on U. $H(B)$ is computed as Shannon's one and $H(R_B)$ is computed as definition 8. Then $H(B) = H(R_B)$.

Proof. Straightforward.

Example 3. Assumed there are an information system with three objects, An equivalence relation matrix defined on the universe is

$$M(R) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

The equivalence classes are $\{x_1, x_2\}$ and $\{x_3\}$. Then the information quantity is

$$H(R) = -\frac{1}{3}\log\frac{2}{3} - -\frac{1}{3}\log\frac{2}{3} - \frac{1}{3}\log\frac{1}{3} = -\frac{2}{3}\log\frac{2}{3} - \frac{1}{3}\log\frac{1}{3} .$$

The computation is the same as Shannon' one in this case.

Theorem 3 Given an information system <*U, A, V, f*>, $E, B \subseteq A$, R_E, R_B is two equivalence relation generated by attributes E and B. $[x_i]_E$ and $[x_i]_B$ is the equivalence classes induced by E and B. The joint entropy of E and B is

$$H(EB) = H(R_E R_B) = -\frac{1}{n} \sum_{i=1}^{n} \log \frac{|[x_i]_E \bigcap [x_i]_B|}{n}$$

Theorem 4 Given an information system <*U, A, V, f*>, $E, B \subseteq A$, R_E, R_B is two equivalence relation generated by attributes E and B. $[x_i]_E$ and $[x_i]_B$ is the equivalence classes induced by E and B. The conditional entropy E conditioned to B $H(E|B)$ is

$$H(E|B) = H(R_E | R_B) = -\frac{1}{n} \sum_{i=1}^{n} \log \frac{|[x_i]_E \bigcap [x_i]_B|}{|[x_i]_B|} .$$

Here the novel computational formulae of Shannon's information will bring great advantage to generalize them to fuzzy cases.

3.2 Information measure on fuzzy equivalence relations

As we know, fuzziness exists in many real-world applications. Pawlak's rough set model just works in the crisp case. D. Dubois etc. generalized the model to the fuzzy case. In this section we will present a generalization of Shannon's entropy. The novel measure has a same form as Shannon's one and can work in the case where a fuzzy equivalence relation is defined.

Given a finite set U, $\tilde{A}$ is a fuzzy or real-valued

attribute set, which generates a fuzzy equivalence relation $\tilde{R}_{\mathrm{A}}$ on *U*. The fuzzy relation matrix $M(\tilde{R}_{\mathrm{A}})$ is denoted by

$$M(\tilde{R}_{\mathrm{A}})=\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{pmatrix}$$

where $r_{ij} \in [0, 1]$ is the relation value of x_i and x_j.

Definition 9 The fuzzy quotient set generated by the fuzzy equivalence relation is defined as

$$U / \tilde{R} = \{[x_i]_{\tilde{R}}\}_{i=1}^{n}$$

where $[x_i]_{\tilde{R}} = \left\{ \frac{r_{i1}}{x_1} + \frac{r_{i2}}{x_2} + \cdots + \frac{r_{in}}{x_n} \right\}$.

Definition 10 The cardinality $|[x_i]_{\tilde{R}}|$ of $[x_i]_{\tilde{R}}$ is defined as

$$|[x_i]_{\tilde{R}}| = \sum_{j=1}^{n} r_{ij} \ .$$

As shown in example 1, $[x_1]_{R_1} = \left\{ \frac{1}{x_1} + \frac{0.9}{x_2} + \frac{0}{x_3} \right\}$ then $|[x_1]| = 1 + 0.9 + 0 = 1.9$.

Definition 11 Information quantity of the fuzzy attribute set or the fuzzy equivalence relation is defined as

$$H(\tilde{A}) = H(\tilde{R}_A) = -\frac{1}{n} \sum_{i=1}^{n} \log \lambda_i \ ,$$

where $\lambda_i = \frac{|[x_i]_{\tilde{R}}|}{n}$.

This measure has the same form as the Shannon's one defined as definition 8. But it has been generalized to the fuzzy case.

The formula of information measure forms a map: $H : R \to \Re^{+}$, where *R* is a equivalence relation matrix, $\Re^{+}$ is the non-negative real-number set. This map builds a foundation on that we can compare the discernibility power, partition power or approximating power of multiple fuzzy equivalence relations. Entropy value increases monotonously with the discernibility power or the knowledge's fineness. So the finer partition is, the greater entropy is, and the more significant attribute set is.

Definition 12 Given a fuzzy information system <*U*, $\tilde{A}$ *V, f*>, $\tilde{A}$ is the fuzzy attribute set. $\tilde{B}$, $\tilde{E}$ are two subsets of $\tilde{A}$. $[x_i]_{\tilde{B}}$ and $[x_i]_{\tilde{E}}$ are fuzzy equivalence classes containing x_i generated by $\tilde{B}$, $\tilde{E}$, respectively. The joint entropy of $\tilde{B}$ and $\tilde{E}$ is defined as

$$H(\tilde{E}\tilde{B}) = H(\tilde{R}_E \tilde{R}_B) = -\frac{1}{n} \sum_{i=1}^{n} \log \frac{|[x_i]_{\tilde{E}} \cap [x_i]_{\tilde{B}}|}{n}$$

Definition 13 Given a fuzzy information system <*U*, $\tilde{A}$ *V, f*>, $\tilde{A}$ is the fuzzy attribute set. $\tilde{B}$, $\tilde{E}$ are two subsets of $\tilde{A}$. $[x_i]_{\tilde{B}}$ and $[x_i]_{\tilde{E}}$ are fuzzy equivalence classes containing x_i generated by $\tilde{B}$, $\tilde{E}$, respectively. The conditional entropy of $\tilde{E}$ conditioned to $\tilde{B}$ is defined as

$$H(\tilde{E} \mid \tilde{B}) = -\frac{1}{n} \sum_{i=1}^{n} \log \frac{|[x_i]_{\tilde{E}} \cap [x_i]_{\tilde{B}}|}{|[x_i]_{\tilde{B}}|}$$

Theorem 5 $H(\tilde{E}|\tilde{B}) = H(\tilde{B}\tilde{E}) - H(\tilde{B})$

Theorem 6

1) $H(\tilde{R}_A) \geq 0$, "=" holds if and only if $r_{ij} = 1$, $\forall i, \forall j$;
2) $H(\tilde{R}_A \tilde{R}_B) \geq \max\{H(\tilde{R}_A),\ H(\tilde{R}_B)\}$;
3) $\tilde{R}_A \subseteq \tilde{R}_B \Leftrightarrow H(\tilde{R}_A \tilde{R}_B) = H(\tilde{R}_A)$.
4) $\tilde{R}_A \subseteq \tilde{R}_B \Leftrightarrow H(\tilde{R}_B \mid \tilde{R}_A) = 0$

Proof. Straightforward.

3.3 Information measures on fuzzy probability equivalence relation

Shannon's entropy and the proposed measure work on the assumption that all the objects are equality-probable. In this section we will give a generalization where a probability distribution is defined on *U*.

Given a fuzzy probability information system <*U*, $\tilde{A}$ *V, f, P*>, $\tilde{A}$ is the fuzzy attribute set , which generates a family of fuzzy equivalence relations on *U*, *P* is the probability distribution over *U*, $p(x_i)$ is the probability of object x_i. An arbitrary fuzzy equivalence relation $\tilde{R}_B \subseteq U \times U$ generated by attributes $\tilde{B}$ is denoted by a relation matrix $M(\tilde{R}_B)$:

$$M(\tilde{R}_B) = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{pmatrix}$$

where $r_{ij} \in [0, 1]$ is the relation value of x_i and x_j. The fuzzy quotient set by the fuzzy equivalence relation is.

$U / \tilde{R} = \{[x_i]_{\tilde{R}}\}_{i=1}^{n}$, where $[x_i]_{\tilde{R}} = \left\{ \frac{r_{i1}}{x_1} + \frac{r_{i2}}{x_2} + \cdots + \frac{r_{in}}{x_n} \right\}$.

Definition 14 The expected cardinality λ_i of a fuzzy equivalence class $[x_i]_{\tilde{R}}$ is defined as

$$\lambda_i = \sum_{j=1}^{n} p(x_j) \cdot r_{ij}$$

Definition 15 The information quantity of fuzzy attribute set $\tilde{B}$ or fuzzy equivalence relation $\tilde{R}_B$ is defined as

$$H(\tilde{B}, P) = -\sum_{i=1}^{n} p(x_i) \log \lambda_i$$

This measure is identical with Yager's entropy [26] in the form, but different in the goal. The information measure we give is to compute the discernibility power of a fuzzy attribute set or a fuzzy equivalence relation where a probability distribute is defined on *U*. while Yager's entropy is to measure the semantics of a fuzzy similarity relation.

Definition 16 Given a fuzzy information system <*U*, $\tilde{A}$ *V, f, P*>, $\tilde{A}$ is the fuzzy attribute set, *P* is the probability distribution on *U*. $\tilde{B}$, $\tilde{E}$ are two subsets of $\tilde{A}$. $[x_i]_{\tilde{B}}$ and $[x_i]_{\tilde{E}}$ are fuzzy equivalence classes containing x_i generated by $\tilde{B}$, $\tilde{E}$, respectively. The fuzzy equivalence relations induced by $\tilde{B}$, $\tilde{E}$ are denoted by $\tilde{R}$ and $\tilde{S}$ .The joint entropy of $\tilde{B}$ *and* $\tilde{E}$ is defined as

$$H(\tilde{E}\,\tilde{B}, P) = H(\tilde{R}\,\tilde{S}, P) = -\sum_{i=1}^{n} p(x_i) \log \hbar_i ,$$

where $\hbar_i = \sum_{j=1}^{n} p(x_j)(r_{ij} \wedge s_{ij})$.

Definition 17 The conditional entropy of $\tilde{E}$ conditioned to $\tilde{B}$ is defined as

$$H(\tilde{E} \mid \tilde{B}, P) = -\sum_{i=1}^{n} p(x_i) \log \frac{\hbar_i}{\lambda_i} ,$$

where $\lambda_i = \sum_{j=1}^{n} p(x_j) \cdot r_{ij}$ and $\hbar_i = \sum_{j=1}^{n} p(x_j)(r_{ij} \wedge s_{ij})$.

Theorem 7 $H(\tilde{E}|\tilde{B}, P) = H(\tilde{B}\tilde{E}, P) - H(\tilde{B}, P)$

Proof. $H(\tilde{B}\tilde{E}, P) - H(\tilde{B}, P)$

$$\begin{aligned} &= -\sum_{i=1}^{n} p(x_i) \log \hbar_i - (-\sum_{i=1}^{n} p(x_i) \log \lambda_i) \\ &= -\sum_{i=1}^{n} p(x_i) \log \frac{\hbar_i}{\lambda_i} \\ &= H(\tilde{E} \mid \tilde{B}, P) \end{aligned}$$

The forms of the proposed information measures are identical with that of Shannon's ones, and they can be used to measure the information generated by a fuzzy attribute set, a fuzzy equivalence relation or a fuzzy partition. In the follows, the proposed information measures will be applied to fuzzy probability approximation Space.

4. Information measures on fuzzy probability

approximation space

The above section presents an information measure for fuzzy equivalence relations when a probability distribution is defined. Here we will apply it to the fuzzy probability approximation space.

Theorem 8 Given a fuzzy probability information system $<U, \tilde{A}, V, f, P>$, $\tilde{A}$ is the fuzzy attribute set, P is the probability distribution on U. $\tilde{B}, \tilde{E}$ are two subsets of $\tilde{A}$. $[x_i]_{\tilde{B}}$ and $[x_i]_{\tilde{E}}$ are fuzzy equivalence classes containing x_i generated by $\tilde{B}, \tilde{E}$, respectively. The fuzzy equivalence relations induced by $\tilde{B}, \tilde{E}$ are denoted by $\tilde{R}$ and $\tilde{S}$, respectively. Then we have:

1) $\forall \tilde{B} \subseteq \tilde{A}: H(\tilde{B}, P) \geq 0$;

2) $H(\tilde{E}\,\tilde{B}, P) \geq \max\{H(\tilde{E}, P), H(\tilde{B}, P)\}$

3) $\tilde{B} \supseteq \tilde{E}$ or $\tilde{R}_B \subseteq \tilde{R}_E$: $\mathrm{H}(\tilde{B}\tilde{E}, P) = H(\tilde{B}, P)$

4) $\tilde{B} \supseteq \tilde{E}$ or $\tilde{R}_B \subseteq \tilde{R}_E$: $H(\tilde{E} \mid \tilde{B}, P) = 0$

Theorem 9 Given a fuzzy probability information system $<U, \tilde{A}, V, f, P>$, $\tilde{B} \subseteq \tilde{A}$, $a \in \tilde{B}$, $H(\tilde{B}, p) = H(\tilde{B} - a, p)$ if a is redundant; $H(\tilde{B}, p) > H(\tilde{B} - a, p)$ if $\tilde{B}$ is independent. $\tilde{B}$ is a reduct if $\tilde{B}$ satisfies:

1) $H(\tilde{B}, p) = H(\tilde{A}, p)$

2) $\forall a \in \tilde{B}: H(\tilde{B}, p) > H(\tilde{B} - a, p)$

Definition 18. The significance of an attribute a in B is defined as

$$SIG(a, \tilde{B}) = H(\tilde{B}, p) - H(\tilde{B} - a, p).$$

Theorem 10 Given a fuzzy probability information system $<U, \tilde{A}, V, f, P>$, $\tilde{A} = \tilde{C} \cup \tilde{d}$. $\tilde{B}$ is a subset of $\tilde{C}$. $\forall a \in \tilde{B}$, $H(d \mid \tilde{B} - a, p) = H(d \mid \tilde{B}, p)$ if a is redundant in $\tilde{B}$ relative to d; $H(d \mid \tilde{B} - a, p) > H(d \mid \tilde{B}, p)$ if $\tilde{B}$ is independent. $\tilde{B}$ is a reduct of $\tilde{C}$ relative to $\tilde{d}$ if $\tilde{B}$ satisfies:

1) $H(\tilde{B} \mid \tilde{d}, p) = H(\tilde{C} \mid \tilde{d}, p)$;

2) $\forall a \in \tilde{B}: H(d \mid \tilde{B} - a, p) > H(d \mid \tilde{B}, p)$.

Definition 19. The relative significance of an attribute a in B is defined as

$$SIG(a, \tilde{B}, d) = H(d \mid \tilde{B} - a, p) - H(d \mid \tilde{B}, p).$$

5. Conclusions

The contribution of the paper is two-fold. On one side, we generalize the fuzzy approximation space to fuzzy probability approximation space by introducing a probability distribution on U. Furthermore, we propose novel information measures on fuzzy equivalence relations to compute the information quantity in fuzzy probability approximation space.

The proposed fuzzy probability approximation space combines three types of uncertainty: randomicity, fuzziness and roughness together. It's shown that the fuzzy probability approximation space will degrade to fuzzy approximation space when the equality-probability assumption holds. If equivalence relations and the subset to be approximated both are crisp, then approximation space is Pawlak's one. The proposed measures integrate fuzziness, probability with roughness, which is showed a rational generalization of other cases. The methods to measure information in Pawlak's approximation space, fuzzy approximation space and fuzzy probability approximation space are presented in uniform forms based on the generalizations.

Reference

[1] W. Swiniarski, Roman; Hargis, Larry. Rough sets as a front end of neural-networks texture classifiers. Neurocomputing Vol. 36, No. 1-4, 2001, pp. 85-102

[2] W. Swiniarski; A. Skowron. Rough set methods in feature selection and recognition. Pattern Recog. Letters. Vol. 24, No.6, 2003, pp. 833-849

[3] Mi, Ju-Sheng; Wu, Wei-Zhi; Zhang, Wen-Xiu. Approaches to knowledge reduction based on variable precision rough set model. Information Sciences Volume: 159, Issue: 3-4, February 15, 2004, pp. 255-272

[4] Tsumoto, Shusaku. Automated extraction of hierarchical decision rules from clinical databases

using rough set model. Expert Systems with Applications Volume: 24, Issue: 2, 2003, pp. 189-197

[5] N. Zhong; J. Dong; S. Ohsuga. Rule discovery by soft induction techniques. Neurocomputing Volume: 36, Issue: 1-4, February, 2001, pp. 171-204

[6] T. P. Hong; L. Tseng; S. Wang. Learning rules from incomplete training examples by rough sets. Expert Systems with Applications. Vol. 22, No. 4, 2002, pp.285-293

[7] Polkowski, L.; Skowron, A. Rough Mereology: A New Paradigm for Approximate Reasoning. Intern. Journal of Approximate Reasoning Vol. 15,No. 4, 1996, 333-365

[8] Pawlak, Zdzislaw. Rough sets, decision algorithms and Bayes' theorem. European Journal of Operational Research Volume: 136, Issue: 1, 2002, pp. 181-189

[9] Pawlak, Z. Granularity of knowledge, indiscernibility and rough sets. Proceedings of 1998 IEEE intern. Conf. on fuzzy systems, 106-110, 1998

[10] Zadeh, L.A. Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. fuzzy sets and systems, 19, 111-127, 1997

[11] D. Dubois, H. Prade. Rough fuzzy sets and fuzzy rough sets. International Journal of general systems. 17 (2-3),1990, 191-209

[12] D. Dubois, H. Prade. Putting fuzzy sets and rough sets together, in: R. Slowiniski (Ed.), Ittelligent Decision support, Kluwer Academic, Dordrecht, 1992, 203-232

[13] Morsi, Nehad N.; Yakout, M.M. Axiomatics for fuzzy rough sets. Fuzzy Sets and Systems Volume: 100, Issue: 1-3, November 16, 1998, pp. 327-342

[14] R. Anna Maria; Kerre, Etienne E. A comparative study of fuzzy rough sets. Fuzzy Sets and Systems Volume: 126, Issue: 2, March 1, 2002, pp. 137-155

[15] Wu, Wei-Zhi; Mi, Ju-Sheng; Zhang, Wen-Xiu. Generalized fuzzy rough sets. Information Sciences Volume: 151, May, 2003, pp. 263-282

[16] W. Wu; W. Zhang. Constructive and axiomatic approaches of fuzzy approximation operators. Information Sciences Vol.159, No.3-4, 2004, 233-254

[17] Mi, Ju-Sheng; Zhang, Wen-Xiu. An axiomatic characterization of a fuzzy generalization of rough sets. Information Sciences Vol.160, No.1-4, 2004, 235-249

[18] Wang, Yi-Fan. Mining stock price using fuzzy rough set system. Expert Systems with Applications Volume: 24, Issue: 1, 2003, pp. 13-23

[19] S. Padmini; R. Miguel et al . Vocabulary mining for information retrieval: rough sets and fuzzy sets. Information Processing and Management. Vol. 37, No. 1, 2001, pp. 15-38

[20] Fernández Salido, J.M.; Murakami, S. Rough set analysis of a general type of fuzzy data using transitive aggregations of fuzzy similarity relations. Fuzzy Sets and Systems Vol. 139, No. 3, 2003, pp. 635-660

[21] Q. Shen and A. Chouchoulas. A rough-fuzzy approach for generating classification rules. Pattern Recognition, 35(11):2425-2438, 2002

[22] C. Shannon, W. Weaver. The mathematical theory of communication, university of Illinois press, Champaign, IL, 1964

[23] B. Forte. Measure of information: the general axiomatic theory, RIRO R2 3 (1969) 63-90

[24] J.Kampe de Feriet, B. Forte. Information etc probabilite CRAS Paris, Ser A 265(1967) 110-114,143-146

[25] L. Zadeh, Probability measures of fuzzy events, J.Math. Anal. Appl. 23 (1968) 421-427

[26] R. Yager. Entropy measures under similarity relations. Internat. J. General systems 20 (1992) 341-358

[27] E. Hernandez, J. Recasens. A reformulation of entropy in the presence of indistinguishability operators. Fuzzy sets and systems. 128 2002 185-196

[28] Radko Mesiar, Jan Rybarik. Entropy of fuzzy

partitions: a general model. Fuzzy sets and systems 99, 1998, 73-79

[29] Carlo Bertoluzza, Viviana Doldi, Gloria Naval. Uncertainty measure on fuzzy partitions. Fuzzy sets and systems 142(2004) 105 -116

[30] Richard Jensen, Qiang Shen. Fuzzy-rough attribute reduction with application to web categorization. Fuzzy sets and systems. 141 (2004) 469-485

[31] Peters, J.F.; Pawlak, Z.; Skowron, A.; A rough set approach to measuring information granules. 2002 Computer Software and Applications Conference, Proceedings.pp1135 – 1139

[32] Slowinski, R.; Vanderpooten, D.; A generalized definition of rough approximations based on similarity. Knowledge and Data Engineering, IEEE Transactions on , Vol. 12 , No.2 , 2000, 331 – 336

[33] Q. Shen, A. Chouchoulas. A Modular Approach to Generating Fuzzy Rules with Reduced attributes for the monitoring of Complex Systems. Engineering applications of artificial Intelligence, vol.13, pp.263-278,2000

[34] Z. Pawlak. Rough classification. Intern. J. human-computer studies. 1999,51, 369-383

[35] R.Yager. On the Entropy of Fuzzy Measures. IEEE Trans. on fuzzy systems. Vol. 8, No. 4, 2000, 453-461

[36] R.Yager. Uncertainty Representation Using Fuzzy Measures. IEEE Transaction on systems, man and cybernetics — Cybernetics, Vol.32, No.1, 13-20, 2002

[37] Greco, S., Matarazzo, B., Słowiński, R.: Rough approximation by dominance relations, International Journal of Intelligent Systems, 17 (2002) no. 2, 153-171

[38] Greco, S., Matarazzo, B., Słowiński, R.: Rough sets methodology for sorting problems in presence of multiple attributes and criteria. European J. of Operational Research, 138 (2002), no. 2, 247-259

[39] Y. T. Yao and Y.Y. Yao, Induction of Classification Rules by Granular Computing, RSCTC 2002 pp.331-338

[40] W. Pedrycz, Shadowed sets: bridging fuzzy and rough set, in : S. K. Pal, A. Skowron (Eds.), Rough Fuzzy Hybridization: A New trend in Decision Making, Springer, Berlin, 1999

[41] G. Y. Wang, H. Yu, D. C. Yang. Decision table reduction based on conditional information entropy. Chinese J. computers. Vol. 25, No. 7, 2002, pp 1-9

[42] A. Zadeh "Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems," Soft Computing 2 (1998) 23 – 25

[43] Tsau Young Lin: Granular Computing. RSFDGrC 2003: 16-24

[44] T. Y., Lin "From Rough Sets and Neighborhood Systems to Information Granulation and Computing in Words," European Congress on Intelligent Techniques and Soft Computing, September 8-12, 1997, 1602-1606

[45] T. Y. Lin, "Granular Computing on Binary Relations I: Data Mining and Neighborhood Systems." In: Rough Sets In Knowledge Discovery, A. Skowron and L. Polkowski (eds), Physica-Verlag, 1998, 107-121

Document Clustering and Summarization Using Biomedical Ontologies

Xiaohua Hu, Illhoi Yoo, Protima Banerjee
College of Information Science and Technology, Drexel University, 3141 Chestnut Street
Philadelphia, PA, USA, 19104

Abstract - **In this paper we present an ontology-based document clustering and summarization system, called BioLit-CS (Biomedical Literature Clustering and Summarization). The basic idea of our summarization method is first to integrate the ontology concepts into the vector representation of the document set, then to cluster the document set in document vector into topical groups. Within each topical group, saliency scores for key concepts and sentences are calculated based on the mutual reinforcement principle. The key concepts and sentences are ranked according to their saliency scores. Then, some (TOP n) of them are selected for inclusion in the top key concept list and the summaries of the documents. We use Relevance Novelty to minimize the redundancy of summary and to maximize both relevance and diversity for extracted sentences. The experimental results on a yeast gene related document set indicate that our system is very effective at generating a concise and informative summary for multiple documents with multiple topics.**

Index Terms - **ontology, text clustering, text summarization, multi-document summarization, data mining, text mining**

I. Introduction

The rapid electronic dissemination of research breakthrough has greatly accelerated the current pace of genomic and proteomics research. A lot of genomic knowledge and discovery is published and collected in huge biomedical literature databases such as MedLine. The number of articles or abstracts in these databases is growing at an unprecedented rate. Medline is the largest biomedical bibliographics database with more than 12 million abstracts collected from more than 4000 journals in biomedical areas. More than 10,000 documents are added to Medline every week. The sheer size of Medline can be daunting to many scientists involved in biomedical research. Biomedical researchers have suffered from dramatic information overload due to the unprecedented growth of biomedical literatures. One way to catch up with the latest information and to tackle the information overload is the use of a text summarization system, which can generate a semantically concise, coherent and informative summary to help domain experts quickly absorb and assimilate the latest information in their fields.

Generally speaking, there are two approaches in text summarization: text extraction based methods and template based methods. Text extraction based methods, after data preprocessing, extract text based on user's input/interest and/or rank the extracted text (usually sentences) based on some statistical or linguistic measures; a lot of heuristics that are empirically acquired are usually used. Template based methods first manually construct domain-specific templates and then fill the templates from the text. In both methods sentences are used as the basic processing units because a sentence is the discourse unit with the best balance of semantic granularity and self contained cohesiveness [1]. The sentences are ranked based on salient scores. The highly ranked sentences are included into a summary.

In this paper we present an ontology-based Biomedical Literature multi-document clustering and summarization system BioLit-CS. The basic idea of our summarization method is to first integrate the ontology concepts into the word vector representation of the document set, then cluster the documents in the vector representation into a topical group. Within each topical group, saliency scores for key terms and sentences are calculated based on the mutual reinforcement principle [2]. The key terms and sentences are then ranked according to their saliency scores and are selected for inclusion in the top key terms list and summaries of the documents. We use Maximal Marginal Relevance [3], [4] to minimize the redundancy of the extracted sentences in the summary.

The rest of this paper is organized as follows. In section 2 we review some of the related work in multidocument summarization, text clustering and biological ontology. In Section 3, we first introduce the architecture of our system BioLit-CS and then discuss the technical details of ontology-based clustering and summarization. We show the experimental results in Section 4 and conclude with discussion and future research plan in Section 5.

II. Related Work

Here, we review some related works in the multi-document summarization, text clustering and biomedical ontologies fields and provide background information about them.

A. Multidocument Summarization

Text summarization has been studied since Luhn's work [5] in 1958. A lot of approaches have been introduced. For example, there are statistical methods based on the bag-of-words model, linguistic methods using natural language processing, knowledge-based methods using concepts and their relations and summary generation methods. The first three approaches try to seek the most important information (usually sentences or terms) for a condensed version of the

documents while the last approach generates completely a new summary that consists of informative terms, phrases, clauses and sentences. The main difficulty of the last approach is figuring out how to combine them to make sentences that are grammatically correct.

In the bioinformatics field many multi-document summarization systems have also been introduced. TextQuest [6] is designed to summarize documents retrieved in response to a keyword(s) based search on PubMed. However, it does not retain the association between the genes and the retrieved documents. MedMiner [7] can provide summarized literature information on genes but it is limited when finding relations between two genes only. Also, it returns a few hundred sentences for the summary. Shatkay et al. [8] suggested a system, which attempts to find functional relations among genes on a genome-wide scale. However, this system requires the user to specify a representative document for each gene which describes the gene very well. Looking for the representative document may take a lot of time, effort and knowledge on the part of the user. In addition, as genes have multiple biological functions, it is very rare to find a document that covers all aspects of a gene across various biological domains. GEISHA [9] is based on the comparison of the frequency of abstracts linked to different gene clusters. Interpretation by the end user of the biological meaning of the terms is facilitated by embedding them in the corresponding significant sentences and abstracts and by establishing relations with other, equally significant terms.

However, those approaches deal with all of the words in documents except stop words. A main drawback of these approaches is that many (semantically) unimportant words are involved with text summarization so that the quality and the performance of text summarization decrease because those words act as noise on summary processing. Unlike traditional text summarization approaches, the ontology-based text summarization method uses two kinds of ontology concepts: the concepts that are found in documents, and then the concepts that are found to be semantically relevant to those concepts through tracking their relationships. Ontology concepts as semantically salient terms are searched and valued. Therefore, more semantically concise summaries with better semantic meaning are expected.

B. Text Clustering

Existing text clustering solutions use all of the words in the documents except the stop words for their term vectors. Thus, it is not uncommon for such solutions to generate thousands of dimensions in the vector representation of documents. Moreover, they handle terms not semantically but only syntactically; thus, they ignore the similarity of terms and relationships between words such as synonyms, hyponyms and hypernyms defined in terminological resources in ontology. For example, semantically identical but differently spelled words (e.g., cancer, malignant tumor) are treated as completely different words in traditional document clustering approaches. Such term handling hampers document similarity measure processing. A good way to solve such a problem is the use of ontology on document clustering [10]. In our architecture, enriching the term vectors with concepts from ontology has three benefits. Firstly, it naturally resolves the synonym problem. Secondly, it can identify documents with different topic using high level (more general) concepts. Lastly, because the concepts that are found in the documents and the concepts that are relevant to those concepts are used on the vector construction, the dimensions are remarkably reduced, which in turn improves the clustering accuracy and efficiency.

C. Biological Ontologies

Biology researchers have suffered from inconsistent descriptions of gene products and ambiguous term definitions from disparate biology databases. This is called "communication problem" [11], which hampers the semantic computational processing of bioliterature, such as text summarization or document clustering. One of the promising solutions to the problem is the use of ontologies, which have gotten much attention recently in semantic web and bioinformatics communities. This is because ontologies explicitly conceptualize a domain without ambiguity, thus providing better understanding of the domain; they include a structured, controlled vocabulary with definition, the taxonomy of the vocabulary and all of the possible relationships among concepts.

There are many biology/medical ontologies, such as Gene Ontology (GO) [12], UMLS, TAMBI Ontology, EcoCyc Ontology, etc. Each ontology is designed for a specific purpose. For example, GO is about gene product function. Current GO is from the result of the integration of 16 biology databases [12]. GO terms are taxonomically grouped into three areas: molecular function, biological process and cellular component which are considered independent of each other. GO terms are structured in a directed acyclic graph because it is very possible for a gene product that has many molecular functions to be used in many biological processes and to be related to many cellular components. For example, a GO term has relationships with more than 400 GO terms. Although GO terms are in the form of a graph, all GO terms are rooted (hierarchically arranged) to in GO_Ontology concept. However, a GO term may have many parents and/or many children in different levels. In this paper we focus on GO ontology because we will cluster and summarize document set related to genes and gene products.

III. Overview of Bio-CS Architecture

In order to summarize documents properly, two problems should be carefully handled: (1) documents are highly redundant in terms of information; and (2) documents contain various kinds of information. One of the promising approaches for the problems above is to cluster the document set because through document clustering similar documents are grouped together while dissimilar documents are grouped into different document clusters.

Our approach follows this philosophy but with significant enhancement by integrating ontology into the clustering procedure. The principal idea of our approach is based on the usage of ontology for generating alternative representations of the given document set. The benefits of integrating ontology into document clustering and summarization can be three-fold;

- Ontology can clearly identify relationships among terms found in documents. Thus, by using concepts found in documents and their relationships, the document vectors inspired by ontology are able to contain a higher level of semantic meaning as well as the actual semantic meaning. For example, if a document that is relevant to sedan or convertible is converted into a vector representation using proper ontology, the vector representation can hold the higher level meaning of sedan or convertible, e.g. car or vehicle. Therefore, the ontology-based document vector can semantically represent the original document well and uncover the hidden meaning of documents.
- Because only important terms (or ontology terms) are used in the document vector, which reduces the cardinality of document dimensions significantly, the clustering performance can be greatly increased. It should be noted that the vector elements of unimportant terms act as noise on calculating document similarity/distance.
- Using different ontologies of different domains we can cluster documents in different ways because the same documents can be interpreted from different points of views. For instance, using generic ontology and molecular ontology biology documents can be clustered in the generic view or in molecular point of view.

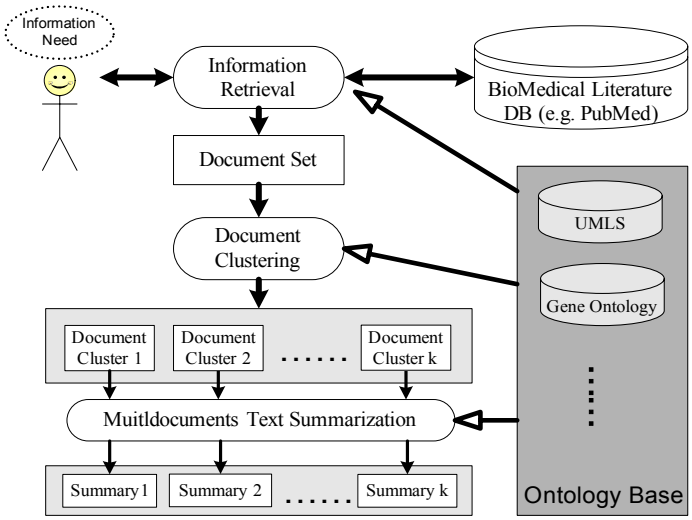


Figure 1. The Architecture of Biomedical Literature Clustering and Summarization (Bio-CS)

The architecture of our system BioLit-CS, which follows the pipeline architecture, is shown in Figure 1. The technical details of major steps are described in the following subsections.

A. Ontology-based Biomedical Literature Clustering

Instead of summarizing the whole document set relevant to a gene cluster, each gene cluster document set is first clustered using a clustering algorithm for better text summarization. Our rationale of document clustering before summarization is that the document set of a gene cluster contains various biological topics because genes in a cluster carry out multiple functions. Thus, clustering the document set guarantees a text summarization system the best input source because each document cluster only consists of similar documents. For better document summarization, document clustering can be a prerequisite.

The traditional document clustering model, such as the bag-of-word model, is often unsatisfactory since the model ignores the relationships among semantically similar terms; for example "car" and "automobile" or "cancer" and "malignant tumor" are treated as completely different terms in the model. Therefore, traditional clustering approach suffers three main problems. First, traditional document clustering is mostly seen as an objective method, which delivers rigidly defined results. This, however, runs contrary to the fact that different people have quite different needs with regard to document clustering because they may view the same documents from completely different perspectives (e.g., a clinical view vs. genetic view). Second, traditional document clustering typically takes place in the high-dimensional space of a word vector whose entries are attributes/properties for a document. However, using unimportant terms as vector entries negatively affects the clustering in high-dimensional spaces on clustering – besides the computational inefficiencies – because each entry is treated as the same regardless of its semantic importance and thus has the same distance from all other data points. Third, traditional document clustering *per se* does not provide an explanation for why a specific document is grouped into a particular cluster.

Our approach deals with those problems by deeply integrating ontology in the clustering procedure. For the first problem using different ontologies from different domains provides multiple subjective perspectives about document clustering into the same document set. The second problem can be easily solved if ontology is used on document clustering because only ontology concepts as importance terms are involved in vector representation; thus, the cardinality of dimensions remarkably decrease. As a result, the performance and the efficiency of document clustering is greatly improved. For the third problem involving high-level concepts in vector representation by analyzing hyponyms/hypernyms relationships among concepts provides reasonable explanation of document clustering because those high-level concepts with much higher salient scores than normal concepts play an important role in distance/similarity measurement of document clustering. The explanation is based on those high-level concepts. Therefore, we expect better clustering performance in terms of the semantic, a computational benefit and flexibility with ontology on document clustering. Consequently, we believe the most important part in document clustering is the conversion from documents into document vectors. Without proper conversion, the document vectors do not represent the original documents well. Ontology plays a crucial part in the conversion.

Our algorithm converts original documents into semantically well-represented document vectors for clustering. This is done by, firstly, calculating global and local measures of ontology concepts found in the documents. Then, each ontology concept is valued based on our own measure (called,

"thorough frequency") that is figured out by its global measure and its descendants' global measures. Finally for some qualified concepts, their parent concepts are involved in document vectors in order to represent high-level concepts of those terms in document vectors. As a result, ontology-based document vector can represent the original documents well by uncovering hidden high-level semantics. The big difference between traditional document vector conversion method and our ontology-based document vector conversion is that our conversion method involves not only terms found in documents in document vectors but also their parent concepts whose values are figured out by sophisticated ontology concept frequency measurement ("thorough frequency" in our term) using the ontology semantic net.

Algorithm: Ontology-based document clustering
Input: Document set (*D*); Ontology; any external clustering algorithm
Output: Document clusters
Procedural:
STEP 1: Calculating concept measures for C_i over documents (d_p)

For $d_p \in D$
  For C_i
    $LF(C_{i_{d_p}})$
    $GF(C_i)$
  End For
End For

STEP 2: Valuing C_i from leaves to the root

For the leaves to the root

$$C_j \in \{C_j \mid PC(C_i, C_j)\}$$

$$TF(C_i) = GF(C_i) + \sum_{C_j \in \{C_j \mid PC(C_i,C_j)\}} GF(C_j)$$

End For

STEP 3: Constructing *DV*

$$DVE = \left\{\{C_i\} + \left\{C_j \mid PC(C_j, \delta_{TF}(C_i))\right\}\right\} \quad (i = 1,...,m)$$

For $d_p \in D$

For $C_i \in \left\{C_i \mid LF(C_{i_{d_p}})\right\}$

$$DV_p \;\&= \left\{TF(C_j \mid PC(C_j, \delta_{TF}(C_i)))\right\} + \left\{LF(C_{i_{d_p}})\right\}$$

  End For
End For

STEP 4: Applying *DV* to a clustering algorithm and Storing clusters to files

In Step 1 for each ontology term (C_i) its measure is calculated over documents. The measure could be one that the Information Retrieval community has used, such as term frequency, document frequency, information gain, Z-score [13], etc. However, TF*IDF should not be used here because we use abstracts of papers, most GO concepts are found only once in the abstracts in which the concepts exist and GO concepts that are found frequently over documents should be regarded as salient concepts; TF*IDF assumes that salient terms are not found too frequently over all documents due to the nature of inversed document frequency (e.g., 'the', 'that', etc) and are found frequently in the documents in which the term exist due to the nature of term frequency.

The frequencies are calculated in both the global (corpus) level and local (document) level. Global frequencies are summed up whenever the same concepts are found. For a document $d_p \in$ D (p = 1,…, n) the global frequency of a concept C_i is defined as:

$$GF(C_i) = \sum_{p=1,...,n} LF(C_{i_{d_p}})$$

where $GF(C_i)$ is the global frequency of a concept C_i and $LF(C_i)$ is the local frequency of a concept C_i and $C_{i_{d_p}}$ is a C_i that is found in d_p. The global frequencies are used for the calculation of thorough frequencies in Step 2.

In Step 2 all the parent level concepts of concepts that are found in documents are valued by their children's global frequencies plus their own global frequencies. For instance, if a concept's *GF* is 5 and the summation of its all children's *GF*s are 10, the new frequency (here, called "thorough frequency" in our term) of the concept is 15. This procedure starts from leaf level to the root. The thorough frequency (*TF*) of a concept (C_i) is mathematically defined as:

$$TF(C_i) = GF(C_i) + \sum_{C_j \in \{C_j \mid PC(C_i,C_j)\}} GF(C_j)$$

where $TF(C_i)$ is the thorough frequency of concept C_i and $PC(C_i, C_j)$ means C_i is the parent level concept of C_j (a child). Through *TF* of a concept we estimate the importance of a concept in documents in terms of semantic. This is feasible because all relationships relevant to a concept are identified through the ontology semantic net.

In Step 3, as the core of this procedure, the document vector is constructed. The vector elements/entries consist of the distinct concepts found in the whole document set plus all parent concepts of qualified concepts. The reason why their all of their parent concepts are also selected is to hold all the semantic meanings in ontology in document vector representation. Thus, document vector elements (*DVE*) are defined as:

$$DVE = \left\{\{C_i\} + \left\{C_j \mid PC(C_j, \delta_{TF}(C_i))\right\}\right\} \quad (i = 1,..,m)$$

where m is the distinct number of concepts found in the corpus and $\delta_{TF}(C_i)$ includes only C_i whose *TF* is bigger than the threshold value. For C_i whose *TF* is smaller than threshold value $\delta_{TF}(C_i)$ outputs nothing and thus, $PC(C_j, \delta_{TF}(C_i))$ also outputs nothing (or an empty set).

Instead of including all parent concepts of all distinct concepts only the parent concepts of qualified concepts are included into document vectors. In order to qualify concepts their thorough frequencies are used because we assume salient

concepts have big enough thorough frequencies; the assumption is based on the fact that semantically salient ontology concepts are frequently found over documents because the documents are related to a gene cluster.

After selecting the concepts to be added to the document vectors we should consider which values should be assigned to the vector elements (selected concepts) as salient scores. For non-parent concepts, their local frequencies are assigned to a vector. For parent concepts as a whole, their thorough frequencies are used; if a local frequency already exists, it is replaced with the thorough frequency. This is defined as:

$$DV_p = \left\{ TF(C_j \mid PC(C_j, \delta_{TF}(C_i))) \right\} + \left\{ LF(C_{i_{d_p}}) \right\} \quad (p = 1, \ldots, n)$$

where n is the number of documents.

The rationale is that, for example, if a document talks about "nucleus", the document is relevant to "intracellular" as an upper level concept and also "cell" as a more upper level concept (see Figure 2). With such information the document vectors can represent the semantics of the original document well. In addition, using such information makes the similarity and dissimilarity of documents clear because parent concepts have more salient values than non-parent concepts. This is possible because all possible relationships among concepts are analyzed. For example, suppose there are five documents about extracelluar, intracellular, membrane, DNA and RNA and those documents are encoded into document vector using proper ontology such as Gene Ontology in the same way in Steps 1, 2 & 3. Table 1 shows local and global frequencies of all concepts. This is a typical document vector conversion of traditional methods except the section of global frequencies. Table 2 shows the ontology-based document vector conversion. Note each frequency is based on the ontology in Figure 2 and document vector in Table 2 contains only local frequency values and thorough frequency values.

Because the high-level (parent) concepts have more frequencies, the documents are easily semantically distinguished by the clustering algorithm; this can be easily explained using Euclidean distance. Only 2 document vector elements (DNA and RNA) mainly affect the calculation of the distance between DNA document and RNA document because they are split in Level 3 (in Figure 2) from the same parent (see Table 2 for the difference between the documents in vector elements). The distance above can be naturally smaller than the distance between extracelluar document and DNA document because they are broken down in Level 2 (in Figure 2); more vector elements and higher frequencies are involved during the distance calculation (see Table 2).

In Step 4 the document vectors are used as the input for any clustering algorithm. Using the clustering results of the clustering algorithm, document clusters are generated. For the document clustering, X-means [14], an extension of K-means, is used because X-means improves two major shortcomings of K-means. It scales better and automatically detects the number of clusters (k problem) using Bayesian Information Criterion.

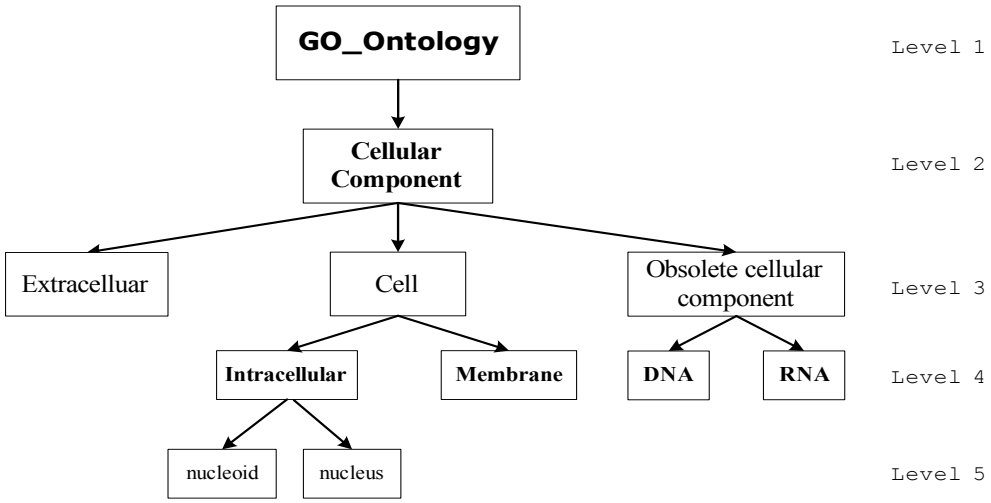


Figure 2. A part of GO Ontology

Table 1. A Document Vector containing Local Frequencies of Concepts

	Extra.	Intra.	Cell	Memb.	nucleoid	nucleus	O.C.C.	DNA	RNA	C.C.	…
Extra. Doc.	6	0	0	0	0	0	0	0	0	0	…
Intra. Doc.	1	5	3	0	3	4	0	0	0	0	…
Memb. Doc.	0	1	3	5	0	0	0	0	0	0	…
DNA Doc.	0	0	0	0	0	0	0	3	1	0	…
RNA Doc.	0	0	1	0	0	0	0	1	4	0	…
Global Frequencies	**7**	**6**	**7**	**5**	**3**	**4**	**0**	**4**	**5**	**0**	**…**

Table 2. A Document Vector containing Local or Thorough Frequencies of Concepts

	Extra.	Intra.	cell	Memb.	nucleoid	nucleus	O.C.C.	DNA	RNA	C.C.	…
Extra. Doc.	**6**	0	0	0	0	0	0	0	0	**41**	…
Intra. Doc.	1	**13**	**25**	0	3	4	0	0	0	**41**	…
Memb. Doc.	0	1	**25**	5	0	0	0	0	0	**41**	…
DNA Doc.	0	0	0	0	0	0	**9**	3	1	**41**	…
RNA Doc.	0	0	1	0	0	0	**9**	1	4	**41**	…

Table 3. A Document Vector containing Local and Thorough Frequencies of Concepts for Full Text Documents

	Extra.	Intra.	Cell	Memb.	nucleoid	nucleus	O.C.C.	DNA	RNA	C.C.	…
Extra. Doc.	26	0	*5*	0	0	0	0	0	*2*	0	…
Intra. Doc.	*1*	15	33	0	30	24	0	*4*	*3*	0	…
Memb. Doc.	*4*	*1*	25	28	0	*2*	0	0	0	0	…
DNA Doc.	*2*	0	*3*	0	*1*	0	0	32	*1*	0	…
RNA Doc.	0	0	*2*	0	0	0	0	*1*	41	0	…

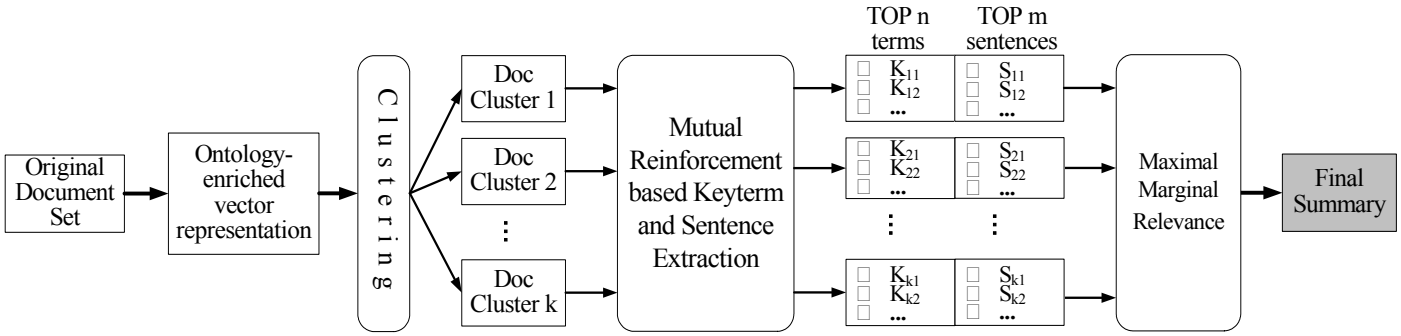


Figure 2. Data flow of Document Clustering and Summarization

One may argue that, for example, Extra. or Intra. document may contain DNA or RNA as an unrelated term which can act as significant noise. The authors agree with that problem. However, our documents are abstracts of papers. Since abstracts are actually summaries of the original documents, it is rare that summaries contain unrelated GO concepts. If the full text is used instead of abstracts, such problem may arise as Table 3 shows (the measures which are italic and underlined are noises). However, this problem can be easily solved; for concepts whose measures are significant over documents, we include their parent concepts in document vectors. Alternatively for each concept Z-score can be used as a measure for this purpose because Z-score indicates the distance from the mean of a distribution normalized by the standard deviation of the distribution. Thus, significant concepts are easily identified from trivial concepts.

B. Ontology-based Biomedical Literature Summarization

After document clustering, as prerequisite for document summarization, each document cluster that is guaranteed to contain homogeneous documents is summarized. In this paper, we use a mutual reinforcement principle [2] to extract key phrase and sentences from the document that are added to the summary. The core of mutual reinforcement principle is that: "a term should have a high saliency score if it appears in many sentences with high saliency scores while a sentence should have a high saliency score if it contains many terms with high saliency score" [2]. We make undirected and weighted bipartite graphs for terms and sentences to extract salient terms and sentences from the graphs on the fly without extensive training process.

For each document cluster, a term set and a sentence set are generated; term set T= $\{t_1, t_2, \ldots, t_n\}$ which includes all terms found in the document set and sentence set S=$\{s_1, s_2, \ldots, s_m\}$ which contains all sentences. Here, bipartite graph is created between term set and sentence set. If a sentence (s_m) contains a term (t_n), an edge is created between s_m and t_n. The elements of two sets are vertices. Each edge may be weighted by the number of relationships between a sentence and a term or by a more elaborated measure (e.g., TF*IDF). A weight w_{mn} indicates the weight on edge between s_m and t_n. Fundamentally the merit of a sentence depends on the terms the sentence contains and the merit of a term relies on the sentences that include the term. The following mathematically represents this principle.

$$Merit(s_m) = \sum_{t_n \in \{t_n \mid edge(s_m, t_n)\}} w_{mn} \quad \text{and}$$

$$Merit(t_n) = \sum_{s_m \in \{s_m \mid edge(s_m, t_n)\}} w_{mn}$$

The function edge(s_m,t_n) indicates a sentence s_m contains a term t_n. This iterative process continues until it reaches a certain number of iterations. Finally TOP n terms and sentences are selected based on their salient scores and added to the summary. There are a lot of numerical computation methods developed to calculate the scores of terms and sentences efficiently. For more detailed discussion, please refer to [16].

Here, we need to take one more step to deal with "summary sentence redundancy". It is very possible that the newly extracted sentences to be added to summary are semantically similar to the previously extracted sentences. Extracting all similar sentences would produce a verbose and repetitive summary. The sentence extraction part of our system is similar to the domain-independent multidocument summarization in [3,4,17] in the way it clusters sentences across documents to help determine which sentences are central to the collection, as well as to reduce redundancy among sentences as it does not make use of comparisons to the centroids of the multidocument set. We will integrate the ideas from Maximum Marginal Relevance measure [3,4] and Cross Sentence Information Subsumption (CSIS) [17] to minimize redundancy and maximize both relevance and diversity for extracted sentences. In order to measure the similarities between two sentences (S_i={k_{i1}, k_{i2},..., k_{ip}} and S_j={k_{j1}, k_{j2},..., k_{jq}}) term sets are generated for each sentence. And then every two terms from different term sets are compared. If two terms are exactly the same, the similarity score is 1. If two terms are different but they are related in the ontology, the similarity score is dependent on the semantic similarity in the ontology. There are many approaches to use the distance between two concepts in ontologies as the basis for their similarity [18]. For example, assuming the commonality between terms k_{iu} and k_{jv} in the ontology is K_p, where K_p is the most specific class that subsumes both k_{iu} and k_{jv} . We can define the semantic similarity as follows:

$$d(k_{iu}, k_{jv}) = \frac{2 * \log P(K_p)}{\log P(K_{iu}) + \log P(K_{jv})}$$

where $P(K_x)$ represents the probability that a randomly selected concept belong to the K_x in the ontology. The similarity measure of S_i and S_j is defined as

$$Sim(S_i, S_j) = \frac{\sum_{u=1}^{m} \sum_{v=1}^{n} w_{iu,jv}}{m+n}$$

$$\text{where } w_{iu,jv} = \begin{cases} 1, \text{ if } k_{iu} = k_{jv} \\ d(k_{iu}, k_{jv}), \text{ if } k_{iu} \text{ is related to } k_{jv} \text{ in the ontology} \\ 0, \text{ if } k_{iu} \text{ \& } k_{jv} \text{ are different literally \& semantically} \end{cases}$$

IV. Experimental Results

We conducted some experiments on a yeast gene data set (http://rana.lbl.gov/EisenData.htm). In our experiment we considered the genes in a function family as a cluster and created 10 data sets. Table 4 shows 10 yeast gene function families as clusters and information about experiment data sets.

The input data set is documents relevant to genes in clusters. For each gene its synonyms are searched; yeast synonym information is found at www.yeastgenome.org/gene_list.shtml. For each gene and its synonyms the relevant documents are fetched from PubMed using our PubMed search tool on the fly. The results of gene cluster 1 and 7 as samples are shown in Table 5.

Table 4. Gene Clustering and Document Clustering

Gene Cluster #	# of genes in the cluster (including synonyms)	# of relevant PubMed documents	# of document clusters for each gene cluster	Gene Function
1	19 (25)	122	2	ATP synthesis
2	19 (35)	519	6	Mitosis
3	19 (69)	262	3	Vaculolar protein targeting
4	20 (30)	501	5	Silencing
5	20 (34)	213	2	Fatty acid metabolism
6	21 (35)	386	6	Meiosis
7	21 (31)	242	3	Phospholipid metabolism
8	22 (30)	203	3	TCA cycle
9	42 (67)	640	6	Chromatin structure
10	42 (75)	1874	15	DNA replication

V. Conclusion

In this paper we present a novel system **Bio-CS** for biomedical literature clustering and summarization. Our system integrates gene ontology, text clustering and text summarization. The experiment results on yeast gene expression data indicate that the **Bio-CS** can clusters can provide a concise and informative textual summary for the gene clusters. One of the challenging issues for summarization is how to organize the extracted sentences in a coherent way. We plan to integrate chronicle ordering to sort the extracted sentence and hope to report our findings in the near future.

Table 5. The top 10 significant terms and the best sentence for each cluster

Cluster #	Cluster common terms	Document Cluster #	Key Terms	Best Sentences
1	ATPase activity; DNA; Cell; membrane	1	RNA; binding; chromosome; cytochrome; protein; telomere	These data were in agreement with the sequence of the hypothetical protein L8003.20 whose primary structure was deduced from DNA sequencing of the yeast chromosome XII.
		2	growth; phosphorylation; protein; translation; transport; vacuolar membrane	We conclude that Yme1p is in part responsible for assuring sufficient F(1)F(0)-ATPase activity to generate a membrane potential in mitochondria lacking mitochondrial DNA and propose that Yme1p accomplishes this by catalyzing the turnover of protein inhibitors of the F(1)F(0)-ATPase.
7	Binding; Biosynthesis; Growth; holin	1	DNA; RNA; cell; membrane; protein; transferase activity	The phospholipid composition of yeast plasma membrane was manipulated by two different methods: (i) by using two auxotrophic strains KA101 (cho1) and MC13 (Cho+) which required phospholipid bases for growth and (ii) by supplementing Saccharomyces cerevisiae (3059) cells with high concentration of choline or ethanolamine.
		2	DNA; RNA; cell; lipid biosynthesis; protein; transcription	Expression of the C. albicans secretory aspartyl proteinase (SAP) and phospholipase B (PLB) virulence genes was determined by reverse transcription-PCR after the addition of caspofungin to cells grown for 15 h in Sabouraud dextrose broth.
		3	centromere; chromosome; lipid biosynthesis; phospholipid; transcription; vacuole	Structural genes of phospholipid biosynthesis in the yeast Saccharomyces cerevisiae are transcriptionally co-regulated by ICRE (inositol/choline-responsive element) promoter motifs.

REFERENCES

[1] Min-Yen Kan and Kathleen R. McKeown (1999) Information Extraction and Summarization: Domain Independence through Focus Types. Columbia University Computer Science Technical Report, CUCS-030-99

[2] Hongyuan Zha. (2002) Generic Summarization and Keyphrase Extraction Using Mutual Reinforcement Principle and Sentence Clustering, Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 113-120.

[3] Goldstein, J., Mittal, V., Carbonell, J. and Kantrowitz, M. 2000. "Multi-document summarization by sentence extraction." In Proceedings of the ANLP/NAACL Workshop on Automatic Summarization, Seattle, WA.

[4] J. Carbonell and J Goldstein (1998). The use of mmr, diversity-based reranking for reordering document and producing summaries. Proc. of the 21th Annual International ACM SIGIR Conference on Research and Development in the Information Retrieval, pp. 335-336.

[5] H.P. Luhn. The automatic creation of literature abstracts. IBM Journal of Research and Development, 2(2):159--165, 1958.

[6] Iliopoulos, A.J. Enright and C.A. Ouzounis (2001). TextQuest: Document Clustering of MEDLINE Abstract For Concept Discovery In Molecular Biology. Pacific Symposium of Biocomputing 2001, 384-395, 2001.

[7] L. Tanabe, U. Scherf, L.H. Smith, J.K. Lee, L. Hunter and J.N Weinstein (1999). MedMiner: An Internet Text-Mining Tool for Biomedical Information, with Application to Gene Expression Profiling. Biotechniques, 27(6): 1210-1217, 1999.

[8] H. Shatkey, S. Edwards, W.J. Wilbur and M. Boguski (2000). Genes, Themes And Microarrays: Using Information Retrieval For Large-Scale Gene Analysis. 8th Int. Conf. on Intelligent Systems Mol. Bio. (ISMB 2000), 8: 317-328, La Jolla, August 2000.

[9] Blaschke, J.C. Oliveros and A. Valencia (2001). Mining Functional Information Associated With Expression Arrays. Funct Integr Genomics, 1(4): 256-268, 2001.

[10] A. Hotho, S. Staab and G. Stumme, Ontologies Improve Text Document Clustering, The 3rd IEEE International Conference on Data Mining, pp. 541-544, Melbourne, Florida, Nov. 19-22, 2003.

[11] S. Schulze-Kremer. Integrating and Exploiting Large-Scale, Heterogeneous and Autonomous Databases with an Ontology for Molecular Biology. In: Molecular Bioinformatics, Sequence Analysis - The Human Genome Project (R. Hofestaedt and H. Lim eds). Shaker Verlag, Aachen, pp. 43-56 (1997).
[12] GO Consortium, http://www.geneontology.org/
[13] Edward I. Altman, The Z-Score Bankruptcy Model: Past, Present, and Future, John Wiley & Sons, New York, 1977.
[14] D. Pelleg and A. Moore, X-means: Extending K-means with efficient Estimation of the number of cluster, in Proceedings of the 17th International Conference on Machine Learning, 2000.
[15] J. Kleinberg (1998). Authoritative Sources in a Hyperlinked Environment. Proc. ACM-SIAM SODA.
[16] X. Hu (2004). Integrating Cluster Ensemble and Text Mining for Gene Expression Analysis, Proceedings of the 2004 IEEE Symposium on Bioinformatics and Bioengineering, 251-259.
[17] D.R. Radev, H. Jing and M. Budzikowska, Centroid-based Summarization of Multiple Documents: Sentence Extraction, Utility-based Evaluation and User Studies, Proceedings of the ANLP/NAACL Workshop on Automatic Summarization, Seattle, WA., 2000, pp. 21-29.
[18] P. Resnik, Using Information Content to Evaluate Semantic Similarity in a Taxonomy, Proceedings of the International Joint Conference on Artificial Intelligence, pp. 448-453, Montreal, Canada, 1995.

GRANULAR COMPUTING: BIAPPROXIMAATION SPACES

A.M. Kozae*, H. M. Abu – Donia**

Department of Mathematics, Faculty of Science, (* Tanta , ** Zagazig) University , Egypt.

Abstract

The purpose of the present work is to construct a new method for approximation of sets using two information systems simultaneously. Some properties and characterizations are given and a comparison with the previous sorts of approximation is obtained.

1. Introduction

A data set is represented as a table where each row represents a case, an event, a patient, or simply an object. Every column represents an attribute (a variable, an observation, a property, etc.) that can be measured for each object; the attribute may be also supplied by a human expert or user. This table is called an information system. More formally, it is a triple (X, T, t) [7] where X is a non-empty finite set of objects called the universe, and T is a non-empty finite set of attributes such that $t : X \longrightarrow V_t$ for all $t \in T$. The set V_t is called the value set of t. The notion of equivalence is recalled first, A binary relation $R \subseteq X \times X$ which is reflexive (i.e., an object is in relation with itself xRx), Symmetric (if xRy then yRx) and transitive (if xRy and yRz then xRz) is called an equivalence relation. The equivalence class of an element $x \in X$ consists of all objects $y \in X$ such that xRy. Let $\zeta = (X, A, t)$ be an information system obtained via data collected from an experiment by a user then there is associated an equivalence relation R_ζ

$$R_\zeta = \{(x, y) \in X \times X : \ \forall \ t_i(x) = t_i(y)\},$$

R_ζ is called indiscernibity relation. If $(x, y) \in R_\zeta$, then objects x and y are indiscernible from each other by attributes from A. The equivalence classes of an indiscraibility relation R on a set X are denoted by $[x]_R$, this class forms a base β_R for a topology on X.

Kelly [8] introduced the concept of bitopological spaces as method of generalizes topological spaces. The field of bitopologies has achieved great success in abstract study [9]. To the best four Knowledge the notions of bitopological structures are not applied in the field of set approximations one of this methods is rough set approach. The purpose of the present work is to construct another approach for rough set analysis using bitopology, we expert that this approach will give a general view for rough set concepts in the case of two information systems resulting from two experiments or the view of two experts. However, we hope that this work is an initial step for the application of bitopological concepts in the fields of applications based on rough set concepts such as reduction of attributes, decision tables used dependency of knowledge bases.

2. Set of pairwise approximation

If the attribute for each object are measured by two experts or users, we have two tables or two information systems $\zeta = (X, T, t)$ and $\zeta' = (X, T, t')$, we can obtain two indiscernibity relations R_ζ and $R_{\zeta'}$. Consequently we have two bases β_{R_ζ} and $\beta_{R_{\zeta'}}$ for two topologies τ_1 and τ_2.

Example 1.1: In the following tables we have two information systems for five cases $X = \{a, b, c, d, e\}$ represent five patients, $T = \{t_1, t_2\}$ attributes represent symptoms of diseases and the values represent the type of symptoms as the following 1 means abdominal pain, 2 means headache, 3 means fever, 4 means diarrhea

	t_1	t_2
a	1	3
b	2	4
c	2	3
d	1	3
e	2	4

table(1)
Results from the medical expert (1)

	t_1	t_2
a	1	3
b	2	4
c	1	3
d	1	3
e	1	3

table(2)
Results from the medical expert (2)

With respect to user (1) the reader will easily notice that cases a and d as well as b and e have exactly the same values of conditions. Also with respect to user (2), the cases $a, b, c$and d have the same values of conditions.

In table (1)

$R_\zeta = \{(x, y) \in X \times X : t_i(x) = t_i(y) \ \forall \ x, y \in X, i, j \in \{1, 2\}\}$

$R_\zeta = \{(a, a), (b, b), (c, c), (d, d), (e, e), (a, d), (d, a), (b, e), (e, b)\}$

$\beta_{R_\zeta} = \{[x]_{R_\zeta} : x \in X\} = \{\{a, d\}, \{b, e\}, \{c\}\}.$

The topology induced by β_{R_ζ} as a base is

$\tau_1 = \{X, \phi, \{a, d\}, \{b, e\}, \{c\}, \{a, b, d, e\}, \{a, d, c\}, \{b, c, e\}\}$

In table (2)

$R_{\zeta'} = \{(x, y) \in X \times X : t_i(x) = t_i(y) \ \forall \ x, y \in X, i, j \in \{1, 2\}\}$

$R_{\zeta'} = \{(a, a), (b, b), (c, c), (d, d), (e, e), (a, c), (c, a), (a, d), (d, a),$
$(a, e), (e, a), (c, d), (d, c), (c, e), (e, c), (d, e), (e, d)\}$

$\beta_{R_{\zeta'}} = \{[x]_{R_{\zeta'}} : x \in X\} = \{\{b\}, \{a, c, d, e\}\}.$

The topology induced by $\beta_{R_{\zeta'}}$ as a base is

$\tau_2 = \{X, \phi, \{b\}, \{a, c, d, e\}\}$

Definition 1.1. If we have two information systems $\zeta = (X, T, t)$ and $\zeta' = (X, T, t')$ and τ_1, τ_2 are two topologies induced by β_{R_ζ} and $\beta_{R_{\zeta'}}$ respectively as a bases we can define pairwise lower approximation for any subset A of X as the following

$$_pL(A) = int_{\tau_1}(A) \cup int_{\tau_2}(A)$$

. Also we define pairwise upper approximation for any subset A of X as the following

$$_pU(A) = cl_{\tau_1}(A) \cap cl_{\tau_2}(A)$$

In Example 1.1, if we let $A = \{b, d\}$. Then $_pL(A) = \{b\}$ and $_pU(A) = \{a, b, d, c\}$

Proposition 1.1. One can easily show the following properties of pairwise approximations:

(1) $_pL(A) \subseteq A \subseteq_p U(A)$

(2) $_pL(X) =_p U(X) = X$ and $_pL(\phi) =_p U(\phi) = \phi$

(3) $_pU(A \cup B) \supseteq \ _pU(A) \ \cup \ _pU(B)$

(4) $_pU(A \cap B) \subseteq \ _pU(A) \ \cap \ _pU(B)$

(5) $_pL(A \cap B) \subseteq \ _pL(A) \ \cap \ _pL(B)$

(6) $_pL(A \cup B) \supseteq \ _pL(A) \ \cup \ _pL(B)$

(7) If $A \subseteq B$ implies $_pU(A) \subseteq \ _pU(B)$ and $_pL(A) \subseteq \ _pL(B)$

(8) $_pU(X \setminus A) = X \setminus \ _pL(A)$ and $_pL(X \setminus A) = X \setminus \ _pU(A)$

(10) $_pL(_pL(A)) \subseteq \ _pU(_pL(A))$

(11) $_pL(_pL(A)) =_p L(A)$

(12) $_pU(_pU(A)) \supseteq \ _pL(_pU(A))$

(13) $_pU(_pU(A)) =_p U(A)$

Proof: We prove the parts (9) and (11) only, other parts are obtains similarly.
(9)

$$\begin{aligned}
({}_pU(B))^c &= (cl_{\tau_1}(B) \cap cl_{\tau_2}(B))^c \\
&= (cl_{\tau_1}(B))^c \cup (cl_{\tau_2}(B))^c \\
&= int_{\tau_1}(B^c) \cup int_{\tau_2}(B^c) \\
&= {}_pL(B^c)
\end{aligned}$$

Similarly **(11)**

$$\begin{aligned}
{}_pL({}_pL(B)) &= {}_pL(int_{\tau_1}(B) \cup int_{\tau_2}(B)) \\
&= int_{\tau_1}(int_{\tau_1}(B) \cup int_{\tau_2}(B)) \cup int_{\tau_2}(int_{\tau_1}(B) \cup int_{\tau_2}(B)) \\
&= int_{\tau_1}(B) \cup int_{\tau_1}(int_{\tau_2}(B)) \cup int_{\tau_2}(int_{\tau_1}(B)) \cup int_{\tau_2}(B) \\
&= int_{\tau_1}(B) \cup int_{\tau_2}(B) \\
&= {}_pL(B)
\end{aligned}$$

In the following example we show that the equality in parts (3), (5), (10) and (12) of Proposition 1.1 are not true in general.

Example 1.2. Consider the two information systems as in Example 1.1.

part (1) Let $A = \{a\}$, $B = \{b\}$, we have ${}_pU(A) = \{a, b\}$, ${}_pU(B) = \{b\}$ and ${}_pU(A \cup B) = \{a, b, d, e\}$. Consequently ${}_pU(A \cup B) \neq {}_pU(A) \cup \ {}_pU(B)$.

part (2) Let $A = \{a, c, d, e\}$, $B = \{a, b, d, e\}$, we have ${}_pL(A) = \{a, c, d, e\}$, ${}_pL(B) = \{a, b, d, e\}$ and ${}_pL(A \cap B) = \{a, d\}$. Consequently ${}_pL(A \cup B) \neq {}_pL(A) \cap \ {}_pL(B)$.

part (3) Let $A = \{a, b, c, d\}$, we have ${}_pL({}_pL(A)) = \{a, b, c, d\}$ and ${}_pU({}_pL(A)) = X$. Consequently ${}_pL({}_pL(A)) \neq_p U({}_pL(A))$

part (4) Let $A = \{e\}$, we have ${}_pL({}_pU(A)) = \phi$ and ${}_pU({}_pU(A)) = \{e\}$. Consequently ${}_pU({}_pU(A)) \neq_p L({}_pU(A))$

The following table show that the difference between approximations by using our approach and Pawlak's approach.

Approximation by using one user	Approximation by using two user
$L(A) \subseteq A \subseteq U(A)$	${}_pL(A) \subseteq A \subseteq {}_pU(A)$
$L(X) = U(X) = X$	${}_pL(X) = {}_pU(X) = X$
$L(\phi) = U(\phi) = \phi$	${}_pL(\phi) = {}_pU(\phi) = \phi$
$U(A \cup B) = U(A) \cup U(B)$	${}_pU(A \cup B) \supseteq {}_pU(A) \cup {}_pU(B)$
$U(A \cap B) \subseteq U(A) \cap U(B)$	${}_pU(A \cap B) \subseteq {}_pU(A) \cap {}_pU(B)$
$L(A \cap B) = L(A) \cap L(B)$	${}_pL(A \cap B) \subseteq {}_pL(A) \cap {}_pL(B)$
$L(A \cup B) \supseteq L(A) \cup L(B)$	${}_pL(A \cup B) \supseteq {}_pL(A) \cup {}_pL(B)$
If $A \subseteq B$ implies $U(A) \subseteq U(B)$ $L(A) \subseteq L(B)$	If $A \subseteq B$ implies ${}_pU(A) \subseteq {}_pU(B)$ ${}_pL(A) \subseteq {}_pL(B)$
$L(X \setminus A) = X \setminus U(A)$ $U(X \setminus A) = X \setminus L(A)$	${}_pL(X \setminus A) = X \setminus {}_pU(A)$ ${}_pU(X \setminus A) = X \setminus {}_pL(A)$
$L(L(A)) = U(L(A))$	${}_pL({}_pL(A)) \subseteq {}_pU({}_pL(A))$
$L(A) = L(L(A))$	${}_pL(A) = {}_pL({}_pL(A))$
$U(U(A)) = L(U(A))$	${}_pU({}_pU(A)) \supseteq {}_pL({}_pU(A))$
$U(A) = U(U(A))$	${}_pU(A) = {}_pU({}_pU(A))$

The emergence of two viewpoints increase sets which is definable internally or externally. One can define the following four basic classes of raugh sets.

Definition 1.2. For any two information systems $\zeta = (X, T, t)$ and $\zeta' = (X, T, t')$. The set $A \subseteq X$ is called:

(1) Roughly pairwise definable iff ${}_pL(A) \neq \phi$ and ${}_pU(A) \neq X$.

(2) Internally pairwise undefinable iff ${}_pL(A) = \phi$ and ${}_pU(A) \neq X$.

(3) Externally pairwise definable iff ${}_pL(A) \neq \phi$ and ${}_pU(A) = X$.

(4) Pairwise exact iff ${}_pL(A) = {}_pU(A) = A$.

We denote the set of all Roughly pairwise definable (resp. Internally pairwise undefinable, Externally pairwise definable and Pairwise exact) sets by $RPD(X, \tau_1, \tau_2)$ (resp. $IPU(X, \tau_1, \tau_2)$, $EPD(X, \tau_1, \tau_2)$ and $PE(X, \tau_1, \tau_2)$). In the case of using one of two information systems $\zeta = (X, T, t)$ and $\zeta' = (X, T, t')$ we denote the set of all Roughly definable (resp. Internally undefinable, Externally definable and exact) sets by $RD(X, \tau_i)$ (resp. $IU(X, \tau_i)$, $ED(X, \tau_i)$ and $E(X, \tau_i)$) where $i = 1, 2$.

Remark 1.1 For any two information systems $\zeta = (X, T, t)$ and $\zeta' = (X, T, t')$. The the relations between the types of sets in Definition 1.2 with respect to two user and one user as the following: for all $i = 1, 2$

(1) $RPD(X, \tau_1, \tau_2) \supseteq RD(X, \tau_i)$

(2) $IPU(X, \tau_1, \tau_2) \subseteq IU(X, \tau_i)$

(3) $EPD(X, \tau_1, \tau_2) \subseteq ED(X, \tau_i)$

(4) $PE(X, \tau_1, \tau_2) \supseteq E(X, \tau_i)$

Example 1.3. In Example 1.1,we have:
$RPD(X, \tau_1, \tau_2) = \{\{b\}, \{c\}, \{a, b\}, \{c, e\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{b, e\}, \{c, d\}, \{a, b, d\},$
$\{a, b, e\}, \{a, c, d\}, \{a, c, e\}, \{a, d, e\}, \{b, c, e\}, \{c, d, e\}, \{a, c, d, e\}, \{a, b, d, e\}\}$.
$RD(X, \tau_1) = \{\{c\}, \{a, d\}, \{b, c\}, \{b, e\}, \{c, d\}, \{c, e\}, \{a, b, d\}, \{a, b, e\}, \{a, c, d\},$
$\{a, d, e\}, \{b, c, e\}, \{a, b, d, e\}\}$.
$RD(X, \tau_2) = \{\{b\}, \{a, c, d, e\}\}$.
$IPU(X, \tau_1, \tau_2) = \{\{a\}, \{d\}, \{e\}, \{a, e\}, \{d, e\}, \phi\}$.
$IU(X, \tau_1) = \{\{a\}, \{b\}, \{d\}, \{e\}, \{a, b\}, \{a, e\}, \{d, e\}, \{b, d\}, \phi\}$.

$IU(X,\tau_2) = \{\{a\},\{c\},\{d\},\{e\},\{a,c\},\{a,d\},\{a,e\},\{d,e\},\{c,d\},\{c,e\},\phi\}$.
$EPD(X,\tau_1,\tau_2) = \{X,\{a,b,c\},\{b,c,d\},\{a,b,c,d\},\{a,b,c,e\},\{b,c,d,e\}\}$.
$ED(X,\tau_1) = \{X,\{a,b,c\},\{a,c,e\},\{b,c,d\},\{c,d,e\},\{a,b,c,d\},\{a,b,c,e\},\{a,c,d,e\},\{b,c,d,e\}\}$.
$ED(X,\tau_2) = \{X,\{a,b\},\{b,c\},\{b,d\},\{b,e\},\{a,b,c\},\{a,b,d\},\{a,b,e\},\{b,c,d\},\{b,c,e\},\{b,d,e\}$
$,\{a,b,c,d\},\{a,b,c,e\},\{a,b,d,e\},\{b,c,d,e\}\}$.
$PE(X,\tau_1,\tau_2) = \{X,\{b\},\{c\},\{a,d\},\{b,e\},\{a,c,d\},\{b,c,e\},\{a,c,d,e\},\{a,b,d,e\}\}$
$E(X,\tau_1) =\{X,\{c\},\{a,d\},\{b,e\},\{a,c,d\},\{b,c,e\},\{a,b,d,e\}\}$
$E(X,\tau_2) = \{X,\{b\},\{a,c,d,e\}\}$

Rough set can be also characterized by the following cofficient $\alpha_p(A) =\mid \frac{{}_pL(A)}{{}_pU(A)} \mid$ called the accuracy of pairwise approximation, where $\mid A \mid$ denotes the cardinality of $A \neq \phi$. Obviously $0 \leq \alpha_p(A) \leq 1$. If $\alpha_p(A) = 1$ the set A is an exact and if $\alpha_p(A) < 1$ the set A is a rough set.

The relation between the degree of accuracy of pairwise approximation by using tow information systems together and approximation by using each of the two information system alone as the following $\alpha_p(A) \geq \max\{\alpha_1(\mathrm{A}), \alpha_2(\mathrm{A})\}$ where $\alpha_p(A)$ is the accuracy of pairwise approximation by two information systems (two users), $\alpha_1(A)$ is the accuracy of pairwise approximation by first information systems (user 1) and $\alpha_2(A)$ is the accuracy of pairwise approximation by second information systems (user 2).

The following table show that the degree of accuracy of approximation $\alpha_p(A)$, $\alpha_1(A)$ and $\alpha_2(A)$ for some sets in Example 1.1

The set	$\alpha_{ij}(A)$	$\alpha_1(A)$	$\alpha_2(A)$
X	1	1	1
$\{a, b\}$	$\frac{1}{4}$	0	$\frac{1}{5}$
$\{a, b, c\}$	$\frac{2}{5}$	$\frac{1}{5}$	$\frac{1}{5}$
$\{b, d, e\}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{5}$
$\{a, c, d, e\}$	1	$\frac{3}{5}$	1

3. Real life application

In the following we will investigate the Middle East setution problem using our approach and compare the results with the results of Pawlak [1] and M. M. Abd El-Monsef [2]. Pawlak and Skowron [3] characterize a rough set by a single membership function for any subset $A \subseteq X$, a rough membership function is defined by

$$\mu_A(x) = \frac{\mid A \cap [x]_R \mid}{\mid [x]_R \mid}$$

where $\mid . \mid$ denotes the cardinality of a set.

Definition 2.1. If we have two information systems $\zeta = (X, T, t)$ and $\zeta' = (X, T, t')$ we can define the pairwise membership function by the relation

$$_p\mu_A(x) = \frac{\mid A \cap ([x]_{R_\zeta} \cup [x]_{R_{\zeta'}}) \mid}{\mid ([x]_{R_\zeta} \cup [x]_{R_{\zeta'}}) \mid}$$

where $[x]_{R_\zeta}$(resp. $[x]_{R_{\zeta'}}$) is the equivalent class of x with respect to information system $\zeta = (X, T, t)$ (resp. $\zeta' = (X, T, t')$) and $\mid . \mid$ denotes the cardinality of a set.

In the folllowing we will showing the political situation of the Middle East problem by using two veiws, the first veiw is befor Irak war and the other after it. Let us consider the nine parties (objects) in this problem.

(1) Egypt (2). Israeli (3). Jordan (4). Lebanaon (5). Palestine

(6) Syria (7). Saudi Arabia (8). Iraq (9). Kuwait (10). Kater.

The relation between those parties are determined by the following twelve issues (attributes).

(a) Return of Golan Heights of Syria.

(b) Israeli military outposts on the Golan Heights.

(c) Israeli accupation zone in south Lebanon.

(d) Free access to all religious centers.

(e) Arab countries grant citizenship to palestinians who choose to remain within their borders.

(f)Israeli retains East Al-Quads.

(g) Isolation and division of Al-Quads.

(h) Autonomous palestinian state on the West Bank and Gaza.

(i) Return of the West Bank and Gaza to Arab rule.

(j) Israeli military autpost along the Jordan river.

(k) Roed map.

(l) The segregation wall.

The following two table (information systems) summarize all the participants opinion on the previous twelve issues before and after Iraq war. If the participant is against the issue we put 0 and if the participant is neutral or favorable toward the issue we assign that by 1

	a	b	c	d	e	f	g	h	i	j	k	l
Egypt	1	0	0	1	0	0	0	1	1	1	1	0
Israeli	0	1	1	1	1	1	1	0	1	1	0	1
Jordan	1	1	0	1	0	0	0	1	1	0	1	1
Lebanon	1	1	0	1	1	0	0	1	1	0	1	0
palestine	1	1	0	1	1	0	0	1	1	0	1	0
syria	1	1	0	1	1	0	0	1	1	1	1	1
KSA	1	1	0	1	1	0	0	1	1	1	1	1
Iraq	1	0	0	1	0	0	0	1	1	1	1	0
Kuwait	0	1	1	1	1	1	1	0	1	1	0	1
Qater	1	0	1	1	0	0	0	1	1	0	1	0

table (*)
Information system before Iraq war

	a	b	c	d	e	f	g	h	i	j	k	l
Egypt	1	0	0	1	0	0	0	1	1	1	1	0
Israeli	0	1	1	1	1	1	1	0	1	1	1	0
Jordan	1	1	1	1	1	0	1	1	1	0	0	1
Lebanon	1	1	0	1	0	0	0	1	1	1	1	1
palestine	1	1	0	1	1	0	0	1	1	0	1	0
syria	1	1	0	1	1	0	0	1	1	1	1	0
KSA	1	1	0	1	1	0	0	1	1	1	1	1
Iraq	0	1	1	1	1	1	1	0	1	1	1	0
Kuwait	0	1	1	1	1	1	1	0	1	1	1	0
Qater	0	1	1	1	1	1	1	0	1	1	1	0

table (**)
Information system after Iraq war

In table (*)

we have the following equivalence class

$[1]_{R_\zeta} = [8]_{R_\zeta} = \{1, 8\}$

$[2]_{R_\zeta} = [9]_{R_\zeta} = \{2, 9\}$

$[6]_{R_\zeta} = [7]_{R_\zeta} = \{6, 7\}$

$[4]_{R_\zeta} = [5]_{R_\zeta} = \{4, 5\}$

$[3]_{R_\zeta} = \{3\}$

$[10]_{R_\zeta} = \{10\}$

In table (**)

we have the following equivalence class

$[1]_{R_{\zeta'}} = \{1\}$

$[2]_{R_{\zeta'}} = [8]_{R_{\zeta'}} = [9]_{R_{\zeta'}} = [10]_{R_{\zeta'}} = \{2, 8, 9, 10\}$

$[3]_{R_{\zeta'}} = \{3\}$

$[4]_{R_{\zeta'}} = \{4\}$

$[5]_{R_{\zeta'}} = \{5\}$

The degree of membership of Israeli with respect to the set

$A = \{\text{Egypt}, \text{Kuwait}, \text{Iraq}, \text{Kater}, \text{Palestine}\}$ with respect to one information system in table (*) is

$$\begin{aligned} \mu_A(2) &= \frac{\mid A \cap [2]_{R_\zeta} \mid}{\mid [2]_{R_\zeta} \mid} \\ &= \frac{\mid \{1,5,8,9,10\} \cap \{2,9\} \mid}{\mid \{2,9\} \mid} \\ &= \frac{1}{2} \end{aligned}$$

The degree of membership of Israeli with respect to the set

$A = \{\text{Egypt}, \text{Kuwait}, \text{Iraq}, \text{Kater}, \text{Palestine}\}$ with respect to tow information system is

$$\begin{aligned} \mu_A(2) &= \frac{\mid A \cap ([2]_{R_\zeta} \cup [2]_{R_{\zeta'}}) \mid}{\mid [2]_{R_\zeta} \cup [2]_{R_{\zeta'}} \mid} \\ &= \frac{\mid \{1,5,8,9,10\} \cap (\{2,9\} \cup \{2,8,9,10\}) \mid}{\mid \{2,9\} \cup \{2,8,9,10\} \mid} \\ &= \frac{3}{4} \end{aligned}$$

Acknowledgments

The authers greatly appreciate the valuable comments and additions of prof.Dr. T. Y. Lin

References

[1] E. A. Rady, A. M. Kozae and M. M. E. Abd El-Monsef "Generalized rough sets" Chaos, Solitons and Fractals, Volume 21, Issue 1, (2004), 49-53

[2] Z. Pawlak "Information systems theoretical foundations" Information Systems, Volume 6, Issue 3, (1981) 205-218

[3] Z. Pawlak, and A. Skowron "Rough membership functions in Fuzzy Logic for the Management of Uncertainty"(L. A. Zadah and J. Kacprzyk ,Eds.), Johnwily and Sons, New York, 251-271, 1994.

[4] Supriya Kumar De and P.R.P. Radha Krishna "Clustering web transactions using rough approximation" Fuzzy Sets and Systems, 27 (2004).

[5] Lixiang Shen and Han Tong Loh "Applying rough sets to market timing decisions" Decision Support Systems, Volume 37, Issue 4, (2004) 583-597.

[6] Francis E. H. Tay and Lixiang Shen "Economic and financial prediction using rough sets model" European Journal of Operational Research, Volume 141, Issue 3, (2002) 641-659.

[7] Z. Pawlak "Rough sets" International Journal of Computer and Information Sciences, 11, (1982) 341-356.

[8] J. Kelly "Bitopological spaces" Proc London Math Soc 3 (1963) 17-89.

[9] O A. El-Tantawy and H. M. Abu-Donia "Some bitopological concepts based on the alternative effects of closure and interior operator" Chaos, Solitons and Fractals, 19 (2004) 1119-1129.

Mathematical Theory of High Frequency Patterns

Tsau Young ('T. Y.') Lin
Department of Computer Science,
San Jose State University,
San Jose, CA 95192-0249, USA
Tel: 408-924-5121, Fax: 408-924-5062
e-mail: tylin@cs.sjsu.edu

Abstract

The principal focus is to examine the foundation of association (rule) mining (AM) via granular computing (GrC). The main results is: The set of all high frequency patterns is the set of set theoretical expresssions of the names of elementary granules or the well form formulas in deicion logic with large meaning set

1. INTRODUCTION

What is data mining? The following informal paraphrase of Fayad et al. (1996)'s definition seems quite universal: *Deriving useful patterns from data.* The keys are data, patterns, derivation system, and useful-ness. We will examine critically the current practices of AM

1.2. Basic Terms in Association Mining (AM)

In AM, two measures, support and confidence, are the main criteria. It is well known among researchers the support is the main hurdle, in other words, high frequency patterns are the main focus. AM is originated from the market basket data (Agrawal, 1993). However, we will be interested in AM for relational tables. For definitive, we assert:

1. A relational table is a bag relation, that is, repetitions of tuples are permissible (Garcia-Monila et al. 2002)
2. An item is an attribute value,
3. A q-itemset is a subtuple of length q,
4. A high frequency pattern of length q is a q-subtuple if its number of occurrences is greater than or equal to a given threshold.

2. EMERGING METHOD - GRANULAR COMPUTING

Bitmap index is a common notion in database theory. The advantage of bitmap representation is computationally efficient (Louis and Lin, 2000), and the drawback is the order of the table has to be fixed (Garcia-Molina, 2002). Based on granular computing, we propose a new method, called granular representations, that avoids this drawback. We will illustrate the idea by examples. The following example is modified from the text cited above (p. 702). A relational table K is viewed as a knowledge representation of a set V, called the universe, of real world entities by tuples of data; see Table 1.

V		BusinesSize	Bmonth	City
v_1		TWENTY	MAR	NY
v_2		TEN	MAR	SJ
v_3		TEN	FEB	NY
v_4	K	TEN	FEB	LA
v_5	→	TWENTY	MAR	SJ
v_6		TWENTY	MAR	SJ
v_7		TWENTY	APR	SJ
v_8		THIRTY	JAN	LA
v_9		THIRTY	JAN	LA
Relational Table K				

BusinesSize	Bmonth	City
100011100	110011000	101000000
011100000	110011000	010011100
011100000	001100000	101000000
011100000	001100000	000100011
100011100	110011000	010011100
100011100	110011000	010011100
100011100	000000100	010011100
000000011	000000011	000100011
000000011	000000011	000100011
Bitmap Table B		

Table 1. K and B are isomorphic

BusinesSize	Granular Representation	Bitmap Representation
TWENTY	$=\{v_1, v_5, v_6, v_7\}$	=100011100
TEN	$=\{v_2, v_3, v_4\}$	=011100000
THIRTY	$=\{v_8, v_9\}$	=000000011
	GDM in Granules	GDM in Bitmaps

Table 2a. Granular Data Model (GDM) for BusinesSize Attribute

Bmonth	Granular Representation	Bitmap Representation
Jan	$=\{v_8, v_9\}$	=000000011
Feb	$=\{v_3, v_4\}$	=001100000
Mar	$=\{v_1, v_2, v_5, v_6\}$	=110011000
APR	$=\{v_7\}$	=000000100
	GDM in Granules	GDM in Bitmaps

Table 2b. Granular Data Model (GDM) for Bmonth attribute

City	Granular Representation	Bitmap Representation
LA	$=\{v_4, v_8, v_9\}$	=000100011
NY	$=\{v_1, v_3\}$	={v1, v3}
SJ	$=\{v_2, v_5, v_6, v_7\}$	=010011100
	GDM in Granules	GDM in Bitmaps

Table 2c. Granular Data Model (GDM) for City attribute

V		BusinesSize	Bmonth	City	BusinesSize	Bmonth	City
v_1		TWENTY	MAR	NY	{v1,v5,v6,v7}	{v1,v2,v5,v6}	{v1,v3}
v_2		TEN	MAR	SJ	{v2,v3,v4}	{v1,v2,v5,v6}	{v2,v5,v6,v7}
v_3		TEN	FEB	NY	{v2,v3,v4}	{v3,v4}	{v1,v3}
v_4	K	TEN	FEB	LA	{v2,v3,v4}	{v3,v4}	{v4,v8,v9}
v_5	→	TWENTY	MAR	SJ	{v1,v5,v6,v7}	{v1,v2,v5,v6}	{v2,v5,v6,v7}
v_6		TWENTY	MAR	SJ	{v1,v5,v6,v7}	{v1,v2,v5,v6}	{v2,v5,v6,v7}
v_7		TWENTY	APR	SJ	{v1,v5,v6,v7}	$\{v_7\}$	{v2,v5,v6,v7}
v_8		THIRTY	JAN	LA	{v8, v9}	$\{v_8,v_9\}$	{v4,v8,v9}
v_9		THIRTY	JAN	LA	{v8, v9}	$\{v_8,v_9\}$	{v4,v8,v9}
Bag Relation K					Granulr Table G		

Table 2. K and G are isomorphic

A bitmap index for an attribute is a collection of bit-vectors, one for each possible value that may appear in the attribute. For the first attribute, **B**usines**S**ize (the amount of business in millions), the bitmap index would have nine bit-vectors. The first bit-vector, for value TWENTY, is 100011100, because the first, fifth, sixth, and seventh tuple have **B**usines**S**ize = TWENTY. The other two, for values TEN and THIRTY, are 011100000 and 000000011 respectively; Table 1 shows both the original table and bitmap table. Bmonth means Birth month; City means the location of the entities.

Next, we will interpret the bit-vectors in terms of set theory. A bit-vector can be viewed as a representation of a subset of V**.** For example, the bit-vector, 100011100, of **B**usines**S**ize = TWENTY says that the first, fifth, sixth, and seventh entities have been selected, in other words, the bit-vector represents the subset $\{v_1, v_5, v_6, v_7\}$. The other two bi-vectors, for values TEN and THIRTY, represent the subsets $\{v_2, v_3, v_4\}$ and $\{v_8, v_9\}$ respectively. We summarize such translations in Table 2a,b,c. and refer to these subsets as **elementary granules.**

Some easy observations:

1. The collection of elementary granules of an attribute (column) forms a partition, that is, all granules of this atribute are pairwise disjoint. This fact was observed by Pawlak (1982) and Tony Lee(1983).
2. From Table 1 & 2, one can easily conclude that the relational table K, the bitmap table B and granular table G are isomorphic. Two tables are isomorphic if one can transform a table to the other by renaming all attribute values in a one-to-one fashion.

3. GRANULAR DATA MODEL (GDM) – TABLE IN FREE FORMAT

The middle columns of Table 2a, 2b and 2c define 3 partitions. The universe and such 3 partions, denoted by (V, $\{E_{\mathbf{B}\text{usines}\mathbf{S}\text{ize}}, E_{\mathbf{B}\text{month}}, E_{\mathbf{C}\text{ity}}\}$), determines the granular table G and vice versa. More generally, a pair (V, E, C) is called a GDM, where E is a set of finite family of partitions, and C consists of the names of all elementary granules. A partition (equivalence relation) of V that is not in the given E is referred to as an **uninterpreted attribute of GDM**, **and its elementary granules are un-interpreted attribute values.**

GDM Theorem. The granular table G and its GDM determine each other.

In view of Isomorphic theorem below, it is sufficient to do AM in GDM.

4. ANALYSIS OF ASSOCIATION MINING (AM)

To understand the mathematical mechanics of AM, let us ecamine how the information has been created and processed. We will take the deductive data mining approach.

First, let us set up some terminology. A symbol is a string of "bits and bytes" that represents a slice of real world, however, such a real world meaning *does not participate* in the formal processing or computing. We term such a processing *computing with symbols*. In AI, such a symbol is termed a semantic primitive. (Feigenbaum, 1981). A symbol is termed a word*,* if the intended real world meaning *participates in the formal processing or computing.* We term such a processing *computing with words.* Note that

mathematicians use words (in group theory) as symbols; their words are our symbols.

4.1. Data Processing and Computing with Words

In traditional data processing (TDP), a relational table is a knowledge representation of a slice of real world. So each symbol of the table represents (to human) a piece of the real world; however, such a representation is not implemented in the system. Nevertheless, DBMS, under *human commands,* does process the data, for examples, Bmonth (attribute), April, March (attribute values) with human-perceived semantics. So in TDP the relational table is a table of words; TDP is human directed *computing with words.*

4.2. Data Mining and Computing with Symbols

In (automated) AM we use the table created in TDP. However, AM algorithms regard the TDP data as symbols; no real world meaning of each word participates in the process of AM. High frequency patterns are completely deduced from the counting of the symbols. AM is computing with symbols. The input data of AM is a relational table of symbols, whose real wolrd meaning does not participate in formal computing.

Under such a circumstance, if we replace the given set of symbols by a new set, then we can derive new patterns by simply replacing the symbols in "old" patterns. Formally, we have (Lin, 2002)

Isomorphic Theorem Isomorphic relational tables have isomorphic patterns.

This theorem implies that the theory of AM is a syntactic theory.

Example From Table 3, it should be clear that the one-to-one correspondences between K and K' induces consistently a one-to-one correspondence between the two sets of distinct attribute values. We describe such a phenomenon by the statement: K and K' are isomorphic.

V		**B**usines**S**ize	**B**month	**C**ity
v_1		TWENTY	MAR	NY
v_2		TEN	MAR	SJ
v_3		TEN	FEB	NY
v_4	K	TEN	FEB	LA
v_5	→	TWENTY	MAR	SJ
v_6		TWENTY	MAR	SJ
v_7		TWENTY	APR	SJ
v_8		THIRTY	JAN	LA
v_9		THIRTY	JAN	LA
Bag Relation K				

U		W't	Name	Material
u_1		20	SCREW	STEEL
u_2		10	SCREW	BRASS
u_3		10	NAIL	STEEL
u_4	K	10	NAIL	ALLOY
u_5	→	20	SCREW	BRASS
U_6		20	SCREW	BRASS
U_7		20	PIN	BRASS
U_8		30	HAMMER	ALLOY
U_9		30	HAMMER	ALLOY
Bag Relation K'				

Table 3 The isomorphism of Table K and K'

K	K'	GDM in Granules	Support
(TWENTY, MAR)	**(20,** SCREW)	$=\{v_1, v_5, v_6, v_7\} \cap \{v_1, v_2, v_5, v_6\}$	3
(MAR, SJ)	(SCREW, BRASS)	$=\{v_1, v_2, v_5, v_6\} \cap \{v_2, v_5, v_6, v_7\}$	3
(TWENTY, SJ)	(20, BRASS)	$=\{v_1, v_5, v_6, v_7\} \cap \{v_2, v_5, v_6, v_7\}$	3

Table 4. Three isomorphic 2-patterns; support =cardinality of granules

In Table 4, we display the high frequency patterns of length 2 from Table K, K' and GDM; the three sets of patterns are isomorphic to each other. So for AM, we can use any one of the three tables. An observation: In using K or K' for AM, one needs to scan the table to get the support, while in using GDM, the support can be read from the cardinality of the granules, no database scan is required – one strength of GDM. Another observation: From the definition of elementary granules, it should be obvious that **subtuples** are mapped to **the intersections of elementary granules**; see Table 4.

5. HIGH FREQUENCY PATTERNS ARE GRANULAR/DECISION FORMULAS

Implicitly AM has assumed high frequency patterns are "expressions" of the input symbols (elements of the input relational table.) Such assumptions are not made in other techniques. In neural network techniques, the input data are numerical, its patterns are not numerical "expressions." They are essentially functions that are derived from activation functions (Park and Sanders, 1989; Lin, 1996).

Let us back to AM, the implicit assumption simplifies the problem. What are the possible "expressions" of the input symbols? There are two possible formalisms, logic formula and set theoretical algebraic expression. In logic form, we have several choices, deductive database systems, datalog, or decision logic among others (Ullman, 1988-89; Pawlak, 1991); we choose decision logic because it is simpler. In set theoretical form, we use GDM (Lin, 2000).

5.1. Decision Logic Based Formula

A *high frequency pattern* in decision logic is a logic formula, whose meaning set (support) has cardinality greater than or equal to the threshold.

5.2. Granular Formulas - Set Theoretical Based Formulas

A *high frequency pattern* in GDM is a granular expression, which is a set theoretical algebriac expression of elementary granules; when the expression is evaluated set theoretically, the cardinailty of the resultant set is greater than or equal to the threshold; we will call the algebraic expressions granular pattern. Note that several distinct algebraic expressions of elementary granules may have the same resultant set.

Some observations: Informally, a logical formula of *granular pattern* is the "logic formula" of the names of elementary granules (Lin, 2000); more pricisely we translate elementary granules, $\cup$ and $\cap$ into their names, "or" and "and" respectively. Next, we note that there are only finitely many distinct subsets that can be generated by the intersections and unions of elementary granules in GDM. If we only consider the disjunct normal form, the total possible high frequency patterns in AM is finite.

6. HIGH FREQUENCY PATTERNS BY LINEAR INEQUALITIES

Let B be the Boolean algebra generated by the elementary granules; the partial order is the set theoretical inclusion $\supseteq$. Then B is the set of all granular expressions. Let O be the smallest element (it is not necessary an empty set) and I is the greatest element (I is the universe V). An element p is an atom, if $p \supseteq O$, and there is no element x such that $p \supseteq x \supseteq O$. Each atom p is an intersection of some elementary granules. Let S(b) be the set of all atom p_j such that $p_j \subseteq b$ and s(b) be its cardinality. From (Birkoff & MacLane, 1977, Chapter 11), we have

Proposition. Every $b \in B$ can be expressed in the form $b = p_1 \cup \ldots \cup p_{s(b)}$.

For convenience, let us define an "operation" of a binary number x and a set S. We write S*x to mean the following:

$$S*x = S, \quad \text{if } x=1 \text{ and } S \neq \varnothing$$
$$S*x = \varnothing, \quad \text{if } x=0 \text{ or } S = \varnothing$$

Let $p_1, p_2, \ldots, p_m$ be the set of all atoms in B. Then a granular expression b can be expressed as

$$p_1*x_1 \cup \ldots \cup p_m* x_m .$$

and its cardinality can be expressed as

$$|b| = \sum |p_i|*x_i$$

where $|\bullet|$ is the cardinality of $\bullet$.

Main Theorem. Let s be the threshold. Then b is a high frequency pattern, if

$$|b| = \sum |p_i|*x_i \geq s \qquad (*)$$

In applications, p_i 's are readily computable; it is the elementary granules of the intersection of all partitions (defined by attributes); see the Table 1 and 2. So we only need to find all binary solutions of x_i. The generators of the solution can be enumerated along the hyperplanes of the inequalities of the constraints.

Observations: Theoretically, this is a remarkable theorem. It says all possible high frequency patterns can be found by solving linear inequalities. However, the practicality of the main theorem is highly depended on the complexity of the problem. If both $|p_i|$ and s are small, then the number of solutions will be out of hands, simply due to the size of the number. We would like to stress that the difficulty is simply due to the size of possible solutions, not the methodology. The result implies that the notion of high frequency patterns may not be tight enough. At this moment, (*) is useful only if the number of attributes under considerations is small.

7. FUTURE TRENDS

7.1. Tighter Notion of Patterns

Let us consider the real world meaning of the patterns of length 2, namely, (TWENTY, MAR) and (20, SCREW). What does this subtuple (TWENTY, MAR) mean? 20 million dollar business on March ? The last statement is not the original meaning of the schema: Originally it means v_1, v_5, v_6 have 20 million dollar busness and they were born in March. This subtuple has no meaning on its own. On the other hand, (20, SCREW) from K' is a valid pattern (most of screws have weight 20). In summary, we have

(TWENTY, MAR) from K has no meaning on its own,
(20, SCREW) from K' has a valid meaning.

Let RW(K) be the Real World that K is representing. The summary implies that the subtuple (TWENTY, MAR), even though occurs very frequenly in the table, there is no real world event correspond to it. The data implies that three entities v_1, v_5, v_6 have common properties encoded by "Twenty" and "Mar." In the table K, they are "naively" summairzed into one concept "(TWENTY, MAR)." Unfortunately, in the real world RW(K), the three occurences of "Twenty" and "Mar" (from three entities, v_1, v_5, v_6) do not integrate into an appropriete new concept "(TWENTY, MAR)." Such "error" occurs, becuase high frequency is an inadequate or inaccurate criterion. We need a tighter notion of patterns.

7.2. Semantic Oriented Data Mining

If we do know how to compute the *semantics,* then the computation should tell us that the repeated two words "TWENTY" and "MAR" can not be combined into a new concept regardless of high repetitions, and should be dropped out. So semantic oriented data mining is needed (Lin & Louie. 2001, 2002). As ontology, semantic web, and computing with words (semantic computing) are heating up, it could be a right time to move onto Semantic Oriented Data Mining.

7.3. New Notions of Patterns and Algorithmic Information Theory

In (Lin, 1993), based on algorithmic information theory or Kolmogorov complexity theory, we proposed that a non-random (compressible string) is a string with patterns and the shortest Turing machine that generates this string is the pattern. We concluded, then, that a finite sequence (a relational table is a finite sequence) with long constant subsequences (the lenght of such constant sequence is the support) is trivially compressible (having a pattern). High frequency patterns are such patterns. Taking the same thought, what would be the next less trivial compressible finite sequences?

CONCLUSIONS

Our analysis on association mining seems fruitful: (1) High frequency patterns are natural generalizations of association rules. (2) All high frequency patterns (generalized associations) can be found by solving linear inequalities. (3) High frequency patterns are rather lean in semantics (**Isomorphic Theorem**). So semantic oriented AM or new notion of patterns may be needed.

REFERENCES

Fayad U. M., Piatetsky-Sjapiro, G. Smyth, P. (1996). From Data Mining to Knowledge Discovery: An overview. In Fayard, Piatetsky-Sjapiro, Smyth, and Uthurusamy eds., *Knowledge Discovery in Databases*, AAAI/MIT Press.

Agrawal, R,. Imielinski, T.,Swami, A. (1993, June). Mining Association Rules Between Sets of Items in Large Databases. In Proceeding of ACM-SIGMOD international Conference on Management of Data, pp. 207-216, Washington, DC.

Barr, A. Feigenbaum, E. A. (1981) The handbook of Artificial Intelligence, Willam Kaufmann.

Garcia-Molina, H., Ullman, J. D., Widom, J. (2002). Database Systems-The Complete Book, Prentice Hall

Park, J. and Sandberg, I. W. (1991) Universal Approximation Using radial-Basis-Function Networks, Neural Computation 3, 246-257

Ullman, J. (1988-89) Principles of Database and Knowledge-base systems Volume I (1988) Volume II (1989), Computer Science Press.

Lee, T. T., (1983). Algebraic Theory of Relational Databases. The Bell System Technical Journal 62(10), 3159-3204

Lin, T. Y. and Louie, E. (2001) Semantics Oriented Association Rules," In: 2002 World Congress of Computational Intelligence, Honolulu, Hawaii, May 12-17, 2002, 956-961 (paper # 5754)

Louie, E., Lin, T. Y. (2000, Oct). Finding Association Rules using Fast Bit Computation: Machine-Oriented Modeling. In: Foundations of Intelligent Systems, Z. Ras and S. Ohsuga (eds), Lecture Notes in Artificial Intelligence #1932, Springer-Verlag, 2000, pp. 486-494. (12th International symposium on methodologies for Intelligent Systems, Charlotte, NC, Oct 11-14, 2000)

Lin, T. Y. (2000). Data Mining and Machine Oriented Modeling: A Granular Computing Approach," Journal of

Applied Intelligence, Kluwer, 13(2), 113-124.

Lin, T. Y. (1996, July). The Power and Limit of Neural Networks. In: *Proceedings of the 1996 Engineering Systems Design and Analysis Conference*, Montpellier, France 7, 49-53.

Lin, T. Y. (1993). Rough Patterns in Data - Rough Sets and Foundation of Intrusion Detection Systems. *Journal of Foundation of Computer Science and Decision Support*, 18(3-4), 225-241.

Pawlak, Z. (1982). Rough sets. *International Journal of Computer and Information Sciences*, 11, 341-356.

Pawlak Z(1991). Rough sets. Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers

Combinatorial Topology and Primitive Concepts in Documents Clustering

Tsau Young ('T. Y.') Lin
Department of Computer Science,
San Jose State University,
San Jose, CA 95192-0249, USA
Tel: 408-924-5121, Fax: 408-924-5062
e-mail: tylin@cs.sjsu.edu

I-Jen Chiang
Graduate Institute of Medical Informatic
Taipei Medical University
205 Wu-Hsien Street
Taipei, Taiwan 110
email: ijchiang@tmu.edu.tw

Extended Abstract

Due to the rapid growth of resources over the Web and the diversity of content within any web page, automatic tools are necessary to help users find, filter, and extract the desired information. Search engines have become indispensable tools for gathering web pages and documents that are relevant to a user's query. However, inconsistent, uninteresting and disorganized search results are often returned. Without conceptual contexts, issues like *polysemy, phrases and term dependency* impose limitations on search technology. Search results can be improved with mechanisms based on categories, subjects, and contents.

Document clustering is considered as a mechanism to improve search results. A good search engine needs to discriminate whether a piece of information is relevant to users' queries within a short time. Short of the ability to extract semantic meaning from a document automatically, one can only hope to find a technique that can classify or cluster Web documents into semantic categories based on extracted features from those documents. Given that multiple concepts can be simultaneously defined in a single Web page, it is hard to limit the number of concept categories in a collection of Web pages. As a result, unsupervised clustering methods are better suited for document categorization on the huge, diverse, and scattered Web.

Our observation is that the frequent itemsets (undirected association rules or simply associations) of key terms in a document collection form mathematically a simplicial complex; previously they have been identified as a hypergraph. The nodes correspond to key terms in a document collection, while simplexes or hyperedges reflect the strong associations among these key terms. Superficially, both hypergraphs and simplicial complexes have captured the essence of the associations of key terms. Yet, the natures of two mathematical systems are quite different, they would yield different theories. Hypergraphs are pure combinatoric concepts, while simplicial complexes are not only combinatoric, but are also topological concepts that are deepest layer of geometric facts. For example, distance is not topological notion. In other words, our clustering is independent of metric; that marks our different from many classical clustering methods.

This paper presents a novel scheme to clustering documents based on simplicial complex in combinatorial topology. The associations among frequently co-occurring terms in documents naturally form a (combinatorial) simplicial complex. We believe each connected component of such a complex represents a primitive concept in the document collection. Based on these primitive concepts, documents can be clustered into meaningful groups. Experiments with three different data sets from web pages and medical literature have shown that the proposed unsupervised clustering approach performs significantly better than traditional clustering algorithms, such as k-means, AutoClass and Hierarchical Clustering (HAC). The results indicate that geometric model is a strong model capturing associations among key terms in text and is useful for automatic document clustering.

A complex bio-network of the function profile of genes

Charles C. H. Liu
Department of Surgery
Cathay Medical Center
Taipei, Taiwan 106
email: chliu@ntu.edu.tw

I-Jen Chiang
Graduate Institute of Medical Informatic
Taipei Medical University
Taipei, Taiwan 110
email: ijchiang@tmu.edu.tw

Jau-Min Wong
Institue of Medical Engineering
National Taipei University
Taipei, Taiwan 100
email: jmwong@ha.mc.ntu.edu.tw

Tsau Young ('T. Y.') Lin
Department of Computer Science
San Jose State University
San Jose, CA 95192-0249
email: tylin@cs.sjsu.edu

Abstract

This paper presents a novel model of concept representation using a multilevel geometric structure, which is called Latent Semantic Networks. *Given a set of documents, the associations among frequently co-occurring terms in any of the documents define naturally a geometric complex, which can then be decomposed into connected components at various levels.*

This hierarchical model of knowledge representation was validated in the functional profiling of genes. Our approach excelled the tradtional approach of vector-based document clustering by the geometrical forms of frequent itemsets generated by the association rules. The biological profiling of genes were a complex of concepts, which could be decomposed into primitive concepts, based on whichthe relevant literature could be clustered in adequate "resolution" of contexts. The hierarchical representation could be validated with tree-based biomedical ontological frameworks, which had been applied for years, and been recently enriched by the online availability of Unified Medical Language System (UMLS) *and* Gene Ontology (GO).

Demonstration of the model and the clustering would be performed on the relevant GeneRIF (References into Function) *document set of one gene. Our geometrical model is suitable for representation of biological information, where hierarchical concepts in different complexity could be explored interactively according to the context of application and the various needs of the reserachers.*

1. Introduction

One of the urgent need of bioinformatics in the post-genomic era is to find "biological themes" or "topics" between genes or gene products, in order to "drink from the fire hose" from vast amounts of literature and experiment results.

One approach of theme finding is to derive knowledge directly without translation by another knowledge source, e.g. a vocabulary system. One of the early successful approaches is direct mining from the source literature. The relationships between genes are constructed by probabilistic modes , such as Bayesian Networks. The most clinically yielding is the PubGene project [4]. However, the interpretation of the results is often qualitative, selectively on some local findings in large graph models. The lack of overall picture is partly due to the exploration of individual genes without preliminary grouping of some closely correlated genes. The result relied on the quality of documents collected as "relevant" to the target genes [8].

Subsequent researches to find "molecular pathway" in raw documents is vigorous use of natural language processing techniques. One of the efforts with a long history of literature mining in other medical domain is the GENIE project, evolved from MEDLEE works [2]. Finely tuned rule-based term tagging and processing improve the efficiency, but the rule sets or knowledge sources they constructed cannot be reused by other applications or be validated by others. Besides, the system is too large for personal document browsing.

The other approaches use external knowledge system, such as keyword hierarchy, to group the raw gene information to more biologically understandable "themes". The

early works are well reviewed by Shatkay in the analysis of microarray data [8]. MedMesh is more recent work addressing on the MeSH systems (Medical Subject Heading) of UMLS (Unified Medical Language System), but much raw document processing is used and the approach was relatively in a "black box" [6]. After the advent of Gene Ontology (GO) system, more tools were developed to apply the ontological framework to impose domain knowledge on analysis of raw data, which were listed under the section of "GO tools" in the official site of the GO Consortium [1].

From the medical point of view, current application of MeSH or GO is still in a very primitive developing stage. One of the main reason lies on the nature of tree-based ontological system. For example, GO divides the functional profiles into three branches from the root – the function domain, the process domain, and the anatomical domain. The first two domaion are closely associated in many application. The third domain is also dependent on the first two "function" domains. In addition, the amount of annotations of genes to the three domains are also unbalanced.

Our research addresses on the limitation of functional analysis of genes, and proposed a new geometric model. In what follows, we start by reviewing related work on the models of the relationships between gene and gene products clustering in section 2. The concepts and definitions of *latent semantic networks* based on geometric forms for the frequent itemsets generated by association rules are given in section 3. The clustering results for clustering of the functioning profile of a gene are described in Section 4; followed by the conclusion.

2. Related Work

Detecting knowledge based on the co-occurence of terms or concepts is one of the basic mechanism of document clustering, and was initially proposed to cluster genes into biologically meaningful groups [4]. However, the characteristics of the "groups" could not be explained by the co-occurrence alone. An approach of getting the biological "meaning" was by annotation with associated MeSH and GO terms, which were both tree-based. Our work approaches the "meaning" problem by proposing a new geometric model of clustering in order to more adequately present the network nature of the functioning profiles of genes.

After Girvan and Newman's work of "community structure" in socal and biological networks [3], the nature of graph structure inherent in a co-occurrence network began to be explored. Wilkinson et al. [6] picked sets of genes correlated to user-selected keywords by partitioningthe components of gene co-occurrence networks functionally correlateed "communities". Wren et al. [9] studied the connections in the gene network to rank the "cohesiveness" of co-occurring genes, diseases, and chemical compounds.

The current published genetic analyses based on "community" networks were calculated based on geometrical measurement in the Euclidean space, which we considered is a fundamental limitation of statistical calculation in document or concept clustering. The clustering of distance measurements between sets of more primitive concepts to form higher hierarchy of concept groups is more applicable in topological spaces than in Euclidean spaces. We proposed a topologically based network more suitable for gene analysis.

3. Geometric Representation of Concept

Term-term inter-relationships that are denoted by their co-occurred associations can automatically model and extract the concepts from a collection of documents. These concepts organize a multilevel and homogenous hierarchy called a *Latent Semantic Network*. The most natural way to represent a latent semantic network is expressed by using the geometric and topologic notations, which can capture the totality of thoughts expressed in this collection of documents; and a "simple component" (which is a *r-connected component*) of a level of hierarchy represents some concept inside this collection.

3.1. Combinatorial Geometry

Let us introduce and define some combinatorial topological concepts. The central idea is n-simplex.

Definition 1 *A n-simplex is a set of independent abstract vertices* $[v_0, \ldots, v_{n+1}]$.

Geometrically 0-simplex is a vertex, 1-simplex an edge (a vertex pair), 2-simplex a triangle, 3-simplex a tetrahedron. A n-simplex is the $n+1$ dimensional analog. This is the smallest convex set in a Euclidean space R^{n+1} containing $n+1$ points $v_0 \ldots, v_{n+1}$ that do not lie in a hyperplane of dimension less than n. For example, there is the standard n-simplex

$$\delta^n = \{(t_0, t_1, \ldots t_{n+1}) \in R^{n+1} \mid \sum_i t_i = 1, t_i \geq 0\}$$

Definition 2 *A face of a n-simplex* $[v_0, \ldots, v_{n+1}]$ *is a r-simplex* $[v_{j_0}, \ldots, v_{j_{r+1}}]$ *whose vertices is a subset* $\{ v_0, \ldots, v_{n+1} \}$ *with cardinality* $r+1$.

Definition 3 *A complex is a finite set of simplices that satisfies the following two conditions:*

- *Any face of a simplex from a complex is also in this complex.*

- *The intersection of any two simplices from a complex is either empty or is a face for both of them.*

The vertices of the complex $v_0, v_1, \cdots, v_n$ is the union of all vertices of those simplces. [7]

Definition 4 *A* hereditary n simplex*, or abbreviated to be* n-H-simplex *is a special complex of n dimensions that consists of one n-simplex and all its faces.*

Definition 5 *A (n, r)-skeleton (denoted by S_r^n) of n-complex is a n-complex whose k-faces($k \leq r$) are removed*

Definition 6 *For any non-empty two simplices A, B are said to be* r-connected *if there exits a sequence of k-simplices $A = S_0, S_1, \ldots, S_m = B$ such that S_j and S_{j+1} has an h-common face for $j = 0, 1, 2, \ldots, m-1$; where $r \leq h \leq k \leq n$.*

Definition 7 *The maximal r-connected subcomplex is called a* r-connected component*. Note For a r-connected component implies there does not exist any r-connected component that is the superset of it.*

3.2. Simple Concept Geometric Structure

In our application each vertex is a key term, so a simplex defines a set of key terms in a collection of documents. Hence, we believe a simplex represents a primitive concept in the collection. For example, the 1-simplex [Wall, Street] represents a primitive concept in financial business. The 0-simplex [Network] might represent many different concepts, however, while it is combined with some other terms would denote latent semantic concepts, such as, these 1-simplices [Computer, Network], [Traffic, Network], [Neural, Network], [Comunication, Network], and so on, demonstrate distinct concepts and identify more obvious semantic than 0-simplex. Of course, the 1-simplex [Neural, Network] is not conspicuous than the 2-simplices [Artifical Neural Network] and [Biology, Neural, Network].

A collection of documents most likely consists of several distinct primitive concepts. Such a collection of primitive concepts is combinatorial a complex.

An idea (in the forms of complex of keywords) may consist of a lot of primitive concepts (in the form of simplices) that are embedded in a document collection. Some primitive concepts may share a common primitive concept, some may not. This situation may be captured by a combinatorial complex of key terms: An idea in the forms of a complex of keywords may consist of a lot of primitive concepts in the form of simplices. Some primitive concepts (simplices) may share a common concept (a common face), some may not.

Example 1 *In Figure 4, we have an idea that consist of twelve terms that organized in the forms of 3-complex. Two $Simplex(a, b, c, d)$ and $Simplex(w, x, y, z)$ are two maximal H-simplices with the highest rank 3. Considering $(3, 1)$-skeleton, S_1^3, by removing all 0-simplices, the other simplices in it can be listed.*

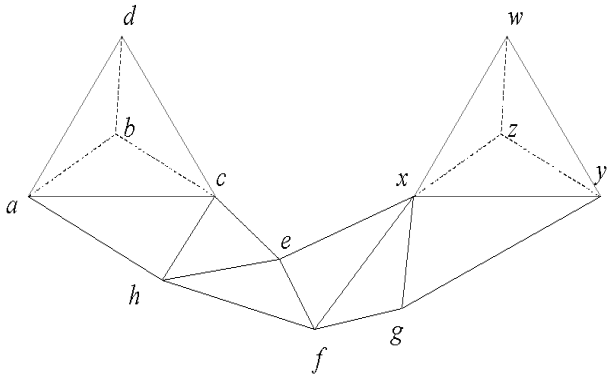


Figure 1. A complex with twelve vetrics.

- $\mathrm{Simplex(a, b, c, d)}$ *and its ten subsimplices:*
  - $\mathrm{Simplex(a, b, c)}$
  - $\mathrm{Simplex(a, b, d)}$
  - $\mathrm{Simplex(a, c, d)}$
  - $\mathrm{Simplex(b, c, d)}$
  - $\mathrm{Simplex(a, b)}$
  - $\mathrm{Simplex(a, c)}$

    ....
- $\mathrm{Simplex(a, c, h)}$ *and its three subsimplices:*
  - $\mathrm{Simplex(a, c)}$
  - $\mathrm{Simplex(a, h)}$
  - $\mathrm{Simplex(c, h)}$

  ...

There does not exist any common faces between any two simplices, so that eight maximal connected components are in S_2^3. So does S_3^3, there are only two maximal connected components in it because the maximum rank of simplices in it is 3.

A maximal connected component of a skeleton represents a complex of association rules, i.e., a set of concepts. If a maximal connected component of a skeleton contains only one simplex, this component is said to organize a primitive concept.

Definition 8 *A maximal connected component is said to be* independent *if it is composed of a single simplex, i.e., there is no common face between two maximal connected components.*

3.3. Issue

From a collection of documents, a complex of association rules can be generated. A skeleton of a complex is closed, because all subcomplexes of a complex are also in the skeleton according to subsimplices in each composite simplex of a complex in a skeleton are also included in the simplex, which satisfies the *apriori* property. As seen in Example 1, all connected components in S_k^n are contained in S_r^n, where $k \geq r$. Based on that, the goal of this paper is to establish the following belief.

Claim A maximal independent connected component of a skeleton represents a primitive *concept* in this collection of documents.

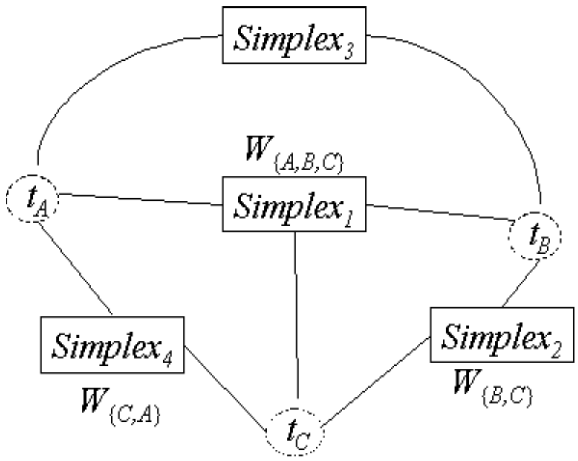


Figure 2. A simple skeleton S_1^3 of example is composed of three terms $\{t_A, t_B, t_C\}$ from a collection of documents, where each simplex is identified by its tfidf value and all 0-simplices have been removed (the nodes are drawn by using dash circles).

Example 2 *Given a skeleton, S_1^2, of association rules depicted in Figure 2, it is a 2-complex composed of the term set V=$\{t_A, t_B, t_C\}$ in a collection of documents. In the skeleton, all 0-simplices are neglect, i.e., the terms depicted in dash lines. The simplex set S =$\{\mathrm{Simplex}_1, \mathrm{Simplex}_2, \mathrm{Simplex}_3, \mathrm{Simplex}_4\}$ ($\mathrm{Simplex}_1$ is a 2-simplex and $\mathrm{Simplex}_2$, $\mathrm{Simplex}_3$ as well as $\mathrm{Simplex}_4$ are 1-simplices) represents generated frequent itemsets from V, and W =$\{w_{A,B}, w_{C,A}, w_{B,C}, w_{A,B,C}\}$ denote their corresponding supports.*

This complex is also a pure 2-simplex, i.e. triangle, with one maximal independent connected component. The boundary of 2-H-simplex has four 0-faces (0-simplexes) and three 1-faces (1-simplexes). Since all the simplexes are in the complex, it is a closed *complex. Therefore, we can say this complex represent a concrete concept. In general, the n-simplex has the following geometric property.*

Property 1 *The boundary of a n-H-simplex has $n+1$ 0-faces (vertices), $\frac{n(n+1)}{2}$ 1-faces (edges), and $\binom{n+1}{i+1}$ i-faces ($i \leq n$), where $\binom{n}{k}$ is a binomial coefficient.*

This geometric representation properly satisfies the *apriori* property of association rules: if the support of an item set $\{t_1, t_2, \cdots, t_n\}$ is bigger that a minimum support, so are all the nonempty subsets of it. In a complex, the universe of vertices organizes 1-simplices, i.e., frequent 1-itemsets, the universe of 1-simplex represents all possible frequent 1-itemsets and frequent 2-itemsets, and so on.

According to Example 1, it is obvious that simplices within the higher level skeleton S_r^n is contained in the lower level skeleton S_k^n with the same n-complex, $r \geq k$. Figure 3 shows the network hierarchy of the example, each skeleton is represented as a layer. For the purpose of simplicity, skeletons induced from r-complex, in which $0 \leq r < 3$, are neglected. The most distinct concepts of all (without a common concept between them) are existed in the topmost layer, although they could be empty concepts, which means there does not exist any non-overlapped concepts. In this example, the H-simplices $\mathrm{Simplex(a, b, c, d)}$ and $\mathrm{Simplex(w, x, y, z)}$ are two *maximal independent connected components* that demonstrate two discriminating primitive concepts. The H-simplices at the lower layers could have a common face between them. Therefore, the concepts denoted by those H-simplices are vague discriminated as shown in Figure 4 in that an overlapped concept induced by a common face is existed. As seen in the skeleton S_1^3, the maximal connected components generated from simplex $\mathrm{Simplex(a, b, c, d)}$ and simplex $\mathrm{Simplex(a, c, h)}$ have a common face $\mathrm{Simplex(a, c)}$ that makes some documents not able to properly discriminated in accordance with the generated association rules from term a and term c, so are the other maximal connected components in the skeleton. Because of the intersection produced by such subsimplices, some documents would be vague classified into two clusters. The lower the skeleton layer is, the serious the concept overlapping situation is.

4. Finding Maximal Connected Components

For the context of latent semantic ideas within a collection of documents, it is naturally that some similar concepts would be cross-referenced among the collection, espe-

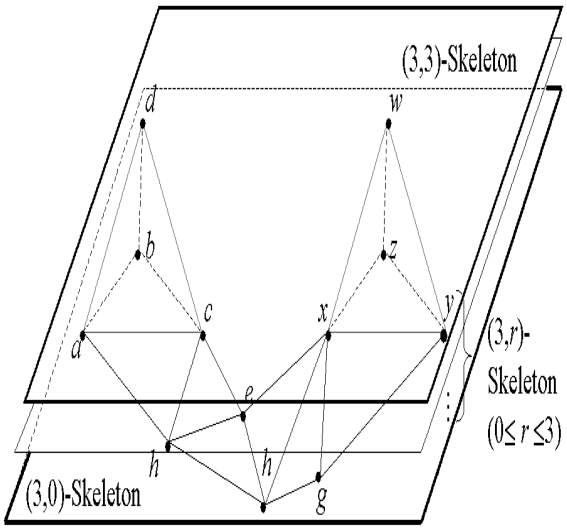


Figure 3. A simple latent semantic network with its hierarchical structures is generated from Example 1. Obviously the skeleton $(3,3)$-Skeleton at the topmost layer composed of two maximal connected components as two distinct concepts $\mathrm{Simplex(a,b,c,d)}$ and $\mathrm{Simplex(w,x,y,z)}$ is contained in the skeleton at the lower layer. Except the topmost layer, all the concepts are in some sort of vague discrimination. The bottom layer contains only one connected component, which is a 3-complex. All the concepts are mixed together that make several primitive concepts are non-distinguishable in this connected component.

cially for a collection of homogeneous documents. Therefore, some professional used words or phrases are often taken to denote a specific idea. No doubt that we can identify them by the usage of those terms. As we already known the best way to recognize them is according to term-term inter-relationships, which are term associations. Following the above statement, combinatorial geometry based latent semantic networks are the perfect model for illustrating the concepts in a huge variety of high-dimensional data, such a document collection. The algorithm for finding all concepts, i.e., maximal connected components, which is generated from the co-occurred terms in a collection of documents, will be introduced as follows.

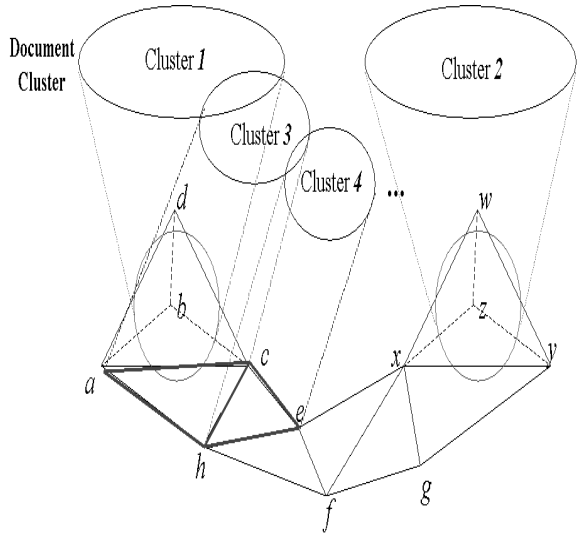


Figure 4. Each cluster of documents is identified by a maximal connected component. Some cluster may overlap with other cluster because of the common face between them.

4.1. Data Structure

In order for the further discussion on the algorithm, let us make the following definitions of the use of geometric notations to represent latent semantic networks on association rules.

Definition 9 *In a latent semanic network, let $\mathcal{V}$ be the set of single terms in a collection of documents, i.e., 0-simplices, and $\mathcal{E}$ be the set of all r-simplices, where $r \geq 0$. If $\mathrm{Simplex_A}$ is in $\mathcal{E}$, its support is defined as $w_(\mathrm{Simplex_A})$, i.e., the tfidf of all terms in $\mathrm{Simplex_A}$ co-occurred in a collection of documents.*

A network, which is a complex in geometry, can be represented as a matrix.

Example 3 *As seen in Example 2, the 2-simplex of the network is the set $\{t_A, t_B, t_C\}$, which is also the maximal connected component that represents a primitive concept in a document collection. As Venn diagram, the incident matrix I and the weighted incident matrix I_W of the network are as follows.*

$$I = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}.$$

$$I_W = \begin{pmatrix} w_{A,B,C} & 0 & w_{A,B} & w_{C,A} \\ w_{A,B,C} & w_{B,C} & w_{A,B} & 0 \\ w_{A,B,C} & w_{B,C} & 0 & w_{C,A} \end{pmatrix}.$$

The rows correspond to the terms and the columns correspond to the simplices.

Each simplex denotes a connected component, i.e., an undirected association rules. If the simplex is a maximal connected component, it defines a maximal frequent itemset. The number of terms in this connected component defines its *rank*, that is, if its rank is r it is equivalent to frequent $r+1$-itemsets.

4.2. Algorithm

As we already known, a r-H-simplex denotes a r-connected component, which is a frequent $r+1$-itemset. If we say a frequent itemset I_i identified by an H-simplex $\mathrm{Simplex_i}$ is a subset of a frequent itemset I_j identified by $\mathrm{Simplex_j}$, it means that $\mathrm{Simplex_i} \subset \mathrm{Simplex_j}$. An H-simplex $\mathrm{Simplex_i}$ is said to be a maximal connected component if no other H-simplex $\mathrm{Simplex_j} \in \mathcal{E}$ is the superset of $\mathrm{Simplex_i}$ for $i \neq j$. Documents can be automatically clustered based on all maximal connected components. It provide a soft-computing that allows overlapped concepts exist within a collection of documents.

All connected components are convex hulls, the intersection of connected components is nothing or a connected component. It would induce an vague region for concept discrimination if the intersection is a non-empty simplex. This common face will induce an unspecified concept between them as we have mentioned before. It is not necessary to consider this common face because it has been considered in its super-simplices.

Example 4 *As shown in Figure 5, one component is organized by the H-simplex* $\mathrm{Simplex_1} = \{t_A, t_B, t_C\}$*, the other is generated by the H-simplex* $\mathrm{Simplex_5} = \{t_C, t_D, t_E\}$.

The boundary of a concept defines all possible term associations in a document collection. Both of them share a common concept that can be taken as a 0*-simplex* $\{t_C\}$*, which is an* 1*-item frequent itemset* $\{t_C\}$.

Property 2 *The intersection of concepts is nothing or a concept that is a maximal H-simplex belonging to all intersected concepts.*

Since there is at most one maximal H-simplex in the intersection of more than one connected components and the dimension or rank of the intersection is lower than all intersected simplices. It is convenient for us to design an efficient algorithm for documents clustering based on all maximal connected components in a complex skeleton by skeleton. It does not need to traverse all complex.

5. Demonstrations

Demonstration were performed on the relevant *GeneRIF (References into Function)* document set, publicly available

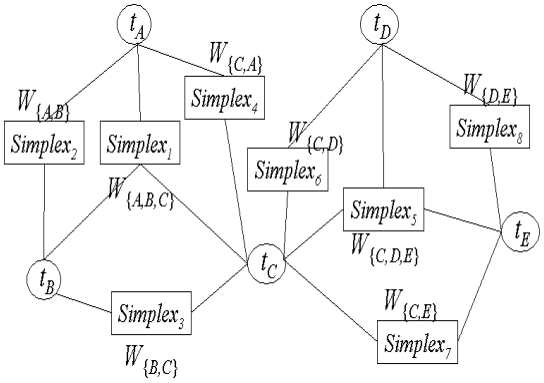


Figure 5. A complex is composed of two maximal connected components generated by two 2-H-simplices $\mathrm{Simplex(t_A, t_B, t_C)}$ and $\mathrm{Simplex(t_C, t_D, t_E)}$. Both of them contain a common face $\mathrm{Simplex(t_C)}$ that produces an undiscriminating concept region.

in the EUtils web service of the NCBI Entrez site. Our geometrical model is suitable for representation of biological information, where hierarchical concepts in different complexity could be explored interactively according to the context of application and the various needs of the reserachers.

The biological background of the experiment is briefly described here, with the terms or the concepts quoted. "CARD15" gene was found equivalent with "NOD2" gene in recent years. This CARD15/NOD2 gene was discovered associated with inflammatory bowel diseases ("IBDs") in 2000, and vigorous correlation studies were performed to elucidate the position on the genome or several candidate "chromosomes". The pathogenesis was proposed later to be "barrier" break in the intestinal ("mucosa") defense mechanism due to the genetic defect, then the focus of researchers shifted to the functioning domain of "inflammation" – "TNF", "TLR4", "NF-KappaB", and "Paneth cell".

The GIF document set of CARD15 gene was queried. The abstracts were retrieved, and the important keywords and synonyms were processed by a dictionary derived from UMLS thethaurus. The co-occurences between the terms were calculated, weighted by TFIDF measurements. In this implementation, the term nodes were ranked by TFIDF weighting, and directed graphes were displayed for additional arrangement of the terms after suggestion by medical domain experts. Our model does not imply directed association.

The nodes of relevant concepts were rendered by the default setting of ATT GraphViz, the layout algorithm of which was according to geometrically even distribution of the nodes and their edges. The nodes with more intercon-

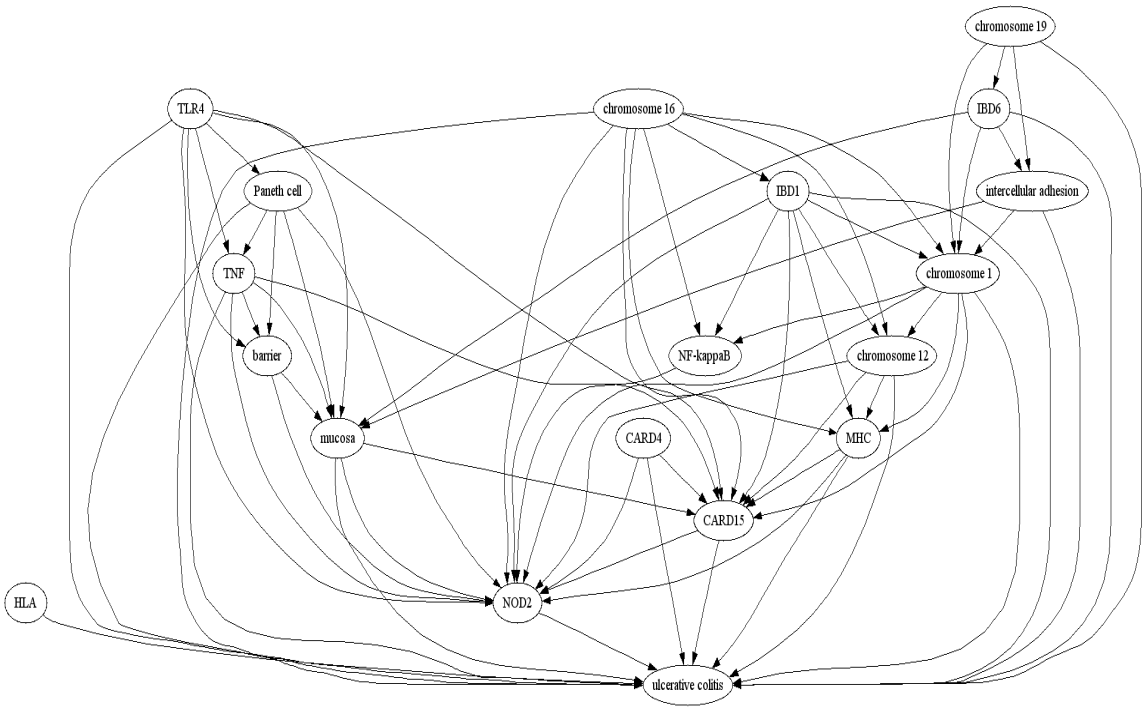


Figure 6. Functional profiles of the CARD15 gene, rendered by GraphViz. The direction of edges are based on TFIDF weighting in this implementation. Our model does not imply directed association.

nections or edges were positioneded together, compatible with the clusters of concepts in our model.

In Figure 6, the whole picture of term co-occurrence was shown. In Figure 7, the threshold of visible co-occurrence (the support) was raised, to show the 4-H-simplex or 5-H-simplex concept clusters. Three groups of 4-connected components or 5-connected components were shown in the left, the midlle, and the right regions, corresponding to the concept clusters of the new focus of "inflammatory process" and the older topics and genetic association and chromosome localization.

The left "inflammatory process" cluster was the 5-frequent itemset with "TLR", "Paneth cell", "TNF", "barrier", and "mucosa". The middle and right clusters were two 4-H-simplex, connected by the intersection of the "chromosome 1" node.

6. Conclusion

Polysemy, *phrases* and *term dependency* are the limitations of web search technology [5]. In the biomedical queries and concept analysis, the problem becomes more severe.

A group of solid term associations can clearly identify a concept. Most methods no matter what is *k-means*, *HCA*, *AutoClass* or *PDDP* classify or cluster documents from the represented matrix of a set of documents. It is inefficient and complicated to discover all term associations from such a high-dimensional and sparse matrix. Given a collection of documents, the associations among frequently co-occurring terms in any of the documents define naturally a geometric complex, which can then be decomposed into connected components at various levels and connected components can properly identify concepts in a collection of documents.

The paper presents a noval approach based on finding maximal connected components for clustering of the functional profile of genes. The r-simplexs, i.e., connected components, can represent the concepts in a collection of relevant documents. It illustrates that geometric complexes are a perfect model to denote association rules in text and is very useful for automatic document clustering and concept grouping, as demonstrated in our experiment in the functional analysis of gene-related documents.

References

[1] GO Consortium. Go tools: Editors, browsers, general go tools and other tools. http://www.geneontology.org/doc/GO.tools.html, 2004.

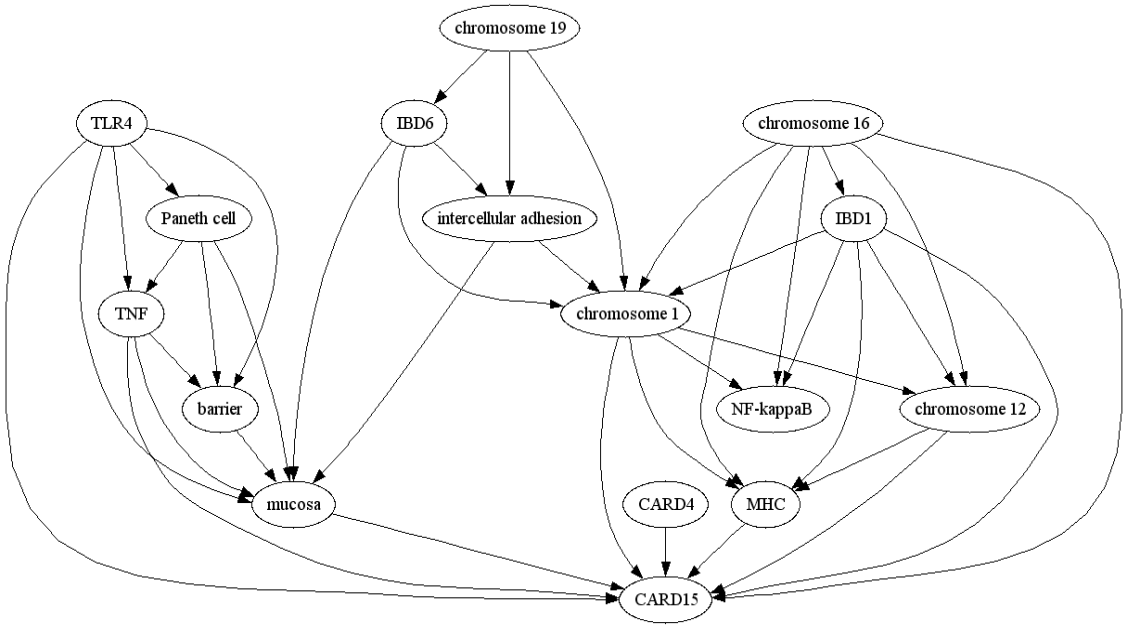


Figure 7. Functional profiles of the CARD15 gene, with the threshold of the co-occurence between concept raised. Three biologically meaningful clusters formed.

[2] C. Friedman, P. Kra, H Yu, M. Krauthammer, and A. Rzhetsky. Genies: a natural-language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics*, 17(suppl 1):S74–82, 2001.

[3] M. Girvan and M. Newman. Community structure in social and biological networks. In *Proceedings of the National Academy of Sciences*, volume 99, page 8271V76, 2002.

[4] T. K. Jenssen, A. Laegreid, J. Komorowski, and E. Hovig. A literature network of human genes for high-throughput analysis of gene expression. *Nature Genetics*, 28(1):21V28, 2001.

[5] A. Joshi and Z. Jiang. Retriever: Improving web search engine results using clustering. In A. Gangopadhyay, editor, *Managing Business with Electronic Commerce: Issues and Trends*, chapter 4. World Scientific, 2001.

[6] P. Kankar, S. Adak, A. Sarkar, K. Murali, and G Sharma. Medmesh summarizer: Text mining for gene clusters. In *Proceedings of the Second SIAM International Conference on Data Mining*, Apr 2002.

[7] J. R. Munkres. *Elements Of Algebraic Topology*. Addison Wesley, Reading MA, 1984.

[8] H. Shatkay, S. Edwards, W. J. Wilbur, and M. Boguski. Genes, themes and microarrays: Using information retrieval for large-scale gene analysis. In *Proc Int Conf Intell Syst Mol Biol*, volume 8, pages 317–28, 2000.

[9] J. D. Wren and H. R. Garner. Shared relationship analysis: ranking set cohesion and commonalities within a literature-derived relationship network. *Bioinformatics*, 20(2):191–8, 2004.

Naïve Rules Do Not Consider Underlying Causality

Lawrence J. Mazlack
Applied Computational Intelligence Laboratory
University of Cincinnati
Cincinnati, Ohio 45221-0030
mazlack@uc.edu

Abstract

Naïve association rules may result if the underlying causality of the rules is not considered. The greatest impact on the decision value quality of association rules may come from treating association rules as causal statements without understanding whether there is, in fact, underlying causality. A complete knowledge of all possible factors (i.e., states, events, constraints) might lead to a crisp description of whether an effect will occur. However, it is unlikely that all possible factors can be known. Commonsense understanding and reasoning accepts imprecision, uncertainty and imperfect knowledge. The events in an event/effect complex may be incompletely known; as well as, what constraints and laws the complex is subject to. Usually, commonsense reasoning is more successful in reasoning about a few large-grain sized events than many fine-grained events. A satisficing solution would be to develop large-grained solutions and only use the finer-grain when the impreciseness of the large-grain is unsatisfactory.

1. Introduction

One of the cornerstones of data mining is the development of association rules. Association rules greatest impact is in helping to make decisions. One measure of the quality of an association rule is its relative decision value. Association rules are often constructed using simplifying assumptions that lead to naïve results and consequently naïve and often wrong decisions. Perhaps the greatest area of concern about the decision value is treating association rules as causal statements without understanding whether there is, in fact, underlying causality.

Causal reasoning occupies a central position in human reasoning. It plays an essential role in human decision-making. Considerable effort over thousands of years has been spent examining causation. Whether causality exists at all or can be recognized has long been a theoretical speculation of scientists and philosophers. Serious questions have been asked whether commonsense perceptions of the world match the underlying reality. These concerns run from the implications of Zeno's paradox [Zeno, 449 B.C.] and Plato's cave [380 B.C.] to Einstein's relativity theory and modern string theory. An introduction to some of these issues may be found in Mazlack [2004].

At the same time, people operate on the commonsense belief that causality exists.

Causal relationships exist in the commonsense world; for example:

> When a glass is pushed off a table and breaks on the floor

it might be said that

> Being pushed from the table *caused* the glass to break.

Although,

> Being pushed from a table is not a ***certain*** *cause* of breakage; sometimes the glass bounces and no break occurs; or, someone catches the glass before it hits the floor.

Counterfactually, usually (but not always),

> *Not* falling to the floor prevents breakage.

Sometimes,

> A glass breaks when an errant object hits it, even though it does not fall from the table.

Positive causal relationships can be described as: *if* α *then* β (or, $\alpha \rightarrow \beta$). For example:

> When an automobile driver fails to stop at a red light and there is an accident it can be said that the failure to stop was the accident's *cause*.

However, negating the causal factor does not mean that the effect does not happen; sometimes effects can be *overdetermined*. For example:

> An automobile that did not fail to stop at a red light can still be involved in an accident; another car can hit it because the other car's brakes failed.

Similarly, simple negation does not work; both because an effect can be overdetermined and because negative statements are weaker than positive statements as the negative statements can become *overextended*. It cannot be said that $\neg\alpha \rightarrow \neg\beta$, for example:

> Failing to stop at a red light is not a ***certain*** *cause* of no accident occurring; sometimes no accident at all occurs.

Some describe events in terms of *enablement* and use counterfactual implication whose negation is implicit; for example [Ortiz, 1999a]:

Not picking up the ticket *enabled* him to miss the train.

There is a multiplicity of definitions of *enable* and *not-enable* and how they might be applied. To some degree, logic notation definitional disputes are involved. These issues are possibly germane to general causality theory. However, it is not profitable to the interests of this paper to consider notational issues; this paper is concerned with the less subtle needs of data analysis.

Negative causal relationships are less sure; but often stated; for example, it is often said that:

Not walking under a ladder prevents bad luck.

Or, usually (but not always),

Stopping for a red light avoids an accident.

In summary, it can be said that the knowledge of at least some causal effects is imprecise for both positive and negative descriptions. Perhaps, complete knowledge of all possible factors might lead to a crisp description of whether an effect will occur. However, it is also unlikely that it may be possible to fully know, with certainty, all of the elements involved. Consequently, the extent or actuality of missing elements may not be known. Additionally, some well described physics as well as neurobiological events appear to be truly random [Freeman, 1995]; and some mathematical descriptions randomly uncertain. If they are, there is no way of avoiding causal imprecision.

Coming to a precise description of what is meant by causality is difficult. There are multiple and sometimes conflicting definitions. For an introductory discussion of these issues, see Mazlack [2004]. Recognizing many things with absolute certainty is problematic. As this is the case, our causal understanding is based on a foundation of inherent uncertainty and incompleteness. Consequently, causal reasoning models must accommodate inherent ambiguity. For an introductory discussion of this, see Mazlack [2003a].

It may well be that a precise and complete knowledge of causal events is not possible or at least uncertain. On the other hand, we have a commonsense belief that causal effects exist in the real world. If we can develop models tolerant of imprecision, it would be useful. Also, to some degree, the degree of importance that some of these items have decreases as grain size increases.

2. Satisficing

People do things in the world by exploiting commonsense *perceptions* of cause and effect. Manipulating perceptions has been explored [Zadeh, 1999] but is not the focus of this paper. The interest here is how perceptions affect commonsense causal reasoning, granularity, and the need for precision.

When trying to precisely reason about causality, complete knowledge of all of the relevant events and circumstances is needed. In commonsense, every day reasoning, approaches are used that do not require complete knowledge. Often, approaches follow what is essentially a *satisficing* [Simon, 1955] paradigm. The use of non-optimal mechanisms does not necessarily result in ad hocism; Goodrich [2000] states:

> "Zadeh [1998] questions the feasibility (and wisdom) of seeking for optimality given limited resources. However, in resisting naïve optimizing, Zadeh does not abandon the quest for justifiability, but instead resorts to modifications of conventional logic that are compatible with linguistic and fuzzy understanding of nature and consequences."

Commonsense understanding of the world tells us that we have to deal with imprecision, uncertainty and imperfect knowledge. This is also the case with scientific knowledge of the world. An algorithmic way of handling imprecision is needed to computationally handle causality. Models are needed to algorithmically consider causes and effects. These models may be symbolic or graphic. A difficulty is striking a good balance between precise formalism and commonsense imprecise reality.

3. Complexes

When events happen, there are usually other related events. The entire collection of events can be called a complex. The events can be called the elements of the complex.

A "mechanism" [Simon, 1991] or a "causal complex" [Hobbs 2001, 2003] is a collection of events whose occurrence or non-occurrence results in a consequent event happening. Hobbs' causal complex is the *complete* set of events and conditions necessary for the causal effect (consequent) to occur. Hobbs suggests that human casual reasoning that makes use of a causal complex does not require precise, complete knowledge of the complex. (Different workers may use the terms "mechanism and "causal complex" differently; I am using them as these author's use them.)

Each complex, taken as a whole, can be considered to be a granule. Larger complexes can be decomposed into smaller complexes; going from large-grained to small-grained. For example, when describing starting an automobile, A large-grained to small-grained, nested causal view would start with

When an automobile's ignition switch is turned on, this *causes* the engine to start.

But, it would not happen if a large system of other nested conditions were not in place.

There has to be available fuel. The battery has to be operational. The switch has to be connected to the battery so electricity can flow through it. The wiring has to connect the switch to the starter and ignition system (spark plugs, etc.). The engine has to be in good working order; and so forth.

Turning the ignition switch on is one action in a complex of conditions required to start the engine. One of the events might be used to represent the collection of equal grain sized events; or, a higher level granule might be specified with the understanding that it will invoke a set of finer-grained events. In terms of nested granules, the largest grained view is: turning on the switch is the sole causal element; the complex of other elements represents the finer-grains. These elements in turn could be broken

down into still finer-grains; for example, "available fuel" could be broken down into:

> fuel in tank, operating fuel pump, intact fuel lines, and so forth.

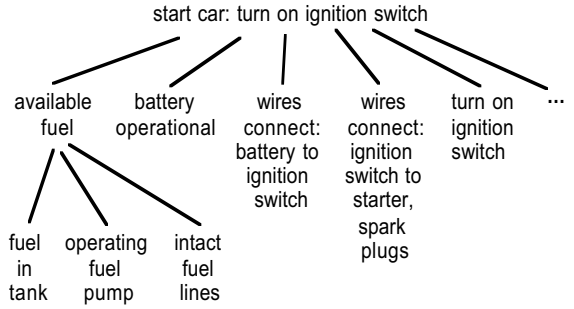


Figure 1. Nested causal complex.

Sometimes, it is enough to know what happens at a large-grained level; at other times it is necessary to know the fined grained result. For example, if

> Bill believes that turning the ignition key of his automobile *causes* the automobile to start.

It is enough if

> Bill engages an automobile mechanic when his automobile does not start when he turns the key on.

However,

> The automobile mechanic needs to know a finer-grained view of an automobile's causal complex than does Robin.

Instead of being concerned with all of the fined grained detail, a better approach may be to incorporate granulation using rough sets and/or fuzzy sets to soften the need for preciseness. And then accept impreciseness in the description. Each complex can be considered to be a granule. Larger complexes can be decomposed into smaller complexes. Thus, going from large-grained to small-grained.

Hobbs [2001] uses first order logic to describe his causal complexes. Pearl [2000] develops probabilistic causal networks of directed graphs (DAGs).

The causal complexes explicitly considered by Hobbs and Pearl have a required structure that may be overly restrictive for commonsense causal understanding, namely:

- If *all* of the events in the causal complex appropriately happen, then the effect will occur
- There is nothing in the causal complex that is irrelevant to the effect

These requirements are probably too precise and extensive to be realized in a commonsense world. Sometimes, only some of the events need to happen. For example,

> Someone may be able to save more money:
> - If their taxes are lowered or
> - If they earn more money.

Either even may lead to greater savings. However,

> Neither may result in increased savings if they also have to pay a large divorce settlement.

So, if all of the events happen, the effect may happen. If some of the events happen, the effect may happen. In the commonsense world, we rarely whether all of the events are in a complex are necessary. For example,

> A man may want to attract the attention of a woman. He may do a large number of things (e.g., hair, clothes, learn to dance, etc.). If he does attract the woman, he may never know which things were relevant and which were not

An issue is how to distinguish between what is in a complex and what is not. Another issue is how to distinguish between the things that deserve to be called "causes" and those that do not. Hobbs suggests that a consideration of causal complexes can be divided into:

- Distinguishing what events are in a causal complex from those outside of it. [Lewis, 1973] [Oritz, 1999b] [Simon, 1952, 1991] [Pearl, 2000]
- Within a causal complex, recognizing the events that should be identified as causes from those that are not. [Macke, 1993] [Shoham, 1990]

Nested granularity may be applied to causal complexes. A complex may be several larger grained elements. In turn, each of the larger grained elements may be a complex of more fine grained elements. Recursively, in turn, these elements may be made up still finer grained elements. In general, people are more successful in applying commonsense reasoning to a few large grain sized events than the many fine grained elements that might make up a complex.

A question concerning complexes is: To what extent can we increase the causal grain size and still have useful causal information? Conversely, can we start with a large-grained causal event and then derive the finer-grained structure? Can we measure and/or control the imprecision involved in changing grain size? If we start with a large-grained structure and resolve it, will our computational complexity burdens be reduced?

Complexes often may be best handled on a black-box, large grained basis. That is, it can be recognized that a complex exists; but we do not necessarily need to deal with the details internal to the complex.

3. Recognizing Causality Is Of Interest In Many Domains

Recognizing causality is of interest in many areas. Of particular interest to this paper are areas where the analysis is non-experimental. The world is taken as it is and not subject to experimentation. In the computational sciences, data mining is of concern. An area not well known to people working in the computational sciences is economics.

Perhaps, the applied area that has the greatest history of attempting to deal with causality and non-observational data is economics. Econometrics is distinguished from statistics by econometrics interest in establishing causation [Hoover, 2003]. How and if causality can be recognized has been a significant area of discussion. Some of this discussion mirrors discussion that has gone on in the

computational sciences. Hoover provides a good entry to the discussion of causality in economics.

Hume [1777/1902, p 165], as a philosopher, suggested that causal statements are really about constant conjunction and time ordering. However, when speaking as an economist, Hume [1742/1985, p 304] was less insistent on causal ordering: "it is of consequence to know the principle whence any phenomenon arises, and to distinguish between a cause and a concomitant effect." The issue of causal ordering is also often of importance to those modeling causality in data discovery.

Data mining analyzes data previously collected; it is non-experimental. There are several different data mining products. The most common are *conditional rules* or *association rules*. Conditional rules are most often drawn from induced trees while association rules are most often learned from tabular data.

IF Age < 20
THEN vote frequency is: often
with {belief = high}

IF Age is old
THEN Income < $10,000
with {belief = 0.8}

Figure 2. Conditional rules.

Customers who
*buy **beer** and **sausage***
*also tend to buy **hamburger***
with {confidence = 0.7}
in {support = 0.15}

Customers who
buy strawberries
*also tend to buy **whipped cream***
with {confidence = 0.8
in {support = 0.2}

Figure 3. Association rules.

At first glance, conditional and association rules seem to imply a causal or cause-effect relationship. That is:

> A customer's purchase of both sausage and beer ***cause**s* the customer to also buy hamburger.

Unfortunately, all that is discovered is the *existence* of a statistical relationship between the items. They have a degree of joint occurrence. The *nature* of the relationship is not identified. Not known is whether the presence of an item or sets of items causes the presence of another item or set of items; or the converse, or some other phenomenon causes them to occur together.

Purely accidental relationships do not have the same decision value, as do causal relationships. For example,

> IF it is true that buying both ***beer*** and ***sausage*** somehow causes someone to *buy* ***beer***,
>
> • THEN: A merchant might profitably put ***beer*** (or the likewise associated ***sausage***) on sale
>
> • AND at the same time: Increase the price of ***hamburger*** to compensate for the sausages' reduce sale price.

On the other hand, knowing that

> ***Bread*** and ***milk*** are often purchased in the same store visit

may not be as useful decision making information as both products are commonly purchased on every store visit. A knowledge of frequent co-occurrences of *bread* and *milk* purchases might lead us to placing the *bread* and *milk* at opposite ends of the store to force shoppers to visit more of the store and consequently make more impulse buying decisions. However, there is a limit to how often when such a physical distance distribution can be reasonably effected. What is most valuable is knowledge of true causal relationships.

Tangentially, what might be of interest is discovering if there is a causal relationship between the purchase of *bananas* and something else. (It turns out that *bananas* are the most frequently purchased food item at Wal-Mart [Nelson, 1998]).

When typically developed, rules do not *necessarily* describe causality. The association rule's confidence measure is simply an estimate of conditional probability. The association rule's support indicates how often the joint occurrence happens (the joint probability over the entire data set). The strength of any causal dependency may be very different from that of a possibly related association value. In all cases

> confidence ≥ causal dependence

All that can be said is that associations describe the strength of joint co-occurrences.

Sometimes, the association might be causal; for example, if

> Someone eats salty peanuts and then drinks beer.

or

> Someone drinks beer and then becomes inebriated.

there may be a causal relationship. On the other hand, if

> A rooster grows and then the sun rises.

or

> Someone wears a 'lucky' shirt and then wins a lottery.

there may not be a causal relationship. Recognizing true causal relationships would greatly enhances the decision value of data mining results.

The most popular market basket association rule development method identifies rules of particular interest by screening for joint probabilities (associations) above a specified threshold.

4.1 Association Rules Without An Underlying Causal Basis Can Lead To Naïve Decisions

Association rules are used is to aid in making retail decisions. However, simple association rules may lead to errors. Errors might occur; either if causality is recognized where there is no causality; or if the direction of the causal relationship is wrong [Silverstein, 1998*a*] [Mazlack,

2003*b*]. Errors might occur; either if causality is recognized where there is no causality; or if the direction of the causal relationship is wrong. For example, if

A study of past customers shows that 94% are sick.

- Is it the following rule?

Our customers are sick, so they buy from us.

- Is it the following complementary rule?

If people use our products, they are likely to become sick.

- Is there no relationship between product purchase and illness?

Consequently, from a decision making viewpoint, it is not enough to know that

People both buy our products and are sick.

What is needed is knowledge of what causes what, if anything at all.

If causality is not recognized, the naïve application of association rules can result in bad decisions [Silverstein, 1998*a*]. This can be seen in an example from Mazlack [2003]:

Example:

At a particular store, a customer buys*:*

- *hamburger 33%* of the time
- *hot dogs* 33% of the time
- both *hamburger* and *hot dogs* 33% of the time
- *sauerkraut* only if *hot dogs* are also purchased

This would produce the binary transaction matrix:

	hamburger	hot dog	sauerkraut
t_1	1	1	1
t_2	1	0	0
t_3	0	1	1

Figure 4. Binary transaction matrix for hamburger, hot dog, and sauerkraut purchases.

This in turn would lead to the associations (confidence):

- (*hamburger, hot dog*) = 0.5
- (*hamburger, sauerkraut*) = 0.5
- (*hot dog, sauerkraut*) = 1.0

All of the support levels are adequately high for this application.

If the merchant:

- Reduced price of *hamburger* (as a sale item)
- Raised price of *sauerkraut* to compensate (as the rule *hamburger fi sauerkraut* has a high confidence.
- The offset pricing compensation would not work, as the sales of sauerkraut would not increase with the sales of *hamburger*. Most likely, the sales of *hot dogs* (and consequently, *sauerkraut*) would likely decrease as buyers would substitute *hamburger* for *hot dogs*.

4.2 Association Rules That Do Not Take Into Account Quantities Can Result In Misleading Causal Inferences

Association rules are often formed by reducing all values to binary zeros and ones.

This is an early technique that was and is used in data mining when analyzing market basket data. However, it is essentially flawed. Quantities do matter; some data co-occurrences are conditioned on there being a sufficiency of a co-occurring attribute. Also, some relationships may be non-linear based on quantity [Mazlack, 2003b]

Example*:*

Situation: Customers frequently buy either wine or beer for themselves in varying amounts. However, when buying for a party, they often purchase both beer and wine and they usually purchase in larger quantities.

Actual basket:		*Binary basket:*	
Beer	Wine	Beer	Wine
6	0	1	0
0	1	0	1
12	0	1	0
0	3	0	1
24	4	1	1
24	5	1	1
48	2	1	1

Figure 5. Beer, wine transactions: quantified and binary.

Missed rule: When at least 24 beers purchased, wine also purchased;
Otherwise, there is no relationship between beer and wine.

Naïvely constructing an association rule on non-quantified, binary data would find a rule that misleadingly represents the situation; i.e.,

Misleading rule: When beer is purchased, wine is also purchased
{confidence = 0.6}
{support = 0.43}

This rule is misleading because it naïvely implies that purchase probabilities are uniform; in fact, they are not. Under one set of conditions, *beer* and *wine* are *never* purchased together under one set of conditions; and, under another set of conditions they are *always* purchased together.

In neither case is there a direct causal relationship. In the quantified rule case, the larger quantities of beer and wine are caused by a third factor (a party).

5. Describing Causality

In some ways, someone may object to this paper, as it does not offer much in the way of solutions. It mostly identifies needs. Part of a reply is that there is limited

space and time. Another is that recognizing a need is the first step to finding a solution. Another is that both recognizing and defining causality is still a very complex and difficult problem, even after over 3,000 years of effort.

Various causality descriptions and discovery tools have been suggested. It may eventually turn out that different subject domains may have different methodological preferences. This section is intended to give a selective, non-complete, taste.

5.1 Intuitive Graph Based Approaches

Different aspects of causality have been examined. As in *Figure 6,* the idea of "positive" causation ($\alpha \rightarrow \beta$) is at the core of commonsense causal reasoning. Often a positive causal relationship is represented as a network of nodes and branches [Mazlack, 2003*a*]. In part because of their intuitive appeal, there have been many approaches to recognizing causality that use graphs.

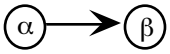


Figure 6. Diagram indicating that a is causally dependent on b.

There are a number of different books describing various aspects of causal graphs. Among them are: Gammerman [1999], Glymour [2001], Hausman [1988], Pearl [2000], Shafer [1996], Spirtes [1993].

5.2 Directed Graphs

Various graph based Bayesian based methods have been suggested to recognize causality. Probably the best known is the class of methods based on Directed Acyclic Graphs (DAGs). The most fully developed approach is Pearl [2000]. Silverstein [1998] followed a similar approach.

Pearl [1991] and Spirtes [1993] claim that it is possible to infer causal relationships between two variables from associations found in observational (nonexperimental) data without substantial domain knowledge. Spirtes claims that directed acyclic graphs could be used if (a) the sample size is large and (b) the distribution of random values is faithful to the causal graph. Robins [1999] argues that their argument is incorrect. Lastly, Scheines [1994] only claims that in some situations will it be possible to determine causality. Their discussion is tangential to the focus of this paper; going deeply into their discussion is outside this paper's scope. It is enough to note that these methods are possibly the most thoroughly developed methods of computational causal analysis.

From the commonsense causal reasoning view, the various directed graph methods have similar liabilities, specifically. Mazlack [2004] discusses and lists and discusses some of the problems.

5.3 Negation And Counterfactuals

Negation or counterfactuals ($\neg\alpha \rightarrow \neg\beta$) also have a place, although it may result in reasoning errors. For example, the rule:

> If a person drinks *wine*, they may become inebriated.

cannot be simply negated to

> If a person ***does not*** drink *wine*, they will ***not*** become inebriated.

One reason is that effects can be *overdetermined;* that is: more than one item can cause an effect. If so, eliminating one cause does not necessarily eliminate the effect. In this case:

> A person may also drink *beer* or *whiskey* to excess and become inebriated.

Events that do not happen can similarly be overdetermined. From a commonsense reasoning view, it is more likely that things do not happen than they do. For example, Oritz [1999a] says that it is not true that

> His closing the barn door caused the horse not to escape.

because the horse might not have attempted to escape even if the door was open. Therefore, a false counterfactual is:

> If he had not closed the barn door, the horse would have escaped.

Similarly, for example, the rule

> If a person smokes, they will get cancer.

cannot be simply negated to

> If a person does not smoke, they will not get cancer.

Again, effects can be overdetermined. In this case,

> People who do not smoke may also get cancer.

Another idea that is sometimes involved in causal reasoning is *causal uncorrelatedness* [Shafer, 1999] where if two variables have no common cause they are causally uncorrelated. This occurs if there are no single events that cause them to both change.

Similarly, Dawid [1999] focuses on the negative; i.e., when α does not affect β. Dawid speaks in terms of *unresponsiveness* and *insensitivity*. If β is *unresponsive* to α if whatever the value of α might be set to, the *value* of β will be unchanged. In parallel, if β is *insensitive* to α if whatever the value α may be set, the *uncertainty* about β will be unaffected. Along the same vein, Shoham [1990, 1991] distinguishes between *causing, enabling,* and *preventing*. The enabling factor is often considered to be a causal factor. Shoham distinguished between background (enabling) conditions and foreground conditions. The background (enabling) conditions are inferred by default. For example [Shoham, 1991]:

> "If information is present that the key was turned and nothing is mentioned about the stated about the state of the battery, then it is inferred that the motor will start, because the battery is assumed, by default to be alive.

Given this distinction, causing is taken to refer to the foreground conditions where enabling and preventing refer to the background conditions (in this example, turning the

key causes the motor to start, the live battery enables it, the dead battery prevents it)."

Other ideas that are sometimes involved in causal reasoning are *causal uncorrelatedness* [Shafer, 1999] where if two variables share no common cause they are causally uncorrelated. This occurs if there are no single events that cause them to both change. Similarly, causal independence occurs when speaking about probabilities.

5.4 Observational And Non-Observational Data

Statistics is the traditional tool used to discover causality when handling experimental data. The standard method in the experimental sciences of recognizing causality is to perform randomized, controlled experiments. This produces experimental data. Depending on their design, randomized experiments may remove reasons for uncertainty whether or not a relationship is casual.

However, the data of greatest interest in the computational sciences, particularly data mining, is non-experimental. This is because analysis is performed on large quantities of warehoused data. In this domain, traditional statistical methods are either not useful an/or are often too computationally complex.

Even if some experimentation is possible, the amount of experimentation in contrast to the amount of data to be mined will be small. This said; some work has been done using chi-squared testing to reduce the search space [Silverstein, 1998].

Several areas can only wholly (economics, sociology) or partially develop non-experimental data. In these areas, investigators can either abandon the possibility of discovering causal relationships; or, claim that causality does not exist. There continue to be efforts to discover causal relationships areas where only non-observational data is available. Among the books considering causality in non-experimental data are: Asher [1983], Blalock [1964], Berry [1984], Hilborn [1997], Shipley [2000].

6. EPILOGUE

One of the corner stones of data mining is the development of association rules. Association rules greatest impact is in helping to make decisions. One measure of the quality of an association rule is its relative decision value. Association rules are often constructed using simplifying assumptions that lead to naïve results and consequently naïve and often wrong decisions. Perhaps the greatest area of concern is treating association rules as causal statements without understanding whether there is, in fact, underlying causality.

Causal relationships exist in the commonsense world. Knowledge of at least some causal effects is imprecise. Perhaps, complete knowledge of all possible factors might lead to a crisp description of whether an effect will occur. However, in our commonsense world, it is unlikely that all possible factors can be known. In commonsense, every day reasoning, we use approaches that do not require complete knowledge.

People recognize that a complex collection of elements causes a particular effect, even if the precise elements of the complex are unknown. They may not know what events are in the complex; or, what constraints and laws the complex is subject to. Sometimes, the details underlying an event are known to a fine level of detail, sometimes not. Generally, people are more successful in reasoning about a few large-grain sized events than many fine-grained events. Perhaps, this can transfer over to computational models of causality.

A lack of complete, precise knowledge should not be discouraging. People do things in the world by exploiting our commonsense *perceptions* of cause and effect. When trying to precisely reason about causality, we need complete knowledge of all of the relevant events and circumstances. In commonsense, every day reasoning, we use approaches that do not require complete knowledge. Often, approaches follow what is essentially a *satisficing* paradigm.

Instead of being concerned with all of the fined grained detail, a better approach may be to incorporate granulation using rough sets and/or fuzzy sets to soften the need for preciseness. And then accept impreciseness in the description. Each complex can be considered to be a granule. Larger complexes can be decomposed into smaller complexes. Thus, going from large-grained to small-grained.

Regardless of causal recognition and representation methodologies, it is important to decision making to understand when association rules have a causal foundation. This avoids naïve decisions and increases the perceived utility of rules with causal underpinnings.

References

H. Asher [1983] **Causal Modeling,** Sage Publications, Newbury Park, California

H. Blalock [1964] **Causal Inferences in Nonexperimental Research,** W. W. Norton, New York

W. Berry [1984] **Nonrecursive Causal Models,** Sage Publications, Newbury Park, California

A. Dawid [1999] "Who Needs Counterfactuals" in **Causal Models and Intelligent Data Management** (ed) A. Gammerman) Springer-Verlag, Berlin

W. Freeman [1995] **Societies Of Brains**, Lawrence Erlbaum, 1995

Gammerman (editor) [1999] **Causal Models and Intelligent Data Management,** Springer-Verlag, Berlin

M. Goodrich, W. Stirling, E. Boer [2000] "Satisficing Revisited," *Minds and Machines,* v 10, 79-109

Glymour [2001] **The Mind's Arrows,** MIT Press (Bradford), London

D. Hausman [1988] **Causal Asymmetries,** Cambridge University Press, Cambridge, U.K.

R. Hilborn, M. Mangel [1997] **The Ecological Detective: Confronting Models With Data,** Princeton University Press, Princeton, New Jersey

J. Hobbs [2001] "Causality," *Proceedings, Common Sense 2001, Fifth Symposium on Logical Formalizations of*

Commonsense Reasoning, New York University, New York, May, 145-155

J. Hobbs [2003] "Causality And Modality: The Case Of 'Would'," to appear in *Journal of Semantics*

K. Hoover [2003] "Lost Causes," *HES Conference,* Presidential Address, Durham, North Carolina

D. Hume [1742/1985] **Essays: Moral, Political, And Literary**, Eugene Miller (ed.), Liberty Classics, Indianapolis, 1985

D. Hume [1777/1902] "An Enquiry Concerning Human Understanding," in L. Selby-Bigge (ed.) **Enquiries Concerning Human Understanding And Concerning The Principles Of Morals,** 2nd edition, Clarendon Press, Oxford, 1902

D. Lewis [1973] **Counterfactuals**, Harvard University Press, Cambridge University Press

L. Mazlack [2003*a*] "Commonsense Causal Modeling In The Data Mining Context," IEEE ICDM FDM Workshop Proceedings, Melbourne, Florida, November 19 - 22, 2003

L. Mazlack [2003*b*] "Causality Recognition For Data Mining In An Inherently Ill Defined World," 2003 BISC FLINT-CIBI International Joint Workshop On Soft Computing For Internet And Bioinformatics, December, 2003

L. Mazlack [2004] "Granular Causality Speculations," NAFIPS 2004, June, Banff

E. Nelson [1998, October 6] "Why WalMart sings, 'Yes, we have bananas'," The Wall Street Journal, B1

C. Ortiz [1999a] "A Commonsense Language For Reasoning About Causation And Rational Action," Artificial Intelligence, v 108, n1-2, p 125-178

C. Oritz [1999b] "Explanatory Update Theory: Applications Of Counterfactual Reasoning To Causation," Artificial Intelligence, v 108, n 1-2, 125-178

Plato [380 B.C.] **Republic,** Book VII, paragraph 514-515, G.M.A. Grube (translator), Hacket Publishing, Indianapolis, Indiana, 1992, 186-187

J. Pearl, T. Verma [1991] ""A Theory Of Inferred Causation," *Principles Of Knowledge Representation And Reasoning: Proceedings Of The Second International Conference,* Morgan Kaufmann, 441-452

J. Pearl [2000] **Causality**, Cambridge University Press, New York, NY

R. Robins, L. Wasserman [1999], "On The Impossibility Of Inferring Causation From Association Without Background Knowledge," in (eds) C. Glymour, G. Cooper, **Computation, Causation, and Discovery** AAAI Press/MIT Press, Menlo Park, 305-321

R. Scheines, P. Spirtes, C. Glymour, C. Meek [1994] **Tetrad II: Tools For Causal Modeling**, Lawrence Erlbaum, Hillsdale, NJ

G. Shafer [1999] "Causal Conjecture," in **Causal Models and Intelligent Data Management** (ed) A. Gammerman) Springer-Verlag, Berlin

G. Shafer [1996] **The Art of Causal Conjecture,** MIT Press, Cambridge, Massachusetts

Y. Shoham [1990] "Nonmonotonic Reasoning And Causation," *Cognitive Science,* v14, 213-252

B. Shipley [2000] **Cause and Correlation in Biology,** Cambridge University Press, Cambridge, U.K.

Y. Shoham [1991] "Remarks On Simon's Comments," *Cognitive Science*, v15, 301-303

C. Silverstein, S. Brin, R. Motwani [1998*a*] "Beyond Market Baskets: Generalizing Association Rules To Dependence Rules," Data Mining And Knowledge Discovery, v 2, 39-68

C. Silverstein, S. Brin, R. Motwani, J. Ullman [1998*b*] "Scalable techniques for mining causal structures," *Proceedings 1998 VLDB Conference,* New York, NY, August 1998, 594-605

H. Simon [1952] "On The Definition Of The Causal Relation," *The Journal Of Philosophy,* v 49, 517-528. Reprinted in Herbert A. Simon, **Models Of Man**, John Wiley, New York, 1957

H. Simon [1953] "Causal ordering And Identifiability," Reprinted in Herbert A. Simon, **Models Of Man**, John Wiley, New York, 1957

H. Simon [1955] "A Behavioral Model Of Rational Choice," *Quarterly Journal Economics,* v 59, 99-118

H. Simon [1991] "Nonmonotonic Reasoning And Causation: Comment," *Cognitive Science,* v 15, 293-300

P. Spirtes, C. Glymour, R. Scheines [1993] **Causation, Prediction, and Search,** Springer-Verlag, New York

L. Zadeh [1998] "Maximizing Sets And Fuzzy Markov Algorithms," *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews,* v 28, 9-15

L. Zadeh [1999] "From computing With Numbers To Computing With Words - From Manipulation Of Measurements To Manipulation Of Perceptions," *IEEE Transactions on Circuits and Systems,* v 45, n 1, 108-119

Zeno (of Elea) [449 B.C.] **Paradox**, in G. Kirk and J. Raven (eds), **The Presocratic Philosophers: A Critical History with a Selection of Texts**, Cambridge, Cambridge University Press (1957)

Data Preprocessing and Data Mining as Generalization Process

E. Menasalvas[(1)], Anita Wasilewska[(2)]
[(1)] Facultad de Informatica.
Universidad Politecnica de Madrid. Spain.
emenasalvas@fi.upm.es
[(2)] Department of Computer Science,
State University of New York,
Stony Brook, NY, USA .
anita@cs.sunysb.edu

Abstract

We define here a model in which data preprocessing and data mining processes are are described as two different types of generalization.

1 Introduction

The preprocessing of data is the initial and often crucial step of the data mining process. We show here that the Generalization Model presented in [8] is strong enough to express not only the data mining stage of data mining process but the preprocessing stage as well. Moreover, we show that preprocessing stage and data mining stage generalize data in a different way and that in fact, the generalization proper (i.e. strong generalization in our model) occurs only at the data mining stage. The preprocessing operations are expressed in the model as a weak generalization. We show that they lead to the strong information generalization in the next, data mining proper stage and improve the quality (granularity) of the generalization process.

2 Generalization Model

It is natural that when building a model of the data mining process one has to include data preprocessing methods and algorithms, i.e. one has to model within it preprocessing stage as well as the data mining proper stage. In order to achieve this task we introduce the preprocessing and the data mining generalization relations (definitions 2.11, 2.10, respectively. We show that they are particular cases of the information generalization relation as defined in our generalization model (definition 2.1). We also prove (theorem 2.3) that the preprocessing relation is a special case of the weak information generalization relation and it is disjoint with our data mining generalization relation. This means that within the framework of our general model we were able to distinguish (as we should have) the preprocessing generalization from the generalization that occurs in the data mining proper stage.

Definition 2.1 A Generalization Model *is a system*

$$\mathcal{GM} = (U,\ \mathcal{K},\ \mathcal{G},\ \preceq)$$

where

$U \neq \emptyset$ *is the* **universe**,

$\mathcal{K} \neq \emptyset$ *is the set of* **generalization states**,

$\prec\ \subseteq\ \mathcal{K} \times \mathcal{K}$ *is* **a generalization relation***;*

We assume that the relation $\preceq$ is transitive.

$\mathcal{G} \neq \emptyset$ *is the set of* **generalizations operators** *such that for every $G \in \mathcal{G}$, for every $K, K' \in \mathcal{K}$,*

$$G(K) = K' \ \ if\ and\ only\ if\ \ K \preceq K'.$$

Definition 2.2 A Strong Generalization Model *is the generalization model (definition 2.1) in which the information generalization relation is not reflexive. We denote the generalization relation of the strong model by $\prec$ and call it a strong generalization relation.*
A Weak Generalization Model *is the generalization model (definition 2.1) in which the information generalization relation is reflexive. We denote the generalization relation of the weak model by $\preceq$ and call it a weak generalization relation.*

Any Data Mining process starts with a certain initial set of data. The model of such a process depends on representation of this data, i.e. it starts with an initial information system

$$I_0 = (U_0, A_0, V_{A_0}, f_0)$$

and we adopt the **universe** U_0 **as the universe of the model**, i.e.

$$\mathcal{GM} = (U_0,\ \mathcal{K},\ \mathcal{G},\ \preceq).$$

In preprocessing stage of data mining process we might perform the following standard operations:

1. eliminate some records, obtaining as result a new information system with an universe $U \subseteq U_0$, or

2. eliminate some attributes, obtaining as result the information system I with the set of attributes $A \subset A_0$, or

3. perform some operations on values of attributes: normalization, clustering, application of concept hierarchy on, etc..., obtaining some set V_A of values of attributes that is similar, or equivalent to V_0. We denote it by

$$V_A \sim V_0.$$

Given an attribute value $v_a \in V_A$ and a corresponding attribute $v_a^0 \in V_0$ (for example v_a being a normalized form of v_a^0 or v_a being a more general form as defined by concept hierarchy of v_a^0) we denote this correspondence by

$$v_a \sim v_a^0.$$

We call any information system I obtained by any of the above operation **a subsystem of** I_0. We put it formally in the following definition.

Definition 2.3 *Given two information systems* $I_0 = (U_0, A_0, V_{A_0}, f_0)$ *and*
$I = (U, A, V_A, f)$, *we say that* I **is a subsystem of** I_0 *and denote it as*

$$I \subseteq I_0$$

if and only if the following conditions are satisfied

(i) $U \subseteq U_0$,

(ii) $A \subseteq A_0$, $V_A \sim V_0$, *and*

(iii) *the information functions* f *and* f_0 *are such that*

$$\forall x \in U \forall a \in A(f(x,a) = v_a$$

$$\Leftrightarrow \exists v_a^0 \in V_0(f_0(x,a) = v_a^0 \cap v_a^0 \sim v_a)).$$

In the data analysis, preprocessing and data mining although we start the process with the information table (i.e. we define the lowest level of information generalization as the relational table) the meaning of the intermediate and final results are considered to be of a higher level of generalization. We represent those levels of generalization by a sets of objects of the given (data mining) universe U, as in [1], [6].

This approach follows the granular view of the data mining and is formalized within a notion of knowledge generalization system, defined in [8] as follows.

Definition 2.4 A knowledge generalization system *based on the information system* $I = (U, A, V_A, f)$ *is a system*

$$K_I = (\mathcal{P}(U), A, E, V_A, V_E, g)$$

where

E *is a finite set of* **knowledge attributes** *(k-attributes) such that* $A \cap E = \emptyset$.

V_E *is a finite set of* **values of k- attributes**.

g *is a partial function called* **knowledge information function***(k-function)*

$$g : \mathcal{P}(U) \times (A \cup E) \longrightarrow (V_A \cup V_E)$$

such that

(i) $g \mid (\bigcup_{x \in U}\{x\} \times A) = f$

(ii) $\forall_{S \in \mathcal{P}(U)} \forall_{a \in A}((S,a) \in dom(g) \Rightarrow g(S,a) \in V_A)$

(iii) $\forall_{S \in \mathcal{P}(U)} \forall_{e \in E}((S,a) \in dom(g) \Rightarrow g(S,e) \in V_E)$

Any set $S \in \mathcal{P}(U)$ i.e. $S \subseteq U$ is often called **a granule** or **a group** of objects.

Definition 2.5 *The set*

$$Gr_K = \{S \in \mathcal{P}(U) : \exists b \in (E \cup A)((S,b) \in dom(g))\}$$

is called **a granule universe** *of* K_I.

Observe that g is a total function on Gr_K.

Definition 2.6 *We call the system* $K = (Gr_K, E, V_E, g)$ *a* **granule knowledge generalization** *system.*

The condition **(i)** of definition 2.4 says that when $E = \emptyset$ the k-function g is total on the set $\{\{x\} : x \in U\} \times A$ and

$$\forall x \in U \forall a \in A(g(\{x\}, a) = f(x,a)).$$

Definition 2.7 *The set*

$$\mathcal{P}^{obj}(U) = \{\{x\} : x \in U\}$$

is called an **object universe**. *The knowledge generalization system*

$$K^{obj} = (\mathcal{P}^{obj}(U), A, \emptyset, V_A, \emptyset, g) = (\mathcal{P}^{obj}(U), A, V_A, g)$$

is called an **object knowledge generalization** *system.*

Theorem 2.1 *For any information system*

$$I = (U, A, V_A, f),$$

the object knowledge generalization system

$$K_I^{obj} = (\mathcal{P}^{obj}(U), A, V_A, g)$$

is isomorphic with I. *We denote it by*

$$I \simeq K_I^{obj}.$$

The function

$$F : U \longrightarrow \mathcal{P}^{obj}(U), \;\; F(x) = \{x\}$$

establishes (by condition **(i)** of definition 2.4) the required isomorphism of K_I^{obj} and I.

Given initial information system $I_0 = (U_0, A_0, V_{A_0}, f_0)$, the object knowledge generalization system (definition **??**)

$$K_{I_0}^{obj} = (\mathcal{P}^{obj}(U_0), A, V_A, g)$$

isomorphic with I_0 i.e. $K_{I_0}^{obj} \simeq I_0$ is also called **the initial knowledge generalization** system.

Data Mining process in the preprocessing stage consists of transformations the initial I_0 into some of $I \subseteq I_0$ and subsequently, in the data mining proper stage, of transformations of knowledge generalizations systems K_I based on $I \subseteq I_0$. The transformations in practice are defined by different Data Mining algorithms, and in our model by appropriate generalization operators. Any data mining transformation starts, for unification purposes with corresponding initial knowledge generalization systems $K_I^1 \simeq I$. We hence adopt the following definition of the set $\mathcal{K}$ of knowledge states.

Definition 2.8 *We adopt the set*

$$\mathcal{K} = \{K_I : \; I \subseteq I_0\}$$

of all knowledge generalization systems based on the initial information system (input data) I_0 as the set of **knowledge states of** $\mathcal{GM}$.

The set $\mathcal{K}^{prep} \subseteq \mathcal{K}$ such that

$$\mathcal{K}^{prep} = \{K_I^{obj} : \; K_I^{prep} \simeq I \; and \; I \subseteq I_0\}$$

is called **a set of preprocessing knowledge states**, *or preprocessing knowledge systems of $\mathcal{GM}$.*

Definition 2.9 *Given set $\mathcal{K}$ of knowledge states (definition 2.8) based on the input data I_0 and $K, K' \in \mathcal{K}$ i.e.*

$$K = (\mathcal{P}(U_0), A, E, V_A, V_E, g),$$

$$K' = (\mathcal{P}(U_0), A', E', V_{A'}, V_{E'}, g').$$

Let $G_K, G_{K'}$ be granule universes (definition 2.5) of K, K' respectively. We define **a weak generalization relation**

$$\preceq \subseteq \mathcal{K} \times \mathcal{K}$$

as follows:

$$K \preceq K' \;\; if \; and \; only \; if$$

(i) $|G_{K'}| \leq |G_K|$,

(ii) $A' \subseteq A$.

If $K \preceq K'$ we say that the system K' **is more or equally general as** K.

Observe that the relation $\preceq$ is reflexive and transitive, but is not antisymmetric, as systems K and K' such that $K \preceq K'$ may have different sets of knowledge attributes and knowledge functions.

Definition 2.10 *Let $\preceq \subseteq \mathcal{K} \times \mathcal{K}$ be relation defined in the definition 2.9.*
A relation

$$\prec_{dm} \subseteq \preceq$$

such that it satisfies additional conditions:

(iii) $|G_{K'}| < |G_K|$,

(iv) $\exists S \in G_{K'}(|S| > 1)$

is called **a data mining generalization relation.**

Theorem 2.2 *The relation $\prec_{dm}$ is not reflexive, and the following properties hold.*

(1) *The weak generalization relation of definition 2.9 is the weak information generalization relation of the generalization model (definition 2.1),*

(2) $\prec_{dm} \subset \preceq$,

(3) *$\prec_{dm}$ is a strong information generalization of the definition 2.2 and if $K \prec_{dm} K'$ we say that the system K'* **is more general then** K.

The preprocessing of data is the initial (an crucial) step of the data mining process. We show now that we can talk about preprocessing operations within our generalization model. The detailed analysis of preprocessing methods and techniques within it will be a subject of separate paper.

Definition 2.11 *Let $\mathcal{K}^{prep} \subseteq \mathcal{K}$ be a the set of preprocessing states (definition 2.8). A relation $\preceq_{prep} \subseteq \preceq$ defined as follows:*

$$\preceq_{prep} = \{(K, K') \in \preceq : \;\; K, K' \in \mathcal{K}^{prep}\}$$

is called **a preprocessing generalization relation**.

Theorem 2.3 *The preprocessing generalization relation is a weak generalization relation and is not a data mining generalization relation i.e.*

$$\preceq_{prep} \cap \prec_{dm} \; = \; \emptyset.$$

Within our framework the systems K, K' such that $K \preceq_{prep} K'$ are, in fact, equally general. So why do we call some preprocessing operations a "generalization"? There are two reasons. One is that traditionally some preprocessing operations have been always called by this name. For example we usually state that we "generalize" attributes by clustering, by introducing attributes hierarchy, by aggregation, etc. as stated on page 114 of the most comprehensive, as far, Data Mining book ([2]).

> ...."Data transformation (preprocessing stage) can involve the following .....
> **Generalization** of the data , where low-level or "primitive" (raw) data are replaced by higher -level concepts through the use of concept hierarchies. For example, categorical attributes ..... can be generalized to higher level concepts. Similarly, values for numeric attributes,like ... may be mapped to higher level concepts." ....

The second, more important reason to call some preprocessing operations a (weak) generalization is that they lead to the "strong" information generalization in the next, data mining proper stage and we perform them in order to improve the quality (granularity) of the generalization.

3 Generalization Models for Data Preprocessing and Data Mining Process

It is natural that when building a model of the data mining process one has to include data preprocessing methods and algorithms, i.e. one has to model within it preprocessing stage as well as the data mining proper stage. In order to achieve this task we choose the notion of weak information generalization relation as a component of our (the most general) notion of the generalization model (definition 2.1). We have then introduced the preprocessing and the data mining generalization relations (definitions 2.11, 2.10, respectively) and proved (theorem 2.3) that the preprocessing relation is a special case of the weak information generalization relation and it is disjoint with our data mining generalization relation. This means that within the framework of our general model we were able to distinguish (as we should have) the preprocessing generalization from the data mining proper stage generalization.

Consequently we define here the semantic models of data preprocessing, data mining, and data mining precess. They are all particular cases of our generalization model (definition 2.1).

Definition 3.1 *When we adopt the preprocessing generalization relation $\preceq_{prep}$ (definition 2.11) as the information generalization relation of the generalization model $\mathcal{GM} = (U,\ \mathcal{K},\ \mathcal{G},\ \preceq)$ (definition 2.1) we call the model thus obtained a* **Preprocessing Model** *and denote it* $\mathbf{PM}$*, i.e.*

$$\mathbf{PM} = (U,\ \mathcal{K}^{prep},\ \mathcal{G}_{prep},\ \prec_{prep})$$

where

- *$\mathcal{K}^{prep}$ is the set of preprocessing knowledge states (definition 2.8),*
- *$\mathcal{G}_{prep} \subseteq \mathcal{G}$ called a set of preprocessing generalization operators (to be defined separately).*

The data mining proper stage is determined by the data mining generalization relation and is defied formally as follows.

Definition 3.2 *Let $\prec_{dm}$ be the data mining generalization relation (definition 2.10). A* **Data Mining Model** *is a system*

$$\mathbf{DM} = (U,\ \mathcal{K},\ \mathcal{G}_{dm},\ \prec_{dm})$$

where

$$\mathcal{G}_{dm} \subseteq \mathcal{G}$$

for $\mathcal{G}_{dm} \neq \emptyset$ being a set of data mining generalization operators defined in the next section.

Now, we express the whole data mining process within our generalization model as follows.

Definition 3.3 *A* **Data Mining Process Model** *is a system*

$$\mathbf{DMP} = (U,\ \mathcal{K},\ \mathcal{G}_p,\ \preceq_p),$$

where

(i) $\preceq_p = \preceq_{prep} \cup \prec_{dm}$,

(ii) $\mathcal{G}_p = \mathcal{G}_{prep} \cup \mathcal{G}_{dm}$,

The set $\mathcal{G}_{dm}$ of data mining is defined in detail in [8], the set $\mathcal{G}_{prep}$ of data preprocessing operators will be a subject of a separate paper.

References

[1] M. Hadjimichael, A. Wasilewska. *A Hierarchical Model for Information Generalization.* Proceedings of the 4th Joint Conference on Information Sciences, Rough Sets, Data Mining and Granual Computing (RSDMGrC'98), North Carolona, USA, vol.II, 306–309.

[2] J. Han, M. Kamber. *Data Mining: Concepts and Techniques* Morgan, Kauffman, 2000

[3] M. Inuiguchi, T. Tanino. *Classification versus Approximation oriented Generalization of Rough Sets* Bulletin of International Rough Set Society, Volume 7, No. 1/2.2003

[4] T.Y. Lin. *Database Mining on Derived Attributes Proceedings of Third International Conference RSCTC'02*, Malvern, PA, USA, October 2002, pp. 14 - 32. Springer Lecture Notes in Artificial Intelligence.

[5] Juan F.Martinez, Ernestina Menasalvas, Anita Wasilewska, Covadonga Fernández, M. Hadjimichael. *Extension of Relational Management System with Data Mining Capabilities* Proceedings of Third International Conference RSCTC'02, Malvern, PA, USA, October 2002, pp. 421- 428. Springer Lecture Notes in Artificial Intelligence.

[6] Ernestina Menasalvas, Anita Wasilewska, Covadonga Fernández *The lattice structure of the KDD process: Mathematical expression of the model and its operators* International Journal of Information Systems) and FI (Fundamenta Informaticae; special isue2001, pp. 48 - 62.

[7] Ernestina Menasalvas, Anita Wasilewska, Covadonga Fernández, Juan F. Martinez. *Data Mining- A Semantical Model* Proceedings of 2002 World Congres on Computational Intelligence, Honolulu, Hawai, USA, May 11- 17, 2002, pp. 435 - 441.

[8] Ernestina Menasalvas, Anita Wasilewska. *Data Mining Operators* Proceedings of ICDM'04, The Fourth IEEE International Conference on Data Mining, Brighton, UK, Nov 1-4, 2004 - to appear.

[9] Pawlak, Z. *Information systems - theoretical foundations* Information systems, 6 (1981), pp. 205-218

[10] Pawlak, Z. *Rough Sets- theoretical Aspects Reasoning About Data* Kluwer Academic Publishers 1991

[11] Skowron, A. *Data Filtration: A Rough Set Approach* Proceedings de Rough Sets, Fuzzy Sets and Knowledge Discovery. (1993). Pag. 108-118

[12] A. Wasilewska, Ernestina Menasalvas Ruiz, María C. Fernández-Baizan. *Modelization of rough set functions in the KDD frame* 1st International Conference on Rough Sets and Current Trends in Computing (RSCTC'98) June 22 - 26 1998, Warsaw, Poland.

[13] Wojciech Ziarko, Xue Fei. *VPRSM Approach to WEB Searching* Proceedings of Third International RSCTC'02 Conference, Malvern, PA, USA, October 2002, pp. 514- 522. Springer Lecture Notes in Artificial Intelligence.

[14] Wojciech Ziarko. *Variable Precision Rough Set Model* Journal of Computer and Systen Sciences, Vol.46. No.1, pp. 39-59, 1993.

[15] J.T. Yao, Y.Y. Yao. *Induction of Classification Rules by Granular Computing* Proceedings of Third International RSCTC'02 Conference, Malvern, PA, USA, October 2002, pp. 331-338. Springer Lecture Notes in Artificial Intelligence.

The iterative and interactive data mining process: the information systems development and knowledge management perspectives

Mykola Pechenizkiy
Dept. of CS and ISs
University of Jyväskylä
Finland
mpechen@cs.jyu.fi

Seppo Puuronen
Dept. of CS and ISs
University of Jyväskylä
Finland
sepi@cs.jyu.fi

Alexey Tsymbal
Dept. of CS
Trinity College Dublin
Ireland
tsymbalo@tcd.ie

Abstract

Data mining (DM) and knowledge discovery are intelligent tools that help to accumulate and process data and make use of it. We review several existing frameworks for data mining that originate from different paradigms. These DM frameworks address various DM algorithms for the different steps of the DM process. However, usually each DM framework explains the nature of one particular type of the algorithms. Recent research has shown that many real-world problems require the integration of several DM algorithms originating from different paradigms in order to produce a better solution. In this paper we introduce our vision how DM process modeling can take advantage of the research made in the areas of Information Systems Development and Knowledge Management.

1. Introduction

Data mining (DM) and knowledge discovery are intelligent tools that help to accumulate and process data and make use of it [6]. Data mining bridges many technical areas, including databases, statistics, machine learning, and human-computer interaction. The set of data mining processes used to extract and verify patterns in data is the core of the knowledge discovery process [24]. These processes include data cleaning, feature transformation, algorithm and parameter selection, and evaluation, interpretation and validation (Figure 1).

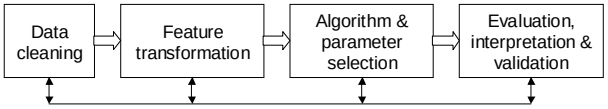


Figure 1. Data mining process (adapted from [24])

The idea of learning from data is far from being new. However, likely due to developments in the database management field and due to the great increase of data volumes being accumulated in databases the interest in DM has become very intense. Numerous DM algorithms have recently been developed to extract knowledge from large databases. Currently, most research in DM focuses on the development of new algorithms or improvement of the speed or the accuracy of the existing ones [24].

Relatively little has been published about theoretical foundations of DM. A few theoretical approaches to DM were proposed in [16]. A motivation for DM foundations development and requirements for a theoretical DM framework were also considered in [16]: a theoretical framework should be simple and easy to apply; it should contribute to DM algorithms and DM systems development; it should be able to model typical DM tasks like clustering, classification and rule discovery; it should recognize that DM is an iterative and interactive process, where a user has to be involved.

In this paper (in Section 2) we consider several existing frameworks for data mining based on statistical, data compression, machine learning, philosophy of science, and database paradigms. We consider their advantages and limitations analyzing what these approaches are able to explain in the data mining process and what they do not. We believe that a reader will notice that each of the above DM frameworks is limited mainly to addressing one particular type of DM algorithms and that they rarely address the issues of iteration and interactivity.

We introduce our vision how DM process modeling can take advantage of the research made in the areas of Information Systems Development (ISD) and Knowledge Management (KM). In Section 3 we refer to the traditional information system (IS) framework presented in [11] that is widely known in the IS community and is a synthesis of many other frameworks considered before it. For us this framework is more substantial than the others since it also focuses on the development process of information systems. We consider Nunamaker's information systems development research framework [21] in the context of DM. We demonstrate how theoretical, constructive and experimental approaches can be applied iteratively and/or in parallel for the development of an artefact (a data-mining tool).

In Section 4 we consider DM research in the context of a complex adaptive system that creates, receives, stores, retrieves, transforms, and transmits (meta-) knowledge to improve its ability to adapt to the environment and to develop (or better utilize available) DM techniques.

We conclude briefly in Section 5 with a summary and further research topics.

2. Review of some existing theoretical frameworks for data mining

It this section we review basic DM frameworks and show that they deal mainly with DM techniques as such. Philosophy of science may help to understand the nature and scope of data mining techniques. However, as we conclude, present-day frameworks for DM lack in describing it as iterative and interactive process and in accounting social dimension of DM, i.e. involvement of a user.

2.1. Statistical paradigms

Generally, it is possible to consider the task of data mining from the statistical point of view, emphasizing the fact that DM techniques are applied to larger datasets than it is commonly done in applied statistics [10]. Thus the analysis of the appropriate statistical literature, where strong analytical background is accumulated, would solve most data mining problems. Many data mining tasks naturally may be formulated in statistical terms, and many statistical contributions may be used in data mining in a quite straightforward manner [9].

According to [5] there exist two basic statistical paradigms that are used in theoretical support for DM. The first paradigm is so-called “Statistical experiment”. It can be seen from three perspectives: Fisher’s version that uses the inductive principle of maximum likelihood, Neyman-E.S. Pearson-Wald’s version that is based on the principle of inductive behavior, and Bayesian version that is based on the principle of maximum posterior probability. An evolved version of “Statistical experiment” paradigm is “Statistical learning from empirical process” paradigm [23]. Generally, many data mining tasks can be seen as the task of finding the underlying joint distribution of variables in the data. Good examples of this approach would be a Bayesian network or a hierarchical Bayesian model, which give a short and understandable representation of the joint distribution. Data mining tasks dealing with clustering and/or classification fit easily into this approach.

The second statistical paradigm is called “Structural data analysis” and can be associated with singular value decomposition methods, which are heavily used, for example, in text mining applications.

A deeper consideration of data mining and statistics can be found in [8]. Here, we just point out that the volume of the data being analysed and different educational background of researchers are not the most important issues that constitute the difference between the areas. Data mining is an applied area of science and limitations in available computational resources is a big issue when applying results from statistics to data mining. An important point is that the theoretical framework of statistics does not concern much about data analysis as an iterative process that generally includes several steps. However, there are persons (mainly with strong statistical background) who consider DM as a branch of statistics, because many DM tasks may be perfectly represented in terms of statistics.

2.2. The data compression paradigm

A data compression approach to data mining can be stated in the following way: compress the dataset by finding some structure or knowledge for it, where knowledge is interpreted as a representation that allows coding the data using a fewer amount of bits. For example, the minimum description length (MDL) principle [17] can be used to select among different encodings accounting to both the complexity of a model and its predictive accuracy.

Machine learning practitioners have used the MDL principle in different interpretations to recommend that even when a hypothesis is not the most empirically successful among those available, it may be the one to be chosen if it is simple enough. The idea is in trading between consistency with training examples and empirical adequacy by predictive success as it is, for example, with accurate decision tree construction. Bensusan [2] connects this to another methodological issue, namely that theories should not be *ad hoc* that is they should not overfit the examples used to build it. Simplicity is the remedy for being *ad hoc* both in the recommendations of philosophy of science and in the practice of machine learning.

The data compression approach has also connections with the rather old Occam’s razor principle that was introduced in the 14^{th} century. The most commonly used formulation of this principle in data mining is "when you have two competing models which make exactly the same predictions, the one that is simpler is better".

Many (if not every) data mining techniques can be viewed in terms of the data compression approach. For example, association rules and pruned decision trees can be viewed as ways of providing compression of parts of the data. Clustering approaches can also be considered as a way of compressing the dataset. There is a connection to the Bayesian theory for modeling the joint distribution – any compression scheme can be viewed as providing a distribution on the set of possible instances of the data.

2.3. The machine learning paradigm

The machine learning (ML) paradigm “let the data suggest a model” can be seen as a practical alternative to

the statistical paradigm “fit a model to the data”. It is certainly reasonable in many situations to fit a small dataset to a parametric model based on a series of assumptions. However, for applications with large volumes of data under analysis the ML paradigm may be beneficial because of its flexibility within a nonparametric, assumption-free nature.

We would like to focus here on a constructive induction approach. Constructive induction is a learning process that consists of two intertwined phases, one of which is responsible for the construction of the “best” representation space and the second concerns generating hypotheses in the found space [15]. Constructive induction methods are classified into three categories: data-driven (information from the training examples is used), hypothesis-driven (information from the analysis of the form of intermediate hypothesis is used) and knowledge-driven (domain knowledge provided by experts is used) methods. Any kind of induction strategy (implying induction, abduction, analogies and other forms of non-truth preserving and non-monotonic inferences) may potentially be used. However, the focus is usually on operating higher-level data-concepts and theoretical terms rather than pure data.

Many DM techniques that apply wrapper/filter approaches to combine feature selection, feature extraction, or feature construction processes (as means of dimensionality reduction and/or as means of search for better representation of the problem) and a classifier or other type of learning algorithm may be considered as constructive induction approaches.

2.4. The philosophy of science paradigm

Categorization of subjectivist and objectivist approaches [4] can be considered in the context of DM. The possibility to compare nominalistic and realistic ontological believes gives us an opportunity to consider data that is under analysis as descriptive facts or constitutive meanings. The analysis of voluntaristic as opposed to deterministic assumptions about the nature of every instance constituting the observed data directs our attitude and understanding of that data. One possibility is to view every instance and its state as determined by the context and/or a law. Another position consists in consideration of each instance as autonomous and independent. An epistemological assumption about how a criterion to validate knowledge discovered (or a model that explains reality and allows making predictions) can be constructed may impact the selection of appropriate data mining technique. From the positivistic point of view such a model-building process can be performed by searching for regularities and causal relationships between the constitutive constructs of a model. And anti-positivism suggests analyzing every individual observation trying to understand it and making an interpretation. Probably some of case-based reasoning approaches can be related to anti-positivism’s vision of the reality.

An interesting difference in the views of the reality can be found considering ideographic as opposed to nomothetic methodological disputes. The nomothetic school does not see the real world as a set of random happenings. And if so, there must be rules that describe some regularities. Thus, nomothetic sciences seek for establishing abstract (general) laws that describe indefinitely repeatable events and processes. On the contrary, ideographic sciences are aimed to understand the unique and nonrecurrent events. They have connection to the ancient doctrine that “all is flux”. If everything were always changing, then any generalization intending to be applied for two or more presumably comparable phenomena would never be true. And ‘averages’ of some measures (from the nomothetic way of thinking) usually is not able to represent the behaviour of a single event or entity.

2.5. The database paradigm

A database perspective on data mining and knowledge discovery was introduced in [12]. The main postulate of their approach is: “there is no such thing as discovery, it is all in the power of the query language”. That is, one can benefit from viewing common data mining tasks not as dynamic operations constructing new pieces of information, but as operations finding unknown (i.e. not found so far) but existing parts of knowledge.

In [3] an inductive databases framework for the data mining and knowledge discovery in databases (KDD) modeling was introduced. The basic idea here is that the data-mining task can be formulated as locating interesting sentences from a given logic that are true in the database. Then discovering knowledge from data can be viewed as querying the set of interesting sentences. Therefore the term “an inductive database” refers to such a type of databases that contains not only data but a theory about the data as well [3].

This approach has some logical connection to the idea of deductive databases, which contain normal database content and additionally a set of rules for deriving new facts from the facts already present in the database. This is a common inner data representation. For a database user, all the facts derivable from the rules are presented, as they would have been actually stored there. In a similar way, there is no need to have all the rules that are true about the data stored in an inductive database. However, a user may imagine that all these rules are there, although in reality, the rules are constructed on demand. The description of an inductive database consists of a normal relational database structure with an additional structure

for performing generalizations. It is possible to design a query language that works on inductive databases. Usually, the result of a query on an inductive database is an inductive database as well. Certainly, there might be a need to find a solution about what should be presented to a user and when to stop the recursive rule generation while querying. We refer an interested reader to [3].

2.6. Conclusions on considered frameworks

The reductionist approach of viewing data mining in terms of statistics has advantages of the strong theoretical background, and easy-formulated problems. The data compression and constructive induction approaches have relatively strong analytical background, as well as connections to the philosophy of science. In addition to the above frameworks there exists an interesting microeconomic view on data mining [14], where a utility function is constructed and it is tried to be maximized. The data mining tasks concerning processes like clustering, regression and classification fit easily into these approaches. Another interesting approach based on granular and rough computing can be found in [15]

One way or another, we can easily see the exploratory nature of the frameworks for DM. Different frameworks account different data mining tasks and allow preserving and presenting background knowledge. However, what seems to be lacking in most approaches, are the ways for taking the iterative and interactive nature of the data mining process into account [16]. Furthermore, none of the above frameworks considers data mining in the context of an adaptive system that processes information.

3. The information systems-based paradigm applied to data mining

Information Systems (IS) are powerful instruments for organizational problem solving through formal information processing. In this section we consider a DM system as an adaptive IS that is armed with a number of techniques to be applied for a problem at hand. Since the variety of problems is changing over time, such a system has to be developed continuously towards the efficient utilization of available techniques and improvement of these techniques. We introduce an IS framework and an IS development framework and then consider how data mining can be seen as an iterative and interactive development process within this framework.

3.1. The information systems perspective

The traditional framework presented by Ives et al. [11] is widely known in the IS community. In this framework an IS is considered in an organizational environment that is further surrounded by an external environment. According to this framework an IS itself includes three environments: a user environment, an IS development environment, and an IS operations environment. There are accordingly three processes through which an IS has interaction with its environments: the use process, the development process, and the operation process.

Analogously, a data-mining system can be considered as a system with a user environment, a DM development environment, and a DM operations environment (Figure 2).

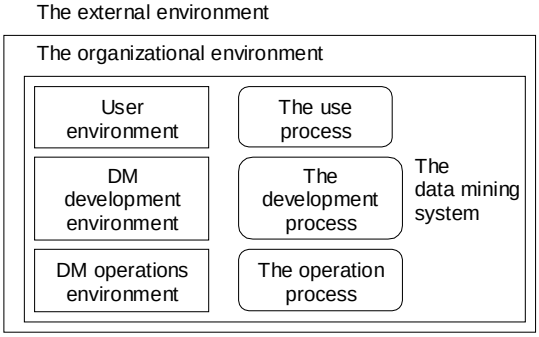


Figure 2. A model for DM research (adapted from [11])

However, in this paper, we focus on the development process of DM system and leave the operation and use processes for further research.

3.2. The IS development perspective

Iivari *et al.* [11] relate the IS development process to the constructive type of research based on their philosophical belief that development always involves creation of some new artefacts – conceptual (models, frameworks) or more technical artefacts (software implementations). The research approach is classified as constructive whereas scientific knowledge is used to produce either useful systems or methods, including development of prototypes and processes. It has been argued that the constructive type of research is important especially for applied disciplines of information systems and computer science [11], and DM may be considered as such a discipline.

In [21] system development is considered as a central part of a multi-methodological information systems research cycle (Figure 3). Theory building involves discovery of new knowledge in the field of study, however it rarely contributes directly to practice. Nevertheless, the new theory often (if not always) needs to be tested in the real world to show its validity, recognize its limitations and make refinements according to observations made during its application. Therefore research methods can be subdivided into basic and applied research, as naturally both are common for any large system development project [21]. A proposed theory

leads to the development of a prototype system in order to illustrate the theoretical framework on the one hand, and to test it through experimentation and observation with subsequent refinement of the theory and the prototype in an iterative manner. Such a view presents the framework of IS as a complete, comprehensive and dynamic research process. It allows multiple perspectives and flexible choices of methods to be applied during different stages of the research process.

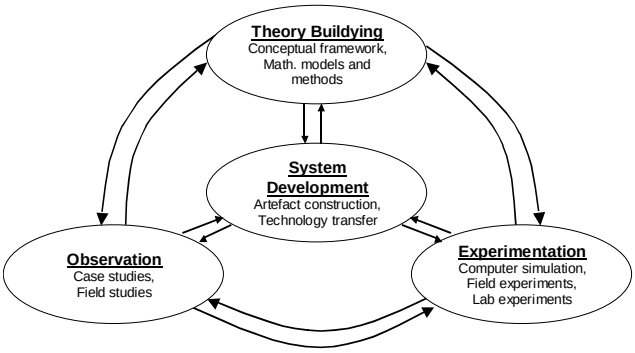


Figure 3. A multimethodological approach to the construction of an artefact for DM (adapted from [21])

3.3 Data mining as artefact development

In this subsection we consider applying theoretical, constructive and experimental approaches with regard to Nunamaker's framework in the context of data mining.

If a stated research problem includes a verb like introduce, improve, maintain, cease, extend, correct, adjust, enhance and so on, the study likely belongs to the area of constructive research. Indeed, these are the kind of actions that researchers in the area of data mining perform, when developing new theories and their applications.

It is obvious that in order to construct a good artefact background knowledge is needed both about the artefact's components, that are basic data mining techniques in the DM context and about components' cooperation, that are commonly selection and combination techniques in the DM context. Beside this some background knowledge is also needed about artefact's external environment, that are different real-world problems, often called just datasets in the DM context.

The evaluation process is an essential part of constructive research. Usually, the experimental approach is used to evaluate a DM artefact. The experimental approach, however, can be beneficial for theory testing and can result in new pieces of knowledge thus contributing to the theory-creating process.

It does not matter is the subject of evaluation a new theory or a new artefact, the general principle of evaluation must hold. This general principle requires that the new theory or artefact must be better than its best challenger so far. A 'goodness' criterion of a built theory or an artefact can be multidimensional and it is sometimes difficult to be defined because of mutual dependence between the compromising variables. However, it is more or less easy to construct a criterion based on such estimates as accuracy of a built model and its performance. From the other hand, it is more difficult or even impossible to include into a criterion such important aspects as interpretability of the artefact's output because estimates of such kind are usually subjective and can be evaluated only by the end-users of a system.

Experimental studies are often divided in the IS community into 'field' or 'laboratory'-based. In the first case different approaches are tested on so-called real-world datasets with real users. In the second case systematically controlled experiments can be organized. Controlled experiments sometimes might produce more beneficial results for theory creating, since unlike real world datasets, synthetically generated data allow to test exactly the desired number of characteristics while keeping all the others unchanged.

Theory testing might be seen at different levels. A low-level task is to evaluate how well a built model works. Another task is to analyse how the built model performs comparing to the other models. Then it is usually necessary to compare the algorithm selected to build the models with other algorithm(s). Finally, when 'laboratory' experiments and evaluation are finished, it is necessary to organize 'field' experiments.

These approaches can be applied iteratively and/or in parallel for the development of an artefact – a data-mining tool, and contribute to theory creation and theory testing.

4. The knowledge management paradigm applied to data mining

In this section we propose to consider DM research in the context of a complex adaptive system that creates, receives, stores, retrieves, transforms, and transmits meta-knowledge to improve its ability to adapt to the environment and to utilize available DM techniques more efficiently and effectively.

4.1 Different types of knowledge and their transformations

One common definition of knowledge is "justified belief that increases an entity's capacity for effective action" [20]. In this section we consider different types of knowledge and their potential in the effective work and

performance of a knowledge discovery system (KDS).

Organizational knowledge can be seen as a hierarchical network of rules about specific data or information that has explanatory, predictive, and functional power. These rules are categorized as procedural and declarative. The procedural rules are "know-how" rules and the declarative rules are "know-what" rules. "Knowing where" and "knowing when" represent spatial and temporal contexts of knowledge validity respectively. "Knowing why" provides a KDS with explanatory facilities when it is necessary to argue why a certain DM strategy is recommended or applied.

Beside these technical issues of knowing with respect to knowledge management in KDS, we recognize three basic types of organizational types of knowing. "Knowing what-for" represents DM goals that reflect business goals, and account knowledge of the application domain. "Knowing who" involves information about "who knows what". As the complexity of the knowledge increases, co-operation between groups (of DM experts, DM practitioners or intelligent knowledge repositories) tends to develop. "Knowing how much" accounts benefits of produced knowledge, resources required, related risks, etc. Although being important the last two knowings are not in the focus of this section.

In Figure 4 we present the concept of knowledge and its transformations adapted from [22].

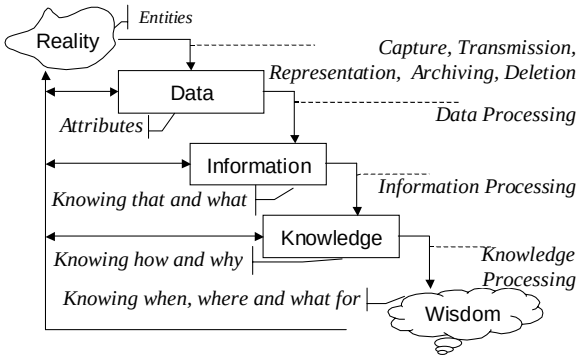


Figure 4. Transformations of data and knowledge concepts (adapted from [22])

Reality is related to entities whereas data are the attributes from those entities. When the current business problem is formulated as a DM task, data represents those attributes. Information is the result of data processing and the information associated with the "knowing that and what" type of knowledge. The concept of knowledge is defined as "knowing how and why" and is the result of information processing. Wisdom is associated with the knowing context of where and when certain knowledge is relevant and valid. All these types of knowing are utilized in many DM techniques. In the time dimension, data naturally deals with the past, information is used in the present and knowledge is to be utilized in the future work.

Observing data, hypothesizing on it, and conducting experiments, new knowledge claims can be produced. These claims are validated, placed into the context and they become new knowledge. However, what is knowledge for one person or system may be used by another as the initial data (facts) for construction of higher-level pieces of knowledge. Therefore, transformations like "data – knowledge – meta-data – meta-knowledge – meta-meta-data – …" are rather natural. Thus, the knowledge discovery transformation of data into knowledge (Figure 4) may be applied at any level of knowledge, as the knowledge – data difference is inessential and subjective in our case. Any level may have a meta-level. Replacing data by meta-data, the transformation produces meta-knowledge instead of knowledge, and so on at the next level. Therefore it is often not so easy to determine whether knowing belongs to meta-data or meta-knowledge. Various meta-learning approaches applied within the instance space of problem space can be related to the Knowledge Management (KM) framework.

In the next subsection we emphasize the view on knowledge as an entity that can be produced, moved, inspected, rejected, and assessed, just as a widget in a factory. We consider the primary knowledge management processes including knowledge creation, knowledge organization, knowledge distribution, and knowledge application.

4.2 The knowledge management process in the context of meta-knowledge

The goal of meta-knowledge management is to make more effective and efficient use of available data mining techniques.

Generally, the problem of knowledge capture, storage, and dissemination is similar to data and information management in ISs, and therefore some executives prefer to view KM as a natural extension to IS functions [1]. According to [25] the most practical way to define KM is to show on the existing IT infrastructure the involvement of: (1) knowledge repositories, (2) best-practices and lessons-learned systems, (3) expert networks [these are DM experts], and (4) communities of practice [these are end-users].

The main idea of the continuous KM process is presented in Figure 5. We separate five key phases of this process. The first phase deals with knowledge identification, acquisition or creation.

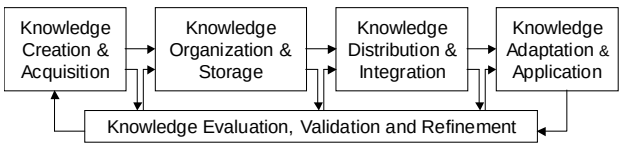


Figure 5. The knowledge management process

The second phase deals with knowledge organization and storage. In our context these processes are related mainly to knowledge representation issues. Minsky [19] discusses pros and cons of connectivist and structural approaches to knowledge representation, concluding that their combination would be natural, since usually at the lower levels of abstraction it tends to have a net architecture and tends to organize clusters and hierarchical structures at the higher levels of abstraction. The third phase is related to knowledge distribution and knowledge integration processes.

Generally, we have four potential sources of knowledge to be integrated: (1) knowledge from an expert in data-mining, knowledge discovery, statistics and related fields; (2) knowledge from a data-mining practitioner; (3) knowledge from laboratory experiments on synthetic data sets; and, finally, (4) knowledge from field experiments on real-world problems.

Beside this, research and business communities, and similar knowledge discovery systems themselves can organize different so-called trusted networks, where the participants are motivated to share their knowledge.

Knowledge sharing, distribution, and integration is beneficial in two perspectives: (1) contributing from "an individual" to acceptance and accumulation of "group" and "organizational" knowledge; (2) external validity, refinement, contextualism and generality of knowledge.

The fourth phase deals with knowledge adaptation and application processes.

The fifth phase deals with the knowledge evaluation, validation and refinement processes. In order to keep the knowledge updated there is a need to have a monitoring process to control whether the discovered meta-knowledge remains valid and a technique for continuous enhancement of knowledge. We consider these issues in the next subsection.

4.3 Meta-knowledge repository lifecycle

Since the repository is created it tends to grow and at some point of growth it naturally begins to collapse under its own weight, requiring major reorganization [25]. Therefore, the repository needs to be continuously updated, and some content needs to be deleted (if misleading), deactivated or archived (if it is potentially useful). Content may become less fragmented and redundant if similar contributions are combined, generalized and restructured.

The process of filtering knowledge claims into accepted or suppressed is commonly applied in KM. This is even more important in meta-knowledge management since a plenty of claims are produced automatically (and therefore usually need to be filtered automatically).

In Section 3.1 we mentioned the "knowing when" and "knowing where" contexts. The basic idea here is that when the environment changes (that in general may happen all the time), all of the general rules without specifying the context could become invalid. Therefore, it is highly desirable to make the knowledge repository adaptive, i.e. some knowledge should exist that would guide an organization to change the repository when the environment calls for it.

Some knowledge claims are naturally in constant competition with the other claims. Disagreements within the knowledge repository need to be resolved by means of generalization of some parts and contextualization of the others. In order to increase the quality and validity of knowledge, it needs to be continually tested, improved or removed (deactivated). Refinement leads to formulating a new knowledge claim, which requires a new process of testing and validation.

Some basic principles of triggers can be introduced in the knowledge repository. Thus, for example, when some knowledge is falsified, the deductively inferred claims from the claims to be deleted should be deleted as well.

We would like to clarify the notions of knowledge validity and knowledge quality with respect to the knowledge refinement process.

The contexts "knowing when" and "knowing where" can be discovered before it appears in a real situation. So-called zooming in and zooming out procedures can be used to find a context where theory can be falsified or supported. The goal of such procedures is in search for balance between generality, compactness, interpretability, and understandability and sensitiveness to the context, exactness, precision, and adequacy of meta-knowledge.

The quality of knowledge can be estimated by its ability to help a KDS produce solutions faster and more effectively. To determine the relative quality of a validated knowledge claim, its value needs to be compared to the values of the other claims according to the existing criteria. In any case knowledge claims have both a degree of utility and a degree of satisfaction. However, the quality of knowledge is often context-dependent. Therefore "where" and "when" context conditions may be important in many situations not only for knowledge validation but also for quality estimation.

The quality of a knowledge claim is further dependent on the accuracy of the criteria used to evaluate it. Such criteria as complexity, usefulness, and predictive power are well formalised and easy to estimate. On the contrary, such criteria as understandability, reliability of source, explanatory power are rather subjective and therefore inaccurate.

5. Conclusions

In this paper we considered several frameworks for data mining based on different paradigms. We also considered advantages and limitations of the existing

frameworks. We introduced our vision how DM process modeling can benefit from the Information Systems Development and Knowledge Management perspectives. The ISD perspective is based on viewing DM as a continuous iterative and interactive process of developing DM techniques and their effective utilization for solving a current problem impacted by the dynamically changing environment. The KM paradigm views DM research in the context of a complex adaptive system that creates, receives, stores, retrieves, transforms, and transmits different types of knowledge.

In this work we have not provided any examples that would demonstrate the applicability of the proposed adaptation of frameworks from IS and KM fields. However, we believe that our work could be helpful in the development of a new higher-level framework for DM, which can be suitable as for advancing research in DM as for DM artefact development activities. In particular, the corresponding IS research methods could be adapted and applied.

We also hope that our work could raise a new wave of interest to the foundations of DM and to the analysis of the DM field from different perspectives, such as ISD and KM.

Acknowledgements

This research is partly supported by the COMAS Graduate School of the University of Jyväskylä, Finland and by the Science Foundation Ireland under Grant No. S.F.I.-02IN.1I111.

6. References

[1] Alavi M., Leidner D. "Knowledge management systems: Issues, challenges, and benefits", *Communications of the AIS 1(7)*, 1999, pp. 2-36.

[2] Bensusan H. "Is machine learning experimental philosophy of science?" In *Proc. of ECAI'2000 Workshop on Scientific Reasoning in AI and Philosophy of Science*, 2000, pp. 9-14.

[3] Boulicaut J., Klemettinen M., and Mannila H. "Modeling KDD Processes within the Inductive Database Framework". In *Proceedings of the First International Conference on Data Warehousing and Knowledge Discovery*, Springer-Verlag, London, UK, 1999, pp. 293-302.

[4] Burrell G., Morgan G. "Sociological paradigms and organizational analysis", Heinemann, London, 1979.

[5] Coppi R. "A theoretical framework for Data Mining: "Informational paradigm", *Computational Statistics and Data Analysis 38*, 2002, pp. 501-515.

[6] Fayyad U.M. "Data Mining and Knowledge Discovery: Making Sense Out of Data", *IEEE Expert 11(5)*, 1996, pp. 20-25

[7] Fayyad U.M., Uthurusamy R "Evolving data into mining solutions for insights". *Communications of the ACM 45(8)*, 2002, pp. 28-31

[8] Friedman J. "Data Mining and Statistics: What's the connection?" In *D. Scott (Ed.) Proc. 29th Symposium on the Interface*, 1999.

[9] Hand D.J. "Data mining: statistics and more?" *The American Statistician, 52*, 1998, pp. 112-118.

[10] Hand D.J. "Statistics and data mining: intersecting disciplines", *SIGKDD Explorations 1*, 1999, pp. 16-19.

[11] Iivari J., Hirscheim R., Klein H. "A paradigmatic analysis contrasting information systems development approaches and methodologies", *Information Systems Research 9(2*), 1999, pp. 164-193

[12] Imielinski T., Mannila H. "A database perspective on knowledge discovery", *Communications of the ACM 39(11)*, 1996, pp. 58-64.

[13] Ives B., Hamilton S., Davis G. "A Framework for Research in Computer-based Management Information Systems", *Management Science 26(9)*, 1980, pp. 910-934.

[14] Kleinberg J., Papadimitriou C., and Raghavan P. "A Microeconomic View of Data Mining," *Data Mining and Knowledge Discovery 2(4)*, 1998, pp. 311-324.

[15] Lin T.Y. "Data Mining: Granular Computing Approach" In N.Zhong, L.Zhou (Eds.) *Proc. 3rd Pacific-Asia Conference, Methodologies for Knowledge Discovery and Data Mining*, LNCS 1547, 1999, pp. 24-33.

[16] Mannila H. "Theoretical Framework for Data Mining", *SIGKDD Explorations 1(2)*, 2000, pp. 30-32.

[17] Mehta M., Rissanen J., Agrawal R. "MDL-Based Decision Tree Pruning", In *Proc. KDD 1995*, 1995, pp. 216-221

[18] Michalski R.S. "Seeking Knowledge in the Deluge of Facts", *Fundamenta Informaticae 30*, 1997, pp. 283-297.

[19] Minsky M. "Logical Versus Analogical or Symbolic Versus Connectionist or Neat Versus Scruffy", *AI Magazine 12(2)*, 1991, pp. 34-51.

[20] Nonaka I. "A Dynamic Theory of Organizational Knowledge Creation", *Organization Science 5(1)*, 1994, pp. 14-37.

[21] Nunamaker W., Chen M., Purdin T. "Systems development in information systems research", *Journal of Management Information Systems 7(3)*, 1990-91, 89-106.

[22] Spiegler I. Knowledge Management: A New Idea or a Recycled Concept? *Communications of the AIS 3(14)*, 2000.

[23] Vapnik V. "The nature of statistical learning". Springer, NY, 1995.

[24] Wu X., Yu P., Piatetsky-Shapiro G., et al. "Data Mining: How Research Meets Practical Development?" *Knowledge and Information Systems 5(2)*, 2000, pp. 248 – 261.

[25] Zack M. "Managing codified knowledge", *Sloan Management Review 40 (4)*, 1999, pp. 45-58.

Actionability as Objective Measure of Rules Interestingness

Zbigniew W. Raś
University of North Carolina
Computer Science Dept.
Charlotte, NC 28223, USA
ras@uncc.edu

Li-Shiang Tsay
University of North Carolina
Computer Science Dept.
Charlotte, NC 28223, USA
ltsay@uncc.edu

Alicja Wieczorkowska
Polish-Japanese Institute of Information Technology
ul. Koszykowa 86
02-008 Warsaw, Poland
awieczor@uncc.edu

Abstract

Subjective measures are used to model interestingness of rules (see [6], [1], [13], [14]). They are user-driven, domain-dependent, and include unexpectedness, novelty and actionability. A rule is actionable if user can do an action to his/her advantage based on this rule [6]. This definition, in spite of its importance, is rather vague and it leaves open door to a number of different interpretations of actionability. In order to narrow it down, a new class of rules (called action rules) constructed from certain pairs of association rules, has been proposed in [10]. A formal definition of an action rule was independently proposed in [4]. These rules have been investigated further in [12] and [11].

To construct action rules it is required that attributes in a decision system are divided into two groups: stable and flexible. Flexible attributes provide a tool for making hints to a user what changes within some values of flexible attributes are needed to re-classify group objects, supporting action rule, to another decision class. The strategy for generating action rules which was proposed in [11] is significantly improved in this paper. The goal of the tree structure used by DEAR-2 is to partition each set of all rules, having the same decision value, into equivalence classes defined by values of stable attributes (two rules belong to the same equivalence class, if values of their the same stable attributes are not conflicting each other). Now, instead of comparing all rules, only rules between some equivalence classes are compared to construct action rules. This strategy significantly reduces the number of steps needed to generate action rules in comparison to DEAR system.

1. Introduction

There are two aspects of interestingness of rules that have been studied in data mining literature, objective and subjective measures (see [6], [1], [13], [14]. Objective measures are data-driven and domain-independent. Generally, they evaluate the rules based on their quality and similarity between them. Subjective measures, including unexpectedness, novelty and actionability, are user-driven and domain-dependent.

A rule is actionable if user can do an action to his/her advantage based on this rule [6]. This definition, in spite of its importance, is too vague and it leaves open door to a number of different interpretations of actionability. In order to narrow it down, a new class of rules (called action rules) constructed from certain pairs of association rules, has been proposed in [10]. A formal definition of an action rule was independently proposed in [4]. These rules have been investigated further in [11].

To give an example justifying the need of action rules, let us assume that a number of customers decided to close their accounts at one of the banks. To find the cause of their action, possibly the smallest and the simplest set of rules describing all these customers is constructed. Next, we search for a new set of rules, describing groups of customers from which no-one left that bank, which classification parts are maximally similar to the classification parts of the rules we have. Now, by comparing these two groups of descriptions, we may find not only the cause why these accounts have been closed but also formulate an action which, if undertaken by the bank, may prevent other customers from closing their accounts. Such actions are stimulated by action

rules and they are seen as precise hints for actionability of rules. For example, an action rule may say that by sending certain offer to a certain group of customers, it is guaranteed that these customers will not close their accounts and they do not move to another bank. Sending that offer by regular mail or giving a call to all these customers are examples of an action associated with that action rule.

The strategy for generating action rules proposed in [11] is significantly improved in the system *DEAR-2* presented in this paper. Initially, all rules discovered in the first step of our new method are partitioned into decision classes (two rules are in the same decision class, if they define the same decision value). In the second step, for each decision value, the algorithm based on tree structure is partitioning all rules having that decision value into additional equivalence classes defined by values of stable attributes (two rules belong to the same equivalence class, if values of their stable attributes do not contradict each other). In the final step, instead of comparing all rules, only rules between some equivalence classes are compared in order to construct action rules. This strategy significantly reduces the number of steps needed to generate action rules in comparison to the strategy (called *DEAR*) proposed in [11].

2. Information System and Action Rules

An information system is used for representing knowledge. Its definition, presented here, is due to Pawlak [7].

By an information system we mean a pair $S = (U, A)$, where:

- U is a nonempty, finite set called the universe,
- A is a nonempty, finite set of attributes i.e. $a : U \longrightarrow V_a$ is a function for $a \in A$, where V_a is called the domain of a.

Elements of U are called objects. For instance, they can be interpreted as customers. Attributes can be interpreted as features, offers made by a bank, characteristic conditions etc.

In this paper we consider a special case of information systems called decision tables [7]. In any decision table together with the set of attributes a partition of that set into conditions and decisions is given. Additionally, we assume that the set of conditions is partitioned into stable conditions and flexible conditions. For simplicity reason, we assume that there is only one decision attribute. *Date of birth* is an example of a stable attribute. *Interest rate* on any customer account is an example of a flexible attribute (dependable on a bank). We adopt the following definition of a decision table:

	a	b	c	d
x_1	0	S	0	L
x_2	0	R	1	L
x_3	0	S	0	L
x_4	0	R	1	L
x_5	2	P	2	L
x_6	2	P	2	L
x_7	2	S	2	H
x_8	2	S	2	H

Table 1. Decision System

A decision table is any information system of the form $S = (U, A_1 \cup A_2 \cup \{d\})$, where $d \notin A_1 \cup A_2$ is a distinguished attribute called decision. The elements of A_1 are called stable conditions, whereas the elements of A_2 are called flexible conditions.

As an example of a decision table we take $S = (\{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}, \{a, c\} \cup \{b\} \cup \{d\})$ represented by Table 1. The set $\{a, c\}$ lists stable attributes, b is a flexible attribute and d is a decision attribute. Also, we assume that H denotes a *high* profit and L denotes a *low* one.

In order to induce rules in which the THEN part consists of the decision attribute d and the IF part consists of attributes belonging to $A_1 \cup A_2$, subtables $(U, B \cup \{d\})$ of S where B is a d-reduct (see [7]) in S should be used for rules extraction. By $L(r)$ we mean all attributes listed in the IF part of a rule r. For example, if $r = [(a, 2) * (b, S) \longrightarrow (d, H)]$ is a rule then $L(r) = \{a, b\}$. By $d(r)$ we denote the decision value of a rule. In our example $d(r) = H$. If r_1, r_2 are rules and $B \subseteq A_1 \cup A_2$ is a set of attributes, then $r_1/B = r_2/B$ means that the conditional parts of rules r_1, r_2 restricted to attributes B are the same. For example if $r_1 = [(b, S) * (c, 2) \longrightarrow (d, H)]$, then $r_1/\{b\} = r/\{b\}$.

In our example, we get the following optimal rules:
$(a, 0) \longrightarrow (d, L)$, $(c, 0) \longrightarrow (d, L)$,
$(b, R) \longrightarrow (d, L)$, $(c, 1) \longrightarrow (d, L)$,
$(b, P) \longrightarrow (d, L)$, $(a, 2) * (b, S) \longrightarrow (d, H)$,
$(b, S) * (c, 2) \longrightarrow (d, H)$.

Now, let us assume that $(a, v \longrightarrow w)$ denotes the fact that the value of attribute a has been changed from v to w. Similarly, the term $(a, v \longrightarrow w)(x)$ means that $a(x) = v$ has been changed to $a(x) = w$. Saying another words, the property (a, v) of object x has been changed to property (a, w).

Let $S = (U, A_1 \cup A_2 \cup \{d\})$ is a decision table and rules r_1, r_2 have been extracted from S. Assume that B_1 is a maximal subset of A_1 such that $r_1/B_1 = r_2/B_1$, $d(r_1) = k_1$, $d(r_2) = k_2$ and $k_1 \leq k_2$. Also, assume that $(b_1, b_2, ..., b_p)$ is a list of all attributes in $L(r_1) \cap L(r_2) \cap A_2$ on which r_1, r_2 differ and $r_1(b_1) = v_1$, $r_1(b_2) = v_2$,..., $r_1(b_p) = v_p$, $r_2(b_1) = w_1$, $r_2(b_2) = w_2$,..., $r_2(b_p) = w_p$.

By (r_1, r_2)-action rule on $x \in U$ we mean expression r:

$$[(b_1, v_1 \longrightarrow w_1) \wedge (b_2, v_2 \longrightarrow w_2) \wedge ... \wedge (b_p, v_p \longrightarrow w_p)](x) \Rightarrow [(d, k_1 \longrightarrow k_2)](x).$$

Object $x \in U$ supports (r_1, r_2)-action rule r in $S = (U, A_1 \cup A_2 \cup \{d\})$, if the following two conditions are satisfied:

- $(\forall i \leq p)[b_i(x) = v_i] \wedge d(x) = k_1$
- if y_1 is the outcome of the rule r applied on x, then there is $y_2 \in U$ such that: $[[b \in L(r_2)] \Longrightarrow [b(y_1) = b(y_2)]] \wedge [d(y_2) = k_2] \wedge (\forall i \leq p)[b_i(y_2) = w_i]$

By the support of action rule r, we mean $RSup_S(r)$ = $card\{x \in U$: x supports r in $S\}$.

By the confidence of action rule r, we mean $Conf_S(r) = RSup_S(r)/Sup_S(r_1)$, where $Sup_S(r_1)$ is the support of r_1 in S.

Another words, object x in S supports (r_1, r_2)-action rule in S, if x supports r_1 and there is y in S which is $L(r_2)$-identical to the outcome of (r_1, r_2)-action rule applied on x and which supports r_2. Two objects x, y in S are B-identical, if $(\forall a \in B)[a(x) = a(y)]$.

To find the confidence of (r_1, r_2)-action rule in S, we divide the number of objects supporting (r_1, r_2)-action rule in S by the number of objects supporting rule r_1 in S.

3. Discovering Extended Action Rules

The notion of an extended action rule was given in [11]. In this section we present a new algorithm for discovering extended action rules. Initially, we partition the set of rules discovered from an information system $S = (U, A_1 \cup A_2 \cup \{d\})$, where A_1 is the set of stable attributes, A_2 is the set of flexible attributes and, $V_d = \{d_1, d_2, ..., d_k\}$ is the set of decision values, into subsets of rules defining the same decision value. Saying another words, the set of rules R discovered from S is partitioned into $\{R_i\}_{i:1 \leq i \leq k}$, where $R_i = \{r \in R : d(r) = d_i\}$ for any $i = 1, 2, ..., k$. Clearly, the objects supporting any rule from R_i form subsets of $d^{-1}(\{d_i\})$.

Let us take Table 1 as an example of a decision system S. We assume that a, c are stable attributes and b, d are flexible. The set R of certain rules extracted from S is given below:

	a	b	c	d
$\{x_1, x_2, x_3, x_4\}$	0			L
$\{x_2, x_4\}$		R		L
$\{x_1, x_3\}$			0	L
$\{x_2, x_4\}$			1	L
$\{x_5, x_6\}$		P		L
$\{x_7, x_8\}$	2	S		H
$\{x_7, x_8\}$		S	2	H

Table 2. Set of rules R with supporting objects

$(a, 0) \longrightarrow (d, L)$, $(c, 0) \longrightarrow (d, L)$,
$(b, R) \longrightarrow (d, L)$, $(c, 1) \longrightarrow (d, L)$,
$(b, P) \longrightarrow (d, L)$, $(a, 2) * (b, S) \longrightarrow (d, H)$,
$(b, S) * (c, 2) \longrightarrow (d, H)$.

We partition this set into two subsets $R_1 = \{[(a, 0) \longrightarrow (d, L)], [(c, 0) \longrightarrow (d, L)], [(b, R) \longrightarrow (d, L)], [(c, 1) \longrightarrow (d, L)], [(b, P) \longrightarrow (d, L)]\}$ and $R_2 = \{[(a, 2) * (b, S) \longrightarrow (d, H)], [(b, S) * (c, 2) \longrightarrow (d, H)]\}$.

Assume now that our goal is to re-classify some objects from the class $d^{-1}(\{d_i\})$ into the class $d^{-1}(\{d_j\})$. In our example, we assume that $d_i = (d, L)$ and $d_j = (d, H)$.

First, we represent the set R as a table (see Table 2).

The first column of this table shows objects in S supporting the rules from R (each row represents a rule). The first 5 rows represent the set R_1 and the last two rows represent the set R_2. In general case, assumed earlier, the number of different decision classes is equal to k.

The next step of the algorithm is to build d_i-tree and d_j-tree. First, from the initial table similar to Table 2, we select all rules (rows) defining the decision value d_i. Similarly, from the same table, we also select all rules (rows) which define decision value d_j.

By d_i-tree we mean a tree $T(d_i) = (N_i, E_i)$, such that:

- each interior node is labelled by a stable attribute from A_1,
- each edge is labelled either by a question mark or by an attribute value of the attribute that labels the initial node of the edge,
- along a path, all nodes (except a leaf) are labelled with different stable attributes,
- all edges leaving a node are labelled with different attribute values (including the question mark) of the stable attribute that labels that node,

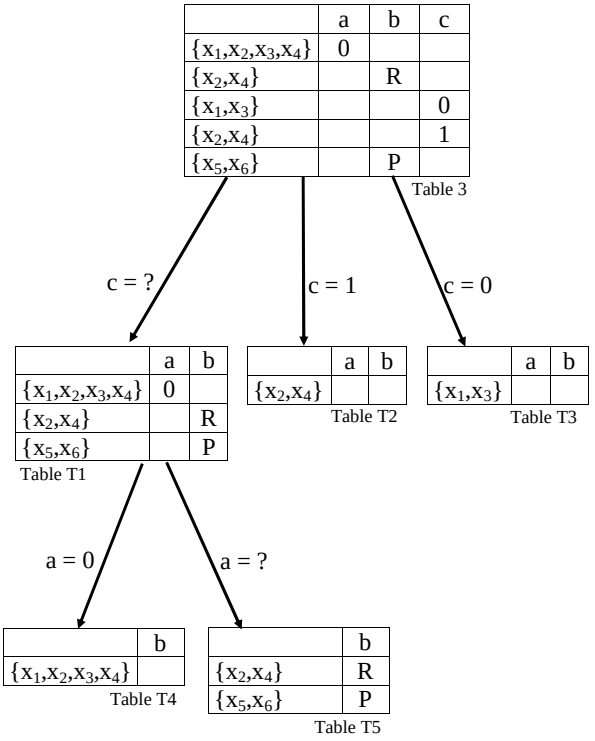


Figure 1. (d, L)**-tree**

- each leaf represents a set of rules which do not contradict on stable attributes and also define decision value d_i. The path from the root to that leaf gives the description of objects supported by these rules.

Now, taking (d, L) from our example as the value d_i, we show how to construct (d, L)-tree for the set of rules represented by Table 2. The construction of (d, L)-tree starts with a table corresponding to the root of that tree (Table 3 in Fig. 1). It represents the set of rules R_1 defining L with supporting objects from S. We use stable attribute c to split that table into 3 sub-tables defined by values $\{0, 1, ?\}$ of attribute c. The question mark means an unknown value.

Following the path labelled by value $c = 1$, we get table $T2$. Following the path labelled by value $c = 0$, we get table $T3$. When we follow the path labelled by value $[c =?][a = 0]$, we get table $T4$. Finally, by following the path having the label $[c =?][a =?]$, we get table $T5$.

Now, let us define (d, H)-tree using Table 4 as its root (see Fig. 2). Following the path labelled by value $[c =?]$, we get the table $T6$. When we follow the path labelled by value $[c = 2]$, we get the table $T7$. Both tables can be easily constructed.

Now, it can be checked that only pairs of rules belonging to tables $\{[T5, T7], [T5, T6], [T2, T6], [T3, T6], [T4, T7]\}$

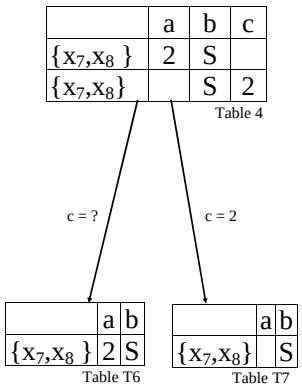


Figure 2. (d, H)**-tree**

DataSet	*Rules*	*Action Rules DEAR*
Breast Cancer	$20sec$	$27min\ 51sec$
Cleveland	$1min\ 09sec$	*Over* $8hrs$
Hepatitis	$54sec$	*Over* $8hrs$

Table 3. Time needed to extract Rules and Action Rules by *DEAR*

can be used in action rules construction. For each pair of tables, we use the same algorithm as in [11] to construct extended action rules.

This new algorithm (called *DEAR-2*) was implemented and tested on many datasets using PC with 1.8 GHz CPU. The time complexity of this algorithm was significantly lower than the time complexity of the algorithm *DEAR* presented in [11]. Both algorithms extract rules describing values of the decision attribute before any action rule is constructed. The next two tables show the time needed by *DEAR* and *DEAR-2* to extract rules and next action rules from three datasets: *Breast Cancer*, *Cleveland*. These three UCI datasets are available at [*http://www.sgi.com/tech/mlc/db/*]. The first one has 191 records described by 10 attributes. Only *Age* is the stable attribute. The second one has 303 records described by 15 attributes. Only two attributes *age* and *sex* are stable. The last one has 155 records described by 19 attributes. Again, only two attributes *age* and *sex* are stable.

The interface to both systems, *DEAR* and *DEAR-2*, is written in Visual Basic. The first picture in Figure 3 shows part of the interface to both systems. The user has an option to generate the coverings (see [7], [8]) for the decision

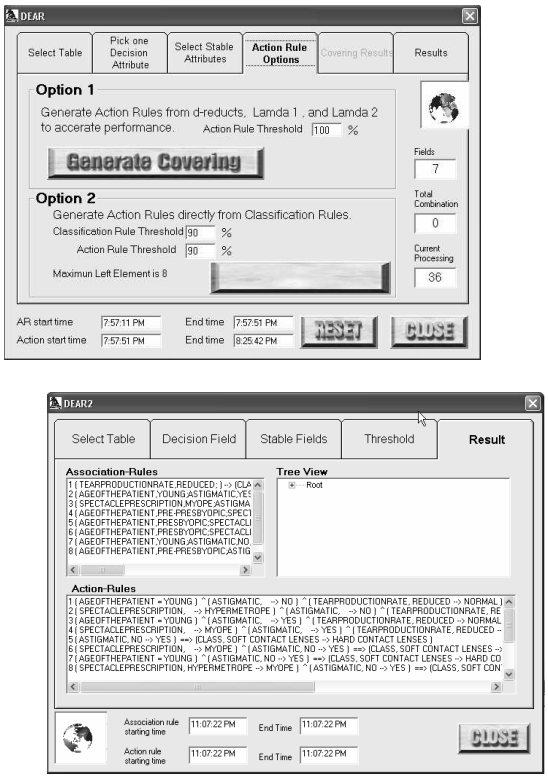


Figure 3. *DEAR* **&** *DEAR-2* **Interface**

DataSet	*Action Rules DEAR 2*
Breast Cancer	$3sec$
Cleveland	$54min\ 20sec$
Hepatitis	$51min\ 53sec$

Table 4. Time needed to extract Action Rules by *DEAR-2*

attribute and next use them in the process of action rules extraction or, if he prefers, he can directly proceed to the rules extraction step. It is recommended, by *DEAR-2*, to generate the coverings for the decision attribute if the information system has many classification attributes. By doing this we usually speed up the process of action rules extraction. The second picture in Figure 3 shows how the results are displayed by *DEAR-2* system.

4. Conclusion

System *DEAR-2* initially generates a set of association rules from S (satisfying two thresholds, the first one for a minimum support and second for a minimum confidence) defining values of a chosen attribute, called decision attribute in S, in terms of the remaining attributes. *DEAR-2* is giving preference to rules which classification part contains maximally small number of stable attributes in S. These rules are partitioned by *DEAR-2* into a number of equivalence classes where each equivalence class contains only rules which classification part has the same values of stable attributes. Each equivalence class is used independently by *DEAR-2* as a base for constructing action rules. The current strategy requires the generation of association rules from S to form a base, before the process of action rules construction starts. We believe that by following the process similar to *LERS* (see [5], [2]) or *ERID* (see [3]) which is initially centered on all stable attributes in S, we should be able to construct action rules directly from S and without the necessity to generate the base of association rules.

References

[1] Adomavicius G., Tuzhilin, A., (1997), Discovery of actionable patterns in databases: the action hierarchy approach, *Proceedings of KDD97 Conference*, Newport Beach, CA, AAAI Press

[2] Chmielewski, M.R., Grzymala-Busse J. W., Peterson N. W., Than S., (1993), The Rule Induction System LERS - a version for personal computers in *Foundations of Computing and Decision Sciences*, Vol. 18, No. 3-4, 1993, Institute of Computing Science, Technical University of Poznan, Poland, 181-212

[3] Dardzińska, A., Raś, Z.W. (2003), On Rule Discovery from Incomplete Information Systems, in *Proceedings of ICDM'03 Workshop on Foundations and New Directions of Data Mining*, (Eds: T.Y. Lin, X. Hu, S. Ohsuga, C. Liau), Melbourne, Florida, IEEE Computer Society, 2003, 31-35

[4] Geffner, H., Wainer, J., (1998), Modeling action, knowledge and control, ECAI 98, *Proceedings of the 13th European Conference on AI*, (Ed. H. Prade), John Wiley & Sons, 532-536

[5] Grzymala-Busse, J. (1997) A new version of the rule induction system LERS, in **Fundamenta Informaticae**, Vol. 31, No. 1, 27-39

[6] Liu, B., Hsu, W., Chen, S., (1997), Using general impressions to analyze discovered classification rules, *Proceedings of KDD97 Conference*, Newport Beach, CA, AAAI Press

[7] Pawlak, Z. (1991) Rough sets-theoretical aspects of reasoning about data, Kluwer, Dordrecht, 1991

[8] Pawlak, Z. (1981) Information systems - theoretical foundations, in **Information Systems Journal**, Vol. 6, 1981, 205-218

[9] Polkowski, L., Skowron A. (1998), Rough sets in knowledge discovery, in *Studies in Fuzziness and Soft Computing*, Physica-Verlag, Springer, 1998

[10] Raś, Z., Wieczorkowska, A., (2000), Action Rules: how to increase profit of a company, in *Principles of Data Mining and Knowledge Discovery*, (Eds. D.A. Zighed, J. Komorowski, J. Zytkow), Proceedings of PKDD'00, Lyon, France, LNCS/LNAI, No. 1910, Springer-Verlag, 2000, 587-592

[11] Raś, Z.W., Tsay, L.-S., (2003), Discovering Extended Action-Rules (System DEAR), in *Intelligent Information Systems 2003*, Proceedings of the IIS'2003 Symposium, Zakopane, Poland, Advances in Soft Computing, Springer-Verlag, 2003, 293-300

[12] Raś, Z., Gupta, S., (2002), Global action rules in distributed knowledge systems, in *Fundamenta Informaticae Journal*, IOS Press, Vol. 51, No. 1-2, 2002, 175-184

[13] Silberschatz, A., Tuzhilin, A., (1995), On subjective measures of interestingness in knowledge discovery, *Proceedings of KDD95 Conference*, AAAI Press

[14] Silberschatz, A., Tuzhilin, A., (1996), What makes patterns interesting in knowledge discovery systems, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 5, No. 6

Definability of Association Rules and Tables of Critical Frequencies

Jan Rauch
EuroMISE centre - Cardio
Department of Knowledge and Information Engineering
Faculty of Informatics and Statistics of University of Economics, Prague
nám. W. Churchilla 4, 130 67 Praha 3, Czech Republic
rauch@vse.cz

Abstract

The former results concerning definability of association rules in classical predicate calculi are summarized. A new intuitive criterion of definability is presented. This criterion concerns important classes of association rules. It is based on tables of critical frequencies of association rules. These tables were introduced as a tool for avoiding complex computation related to the verification of the association rules corresponding to statistical hypotheses tests.

1. Introduction

The goal of this paper is to contribute to the theoretical foundations of data mining. We deal with association rules of the form $\varphi \approx \psi$ where φ and ψ are Boolean attributes derived from the columns of the analysed data matrix $\mathcal{M}$. The association rule $\varphi \approx \psi$ says that φ and ψ are associated in the way given by the symbol $\approx$. The symbol $\approx$ is called *4ft-quantifier* . It corresponds to a condition concerning a four-fold contingency table of φ and ψ in $\mathcal{M}$. Association rules of this form were introduced and studied in [2]. They were further studied in [4, 8], the results were partly published e.g. in [5, 6, 7, 9].

The main presented result concerns definability of association rules in classical predicate calculi. It was shown in [9] that the association rules can be understood as formulae of monadic predicate observational calculi defined in [2]. Monadic predicate observational calculus is a modification of classical predicate calculus: only finite models are allowed and generalised quantifiers are added. 4ft-quantifier $\approx$ is an example of the generalised quantifier.

There is a natural question of classical definability of association rules i.e. the question *which association rules can be expressed by means of classical predicate calculus* (predicates, variables, classical quantifiers $\forall$, $\exists$, Boolean connectives and the predicate of equality). This question is solved by the Tharp's theorem proved in [2].

The Tharp's theorem is but too general from the point of view of association rules. A more intuitive criterion of classical definability of association rules was proved in [4] see also [9]. The first goal of this paper is to show that this criterion can be further simplified for several important classes of association rules.

The simplified criterion is based on tables of critical frequencies (further only *TCF* instead of table of critical frequencies). TCF's were introduced as a tool for avoiding complex computation [2, 4] related to the association rules corresponding to the statistical hypothesis tests. It means that this paper deals with three features of association rules:

- classes of association rules
- tables of critical frequencies
- classical definability of association rules.

The second goal of this paper is to point out to the mutual relations among these features.

A short overview of association rules of the form $\varphi \approx \psi$ is given in section 2. The classes of association rules are introduced in section 3. The definition of TCF is based on classes of association rules and it is given in Sect. 4. Results concerning classical definability of association rules are presented in Sect. 5 and 6. Some concluding remarks are in Sect. 7.

2 Association Rules

The association rules is an expression $\varphi \approx \psi$ where φ and ψ are Boolean attributes and the symbol $\approx$ is the 4ft-quantifier. The Boolean attributes φ and ψ are derived from basic Boolean attributes using propositional connectives $\vee$, $\wedge$ and $\neg$ in the usual way. The *basic Boolean attribute* is the expression $A(\alpha)$ where the symbol α denotes a subset of the

set of all possible values of the attribute A (i.e. column of the analysed data matrix $\mathcal{M}$.)

The basic Boolean attribute $A(\alpha)$ is true in the row o of $\mathcal{M}$ if it is $a \in \alpha$ where a is the value of the attribute A in the row o. The truth values of Boolean attributes φ and ψ are defined in the usual way. The value of the Boolean attribute φ in the row o of the data matrix $\mathcal{M}$ is denoted $\varphi(o, \mathcal{M})$. It is $\varphi(o, \mathcal{M}) = 1$ if φ is true in o and it is $\varphi(o, \mathcal{M}) = 0$ if φ is false in o.

The expressions $A(1)$, $B(1,2)$, and $C(4,5)$ are examples of basic Boolean attributes derived from the attributes - columns of the (very simple) data matrix $\mathcal{M}$ see Fig. 1.

row	attributes			basic Boolean attributes		
of $\mathcal{M}$	A	B	C	$A(1)$	$B(1,2,3)$	$C(4,5)$
o_1	1	9	4	1	0	1
o_2	1	2	6	1	1	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
o_n	2	4	5	0	0	1

Figure 1. Data matrix $\mathcal{M}$

The 4ft-quantifier $\approx$ corresponds to a condition concerning a four-fold contingency table of φ and ψ in $\mathcal{M}$. This table is denoted 4ft(φ, ψ, $\mathcal{M}$) and it is called called *4ft table of φ and ψ in $\mathcal{M}$* , see Table 1.

Table 1. 4ft table 4ft(φ, ψ, $\mathcal{M}$) of φ and ψ in $\mathcal{M}$

$\mathcal{M}$	ψ	$\neg\psi$
φ	a	b
$\neg\varphi$	c	d

Here a is the number of the rows of $\mathcal{M}$ satisfying both φ and ψ, b is the number of the objects satisfying φ and not satisfying ψ, etc. We write 4ft(φ, $\psi, \mathcal{M}$) $= \langle a, b, c, d\rangle$.

The association rule $\varphi \approx \psi$ is true in the analysed data matrix $\mathcal{M}$ if the condition corresponding to the 4ft-quantifier $\varphi \approx \psi$ is satisfied for the 4ft-table 4ft(φ, $\psi, \mathcal{M}$). We write $Val(\varphi \approx \psi, \mathcal{M}) = 1$ if $\varphi \approx \psi$ is true in the data matrix $\mathcal{M}$, otherwise we write $Val(\varphi \approx \psi, \mathcal{M}) = 0$.

Several examples of 4ft-quantifiers follow.

The 4ft-quantifier $\Rightarrow_{p,Base}$ of *founded implication* [2] is defined for $0 < p \leq 1$ and $Base > 0$ by the condition $\frac{a}{a+b} \geq p \wedge a \geq Base$. The association rule $\varphi \Rightarrow_{p,Base} \psi$ means that at least $100p$ per cent of objects satisfying φ satisfy also ψ and that there are at least $Base$ objects of $\mathcal{M}$ satisfying both φ and ψ.

The 4ft-quantifier $\Rightarrow^{!}_{p,\alpha,Base}$ of *lower critical implication* [2] is defined for $0 < p \leq 1$, $0 < \alpha < 0.5$ and $Base > 0$ by the condition $\sum_{i=a}^{a+b} \binom{a+b}{i} p^i (1-p)^{a+b-i} \leq \alpha \wedge a \geq Base$. The association rule $\varphi \Rightarrow_{p,Base} \psi$ corresponds to a statistical test (on the level α) of the null hypothesis $H_0 : P(\psi|\varphi) \leq p$ against the alternative one $H_1 : P(\psi|\varphi) > p$. Here $P(\psi|\varphi)$ is the conditional probability of the validity of ψ under the condition φ.

The 4ft-quantifier $\Leftrightarrow_{p,Base}$ of *founded double implication* [3] is defined for $0 < p \leq 1$ and $Base > 0$ by the condition $\frac{a}{a+b+c} \geq p \wedge a \geq Base$. The association rule $\varphi \Leftrightarrow_{p,Base} \psi$ means that at least $100p$ per cent of rows of $\mathcal{M}$ satisfying φ or ψ satisfy both φ and ψ and that there are at least $Base$ rows of $\mathcal{M}$ satisfying both φ and ψ.

The 4ft-quantifier $\equiv_{p,Base}$ of *founded equivalence* [3] is defined for $0 < p \leq 1$ and $Base > 0$ by the condition $\frac{a+d}{a+b+c+d} \geq p \wedge a \geq Base$. The association rule $\varphi \equiv_{p,Base} \psi$ means that φ and ψ have the same value (either *true* or *false*) for at least $100p$ per cent of all objects of $\mathcal{M}$ and that there are at least $Base$ objects satisfying both φ and ψ.

Fisher's quantifier $\sim_{\alpha,Base}$ [2] is defined for $0 < \alpha < 0.5$ and $Base > 0$ by the condition $\sum_{i=a}^{\min(r,k)} \frac{\binom{k}{i}\binom{n-k}{r-i}}{\binom{r}{n}} \leq \alpha \wedge ad > bc \wedge a \geq Base$. This quantifier corresponds to the statistical test (on the level α) of the null hypothesis of independence of φ and ψ against the alternative one of the positive dependence.

The 4ft-quantifier $\rightarrow_{conf,sup}$ is defined for $0 < conf < 1$ and $0 < sup < 1$ by the condition $\frac{a}{a+b} \geq conf \wedge \frac{a}{a+b+c+d} \geq sup$. It corresponds to the "classical" association rule with confidence $conf$ and support sup [1].

Additional 4ft-quantifiers are defined e.g. in [2, 3, 6]. The data mining procedure 4ft-Miner [10] deals with 14 types of 4ft-quantifiers.

An example of association rule is the expression

$$A(1) \wedge B(1,2,3) \Leftrightarrow_{p,Base} C(4,5)$$

It means that at least $100p$ per cent of rows of data matrix $\mathcal{M}$ satisfying $A(1) \wedge B(1,2,3)$ or $C(4,5)$ satisfy both $A(1) \wedge B(1,2,3)$ and $C(4,5)$ and that there are at least $Base$ rows of $\mathcal{M}$ satisfying both $A(1) \wedge B(1,2,3)$ and $C(4,5)$.

The condition associated to the 4ft-quantifier $\approx$ defines a $\{0,1\}$ - function $Asf_{\approx}$ such that

$$Asf_{\approx}(a,b,c,d) = \begin{cases} 1 & \text{if the condition associated to} \approx \\ & \text{is satisfied for } \langle a,b,c,d\rangle \\ 0 & \text{otherwise.} \end{cases}$$

(Here we write $Asf_{\approx}(a,b,c,d)$ instead of $Asf_{\approx}(\langle a,b,c,d\rangle)$.) This function is called *associated function of the 4ft-quantifier* $\approx$, see [2]. It is defined for all 4ft tables $\langle a,b,c,d\rangle$.

Further we will write only $\approx (a,b,c,d)$ instead of $Asf_{\approx}(a,b,c,d)$. It means that the association rule $\varphi \approx \psi$ is true in the analysed data matrix $\mathcal{M}$ iff $\approx (a,b,c,d) = 1$ where $\langle a,b,c,d\rangle =$ 4ft(φ, $\psi, \mathcal{M}$).

3 Classes of Association Rules

The classes of association rules are defined by classes of 4ft-quantifiers. The association rule $\varphi \approx \psi$ belongs to the *class of implicational rules* if the 4ft-quantifier $\approx$ belongs to the *class of implication quantifiers*. We also say that the association rule $\varphi \approx \psi$ is *implicational rule* and that the 4ft-quantifier $\approx$ is *implicational quantifier*. This is the same for additional classes of association rules.

We are going to present some of classes of association rules. We present more classes than we use in the main results concerning definability of association rules. The reason is to point out to additional interesting properties of association rules. The main results on definability concerns implicational and equivalency rules only.

3.1 Implicational Quantifiers

The class of implicational quantifiers is defined in [2] such that the 4ft-quantifier $\approx$ is **implicational** if

$$\approx(a,b,c,d) = 1 \ \wedge \ \ a' \geq a \ \wedge \ b' \ \leq b$$

implies

$$\approx(a',b',c',d') = 1$$

for all 4ft tables $< a,b,c,d >$ and $< a',b',c',d' >$. The condition $a' \geq a \ \wedge \ b' \leq b$ is the **truth preservation condition for implicational quantifiers**.

Let us assume that $\langle a,b,c,d\rangle$ is the 4ft table of φ and ψ in data matrix $\mathcal{M}$ and that $\langle a',b',c',d'\rangle$ is the 4ft table of φ and ψ in data matrix $\mathcal{M}'$. The truth preservation condition $a' \geq a \ \wedge \ b' \leq b$ means that in data matrix $\mathcal{M}'$ there are more rows satisfying both φ and ψ than in data matrix $\mathcal{M}$ and that in $\mathcal{M}'$ there are fewer rows satisfying φ and not satisfying ψ than in $\mathcal{M}$. In other words this condition means that that the 4ft table $\langle a',b',c',d'\rangle$ is "*better from the point of view of implication*" than the 4ft table $\langle a,b,c,d\rangle$ (i-better, see [2]).

Thus it is reasonable to expect that if the implicational rule $\varphi \approx \psi$ (i.e. the rule expressing implication by $\approx$) is true in data matrix $\mathcal{M}$ then it is also true in data matrix $\mathcal{M}'$ that is better from the point of view of implication. This expectation is ensured for implicational quantifiers by the above given definition.

It is easy to prove that the 4ft-quantifier $\Rightarrow_{p,Base}$ of founded implication (see Sect. 2) is implicational. It is proved in [2] that the 4ft-quantifier $\Rightarrow^{!}_{p,\alpha,Base}$ of lower critical implication (see Sect. 2) is also implicational.

Remark 1: It is also easy to prove for the implicational quantifier $\Rightarrow^*$ that the value $\Rightarrow^*(a,b,c,d)$ depends neither on c nor on d. Thus we write only $\Rightarrow^*(a,b)$ instead of $\Rightarrow^*(a,b,c,d)$ for the implicational quantifier $\Rightarrow^*$.

There are various theoretical results related to the class of implicational quantifiers. Both practically useful and theoretically interesting are results concerning deduction rules of the form $\frac{\varphi \Rightarrow^* \psi}{\varphi' \Rightarrow^* \psi'}$ where φ, ψ are Boolean attributes and $\Rightarrow^*$ is the implicational quantifier [2, 6]. There are also results concerning implicational rules in data with missing information see [2] and also [7].

3.2 Double Implicational Quantifiers

We can try to express the relation of equivalence of Boolean attributes φ and ψ in an analogy to propositional logic. If u and v are propositions and both $u \rightarrow v$ and $v \rightarrow u$ are true, then u is equivalent to v (the symbol "$\rightarrow$" is here a propositional connective of implication). Thus we can try to express the relation of equivalence of attributes φ and ψ using a "double implicational" 4ft-quantifier $\Leftrightarrow^*$ such that $\varphi \Leftrightarrow^* \psi$ if and only if $\varphi \Rightarrow^* \psi$ and $\psi \Rightarrow^* \varphi$, where $\Rightarrow^*$ is a suitable implicational quantifier.

If we apply the truth preservation condition for implicational quantifier to $\varphi \Rightarrow^* \psi$, we obtain $a' \geq a \ \wedge \ b' \leq b$. If we apply it to $\psi \Rightarrow^* \varphi$, we obtain $a' \geq a \ \wedge \ c' \leq c$, ($c$ is here instead of b, see Table 1). This leads to the **truth preservation condition for double implicational quantifiers** $a' \geq a \wedge b' \leq b \wedge c' \leq c$. Thus the class of double implicational quantifiers is defined in [8], (see also [3] and [6]) such that the 4ft-quantifier $\approx$ is **double implicational** if

$$\approx(a,b,c,d) = 1 \ \wedge \ \ a' \geq a \ \wedge b' \ \leq b \ \wedge \ c' \leq c$$

implies

$$\approx(a',b',c',d') = 1$$

for all 4ft tables $< a,b,c,d >$ and $< a',b',c',d' >$.

It is easy to prove that the 4ft-quantifier $\Leftrightarrow_{p,Base}$ of founded double implication see Sect. 2 is double implicational. It can be also proved that the 4ft-quantifier $\Leftrightarrow^{!}_{p,\alpha}$ of *lower critical double implication* [3] defined for $0 < p \leq 1$, $0 < \alpha < 0.5$ and $Base > 0$ by the condition $\sum_{i=a}^{a+b+c} \binom{a+b+c}{i} p^i (1-p)^{a+b+c-i} \leq \alpha \ \wedge \ a \geq Base$ is double implicational see [8].

Remark 2: The value $\Leftrightarrow^*(a,b,c,d)$ does not depend on d for the double implicational quantifier $\Leftrightarrow^*$. Thus we write only $\Leftrightarrow^*(a,b,c)$ instead of $\Leftrightarrow^*(a,b,c,d)$ for the double implicational quantifier $\Leftrightarrow^*$.

However it can be proved that there is no implicational quantifier $\Rightarrow^*$ such that $\Leftrightarrow_{p,Base}(a,b,c) = 1$ if and only if $\Rightarrow^*(a,b) = 1$ and $\Rightarrow^*(a,c) = 1$ and analogously for 4ft-quantifier $\Leftrightarrow^{!}_{p,\alpha}$[8].

This fact led to the definition and study of the class of pure double implicational quantifiers and the class of strong double implicational quantifiers [8].

We say that the quantifier $\Leftrightarrow^*$ is *pure double implicational* if there is an implicational quantifier $\Rightarrow^*$ such that

$$\Leftrightarrow^* (a,b,c) = 1 \text{ if and only if } \Rightarrow^* (a,b) \wedge \Rightarrow^* (a,c)$$

for each 4ft table $< a,b,c,d >$. We say that the quantifier $\Leftrightarrow^*$ is *strong double implicational* if there are two implicational quantifiers $\Rightarrow^*_1$ and $\Rightarrow^*_2$ such that '

$$\Leftrightarrow^* (a,b,c) = 1 \text{ if and only if } \Rightarrow^*_1 (a,b) \wedge \Rightarrow^*_2 (a,c)$$

for each 4ft table $< a,b,c,d >$.

It is easy to prove that each pure double implicational quantifier is strong double implicational and that each strong double implicational quantifier is double implicational. There are interesting properties of pure double implicational and of strong implicational quantifiers [8].

Let us note that the quantifiers $\Leftrightarrow_{p,Base}$ and $\Leftrightarrow^!_{p,\alpha}$ are similar what concerns dealing with the sum $b+c$. This sum is treated in the same way as the frequency b is treated in the quantifiers $\Rightarrow_{p,Base}$ and $\Rightarrow^!_{p,\alpha}$, see above. This led to the definition of the class of Σ-*double implicational quantifiers* [8]. The 4ft-quantifier $\approx$ is **Σ-double implicational** if

$$\approx (a,b,c,d) = 1 \ \wedge \ \ a' \geq a \ \wedge b' + c' \leq b + c$$

implies

$$\approx (a',b',c',d') = 1$$

for all 4ft tables $< a,b,c,d >$ and $< a',b',c',d' >$.

There are again various interesting results related to the class of Σ-double implicational quantifiers. An example is a criterion of correctness of deduction rules of the form $\frac{\varphi \Leftrightarrow^* \psi}{\varphi' \Leftrightarrow^* \psi'}$ where φ, ψ are Boolean attributes and $\Leftrightarrow^*$ is the Σ-double implicational quantifier [6].

3.3 Equivalence Quantifiers

The double implicational quantifier is an attempt to express the equivalence of Boolean attributes φ and ψ in an analogy to propositional logic. We start from the fact that if u and v are propositions and both $u \rightarrow v$ and $v \rightarrow u$ are true, then u is equivalent to v.

There is an other way to express the equivalence of the propositions u and v. The propositions u and v are equivalent if both $u \rightarrow v$ and $\neg u \rightarrow \neg v$ are true. Thus we can try to express the relation of equivalence of the attributes φ and ψ using an "equivalence" 4ft-quantifier $\equiv^*$ such that $\varphi \equiv^* \psi$ if and only if $\varphi \Rightarrow^* \psi$ and $\neg\varphi \Rightarrow^* \neg\psi$, where $\Rightarrow^*$ is the suitable implicational quantifier.

If we apply the truth preservation condition for implicational quantifiers to $\varphi \Rightarrow^* \psi$ we obtain $a' \geq a \ \wedge \ b' \leq b$. If we apply it to $\neg\varphi \Rightarrow^* \neg\psi$, we obtain $d' \geq d \ \wedge \ c' \leq c$, ($c$ is here instead of b and d is instead of a, see table 1). This leads to the **truth preservation condition for equivalency quantifiers** [2, 8]. Thus the class of equivalency quantifiers is defined such that the 4ft-quantifier $\approx$ is **equivalency quantifier** if

$$\approx (a,b,c,d) = 1 \ \wedge \ \ a' \geq a \ \wedge b' \ \leq b \ \wedge \ c' \leq c \ \ \wedge \ \ d' \geq d$$

implies

$$\approx (a',b',c',d') = 1$$

for all 4ft tables $< a,b,c,d >$ and $< a',b',c',d' >$.

Let us emphasize that the class of quantifiers defined by the truth preservation condition for equivalency quantifiers was defined in the frame of development of the GUHA method of exploratory data analysis about 35 years ago see e.g. [2]. This class was denominated as a class of associational quantifiers in [2]. However the term *association rule* is now commonly used for the association rules with confidence and support defined in [1]. We use, therefore, the terms *equivalency quantifier* and *equivalency rule*.

It is easy to prove that the 4ft-quantifier $\equiv_{p,Base}$ of founded equivalence see Sect. 2 is equivalency. It can be also proved that the 4ft-quantifier $\equiv^!_{p,\alpha,Base}$ of *lower critical equivalence* [3] defined for $0 < p \leq 1$, $0 < \alpha < 0.5$ and $Base > 0$ by the condition $\sum_{i=a+d}^{n} \binom{n}{i} p^i (1-p)^n \leq \alpha \wedge \ a \geq Base$ is equivalency see [8].

It is also proved in [2] that the Fisher's quantifier, the χ^2-quantifier $\sim^2_{\alpha,Base}$ and the quantifier $\sim_{\delta,Base}$ of simple deviation are equivalency (i.e. associational in the sense of [2]) The χ^2-quantifier $\sim^2_{\alpha,Base}$ is defined in [2] for $0 \ < \ \alpha \leq 0.5$ and $Base > 0$ by the condition $ad > bc \wedge a \geq \ Base \wedge \frac{(ad-bc)^2}{(a+b)(a+c)(b+d)(b+d)}(a+b+c+d) \geq \chi^2_\alpha$ where χ^2_α is $(1-\alpha)$-quantile of the χ^2 distribution. The quantifier $\sim_{\delta,Base}$ of simple deviation is defined in [2] for $0 \leq \delta$ and $Base > 0$ by the condition $ad > e^\delta bc \wedge a \geq Base$.

It can be however proved that the 4ft-quantifier $\rightarrow_{conf,sup}$ defined by the condition $\frac{a}{a+b} \ \geq \ conf \ \wedge \ \frac{a}{a+b+c+d} \geq sup$ (see Sect. 2) that corresponds to the "classical" association rule is not equivalency [8] (i.e. not associational in the sense of [2]).

We can define classes of various equivalency quantifiers analogously to the classes of pure double implicational quantifiers, of strong double implicational quantifiers and Σ-double implicational quantifiers [8]. There are interesting properties of the just defined equivalency quantifiers see [2, 6].

4 Tables of Critical Frequencies

Further we will denote $\mathcal{N}^+ = \{0,1,2,\ldots\} \cup \{\infty\}$. First we prove the theorem about *partial tables of maximal b.* (Please note that the equivalency quantifier is the associational quantifier according to [2]).

Theorem 1 *Let $\approx$ be an equivalency quantifier. Then there is a non-negative function $Tb_{\approx}$ that assigns to each triple $\langle a, c, d\rangle$ of non-negative natural numbers a value $Tb_{\approx}(a, c, d) \in \mathcal{N}^+$ such that*

1. *For each $b \geq 0$ it is $\approx (a, b, c, d) = 1$ if and only if $b < Tb_{\approx}(a, c, d)$.*
2. *If $a' > a$ then $Tb_{\approx}(a', c, d) \geq Tb_{\approx}(a, c, d)$.*

Proof: Let us define

$$Tb_{\approx}(a, c, d) = \min\{b | \approx (a, b, c, d) = 0\}\,.$$

Let us remember that $\approx$ is equivalency. It means that

$$\approx (a, b, c, d) = 1 \ \wedge \ a' \geq a \wedge b' \leq b \wedge c' \leq c \ \wedge \ d' \geq d$$

implies

$$\approx (a', b', c', d') = 1$$

It means among other

I: *If $\approx (a, b, c, d) = 0$ and $v \leq a$ then also $\approx (v, b, c, d) = 0$.*

II: *If $\approx (a, b, c, d) = 0$ and $w \geq b$ then also $\approx (a, w, c, d) = 0$.*

The point II means that it is $\approx (a, b, c, d) = 0$ for each $b \geq \min\{b | \approx (a, b, c, d) = 0\}$.

We prove that the function defined in the above given way has the properties 1. an 2.

1. *Let us suppose $b \geq 0$ and $\approx (a, b, c, d) = 1$. We have to prove $b < Tb_{\approx}(a, c, d)$. Let us suppose $b \geq Tb_{\approx}(a, c, d) = \min\{b \mid \approx (a, b, c, d) = 0\}$. It however means according to point* **II** *that $\approx (a, b, c, d) = 0$. Thus it must be $b < Tb_{\approx}(a, c, d)$.*

   Let us suppose $b \geq 0$ and $\approx (a, b, c, d) = 0$. We have to prove $b \geq Tb_{\approx}(a, c, d)$. It but follows from the definition of $Tb_{\approx}(a, c, d)$.

2. *Let us suppose $a' > a$ and also $Tb_{\approx}(a', c, d) < Tb_{\approx}(a, c, d)$. Let us denote $e = Tb_{\approx}(a, c, d)$, thus it is $e > 0$. It means $Tb_{\approx}(a', c, d) \leq e - 1$ and thus according to the definition of $Tb_{\approx}(a', c, d)$ it is $\approx (a', e - 1, c, d) = 0$. Due to point* **I** *it is also $\approx (a, e - 1, c, d) = 0$. It is but also $e - 1 < e = Tb_{\approx(a,c,d)}$ and it means $\approx (a, e - 1, c, d) = 1$ according to already proved point 1. It is a contradiction and thus it cannot be both $a' > a$ and $Tb_{\approx}(a', c, d) < Tb_{\approx}(a, c, d)$. It but means that it follows $Tb_{\approx}(a', c, d) \geq Tb_{\approx}(a, c, d)$ from $a' > a$.*

This finishes the proof.

Let us remember that the value of $\Rightarrow^*\ (a, b, c, d)$ depends neither on c nor on d for the implicational quantifier $\Rightarrow^*$ and thus we write only $\Rightarrow^*\ (a, b)$ instead of $\Rightarrow^*\ (a, b, c, d)$, see *Remark 1* in Sect. 3.

The just proved theorem has a direct consequence for the implicational quantifiers.

Theorem 2 *Let $\Rightarrow^*$ be an implicational quantifier. Then there is a non-negative non-decreasing function $Tb_{\Rightarrow^*}$ that assigns to each non-negative integer a a value $Tb_{\Rightarrow^*} \in \mathcal{N}^+$ such that for each $b \geq 0$ it is $\Rightarrow^*\ (a, b) = 1$ if and only if $b < Tb_{\Rightarrow^*}(a)$.*

Proof: Due to the above mentioned Remark 1 we can only put $Tb_{\Rightarrow^*}(a) = Tb_{\Rightarrow^*}(a, 0, 0)$ where $Tb_{\Rightarrow^*}(a, c, d)$ is the function from the theorem 1

We define the notions of tables of maximal b on the basis of just proved theorems.

Definition 1

1. *Let $\approx$ be an equivalency quantifier and let $c \geq 0$ and $d \geq 0$ be the natural numbers. Then the* **partial table of maximal** *b for the quantifier $\approx$ and for the couple $\langle c, d\rangle$ is the function $Tbp_{\approx,c,d}$ defined such that*

   $$Tbp_{\approx,c,d}(a) = Tb_{\approx}(a, c, d)$$

   where $Tb_{\approx}(a, c, d)$ is the function from the theorem 1.

2. *Let $\Rightarrow^*$ be an implicational quantifier. Then the function $Tb_{\Rightarrow^*}$ from the theorem 2 is a* **table of maximal** *b for the implicational quantifier $\Rightarrow^*$.*

3. *Let T be a partial table of maximal b or a table of maximal b. Then a* **step in the table** *T is each such $a \geq 0$ for which it is $T(a) < T(a + 1)$.*

It is important that the function $Tb_{\Rightarrow^*}$ makes it possible to use a simple test of inequality instead of a rather complex computation. For example we can use inequality $b < Tb_{\Rightarrow^!_{p,\alpha,s}}(a)$ instead of condition $\sum_{i=a}^{a+b} \frac{(a+b)!}{i!(a+b-i)!} p^i (1-p)^{a+b-i} \leq \alpha \wedge a \geq s$ for quantifier $\Rightarrow^!_{p,\alpha,s}$ of lower critical implication, see section 2. An other form of the table of critical frequencies for implicational quantifier is defined in [2].

Let us remark that it can be $Tb_{\Rightarrow^*}(a) = \infty$. A trivial example gives the quantifier $\Rightarrow^T$ defined such that $\Rightarrow^T\ (a, b) = 1$ for each a,b. Then it is $Tb_{\Rightarrow^T}(a) = \infty$ for each a.

The partial table of maximal b and table of maximal b are called *tables of critical frequencies*. Further tables of critical frequencies for Σ-double implicational quantifiers and for Σ-equivalence quantifiers are defined and studied in [8].

5 Classical Definability and TCF

5.1 Association Rules and Observational Calculi

Monadic observational predicate calculi (MOPC for short) are defined and studied in [2] as a special case of observational calculi. MOPC can be understood as a modification of classical predicate calculus such that only finite models (i.e. data structures in which the formulas are interpreted) are admitted and more quantifiers than $\forall$ and $\exists$ are used. These new quantifiers are called *generalised quantifiers*. The 4ft-quantifier is a special case of the generalised quantifiers.

Classical monadic predicate calculus (CMOPC for short) is a MOPC with only classical quantifiers. In other words it is a classical predicate calculus with finite models. The formulas $(\forall x)P_1(x)$ and $(\exists x)(\exists y)((x \neq y) \wedge P_1(x) \wedge \neg P_2(y))$ are examples of formulas of CMOPC.

If we add the 4ft-quantifiers to CMOPC we get MOPC the formulas of which correspond to association rules. Examples of such formulas are $(\Rightarrow_{p,Base}\ x)(P_1(x), P_2(x))$ and $(\Leftrightarrow_{p,Base}\ x)(P_1(x) \vee P_3(x), P_2(x) \wedge P_4(x))$. The values of these formulas can be defined in Tarski style see [2]. We suppose that the formulas are evaluated in $\{0,1\}$ - data matrices (i.e. finite data structures), see example in Fig. 2 where predicates $P_1, \ldots, P_n$ are interpreted by columns - functions $f_1, \ldots, f_n$ respectively.

row of $\mathcal{M}$	P_1 f_1	P_2 f_2	$\ldots$ $\ldots$	P_n f_n	$P_1 \vee P_3$ $\max(f_1, f_3)$	$P_2 \wedge P_4$ $\min(f_2, f_4)$
o_1	1	0	$\ldots$	1	0	1
o_2	0	1	$\ldots$	1	1	0
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
o_n	1	0	$\ldots$	0	0	1

Figure 2. Example of $\{0,1\}$ - data matrix

The rule $(\approx x)(P_1(x) \vee P_3(x), P_2(x) \wedge P_4(x))$ can be written in various forms, e.g. $(\approx)(P_1 \vee P_3, P_2 \wedge P_4)$ or $P_1 \vee P_3 \approx P_2 \wedge P_4$. Its evaluation is in any case based on the value $\approx (a, b, c, d)$ where $\langle a, b, c, d\rangle$ is the 4ft-table of $P_1(x) \vee P_3(x)$ and $P_2(x) \wedge P_4(x)$ in the data matrix in question. The same is true for each association rule of the form $(\approx x)(\varphi(x), \psi(x))$.

Let us remark that the association rule of the form like $A(1,2,3) \approx B(4,5)$ can be understood (informally speaking) like the rule $A_1 \vee A_2 \vee A_3 \approx B_4 \vee B_5$ where A_1 is a predicate corresponding to the basic Boolean attribute $A(1)$ etc.

5.2 Definability and Associated Function

The natural question is what association rules are classically definable. We say that the association rule $(\approx\ x)\ (\varphi(x),\ \psi(x))$ - formula of MOPC is classically definable if there is a formula Φ of CMOPC with equality such that Φ is logically equivalent to $(\approx x)(\varphi(x), \psi(x))$. The association rule $(\approx x)(P_1(x) \vee P_3(x), P_2(x) \wedge P_4(x))$ is e.g. classically definable if it is equivalent to the formula created from the predicates $P_1(x)$, $P_2(x)$, $P_3(x)$, $P_4(x)$, propositional connectives $\neg, \vee, \wedge$ classical quantifiers $\exists, \forall$ and from the binary predicate of equality $=$. The precise formal definition is given in [2], see also [9]. If the association rule $(\approx x)(\varphi, \psi)$ is classically definable then we also say that the 4ft-quantifier $\approx$ is classically definable and vice-versa.

The question of classical definability of (not only) association rules is solved by the Tharp's theorem proved in [2]. The Tharp's theorem is but too complex and general from the point of view of association rules. A more intuitive criterion of classical definability of association rules is proved in [4] see also [9]. This criterion is based on the associated function $Asf_{\approx}(a, b, c, d)$ of the 4ft-quantifier $\approx$ (we write sometimes only $\approx\ (a, b, c, d)$ instead of $Asf_{\approx}(a, b, c, d)$, see section 2).

The criterion uses the notion of *interval in* $\mathcal{N}^4$ where $\mathcal{N}$ is the set of all natural numbers. It is defined as the set

$$I = I_1 \times I_2 \times I_3 \times I_4$$

such that it is for $i = 1, 2, 3, 4$ $I_j = \langle k, l\rangle$ or $I_j = \langle k, \infty)$ where $0 \leq k < l$ are natural numbers. The empty set $\emptyset$ is also the interval in $\mathcal{N}^4$.

The criterion of classical definability of association rules is given by the following theorem proved in [4], see also [9].

Theorem 3 *The 4ft-quantifier $\approx$ is classically definable if and only if there are K intervals $I_1, \ldots, I_K$ in $\mathcal{N}^4$, $K \geq 0$ such that it is for each 4ft table $\langle a, b, c, d\rangle$*

$$Asf_{\approx}(a, b, c, d) = 1 \ \textit{iff}\ \langle a, b, c, d\rangle \in \bigcup_{j=1}^{K} I_j\ .$$

5.3 Definability of Equivalency Quantifiers

We use the criterion of classical definability based on associated functions of 4ft-quantifiers to give a very intuitive necessary condition of classical definability of equivalency rules. This condition says that if the equivalency quantifier is definable then its each partial table of maximal b of this quantifier has only finite number of steps. It is proved in the next theorem.

Theorem 4 *Let $\approx$ be an classically definable equivalency quantifier. Then each its partial table of maximal b has only finite number of steps.*

Proof: We suppose that $\approx$ is classically definable quantifier. Thus according to the theorem 3 there are K intervals $I_1, \ldots, I_K$ in $\mathcal{N}^4$, $K \geq 0$ such that it is for each 4ft table $\langle a, b, c, d\rangle$

$$Asf_{\approx}(a, b, c, d) = 1 \text{ iff } \langle a, b, c, d\rangle \in \bigcup_{j=1}^{K} I_j \, .$$

If $K = 0$ then it is $\approx (a, b, c, d) = 0$ for each 4ft table $\langle a, b, c, d\rangle$ and it is $Tbp_{\approx,c,d}(a) = 0$ for each a and for each partial table $Tbp_{\approx,c,d}$ of maximal b of $\approx$. It but means that each such partial table of maximal b has no step.

Let us suppose that $K > 0$ and that

$$I_j = \langle a_j, A_j) \times \langle b_j, B_j) \times \langle c_j, C_j) \times \langle d_j, D_j) \, .$$

Suppose that for c_0 and d_0 the partial table $Tbp_{\approx}(a, c_0, d_0)$ of maximal b has infinitely many steps. It means that for each natural $n > 0$ there are $a > n$ and $b > n$ such that $\approx (a, b, c_0, d_0) = 1$. Thus there must be $m \in 1, \ldots, K$ such that

$$I_m = \langle a_m, \infty) \times \langle b_m, \infty) \times \langle c_m, C_m) \times \langle d_m, D_m)$$

and $c_0 \in \langle c_m, C_m)$ and $d_0 \in \langle d_m, D_m)$.

We suppose that the partial table $Tbp_{\approx}(a, c_0, d_0)$ of maximal b has infinitely many steps, thus there is also $a > a_m$ such that $Tbp_{\approx}(a, c_0, d_0) < Tbp_{\approx}(a + 1, c_0, d_0)$. Thus it is

$$\approx (a, Tbp_{\approx}(a + 1, c_0, d_0), c_0, d_0) = 0 \, .$$

Let us denote $b' = \max(b_m, Tbp_{\approx}(a + 1, c_0, d_0))$, thus it is $\approx (a, b', c_0, d_0) = 0$ because of $\approx$ is equivalency (see also point II in the proof of the theorem 1).

It is however $\langle a, b', c_0, d_0\rangle \in I_m$ and it means that $\approx (a, b', c_0, d_0) = 1$. It is a contradiction that finishes the proof.

5.4 Definability of Implicational Quantifiers

The next theorem shows that the necessary condition of definability of equivalency rules proved in theorem 4 is also the sufficient condition of definability of implicational quantifiers.

Theorem 5 *Let $\Rightarrow^*$ be an implicational quantifier. Then $\Rightarrow^*$ is classically definable if and only if its table of maximal b has only finite number of steps.*

Proof: Let $Tb_{\Rightarrow^*}$ be a table of maximal b of $\Rightarrow^*$.

If $\Rightarrow^*$ is classically definable then we prove that $Tb_{\Rightarrow^*}$ has only finite number of steps in a similar way like we proved in the theorem 4 that the partial table $Tbp_{\approx}(a, c_0, d_0)$ of maximal b has finite number of steps.

Let us suppose that $Tb_{\Rightarrow^*}$ has K steps where $K \geq 0$ is a natural number. We prove that $Tb_{\Rightarrow^*}$ is classically definable.

First let us suppose that $K = 0$. We distinguish two cases: $Tb_{\Rightarrow^*}(1) = 0$ and $Tb_{\Rightarrow^*}(1) > 0$.

If it is $Tb_{\Rightarrow^*}(1) = 0$ then it is also $Tb_{\Rightarrow^*}(0) = 0$ (there is no step). It but means that $\Rightarrow^* (a, b, c, d) = 0$ for each 4ft table $\langle a, b, c, d\rangle$ because of it cannot be $b < 0$. Thus it is $\Rightarrow^* (a, b, c, d) = 1$ if and only if $\langle a, b, c, d\rangle \in \emptyset$. The empty set $\emptyset$ is but also the interval in $\mathcal{N}^4$ and the quantifier $\Rightarrow^*$ is according to the theorem 3 classically definable.

If it is $K = 0$ and $Tb_{\Rightarrow^*}(1) > 0$ then it is $\Rightarrow^* (a, b, c, d) = 1$ if and only if

$$\langle a, b, c, d\rangle \in \langle 0, \infty) \times \langle 0, Tb_{\Rightarrow^*}(1)) \times \langle 0, \infty) \times \langle 0, \infty)$$

and thus the quantifier $\Rightarrow^* (a, b, c, d)$ is definable according to the theorem 3.

Let us suppose that $S > 0$ is a natural number and that

$$0 \leq a_1 < a_2 < \ldots < a_S$$

are all the steps in $Tb_{\Rightarrow^*}$. We will define intervals $I_1, I_2, \ldots, I_{S+1}$ in the following way.

If $Tb_{\Rightarrow^*}(a_1) = 0$ then $I_1 = \emptyset$ otherwise

$$I_1 = \langle 0, a_1 + 1) \times \langle 0, Tb_{\Rightarrow^*}(a_1)) \times \langle 0, \infty) \times \langle 0, \infty) \, .$$

For $j = 2, \ldots, S$ we define

$$I_j = \langle a_{j-1}, a_j + 1) \times \langle 0, Tb_{\Rightarrow^*}(a_j)) \times \langle 0, \infty) \times \langle 0, \infty) \, .$$

The interval I_{S+1} is defined such that

$$I_{S+1} = \langle a_S, \infty) \times \langle 0, Tb_{\Rightarrow^*}(a_S)) \times \langle 0, \infty) \times \langle 0, \infty) \, .$$

It is clear that the intervals $I_1, I_2, \ldots, I_{S+1}$ are defined such that

$$\Rightarrow^* (a, b, c, d) = 1 \text{ iff } \langle a, b, c, d\rangle \in \bigcup_{j=1}^{S+1} I_j$$

and according to the theorem 3 the quantifier $\Rightarrow^*$ is definable. This finishes the proof.

6 Undefinability of Particular Quantifiers

First we prove that the 4ft-quantifiers $\Rightarrow_{p,Base}$ of founded implication, $\Rightarrow^!_{p,\alpha,Base}$ of lower critical implication are not classically definable. We will use the following lemmas.

Lemma 1 *Let $\Rightarrow^*$ be an implicational quantifier that satisfies the conditions*

a) *There is $A \geq 0$ such that for each $a \geq A$ there is b such that $\Rightarrow^* (a, b) = 0$.*

b) *For each $a \geq 0$ and $b \geq 0$ such that $\Rightarrow^* (a, b) = 0$ there is $a' \geq a$ for which it is $\Rightarrow^* (a', b) = 1$.*

Then the table $Tb_{\Rightarrow^*}$ of maximal b of $\Rightarrow^*$ has infinitely many steps.

Proof: If the quantifier $\Rightarrow^*$ satisfies the condition **a)** *then it is $Tb_{\Rightarrow^*}(a) < \infty$ for each $a \geq 0$. If the quantifier $\Rightarrow^*$ satisfies the condition* **b)** *then there is for each $a > A$ such $a' > a$ that $\Rightarrow^* (a', Tb_{\Rightarrow^*}(a)) = 1$. Thus it is $\Rightarrow^* (a, Tb_{\Rightarrow^*}(a)) = 0$ (by the definition of $Tb_{\Rightarrow^*}$) and $\Rightarrow^* (a', Tb_{\Rightarrow^*}(a)) = 1$. It means that between a and a' there must be a step s of the table $Tb_{\Rightarrow^*}$.*

We have proved that for each $a > A$ there is a step $s \geq a$ of the table $Tb_{\Rightarrow^*}$. It but means that the table $Tb_{\Rightarrow^*}$ has infinitely many steps. This finishes the proof.

Lemma 2 *Let us suppose that $\approx$ is an equivalency quantifier and c_0 and d_0 are natural numbers such that the following conditions are satisfied.*

a) *There is $A \geq 0$ such that for each $a \geq A$ there is b such that $\approx (a, b, c_0, d_0) = 0$.*

b) *For each $a \geq 0$ and $b \geq 0$ such that $\approx (a, b, c_0, d_0) = 0$ there is $a' \geq a$ for which $\approx (a', b, c, d) = 1$.*

Then the partial table $Tbp_{\approx}(a, c_0, d_0)$ of maximal b of $\approx$ has infinitely many steps.

Proof: The proof is similar to the proof of the lemma 2.

Lemma 3 *Let us suppose that $0 < p < 1$ and $i \geq 0$ is a natural number. Then it is*

$$\lim_{K \to \infty} \binom{K}{i} p^i (1-p)^{K-i} = 0 \cdot$$

Proof: It is:

$$\binom{K}{i} p^i (1-p)^{K-i} \leq K^i p^i (1-p)^K (1-p)^{-i} =$$

$$= K^i (1-p)^K \left(\frac{p}{1-p}\right)^i .$$

Thus it is enough to prove that for $r \in (0,1)$ and $i \geq 0$ it is

$$\lim_{K \to \infty} K^i r^K = 0 \cdot$$

To prove this it is enough to prove that for $r \in (0,1)$, real x and a natural $i \geq 0$ it is

$$\lim_{x \to \infty} x^i r^x = 0 .$$

It is $\lim_{x\to\infty} x^i = \infty$, $\lim_{x\to\infty} r^x = 0$ and thus according to the l'Hospital's rule it is

$$\lim_{x \to \infty} x^i r^x = \lim_{x \to \infty} \frac{x^i}{r^{-x}} = \lim_{x \to \infty} \frac{(x^i)^{(i)}}{(r^{-x})^{(i)}} =$$

$$= \lim_{x \to \infty} \frac{i!}{(-\ln r)^i r^{-x}} = \lim_{x \to \infty} r^x = 0 ,$$

where $(x^i)^{(i)}$ is an i-th derivation of x^i and analogously for $(r^{-x})^{(i)}$.

Lemma 4 *Let us suppose $a \geq 0$ and $b \geq 0$ are natural numbers. Then it is for each $k \in \langle 0, b \rangle$ and $0 < p < 1$*

$$\lim_{a \to \infty} \binom{a+b}{a+k} p^{a+k} (1-p)^{b-k} = 0 \cdot$$

Proof: It is:

$$\binom{a+b}{a+k} p^{a+k} (1-p)^{b-k} =$$

$$= \binom{a+b}{a+b-(a+k)} p^{a+k} (1-p)^{b-k} =$$

$$= \binom{a+b}{b-k} p^{a+k} (1-p)^{b-k} \leq$$

$$\leq (a+b)^{b-k} p^{a+k} (1-p)^{b-k} .$$

Thus it is enough to prove that it is

$$\lim_{a \to \infty} (a+b)^{b-k} p^a = 0 \cdot$$

The proof of this assertion is similar to the proof of the assertion

$$\lim_{K \to \infty} K^i r^K = 0 \cdot$$

in the lemma 3.

Lemma 5

1. *The 4ft-quantifier $\Rightarrow_{p,Base}$ of founded implication satisfies the condition* **a)** *from the lemma 1 for each $0 < p \leq 1$ and $Base > 0$.*
2. *The 4ft-quantifier $\Rightarrow^!_{p,\alpha,Base}$ of lower critical implication satisfies the condition* **a)** *from the lemma 1 for each $0 < p < 1$, $0 < \alpha < 0.5$ and $Base > 0$.*

Proof:

1. *We have to prove that there is $A \geq 0$ such that for each $a \geq A$ there is b such that $\Rightarrow_{p,Base} (a, b) = 0$ for each $0 < p \leq 1$ and $Base > 0$. Let us remember that the 4ft-quantifier $\Rightarrow_{p,Base}$ is defined by the condition $\frac{a}{a+b} \geq p \wedge a \geq Base \cdot$*

   Let be $A > Base$ and $a \geq A$. Then we choose b' such that $b' > \frac{a - p*a}{p}$. Then it is $\Rightarrow_{p,Base} (a, b') = 0$.

2. *We have to prove that there is $A \geq 0$ such that for each $a \geq A$ there is b such that $\Rightarrow^{!}_{p,\alpha,Base}(a,b) = 0$ for each $0 < p < 1$, $0 < \alpha < 0.5$ and $Base > 0$. Let us remember that the 4ft-quantifier $\Rightarrow^{!}_{p,\alpha,Base}$ is defined by the condition*

$$\sum_{i=a}^{a+b} \binom{a+b}{i} p^i (1-p)^{a+b-i} \leq \alpha \wedge a \geq Base\,.$$

Let be $A > Base$ and $a \geq A$. We show that there is a natural b such that

$$\sum_{i=a}^{a+b} \binom{a+b}{i} p^i (1-p)^{a+b-i} > \alpha\,.$$

It is $\sum_{i=a}^{a+b} \binom{a+b}{i} p^i (1-p)^{a+b-i} > \alpha$ if and only if

$$\sum_{i=0}^{a-1} \binom{a+b}{i} p^i (1-p)^{a+b-i} \leq 1-\alpha$$

because of $\sum_{i=0}^{a+b} \binom{a+b}{i} p^i (1-p)^{a+b-i} = 1$.

According to the lemma 3 there is a natural $V > a$ such that it is

$$\binom{V}{i} p^i (1-p)^{V-i} \leq \frac{1-\alpha}{a}$$

for $i = 0, \ldots, a-1$. Thus it is

$$\sum_{i=0}^{a-1} \binom{V}{i} p^i (1-p)^{V-i} \leq 1-\alpha$$

Let us choose $b = V - a$. It means

$$\sum_{i=0}^{a-1} \binom{a+b}{i} p^i (1-p)^{a+b-i} \leq 1-\alpha$$

and it finishes the proof.

Lemma 6

1. *The 4ft-quantifier $\Rightarrow_{p,Base}$ of founded implication satisfies the condition* **b)** *from the lemma 1 for each $0 < p \leq 1$ and $Base > 0$.*
2. *The 4ft-quantifier $\Rightarrow^{!}_{p,\alpha,Base}$ of lower critical implication satisfy the condition satisfies the condition* **b)** *from the lemma 1 for each $0 < p < 1$, $0 < \alpha < 0.5$ and $Base > 0$.*

Proof:

1. *We have to prove that for each $a \geq 0$ and $b \geq 0$ such that $\Rightarrow_{p,Base}(a,b) = 0$ there is $a' \geq a$ for that it is $\Rightarrow_{p,Base}(a,b) = 1$. The proof is trivial, we use the fact that $\lim_{a\to\infty} \frac{a}{a+b} = 1$.*
2. *We have to prove that for each $a \geq 0$ and $b \geq 0$ such that $\Rightarrow^{!}_{p,\alpha,Base}(a,b) = 0$ there is $a' \geq a$ for that it is $\Rightarrow^{!}_{p,\alpha,Base}(a,b) = 1$.*

   Let us suppose that $\Rightarrow^{!}_{p,\alpha,Base}(a,b) = 0$. It means that $a < Base$ or $\sum_{i=a}^{a+b} \binom{a+b}{i} p^i (1-p)^{a+b-i} > \alpha$.

   According to the lemma 4 there is natural n such that for each e, $e > n$ and $k = 0, \ldots, b$ it is

$$\binom{e+b}{e+k} p^{e+k} (1-p)^{b-k} < \frac{\alpha}{b+1}.$$

   Let us choose $a' = \max\{a, n, Base\}$. Then it is $a' \geq Base$ and also

$$\sum_{i=a'}^{a'+b} \binom{a'+b}{i} p^i (1-p)^{a'+b-i} =$$

$$= \sum_{k=0}^{b} \binom{a'+b}{a'+k} p^{a'+k} (1-p)^{b-k} < \alpha\ .$$

   Thus it is $\Rightarrow^{!}_{p,\alpha,Base}(a',b) = 1$ and it finishes the proof.

Theorem 6 *The 4ft-quantifier $\Rightarrow_{p,Base}$ of founded implication is not classically definable for each $0 < p \leq 1$ and $Base > 0$.*

The 4ft-quantifier $\Rightarrow^{!}_{p,\alpha,Base}$ of lower critical implication is not classically definable for each $0 < p < 1$, $0 < \alpha < 0.5$ and $Base > 0$.

Proof: The table of maximal b of the 4ft-quantifier $\Rightarrow_{p,Base}$ of founded implication has infinitely many steps according to the lemmas 2, 5 and 6. Thus it is not classically definable according to the theorem 5.

The proof for the quantifier $\Rightarrow^{!}_{p,\alpha,Base}$ is analogous.

Now we prove that the the Fisher's quantifier $\sim_{\alpha,Base}$, is not classically definable. Let us remember that it is defined for $0 < \alpha < 0.5$ and $Base > 0$ by the condition

$$\sum_{i=a}^{\min(r,k)} \frac{\binom{k}{i}\binom{n-k}{r-i}}{\binom{r}{n}} \leq \alpha \ \wedge\ ad > bc \ \wedge\ a \geq Base\cdot$$

We use the following results from [2].

Definition 2 *(see [2]) The equivalency quantifier $\approx$ is* **saturable** *if it satisfies:*

1. *For each 4ft-table $\langle a,b,c,d\rangle$ with $d \neq 0$ there is an $a' \geq a$ such that $\approx(a',b,c,d) = 1$.*
2. *For each 4ft-table $\langle a,b,c,d\rangle$ with $a \neq 0$ there is an $d' \geq d$ such that $\approx(a,b,c,d') = 1$.*

3. *For each 4ft-table* $\langle a, b, c, d\rangle$ *there is a 4ft-table* $\langle a', b', c', d'\rangle$ *such that* $a' \geq a$, $b' \geq b$, $c' \geq c$, $d' \geq d$ *and* $\approx (a', b', c', d') = 0$.

Theorem 7 *The Fisher's quantifier* $\sim_{\alpha,Base}$ *is saturable for* $0 < \alpha < 0.5$ *and* $Base > 0$.

Proof: See [2].

Lemma 7 *There are natural numbers* c_0 *and* d_0 *such that the Fisher's quantifier* $\sim_{\alpha,Base}$ *satisfies the conditions a) and b) from the lemma 2.*

Proof: We prove that the conditions a) and b) are satisfied for $c_0 = 1$ *and* $d_0 = 1$. *We have to proof*

a) *There is* $A \geq 0$ *such that for each* $a \geq A$ *there is* b *such that* $\sim_{\alpha,Base} (a, b, 1, 1) = 0$.

b) *For each* $a \geq 0$ *and* $b \geq 0$ *such that* $\approx (a, b, 1, 1) = 0$ *there is* $a' \geq a$ *for which* $\sim_{\alpha,Base} (a', b, 1, 1) = 1$.

Let us choose $b = a + 1$ *for each* $a \geq Base$, *then it is* $ad < bc$ *and thus it is* $\sim_{\alpha,Base} (a, b, 1, 1) = 0$. *It means that the condition* **a** *is satisfied.*

The condition **b** *follows from the fact that the Fisher's quantifier* $\sim_{\alpha,Base}$ *is saturable, see theorem 7.*

Theorem 8 *The Fisher's quantifier* $\sim_{\alpha,Base}$ *is not classically definable for each* $0 < \alpha < 0.5$ *and* $Base > 0$.

Proof: The partial table $Tbp_{\sim_{\alpha,Base}}(a, 1, 1)$ *of maximal* b *of* $\sim_{\alpha,Base}$ *has infinitely many steps according to the lemmas 7 and 2. Thus it is not classically definable according to the theorem 4.*

7 Conclusions

We have presented a simple criterion of classical definability of the important 4ft-quantifiers. This criterion is based on the tables of critical frequencies that are itself important tool for verification of association rules. This criterion depends on the class of association rules (i.e. the class of 4ft-quantifiers) we deal with. We also pointed out to the relations of the classes of association rules to the important deduction rules concerning association rules see Sect. 3.

Let us remark that there are further interesting and practically useful relations of tables of critical frequencies, classes of association rules, logical properties of association rules and properties of association rules in the data with missing information. They are partly published in [2, 5, 6, 7] and in more details investigated in [4, 8].

Acknowledgement: The work described here has been supported by grants LN 00B 107 of the Ministry of Education of the Czech Republic, COST ACTION 274 - TARSKI, and by project IGA 17/04 of University of Economics, Prague.

References

[1] Aggraval R, et al (1996) Fast Discovery of Association Rules. In: Fayyad UM, et al (eds). Advances in Knowledge Discovery and Data Mining. AAAI Press, Menlo Park California

[2] Hájek P, Havránek T (1978) Mechanising Hypothesis Formation - Mathematical Foundations for a General Theory. Springer, Berlin Heidelberg New York

[3] Hájek P, Havránek T, Chytil M (1983) GUHA Method. Academia, Prague (in Czech)

[4] Rauch J (1986) Logical Foundations of Hypothesis Formation from Databases. PhD Thesis, Mathematical Institute of the Czechoslovak Academy of Sciences, Prague (in Czech)

[5] Rauch J (1997) Logical Calculi for Knowledge Discovery in Databases. In: Zytkow J, Komorowski J (eds) Principles of Data Mining and Knowledge Discovery. Springer, Berlin Heidelberg New York

[6] Rauch J (1998) Classes of Four-Fold Table Quantifiers. In: Zytkow J, Quafafou M (eds) Principles of Data Mining and Knowledge Discovery. Springer, Berlin Heidelberg New York

[7] J. Rauch (1998) "Four-Fold Table Calculi and Missing Information" in Proc. Joint Conference on Information Sciences, Durham, North Carolina

[8] Rauch J(1998) Contribution to Logical Foundations of KDD. Assoc. Prof. Thesis, Faculty of Informatics and Statistics, University of Economics, Prague (in Czech)

[9] J. Rauch (2003) Definability of Association Rules in Predicate Calculus. In: Lin T Y, Hu X, Ohsuga S, Liau C J (eds) Foundations and New Directions in Data Mining. IEEE Computer Society, Melbourne, Florida

[10] Rauch J, Šimůnek M (2002) Alternative Approach to Mining Association Rules. In: Lin T Y, Ohsuga S (eds) The Foundation of Data Mining and Knowledge Discovery (FDM02), IEEE Computer Society, Maebashi

[11] Tharp L H.(1973) The characterisation of monadic logic. Journal of Symbolic Logic 38: 481–488

On the Recursion Theoretic Complexity of Privacy Preserving Data Mining

Bhavani Thuraisingham
The University of Texas at Dallas
and
The MITRE Corporation

Data mining is the process of users posing queries and extracting information often previously unknown using machine learning and statistical reasoning techniques. Because of data mining tools, even naive users can now make correlations and associations. If the extracted information is sensitive then there could be security violations. Furthermore, the extracted information could violate the privacy of individuals. That is, data mining is essentially a threat to security and privacy of individuals. Much of the recent work has focused on privacy preserving data mining where the goal is to carry out data mining, but at the same time ensure the privacy of the individuals as much as possible.

In this paper we examine the privacy problem that results from data mining as well as making associations and deductions and explore the recursion theoretic complexity of the privacy problem. We view the privacy problem as an aspect of the inference problem and give a definition of the problem based on deductive databases. We then state and prove the unsolvability of the general privacy problem and then obtain a characterization of this problem with respect to recursion theory. We then provide directions for examining the computational complexity of the privacy problem.

Ensembles of Least Squares Classifiers with Randomized Kernels

Kari Torkkola
Motorola, Intelligent Systems Lab, Tempe, AZ, USA,
Kari.Torkkola@motorola.com

Eugene Tuv
Intel, Analysis and Control Technology, Chandler, AZ, USA,
eugene.tuv@intel.com

Abstract

For the recent NIPS-2003 feature selection challenge we studied ensembles of regularized least squares classifiers (RLSC). We showed that stochastic ensembles of simple least squares kernel classifiers give the same level of accuracy as best single RLSC. Results achieved were ranked among top best at the challenge. We also showed that performance of a single RLSC is much more sensitive to the choice of kernel width than that of an ensemble. As a continuation of this work we demonstrate that stochastic ensembles of least squares classifiers with randomized kernels and OOB-postprocessing often outperform the best single RLSC, and require practically no tuning. We used the same set of very high dimensional classification problems presented at the NIPS challenge. Fast exploratory Random Forests were applied for variable filtering first.

1. Introduction

Regularized least-squares regression and classification dates back to the work of Tikhonov and Arsenin [15], and has been re-advocated and revived recently by Poggio, Smale and others [13, 6, 14, 12]. Regularized Least Squares Classifier (RLSC) is an old combination of quadratic loss function combined with regularization in reproducing kernel Hilbert space, leading to a solution of a simple linear system. In many cases in the work cited above, this simple RLSC appears to equal or exceed the performance of support vector machines and other modern developments in machine learning.

This simplicity of the RLSC approach is a major thread in this paper. We verify the above mentioned findings using the NIPS 2003 Feature Selection Challenge datasets. All these five datasets define binary classification problems.

The combination of RLSC with Gaussian kernels and the usual choice of spherical covariances gives an equal weight to every component of the feature vector. This poses a problem if a large proportion of the features consists of noise. With the datasets of the challenge this is exactly the case. In order to succeed in these circumstances, noise variables need to be removed or weighted down. We apply *ensemble-based variable filtering* to remove noise variables. A Random Forest (RF) is trained for the classification task, and an importance measure for each variable is derived from the forest [4]. Only highest ranking variables are then passed to RLSC. We chose Random Forests (RF) for this task for several reasons. RF can handle huge numbers of variables easily and global relative variable importance is derived as a by-product of the forest construction with no extra computation involved.

In this paper we study empirically how a stochastic ensemble of RLSCs with random kernel widths compares to a single optimized RLSC. Our motivation to do this is the well known fact that ensembles of simple weak learners are known to produce stable models that often significantly outperform an optimally tuned single base learner [3, 9, 4]. Another motivating factor is the elimination of the kernel width and regularization parameter selection procedures altogether. A further advantage of ensembles is the possibility of parallelization. Using much smaller sample sizes to train each expert of an ensemble could be faster than training a single learner using a huge data set.

For an ensemble to be effective, the individual experts need to have low bias and the errors they make should be uncorrelated [2, 4]. Using no regularization with LSC reduces the bias of the learner making it a good candidate for ensemble methods. Diversity of the learners can be accomplished by training base learners using independent random samples of the training data and by using random kernel widths. The latter is the main topic of this paper.

The structure of this paper is as follows. We begin by briefly describing the RLSC, theory behind it, and its con-

nections to support vector machines. We discuss ensembles, especially ensembles of RLSCs and the interplay of regularization and bias in ensembles. The scheme for variable filtering using ensembles of trees is discussed next, after which we describe experimentation with the NIPS2003 feature selection challenge data sets. We discuss our findings regarding ensembles of random kernel LSCs, and conclude by touching upon several possible future directions.

2. Regularized Least-Squares Classification (RLSC)

In supervised learning the training data $(x_i, y_i)_{i=1}^m$ is used to construct a function $f : X \to Y$ that predicts or generalizes well. To measure goodness of the learned function $f(x)$ a loss function $L(f(x), y_{true})$ is needed. Some commonly used loss functions for regression are as follows:

- Square loss or L_2: $L(f(x), y) = (f(x)-y)^2$ (the most common),
- Absolute value, or L_1 loss: $L(f(x), y) = |f(x) - y|$,
- Vapnik's ϵ-insensitive loss: $L(f(x), y) = (|f(x)-y|-\epsilon)_+$,
- Huber's loss function : $\begin{cases} |y - f(x)|^2, & \text{for } |f(x) - y| \le \delta \\ \delta(|y - f(x)| - \delta/2), & \text{otherwise} \end{cases}$

Examples of loss functions for classification are

- Misclassification: $L(f(x), y) = I(sign(f(x)) \neq y)$
- Exponential (Adaboost): $L(f(x), y) = exp(-yf(x))$
- Hinge loss (implicitly introduced by Vapnik) in binary SVM classification: $L(f(x), y) = (1 - yf(x)) \cdot I(yf(x) > 1)$
- Binomial deviance: $L(f(x), y) = log(1 + exp(-2yf(x)))$
- Squared error: $L(f(x), y) = (1 - yf(x))^2$

Given a loss function, the goal of learning is to find an approximation function $f(x)$ that minimizes the expected risk, or the generalization error

$$E_{P(x,y)} L(f(x), y) \tag{1}$$

where P(x,y) is the unknown joint distribution of future observations (x,y).

Given a finite sample from the (X,Y) domain this problem is ill-posed. The regularization approach championed by Tomaso Poggio and rooted in Tikhonov regularization theory [15] restores well-posedness (existence, uniqueness, and stability) by restricting the hypothesis space, the functional space of possible solutions:

$$\hat{f} = arg \min_{f \in H} \frac{1}{m} \sum_{i=1}^{m} L(f(x_i), y_i) + \gamma \|f\|_K^2 \tag{2}$$

The hypothesis space H here is a Reproducing Kernel Hilbert Space (RKHS) defined by kernel K, and γ is a positive regularization parameter.

The mathematical foundations for this framework as well as a key algorithm to solve (2) are derived elegantly in [13] for the quadratic loss function. The algorithm can be summarized as follows:

1. Start with the data $(x_i, y_i)_{i=1}^m$.
2. Choose a symmetric , positive definite kernel, such as
$$K(x, x') = e^{-\frac{||x-x'||^2}{2\sigma^2}}. \tag{3}$$
3. Set
$$f(x) = \sum_{i=1}^{m} c_i K(x_i, x), \tag{4}$$
where $\mathbf{c}$ is a solution to
$$(m\gamma \mathbf{I} + \mathbf{K})\mathbf{c} = \mathbf{y}, \tag{5}$$
which represents well-posed linear ridge regression.

The generalization ability of this solution, as well choosing the regularization parameter γ were studied by Cucker and Smale in [6, 7].

Thus, the regularized least-squares algorithm (RLSC) solves a simple well defined linear problem. The solution is a linear kernel expansion of the same form as the one given by support vector machines (SVM). Note also that SVM formulation naturally fits in the regularization framework (2). Inserting the SVM hinge loss function $L(f(x), y) = (1 - yf(x))_+$ to (2) leads to solving a quadratic optimization problem instead of a linear solution to find coefficients $\mathbf{c}$ in (4).

RLSC with quadratic loss function, that is more common for regression, has also proven to be very effective in binary classification problems [14].

3. Model Averaging and Regularization

3.1. Stability

Generalization ability of a learned function is closely related to its stability. Stability of the solution could be loosely defined as continuous dependence on the data. A stable solution changes very little for small changes in data. A comprehensive treatment of this connection can be found in [2].

Furthermore, it is well known that bagging (bootstrap aggregation) can dramatically reduce variance of unstable learners providing some regularization effect [3]. Bagged ensembles do not overfit, and they are limited by learning power of base learners. Key to the performance is a low bias of the base learner, and low correlation between base learners.

Evgeniou experimented with ensembles of SVMs [8]. He used a few datasets from UCI tuning all parameters separately for both a single SVM and for an ensemble of SVMs to achieve the best performance. He found that both perform similarly. He also found that generalization bounds for ensembles are tighter than for a single machine.

Poggio et al [12] studied the relationship between stability and bagging. They showed that there is a bagging scheme, where each expert is trained on a disjoint subset of the training data, providing strong stability to ensembles of non-strongly stable experts, and therefore providing the same order of convergence for the generalization error as Tikhonov regularization. Thus, at least asymptotically, bagging strongly stable experts would not improve generalization ability of the individual member.

3.2. Ensembles of RLSCs

Since the sizes of the challenge datasets are relatively small, we compare simple stochastic aggregation of LSCs using random kernel widths to the best individually trained RLSC.

We are looking for diverse low biased experts: for RLSC bias is controlled by regularization parameter, and σ in case of Gaussian kernel. Instead of bootstrap sampling from training data which imposes a fixed sampling strategy, we found that often much smaller sample sizes of the order of 30-50% of the data set size improve performance. A further source of diversity is introduced by each expert having a different random kernel width.

Combining the outputs of the experts in an ensemble can be done in several ways. The simplest alternative is majority voting over the outputs of the experts. In binary classification this is equivalent to averaging the discretized (+1,-1) predictions of the experts. In our experiments this performed better than averaging the actual numeric expert outputs before applying their decision function (sign).

A well known avenue to improve the accuracy of an ensemble is to replace the simple averaging of individual experts by a weighting scheme. Instead of giving equal weight to each expert, the outputs of more reliable experts are weighted up. Linear regression can be applied to learn these weights.

To avoid overfitting, the training material to learn this regression should be produced by passing only such samples through an expert, that did not participate in construction of the particular expert. Typically this is done by using a separate validation data set. Since some of the datasets used were very small in size, it was not useful to split the training sets further for this purpose. Instead, since each expert is constructed only from a fraction of the training data set, the rest of the data is available as "out-of-bag samples" (oob).

We experimented with two schemes to construct the training data matrix. Since each expert populates the matrix only with oob-samples, the empty spaces corresponding to the training data of the expert can be filled in either with zeroes, or with the expert outputs by passing the training data through the expert. The latter is optimistically biased, and the former is biased toward zero, the "don't know" condition. In the latter case we also upweighted the entries by the reciprocal of the fraction of missing entries in order to compensate for the inner product of the regression coefficients with the entries to sum to either plus or minus one.

Since expert outputs are correlated (although the aim is to have uncorrelated experts!) PCA regression can be applied to reduce the number of regression coefficients. Partial Least Squares regression could also be used instead of PCA regression.

4. Variable Filtering with Tree-Based Ensembles

Practically for all datasets (except arcene) from the challenge we noticed significant improvement in accuracy when only small (but important) fraction of the original variables was used in kernel construction.

We used fast exploratory tree-based models for variable filtering. One of many important properties of CART [5] is its embedded ability to select important variables during tree construction (greedy recursive partition, where impurity reduction is maximized at every step), and therefore resistance to noise. Variable importance then can be defined as

$$M(x_m, T) = \sum_{t \in T} \Delta I(x_m, t) \tag{6}$$

where $\Delta I(x_m, t)$ is the decrease in impurity due to an actual or potential split on variable x_m at a node t of the optimally pruned tree T. The sum in (6) is taken over all internal tree nodes where x_m was a primary splitter or a surrogate variable. Consequently, no additional effort is needed for its calculation.

Two recent advances in tree ensembles - Multivariate Adaptive Regression Trees (MART) [10, 11] and Random Forests (RF) [4] inherit all nice properties of a single tree, and provide more reliable estimate of this value, as the importance measure is averaged over the trees in the ensemble

$$M(x_i) = \frac{1}{M} \sum_{m=0}^{M} M(x_i, T_m). \tag{7}$$

MART builds shallow trees using all variables, and hence, can handle large datasets with moderate number of variables. RF builds maximal trees but chooses a small random subset of variables at every split, and easily handles thousands of variables in datasets of moderate

size. For datasets massive in both dimensions a hybrid scheme with shallow trees and dynamic variable selection has been shown to have at least the same accuracy but to be much faster than either MART or RF [1].

Note that the index of variable importance defined in the above measures is the global contribution of a variable to the learned model. It is not just a univariate response-predictor relationship.

For the NIPS challenge we used RF to select important variables. Forest was grown using the training data until there was no improvement in the generalization error. Typically, this limit was around 100. As an individual tree is grown, a random sample of the variables is drawn, out of which the best split is chosen (instead of considering all of the variables). The size of this sample was typically $\sqrt{N}$.

5. Experiments with NIPS 2003 Feature Selection Challenge Data Sets

The purpose of the NIPS 2003 challenge in feature selection was to find feature selection algorithms that significantly outperform methods using all features, on all five benchmark datasets. The datasets and their (diverse) characteristics are listed in Table 1.

Of these data sets, only Dorothea was highly unbalanced with approximately 12% of samples in one class, and 88% in the other. The rest of the sets had an approximately balanced class distribution. All tasks are two-class classification problems.

5.1. Variable Selection Experiments

Initial experimentation was performed to determine whether variable selection was necessary at all. We trained ELSCs for madelon and dexter data sets. Results are given in Table 2 as the averages of ten-fold cross validation.

These results clearly indicated that RLSC is sensitive to noise variables in data, and that variable selection based on importances derived from Random Forests works well.

For the rest of the experiments, we adopted the following variable selection procedure. Variables are ranked by a random forest as described in Sec. 4. If there are significant cut-off points in the ranked importance, the variable set before the cut-off point is selected. Figure 1 shows a clear example of such a cut-off point.

For each data set, the smallest possible variable set as indicated by a cut-off point was tried first. If the results were unsatisfactory, the next cut-off point was searched, and so on, until satisfactory results were obtained. The maximum number of variables considered was about 500. Full cross-validation was thus not done over the whole possible range of the number of selected variables.

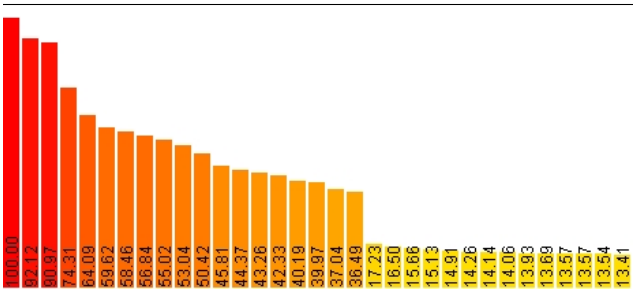


Figure 1. The importance of the top 33 out of 500 variables of Madelon derived from a training set of 2000 cases in 500 trees. Variable importance has a clear cut-off point at 19 variables.

Variable set was thereafter fixed to the one that produced the smallest cross-validation error, with two exceptions: Contrary to other data sets, on arcene the error rate using the validation set did not follow cross-validation error but was the smallest when all variables were used. Arcene is evidently such a small data set that variable selection and classifier training both using the 100 training samples, will overfit. The second exception is dexter, which gave the best results using 500 variables ranked by maximum mutual information with the class labels [16].

At this point we also experimented with variable standardization and weighting variables by their importance. Due to lack of space these experiments are not tabulated, but the decisions are summarized in table 3

5.2. Classification experiments with ELSCs using random kernels

An individual RLSCs has two parameters that need to be determined by cross-validation. These are the kernel width σ^2 and the regularization parameter γ. For a single RLSC, regularization is critical in order not to overfit. The choice of the parameter needs to be made by cross-validation, and appears to be very data dependent. This leads to optimization in a two-dimensional parameter space using cross-validation. As an example, we present this optimization for the Madelon data set in Fig. 2.

An ensemble of stochastic LSCs is less sensitive to kernel width, does not require search for the regularization parameter, is not sensitive to the ensemble size (once it is large enough), and is not very sensitive to the fraction of data sampled to train each LSC [17]. Our motivation in using random kernels, or more precisely, random kernel widths,

Set	Size	Type	Number of of variables	Training Examples	Validation Examples
Arcene	8.7 MB	Dense	10000	100	100
Gisette	22.5 MB	Dense	5000	6000	1000
Dexter	0.9 MB	Sparse integer	20000	300	300
Dorothea	4.7 MB	Sparse binary	100000	800	350
Madelon	2.9 MB	Dense	500	2000	600

Table 1. NIPS2003 Feature Selection Challenge Data

Data set	Variables	Error rate using all variables	Selected variables	Error rate using selected variables
madelon	500	0.254	19	0.093
dexter	20000	0.324	109	0.074

Table 2. Comparison of no variable selection to variable selection.

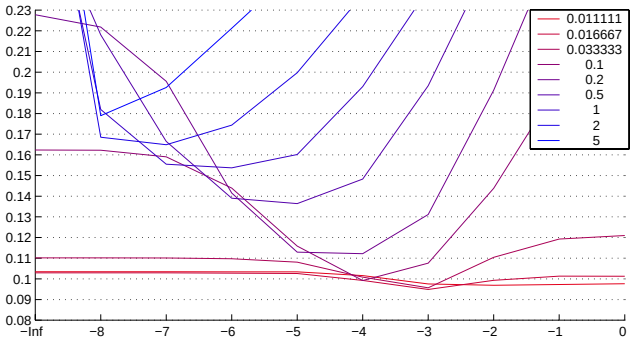


Figure 2. Single RLSC: Cross-validation experimentation in order to find the optimal combination of kernel width and regularization parameter for madelon data set. Vertical axis is the 10-fold cross-validation error rate on training data, horizontal axis is $\log_{10}(\gamma)$, and each curve corresponds to a specific kernel width. Legend displays the multiplier to $d_{av}^2 = 37.5$.

was to get rid of all these tunable parameters in ensemble construction without sacrificing any of the generalization performance.

Naturally, the kernel width cannot be completely random, but in a reasonable range, which is determined by the data. We sampled the σ^2 uniformly in the range of $[d_{med}^2, 4d_{min}^2]$, where d_{med} is the median distance between samples, and d_{min} is the minimum distance between samples. This was found to be a reasonable range for all the five diverse challenge datasets.

Data set	Optimized RLSC	Optimized ELSC	Random kernel ELSC
arcene	0.1331	0.1331	0.1130
gisette	0.0210	0.0210	0.0200
dorothea	0.1183	0.1183	0.1140
madelon	0.0700	0.0667	0.0717
dexter	0.0633	0.0633	0.0700

Table 4. Error rates using the separate validation data set after optimizing σ^2 and γ for a single RLSC, and σ^2 and the fraction of data sampled for each LSC in an ensemble of 200 classifiers. Random kernel ELSC required no parameter tuning.

The ensemble size was fixed to 200, and the fraction of training data to train each LSC was fixed to 0.5. These were near-optimal values for ELSCs according to our earlier experiments [17].

Ensemble output combination was done using PCA-regression. We experimented also with plain regression using a mixture of training/oob samples or just the oob-samples, but the differences were insignificant.

We present the final classification error rates in table 4. Even though there is no significant difference in validation error rates between using a single RLSC with optimized parameters, an ELSCs with optimized parameters, or an ELSC with random kernel width, the fact that the latter can be trained without any necessary parameter/model selection makes it a desirable alternative.

Data set	Original variables	Selected variables	Selection method	Standardize?	Weighting?
madelon	500	19	RF	yes	no
dexter	20000	500	MMI	yes	by MI
arcene	10000	10000	none	no	no
gisette	5000	307	RF	no	no
dorothea	100000	284	RF	no	no

Table 3. Variable selection, standardization, and variable weighting decisions.

6. Future Directions

We describe an approach in this paper that consists of two disjoint systems, Random Forests for variable selection, and ELSC for the actual classification. Even though the two systems nicely complement each other, RF providing fast embedded variable selection and ELSC providing highly capable base learners to compensate for the lack of smoothness of the trees of an RF, an integrated approach would be desirable. We describe an idea towards such a system.

RF could act as one type of supervised kernel generator using the pairwise similarities between cases. Similarity for a single tree between two cases could be defined as the total number of common parent nodes, normalized by the level of the deepest case, and summed up for the ensemble. Minimum number of common parents to define nonzero similarity is another parameter that could be used like width in Gaussian kernels.

Figure 3 illustrates the difference between a Gaussian kernel and the proposed supervised kernel.

An advantage of the method is that it works for any type of data, numeric, categorical, or mixed, even for data with missing values. This is because the base learners of the Random Forest can tolerate these.

A further advantage is that explicit variable selection is bypassed altogether. Important variables will become used in the trees of the forest, and they thus participate implicitly in the evaluation of the kernel.

7. Conclusion

We proposed a relatively straightforward approach to create powerful ensembles of simple least square classifiers with random kernels. We used recent NIPS2003 feature selection challenge data to evaluate performance of such ensembles. The binary classification data sets considered in the challenge originated in different domains with number of variables ranging from moderate to extremely large and moderate to very small number of observations. We used fast exploratory Random Forests for variable filtering as a preprocessing step. The individual learners were trained on small random sample of data with Gaussian kernel width randomly selected from relatively wide range of values determined only by basic properties of the corresponding dissimilarities matrix. The random sample of data used to build individual learner was relatively small. Modest ensemble size (less than 200) stabilized the generalization error. We used consistent parameter settings for all datasets, and achieved at least the same accuracy as the best single RLSC or an ensemble of LSCs with fixed tuned kernel width. Individual learners were combined through simple OOB postprocessing PCA regression.

For high dimensional noisy problems variable filtering with fast exploratory ensembles of random trees (Random Forests with default parameter settings) showed to be very effective preprocessing procedure.

References

[1] A. Borisov, V. Eruhimov, and E. Tuv. Dynamic soft feature selection for tree-based ensembles. In I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh, editors, *Feature Extraction, Foundations and Applications*. Springer, New York, 2004. forthcoming, submitted.

[2] O. Bousquet and A. Elisseeff. Algorithmic stability and generalization performance. In *NIPS*, pages 196–202, 2000.

[3] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[4] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[5] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. CRC Press, 1984.

[6] F. Cucker and S. Smale. On the mathematial foundations of learning. *Bulletin of the American Mathematical Society*, 89(1):1–49, 2001.

[7] F. Cucker and S. Smale. Best choices for regularization parameters in learning theory: on the bias-variance problem. *Foundations of Computational Mathematics*, 2(4):413–428, 2003.

[8] T. Evgeniou. *Learning with Kernel Machine Architectures*. PhD thesis, Massachusetts Institute of Technology, EECS, July 2000.

[9] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37, 1995.

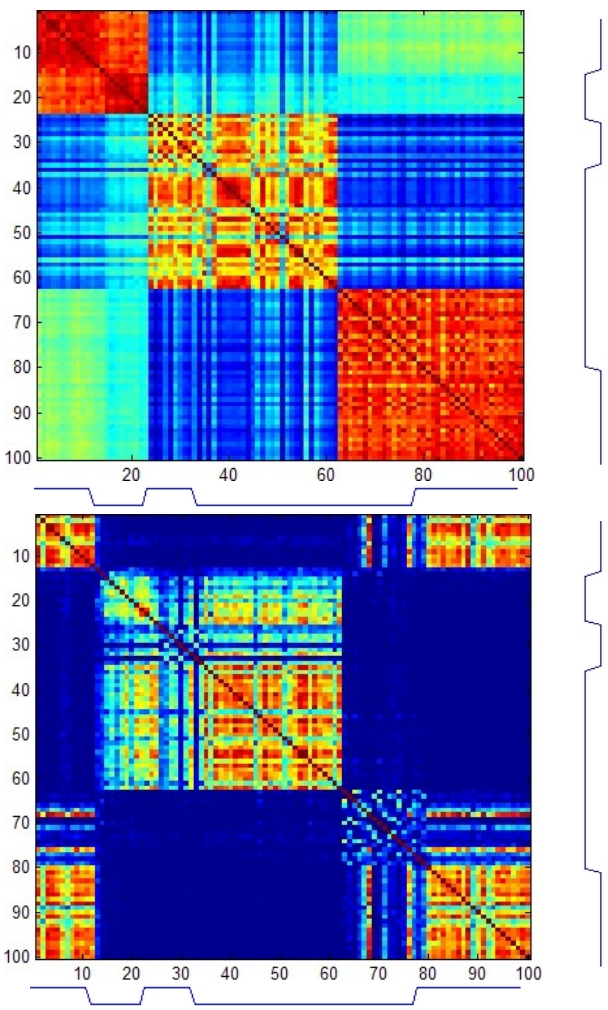


Figure 3. Gaussian kernel compared to a supervised kernel using the Arcene dataset. Left side depicts the 100×100 Gaussian kernel matrix of the dataset clustered in three clusters. Each cluster has samples from both classes. Class identity of a sample is depicted as the graphs below and between the kernel matrix images. For ideal classification purposes, the kernel matrix should reflect the similarity within a class and dissimilarity between classes. This can be seen on the right side of the figure, where the proposed supervised kernel has split the first cluster (top left corner) into the two classes nicely. Splits on the second and third clusters are not that clean but still visible, and much more so than what can be seen in the Gaussian kernel.

[10] J. Friedman. Greedy function approximation: a gradient boosting machine. Technical report, Dept. of Statistics, Stanford University, 1999.

[11] J. Friedman. Stochastic gradient boosting. Technical report, Dept. of Statistics, Stanford University, 1999.

[12] T. Poggio, R. Rifkin, S. Mukherjee, and A. Rakhlin. Bagging regularizes. CBCL Paper 214, Massachusetts Institute of Technology, Cambridge, MA, Feb. 2002. AI Memo #2002-003.

[13] T. Poggio and S. Smale. The mathematics of learning: Dealing with data. *Notices of the American Mathematical Society (AMS)*, 50(5):537–544, 2003.

[14] R. Rifkin. *Everything Old Is New Again: A Fresh Look at Historical Approaches in Machine Learning*. PhD thesis, MIT, 2002.

[15] A. Tikhonov and V. Arsenin. *Solutions of Ill-posed Problems*. W.H.Wingston, Washington, D.C., 1977.

[16] K. Torkkola. Feature extraction by non-parametric mutual information maximization. *Journal of Machine Learning Research*, 3:1415–1438, March 2003.

[17] K. Torkkola and E. Tuv. Ensembles of regularized least squares classifiers for high-dimensional problems. In I. Guyon, S. Gunn, M. Nikravesh, , and L. Zadeh, editors, *Feature Extraction, Foundations and Applications*. Springer, New York, 2004. forthcoming, submitted.

On extracting propositions of nonclassical logics from trained neural networks:A preliminary study

Hiroshi Tsukimoto
Tokyo Denki University
School of Engineering
2-2, Kanda-Nishiki-cho, Chiyoda-ku, Tokyo 101-8457 Japan
tsukimoto@c.dendai.ac.jp

Abstract

Data mining has two requirements, accurate predictions and comprehensible rules. Rule extraction from mathematical formulas such as neural networks or regression formulas is needed for perfect data mining techniques satisfying the above two requirements. Rule extraction from neural networks has been developed. "Rules" in the above sentences basically mean the propositions of classical logic. The propositions of classical logic extracted from trained neural networks only approximately describe the trained neural networks. Trained neural networks contain a lot of information(knowledge) which cannot be described by classical logic. There are several nonclassical logics such as intuitionistic logic or modal logic. The propositions of nonclassical logics can describe trained neural networks in more detail than classical logic. In order to extract propositions of nonclassical logics from trained neural networks, the relations between nonclassical logics and neural networks should be studied. This paper presents a preliminary study towards extracting propositions of nonclassical logics from trained neural networks. This paper shows that trained neural networks can be represented by intuitionistic modal logics. This paper also shows that the relation between a neural network and the rule extracted from the network is represented by an intuitionistic modal logic. The relations between neural networks and nonclassical logics are basically studied based on multilinear function space.

1 Introduction

Data mining has two requirements, accurate predictions and comprehensible rules. The major data mining techniques such as neural networks, statistics, decision trees or association rules,cannot satisfy the above two data mining requirements[17].

Rule extraction from mathematical formulas such as neural networks or regression formulas is needed for perfect data mining techniques. Several researchers have been developing techniques for rule extraction from neural networks[2]. The author also has been developing techniques for rule extraction from mathematical formulas such as neural networks or regression formulas[14],[15],[18]. The technique is called the Approximation Method. Based on the Approximation Method, the author has been developing a data mining technique called Logical Regression Analysis(LRA)[20].

"Rules" in the above sentences basically mean the propositions of classical logic. However, neural networks cannot be represented by the propositions of classical logic. Rules extracted from trained neural networks only approximately represent the trained neural networks. In other words, the neural networks contain a lot of information(knowledge) which cannot be represented by the propositions of classical logics extracted from the trained neural networks.

There are several nonclassical logics such as intuitionistic logic or modal logic[9]. Propositions of nonclassical logics can represent trained neural networks in more detail than classical logic.

For example, let a neural network be trained with time series data. Assume that the trained neural network contains "Event A or event B necessarily happens after event c has happened in some period." This proposition cannot be represented by classical logic, but can be represented by temporal logics, which are modal logics[9].

As the above example showed, we can obtain a lot of linguistic information(knowledge) from trained neural networks by extracting propositions of nonclassical logics.

In order to extract propositions of nonclassical logics from trained neural networks, the relations between nonclassical logics and neural networks should be studied, and it may be necessary to develop new nonclassical logics.

This paper presents a preliminary study towards extract-

ing propositions of nonclassical logics from trained neural networks. As the first stage, this paper studies the relations between nonclassical logics and neural networks.

This paper shows that neural networks can be regarded as the propositions of intuitionistic logic and intuitionistic modal logic[19].

We are interested in the relation between a neural network and the Boolean function(=rule) extracted from the network.

Many Boolean functions can be extracted from a trained neural network. However, the Boolean functions should well approximate the trained neural network. Therefore, the Boolean function which is nearest to the neural network is considered in this paper.

This paper shows that the relation between a neural network and the Boolean function nearest to the network is represented by intuitionistic modal logic$\mathbf{IS5}$[1],[3].

The relations between neural networks and nonclassical logics are basically studied based on multilinear function space.

In the discrete domain, neural networks are multilinear functions. Multilinear function space is the algebraic model of intuitionistic logic. Multilinear function space is the algebraic model of intuitionistic modal logic$\mathbf{IS5}$. Therefore, neural networks can be regarded as the propositions of intuitionistic modal logic$\mathbf{IS5}$. The relation between an neural network and the Boolean function nearest to the neural network is represented by intuitionistic modal logic$\mathbf{IS5}$

Section 2 explains multilinear function space. Section 3 shows that multilinear function space is the algebraic model of intuitionistic logic. Section 4 shows that multilinear function space is the algebraic model of intuitionistic modal logic$\mathbf{IS5}$. Section 5 explains the relation between a neural network and the Boolean function nearest to the neural network. Section 6 shows that the relation between an neural network and the Boolean function nearest to the neural network is represented by intuitionistic modal logic$\mathbf{IS5}$, and shows an example of $voting - records$.

In the continuous domain, similar discussions hold true, which is explained in Section 7.

2 Multilinear function space

2.1 Multilinear functions

Definition 1 *Multilinear functions of n variables are as follows:*

$$\sum_{i=1}^{2^n} p_i x_1^{e_{i1}} \cdots x_n^{e_{in}},$$

where p_i is real, x_i is a variable, and e_i is 0 or 1.

In this paper, n stands for the number of variables.

Example Multilinear functions of 2 variables are as follows:

$$pxy + qx + ry + s.$$

Multilinear functions do not contain any terms such as

$$x_1^{k_1} x_2^{k_2} \cdots x_n^{k_n},$$

where $k_i \geq 2$. A function $f : \{0,1\}^n \to \mathbf{R}$ is a multilinear function, because

$$x_i^{k_i} = x_i$$

holds in $\{0,1\}$ and so there is no term like

$$x_1^{k_1} x_2^{k_2} \cdots x_n^{k_n} (k_i \geq 2)$$

in the functions. In other words, multilinear functions are functions which are linear when only one variable is considered and the other variables are regarded as parameters.

2.2 Neural networks are multilinear functions in the domain $\{0,1\}$

Theorem 2 *When the domain is $\{0,1\}$, neural networks are multilinear functions.*

Proof As described in 2.1, a function whose domain is $\{0,1\}$ is a multilinear function. Therefore, when the domain is $\{0,1\}$, neural networks are multilinear functions.

2.3 Multilinear function space of the domain $\{0,1\}$ is the linear space spanned by the atoms of Boolean algebra of Boolean functions

Definition 3 *The atoms of Boolean algebra of Boolean functions of n variables are as follows:*

$$\phi_i = \prod_{j=1}^{n} e(x_j) \ (i = 1, ..., 2^n), \tag{1}$$

where $e(x_j) = \overline{x_j}$ or x_j.

Example The atoms of Boolean algebra of Boolean functions of 2 variables are as follows:
$x \wedge y$, $x \wedge \overline{y}$, $\overline{x} \wedge y$, $\overline{x} \wedge \overline{y}$.

Theorem 4 *The space of multilinear functions ($\{0,1\}^n \to \mathbf{R}$) is the linear space spanned by the atoms of Boolean algebra of Boolean functions.*

Proof Any Boolean function can be represented as the linear combination of the atoms, that is,

$$\sum_{i=1}^{2^n} c_i \phi_i, \tag{2}$$

where ϕ_i in formula (2) is an atom, c_i is 0 or 1, and $\sum$ means logical disjunction.

Let logical conjunction, logical disjunction and negation be represented by elementary algebra. In the domain $\{0,1\}$, logical conjunction $x \wedge y$ equals xy, which is the product of elementary algebra, and negation $\bar{x}$ equals $1-x$ of elementary algebra. Logical disjunction is calculated using de Morgan's law.

Table 1 shows the elementary algebra representations of logical operations. In the table, l.o. stands for logical operation, and e.a.r. stands for elementary algebra representation.

Table 1. Elementary algebra representations of logical operations

	conjunction	disjunction	negation
l.o.	$x \wedge y$	$x \vee y$	$\bar{x}$
e. a. r.	xy	$x+y-xy$	$1-x$

Table 2. Elementary algebra representations of logical operations of atoms

	conjunction	disjunction	negation
l.o.	$\phi_i \wedge \phi_j$	$\phi_i \vee \phi_j$	$\overline{\phi_i}$
e.a.r.	$\phi_i\phi_j$	$\phi_i+\phi_j$	$1-\phi_i$

Table 2 shows the elementary algebra representations of logical operations of atoms. The representation of formula (2) by elementary algebra is the same as formula (2) when $\prod$ in formula (1) is interpreted as the product of elementary algebra, $\sum$ is interpreted as the sum of elementary algebra, and $\bar{\phi}$ is interpreted as $1-\phi$.

By extending the coefficients c_i in formula (2) from $\{0,1\}$ to real, the functions become real linear functions as follows:

$$\sum_{i=1}^{2^n} a_i\phi_i, \tag{3}$$

where a_i is real and $\sum$ means the sum of elementary algebra.

A function in formula (3) is the multilinear function (of variables), because a function in formula (3), that is, the linear function of the atoms of Boolean algebra of Boolean functions can be developed to a multilinear function, and a multilinear function can be expanded by the atoms uniquely.

Example A linear function of the atoms of 2 variables is

$$axy + bx\overline{y} + c\overline{x}y + d\overline{xy}.$$

This function is transformed to the following:

$$pxy + qx + ry + s,$$

where

$p = a-b-c+d, \quad q = b-d, \quad r = c-d, \quad s = d.$

A multilinear function

$$pxy + qx + ry + s$$

can be transformed into

$$axy + bx\overline{y} + c\overline{x}y + d\overline{xy},$$

where

$a = p+q+r+s, \quad b = q+s, \quad c = r+s, \quad d = s.$

Now, it has been shown that the multilinear function space of the domain $\{0,1\}$ is the linear space spanned by the atoms of Boolean algebra of Boolean functions. The dimension of the space is 2^n.

Multilinear function space is made into a Euclidean space, which is easily verified[11].

2.4 Vector representations of logical operations

The vector representations of logical functions are called logical vectors. $\mathbf{f}((f_i)), \mathbf{g}((g_i)), ..$ stand for logical vectors. Note that f stands for a function, while f_i stands for a component of a logical vector $\mathbf{f}$.

Example

$$0.6xy + 0.1x + 0.1y + 0.1$$

is transformed to

$$0.9xy + 0.2x\overline{y} + 0.2\overline{x}y + 0.1\overline{xy}.$$

Therefore, the logical vector is

$$(0.9, 0.2, 0.2, 0.1).$$

Vector representations of logical operations are as follows:

$$\mathbf{f} \wedge \mathbf{g} = (f_i g_i),$$

$$\mathbf{f} \vee \mathbf{g} = (f_i + g_i - f_i g_i),$$

$$\overline{\mathbf{f}} = (1 - f_i).$$

When multilinear functions are Boolean functions, the above vector representations of logical operations are the same as the representations below.

$$\mathbf{f} \wedge \mathbf{g} = (Min(f_i, g_i)),$$

$$\mathbf{f} \vee \mathbf{g} = (Max(f_i, g_i)),$$

$$\overline{\mathbf{f}} = (1 - f_i).$$

The above representations appear in intermediate logic LC [5] in Section 3.

Example Let f be $x \vee y$ and let g be $\bar{x} \wedge \bar{y}$. The logical conjunction of f and g is as follows:

$$(x \vee y) \wedge (\bar{x} \wedge \bar{y}) = (x \wedge \bar{x} \wedge \bar{y}) \vee (y \wedge \bar{x} \wedge \bar{y}) = 0$$

The logical vectors of f and g, that is, **f** and g are as follows:

$$\mathbf{f} = (1,1,1,0), \quad \mathrm{g} = (0,0,0,1),$$

where the bases are

$$x \wedge y = (1,0,0,0), x \wedge \bar{y} = (0,1,0,0),$$
$$\bar{x} \wedge y = (0,0,1,0), \bar{x} \wedge \bar{y} = (0,0,0,1).$$

The logical conjunction of **f** and **g** is as follows using the above definition.

$$\begin{aligned} \mathbf{f} \wedge \mathrm{g} &= (Min(f_i, g_i)) \\ &= (Min(1,0), Min(1,0), Min(1,0), Min(0,1)) \\ &= (0,0,0,0) \\ &= \mathbf{0} \end{aligned}$$

3 Multilinear function space is a model of nonclassical logics

3.1 Multilinear function space is a model of intuitionistic logic

Heyting algebra, which is the algebraic model of intuitionistic logic, is defined as follows[4].

Definition 5 *A Heyting algebra is a distributive lattice with respect to $\wedge, \vee$ and with $\top$ and $\bot$, and satisfies the following formulas.*

$$\begin{aligned} f \wedge (f \supset g) &= f \wedge g, \\ (f \supset g) \wedge g &= g, \\ (f \supset g) \wedge (f \supset h) &= f \supset (g \wedge h), \\ \bot \wedge f &= \bot, \\ \bot \supset \bot &= \top. \end{aligned}$$

Complement f' is defined by

$$f' = f \supset \bot.$$

Theorem 6 $< [0,1], \wedge, \vee, \top, \bot >$ *is a Heyting algebra with the following definitions:*

$$\begin{aligned} x \wedge y &= Min(x,y), \\ x \vee y &= Max(x,y), \\ x \supset y &= \begin{cases} 1 (x \leq y) \\ y (x > y), \end{cases} \\ \top &= 1, \\ \bot &= 0. \end{aligned}$$

Note that x and y are used instead of f and g, because x and y stand for real numbers.

It can be easily verified that the above definitions satisfy Definition 5.

Theorem 7 *A subset of $[0,1]^m$ (m is dimension) of multilinear function space is an algebraic model of intuitionistic logic with the following definitions.*

$$\begin{aligned} \mathbf{f} \wedge \mathrm{g} &:= (\min(f_i, g_i)) \\ \mathbf{f} \vee \mathrm{g} &:= (\max(f_i, g_i)) \\ \mathbf{f} \supset \mathrm{g} &:= (f_i \supset g_i) \\ f_i \supset g_i &:= \begin{cases} 1 & (f_i \leq g_i) \\ g_i & (f_i > g_i) \end{cases} \end{aligned}$$

Proof

The multilinear function space is a linear space spanned by the atoms of Boolean algebra of Boolean functions, and therefore, a subset of the space $[0,1]^m$ (m is dimension) is a direct product of the interval[0,1]. If an interval is a model of a logic, the direct product of the intervals is also a model of the logic[8]. Therefore, since an interval[0,1] is an algebraic model of intuitionistic logic, a direct product of intervals$[0,1]^m$ is also an algebraic model of intuitionistic logic. Therefore, the subset of the multilinear function space $[0,1]^m$ is an algebraic model of intuitionistic logic.

3.2 Nonclassical logics complete for multilinear function space

Intuitionistic logic is not complete for the interval $[0,1]$ as explained later. This subsection briefly explains the logics complete for the interval $[0,1]$.

The logics which are complete for the interval $[0,1]$ are also complete for the direct product of the interval $[0,1]$, that is, $[0,1]^m$[8]. The logics are continuously valued logics. There are three logics which are complete for the interval, that is, an intermediate logic LC (LC for short), Łukasiewicz logic and product logic.

Logical conjunctions and logical implications are defined as follows [7].

1.LC

$conjunction : x \wedge y = \min(x,y)$

$$implication : x \to y = \begin{cases} 1 & x \leq y \\ y & \text{otherwise} \end{cases}$$

2. Łukasiewicz logic

$conjunction : x \wedge y = \max(0, x+y-1)$

$implication : x \to y = \min(1, 1-x+y)$

3. product logic

$conjunction : x \wedge y = xy$

$$implication : x \to y = \begin{cases} 1 & x \leq y \\ y/x & \text{otherwise} \end{cases}$$

In sequent calculus, there are three structure rules as follows:

$$\text{contraction}\ \frac{x, x \to y}{x \to y},$$

$$\text{weakening}\ \frac{x \to y}{x, z \to y},$$

$$\text{exchange}\ \frac{x, y \to z}{y, x \to z}.$$

Logics which do not have some of the above rules are called substructural logics. In terms of contraction, LC satisfies contraction, Łukasiewicz logic and product logic do not satisfy contraction[10].

Due to space limitations, all three logics cannot be explained. Only LC is briefly explained. Similar explanations hold for Łukasiewicz logic and product logic[7].

Intermediate logics are weaker than classical logic and stronger than intuitionistic logic[4]. LC is an intermediate logic, which was presented by Dummett[5]. The logic is defined as follows[4].

$$\text{LC=intuitionistic logic} + (\varphi \to \psi) \vee (\psi \to \varphi),$$

where φ and ψ are logical formulas. LC stands for Logic of Chain, which comes from the fact that the model of the logic is a chain, that is, a linearly ordered set.

Intuitionistic logic is not complete for the model, that is, intuitionistic logic cannot prove the relation below

$$(\varphi \to \psi) \vee (\psi \to \varphi), \tag{4}$$

which corresponds to the following formula which holds in any interval:

$$(x \leq y) \vee (y \leq x),$$

where x and y are points in the interval, while LC has the axiom (4) and so LC is complete for the interval.

LC, Łukasiewicz logic or product logic is complete for [0,1], while intuitionistic logic is not complete for [0,1]. Therefore, in terms of detailed description, LC, Łukasiewicz logic or product logic is better than intuitionistic logic. However, the modal extension of LC, Łukasiewicz logic or product logic has not been developed. Therefore, in this paper, intuitionistic modal logics are discussed hereinafter.

4 Multilinear function space is a model of intuitionistic modal logic

Before the explanation of intuitionistic modal logic $\mathbf{IS5}$, classical modal logic $\mathbf{S5}$ is briefly explained.

4.1 Classical modal logic $\mathbf{S5}$

Modal logics deal with modalities such as necessity, possibility, belief, knowledge, or tense. Necessity is denoted by I, and possibility is denoted by C in this paper.

Modal logic $\mathbf{K}$ is the logic that has the following inference rule in addition to the axioms of classical propositional logic.

$$\frac{A \to B}{\mathrm{I}A \to \mathrm{I}B}$$

Modal logic $\mathbf{S5}$ has the followings:

$$\mathrm{I}A \to A,$$

$$\mathrm{C}A \to \mathrm{IC}A.$$

There are several models of modal logics such as Kripke models or algebraic models. Modal algebras are the algebraic models of modal logics.

$\mathbf{S5}$ algebra is the algebraic model of $\mathbf{S5}$. When a unitary operator I on Boolean algebra satisfies the following items, then it is $\mathbf{S5}$ algebra:

Definition 8

$$I(f \wedge g) = If \wedge Ig$$

$$I\top = \top$$

$$Cf = (If')'(\textit{Definition of} C)$$

$$C(f \vee g) = Cf \vee Cg$$

$$If \leq f$$

$$Cf \leq ICf$$

4.2 $\mathbf{IS5}$ algebra

This subsection explains $\mathbf{IS5}$ algebra, which is the algebraic model of intuitionistic modal logic $\mathbf{IS5}$.

Definition 9 *When a unitary operator I on Heyting algebra satisfies the following items, then it is intuitionistic $\mathbf{S5}$ ($\mathbf{IS5}$) algebra[1], [3]:*

$$\begin{aligned} I(f \wedge g) &= If \wedge Ig, \\ I\top &= \top, \\ C(f \vee g) &= Cf \vee Cg \\ C\bot &= \bot \\ If &\leq f, \\ Cf &\leq ICf. \end{aligned}$$

While C is defined as in classical modal logics, C is not necessarily defined as $Cf = (If')'$ in intuitionistic modal logics. For example, in [6], I and C are required to be non-interdefinable[1].

I means necessity and C means possibility.

Definition 10 *I is defined as follows:*

$$If := (I(f_i)),$$

$$I(f_i) = \begin{cases} f_i (f_i \geq a) \\ 0 (f_i < a), \end{cases}$$

where $0 < a < 1$.

Cf is defined as follows:

$$C(f_i) = \begin{cases} 1 (f_i > 0) \\ 0 (f_i = 0). \end{cases}$$

This definition satisfies $Cf = (If')'$, although C is not necessarily defined as $Cf = (If')'$ in intuitionistic modal logics. If C is defined as

$$C(f_i) = \begin{cases} 1 (f_i \geq a) \\ f_i (f_i < a), \end{cases}$$

where $0 < a < 1$, $Cf = (If')'$ is not satisfied. The latter definition does not satisfy $Cf \leq ICf$, but satisfies $If \leq IIf$. That is, if C is defined in the latter manner, it is another modal logic.

From the above definitions, $If \leq f$, that is, the information of If is greater than the information of f, and so it is reasonable that If represents 'f is necessary'. $f \leq Cf$, that is, the information of Cf is less than the information of f, and so it is reasonable that Cf represents 'f is possible'.

Theorem 11 *Hyper-rectangle $\prod_{i=1}^{m}[0,1]$ of the space of multilinear functions with Definition 10 satisfies Definition 9. This can be easily verified.*

From the above theorem, any hyper-rectangle $\prod_{i=1}^{m}[0,1]$ of the space of multilinear functions with Definition 10 is the model of $\mathbf{IS5}$.

5 The relation between a neural network and the Boolean function nearest to the neural network

Many Boolean functions can be extracted from a trained neural network. However, the Boolean functions should well approximate the trained neural network. Therefore, the Boolean function which is nearest to the neural network is considered hereinafter.

This section deals with a unit of a neural network. The discussion in this section is easily expanded to a neural network with hidden units, which is explained later.

Let $f(x_1, ..., x_n)$ stand for a unit of a neural network, and $(f_i)(i = 1, ..., 2^n)$ be the values of the unit. Let the values of the unit be the interval [0,1], because the values of a unit of a neural network are [0,1].

Let $g(x_1, ..., x_n)$ stand for a Boolean function, and $(g_i)(g_i = 0 \text{ or } 1, i = 1, ..., 2^n)$ be the values of the Boolean function.

The Boolean function nearest to the unit of a neural network is as follows:

$$g_i = \begin{cases} 1 (f_i \geq 0.5), \\ 0 (f_i < 0.5). \end{cases}$$

This Boolean function is the nearest in Euclidean distance. The Boolean function is represented as follows:

$$g(x_1, ..., x_n) = \sum_{i=1}^{2^n} g_i \phi_i,$$

where g_i is calculated by the above formula, and ϕ_i is an atom.

The rule extraction method based on the above approximation is called the Approximation Method[15],[18],[20].

Example

Let a unit of a neural network be as follows:

$$f(x, y) = S(2.51x - 4.80y - 0.83),$$

where $S(\cdot)$ stands for sigmoid function. The values of $f(0,0), f(0,1), f(1,0)$, and $f(1,1)$ are as follows:
$f(0,0) = S(2.51 \cdot 0 - 4.80 \cdot 0 - 0.83) = S(-0.83)$
$f(0,1) = S(2.51 \cdot 0 - 4.80 \cdot 1 - 0.83) = S(-5.63)$
$f(1,0) = S(2.51 \cdot 1 - 4.80 \cdot 0 - 0.83) = S(1.68)$
$f(1,1) = S(2.51 \cdot 1 - 4.80 \cdot 1 - 0.83) = S(-3.12).$

$S(-0.83) \simeq 0, S(-5.63) \simeq 0, S(1.68) \simeq 1$, and $S(-3.12) \simeq 0$. Therefore, the values of the nearest Boolean function $g(x, y)$ are as follows:

$$g(0,0) = 0, g(0,1) = 0, g(1,0) = 1, g(1,1) = 0.$$

The Boolean function $g(x, y)$ is as follows:

$$\begin{aligned} g(x,y) &= g(0,0)\bar{x}\bar{y} \vee g(0,1)\bar{x}y \vee g(1,0)x\bar{y} \vee g(1,1)xy \\ g(x,y) &= 0\bar{x}\bar{y} \vee 0\bar{x}y \vee 1x\bar{y} \vee 0xy \\ g(x,y) &= x\bar{y}. \end{aligned}$$

The relation between a unit of neural network and the Boolean function nearest to the unit of the neural network holds true in the case of the relation between a neural network and the Boolean function nearest to the neural network, which is obvious.

Table 3. Proof

f_i	≤ 0.5	$0.5 \leq$
If_i	0	f_i
CIf_i	0	1

6 The relation between a neural network and the Boolean function nearest to the neural network is represented by intuitionistic modal logic IS5

6.1 The relation between a neural network and the Boolean function nearest to the neural network

As explained previously, a multilinear function $f(=(f_i))$ is approximated to a Boolean function $g(=(g_i))(g_i = 0 \text{ or } 1)$ by the following method:

$$g_i = \begin{cases} 1(f_i \geq 0.5), \\ 0(f_i < 0.5). \end{cases}$$

Theorem 12

$$g = CIf$$

Proof The above theorem can be easily verified by Definition 10, where $a = 0.5$. See Table 3.

Therefore, the relationship between a multilinear function and the nearest Boolean function can be represented as intuitionistic modal logic **IS5**.

Example

As explained in Section 5,

$$x\bar{y}$$

is the nearest Boolean function to

$$f(x, y) = S(2.51x - 4.80y - 0.83).$$

Therefore,

$$x\bar{y} = CIf,$$

which represents '$x\bar{y}$ means that it is possible that f is necessary.'

6.2 An example of a neural network

Let N_1 and N_2 be two trained neural networks, which have hidden layers, two inputs x and y, two hidden units, and one output. See Fig.1. The output function of each unit is sigmoid function. Table 4 shows the training results of weight parameters and biases of N_1. Table 5 shows the training results of weight parameters and biases of N_2.

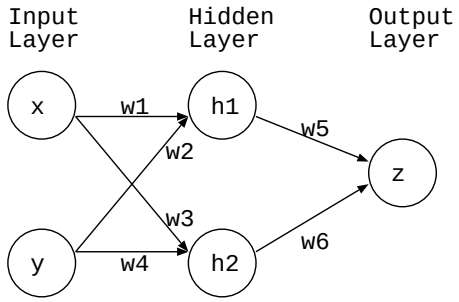


Figure 1. Neural network

Table 4. Training results 1

unit	w1(w3, w5)	w2(w4,w6)	bias
hidden unit 1	-4.87	-4.86	-6.70
hidden unit 2	-2.86	-2.88	3.50
output unit	7.61	-13.83	4.50

N_1 is as follows:

$S(7.61S(-4.87x - 4.86y - 6.70) - 13.83S(-2.86x - 2.88y + 3.50) + 4.50)$.

From the above formula, the logical vector is calculated as follows:

$$(0.98, 0.01, 0.01, 0.00),$$

where the coordinate system is

$$xy, x\bar{y}, \bar{x}y, \bar{x}\bar{y}$$

The vector representation of nearest Boolean function is

$$(1, 0, 0, 0).$$

The Boolean function is

$$xy$$

Therefore,

$$xy = CIN_1$$

The logical vector of N_2 is calculated in the same way as follows:

$$(0.02, 0.98, 0.98, 0.99).$$

The nearest Boolean function is

$$\bar{y} \vee \bar{x}y \vee \bar{x}\bar{y} = \bar{x} \vee \bar{y}.$$

Table 5. Training results 2

unit	w1(w3, w5)	w2(w4,w6)	bias
hidden unit 1	4.80	4.72	-2.31
hidden unit 2	-3.49	-3.56	1.67
output unit	5.81	-4.62	-0.42

Therefore,

$$\bar{x} \vee \bar{y} = CIN_2$$

As showed above, trained neural networks can be represented by nonclassical logics.

The interpretations of I(necessity) and C(possibility) depend on the data.

6.3 An example of *voting-records* data

voting-records data consists of the voting records of the U.S. House of Representatives in 1984. Attributes are the following policies and the attribute values are yes and no.

a; handicapped-infants: y,n
b; water-project-cost-sharing: y,n
c; adoption-of-the-budget-resolution: y,n
d; physician-fee-freeze: y,n
e; el-salvador-aid: y,n
f; religious-groups-in-schools: y,n
g; anti-satellite-test-ban: y,n
h; aid-to-nicaraguan-contras: y,n
i; mx-missile: y,n
j; immigration: y,n
k; synfuels-corporation-cutback: y,n
l; education-spending: y,n
m; superfund-right-to-sue: y,n
n; crime: y,n
o; duty-free-exports: y,n
p; export-administration-act-south-africa: y,n

Classes are Democrat and Republican. The number of samples is 435.

The neural network for learning $voting - records$ data consists of 16 inputs, 3 hidden units and 2 output units. The learning method is the back-propagation method, the repetition time is 2500, and the error after the learning is 0.005.

As explained in the preceding subsection, the whole network can be described using intuitionistic modal logic$\mathbf{IS5}$. Here, another method is explained. First, each unit is described by using intuitionistic modal logic$\mathbf{IS5}$, and second, the descriptions are merged.

The Boolean functions nearest to the hidden units are as follows: t_i stands for the output of a hidden unit. $a, ..., p$ stand for inputs, for example, $a = 1$ when handicapped-infants is y and $a = 0$ when handicapped-infants is n.

hidden unit 1:0
hidden unit 2:$\bar{c}d\bar{k}$
hidden unit 3: $dj(\bar{i} \vee \bar{k})$
output 1 (Republican):$t_2 \vee t_3$
output 2 (Democrat):$\bar{t_2}\bar{t_3}$

By substituting the results of hidden units to output units, the followings are obtained:

$$\text{Republican:} d(\bar{c}\bar{k} \vee \bar{i}j \vee j\bar{k}),$$

$$\text{Democrat:} \bar{d} \vee c\bar{j} \vee ik \vee \bar{j}k.$$

The accuracy of the result is 94.4%.

The above results are Boolean functions approximating the trained neural network. The relation between a unit of a neural network and the Boolean function nearest to the unit of the network is represented by intuitionistic modal logic$\mathbf{IS5}$ as explained previously. For example, hidden unit 2 is described as follows:

$$\bar{c}d\bar{k} = \mathrm{CI}t_2.$$

The other units are described in the same manner.

Hidden unit 2 is described as

$$\bar{c}d\bar{k} = \mathrm{CI}t_2,$$

and hidden unit 3 is described as

$$dj(\bar{i} \vee \bar{k}) = \mathrm{CI}t_3.$$

Therefore,

By the disjunction,

$$\bar{c}d\bar{k} \vee dj(\bar{i} \vee \bar{k}) = \mathrm{CI}t_2 \vee \mathrm{CI}t_3 \qquad (5).$$

Output unit 1(o_1)(Republican) is described as

$$t_2 \vee t_3 = \mathrm{CI}o_1,$$

and

$$\mathrm{CI}(t_2 \vee t_3) = \mathrm{CICI}o_1.$$

Since $\mathrm{I}(f \vee g) = \mathrm{I}f \vee \mathrm{I}g,$

$$\mathrm{C}(\mathrm{I}t_2 \vee \mathrm{I}t_3) = \mathrm{CICI}o_1,$$

and

$$\mathrm{CI}t_2 \vee \mathrm{CI}t_3 = \mathrm{CICI}o_1.$$

Moreover, from formula (5)

$$\bar{c}d\bar{k} \vee dj(\bar{i} \vee \bar{k}) = \mathrm{CICI}o_1.$$

I stands for necessity and C stands for possibility, and so the above formula can be interpreted with a modality. However, the interpretation is not so interesting. Only two modal operators cannot present interesting interpretations. In order to obtain interesting results, many modal operators and many modal logics should be studied.

7 In the case of continuous domains

So far, discrete domains have been discussed. This section briefly explains continuous domains. The continuous domain can be reduced to [0,1] domain by a certain normalization, and so only [0,1] domain is discussed. The multilinear function space of [0,1] domain can be made into a Euclidean space, and so the similar discussions hold true. Details can be found in [11] and [18].

7.1 τ

Definition 13 *τ_x is defined as follows:*

Let $f(x)$ be a real polynomial function. Consider the following formula:
$f(x) = p(x)(x - x^2) + q(x)$,
where $q(x) = ax + b$, where a and b are real, that is, $q(x)$ is the remainder.

τ_x is defined as follows:
$\tau_x : f(x) \to q(x)$.

The above definition implies the following property:
$\tau_x(x^k) = x$,
where $k \geq 2$.

Definition 14 *In the case of n variables,*

τ is defined as follows:

$$\tau = \prod_{i=1}^{n} \tau_{x_i}.$$

For example,

$$\tau(x^2y^3 + y + 1) = xy + y + 1.$$

7.2 The multilinear function space in the domain [0,1] is a Euclidean space

Definition 15 *An inner product in the case of n variables is defined as follows:*

$$< f, g >= 2^n \int_0^1 \tau(fg)dx,$$

where f and g are multilinear functions. The above definition can satisfy the properties of inner product[11],[12],[16].

Definition 16 *Norm $|f|$ is defined as follows:*

$$|f| = \sqrt{< f, f >}.$$

The distance between functions is roughly measured by the norm. For example, x^2 is different from x. However, by the norm, the distance between the two functions is 0, because τ in the norm

$$\sqrt{< f, g >} = \sqrt{2^n \int_0^1 \tau(fg)dx}$$

identifies $x^k (k \geq 2)$ with x. Therefore, the two functions are identified as being the same one in the norm. The norm can be regarded as a qualitative norm, because, roughly speaking, the norm identifies increasing functions as direct proportions and identifies decreasing functions as inverse proportions, and the norm ignores the function values in the intermediate domain between 0 and 1.

Theorem 17 *The multilinear function space in the domain [0,1] is a Euclidean space with the above inner product: Proof can be found in [11], [12] and [16].*

The orthonormal system is as follows:

$$\phi_i = \prod_{j=1}^{n} e(x_j) \ (i = 1, ..., 2^n),$$

where $e(x_j) = 1 - x_j$ or x_j. It is easily understood that these orthonormal functions are the expansion of atoms in Boolean algebra of Boolean functions. In addition, it can easily be verified that the orthonormal system satisfies the following properties:

$$< \phi_i, \phi_j >= \begin{cases} 0 (i \neq j), \\ 1 (i = j), \end{cases}$$

$$f = \sum_{i=1}^{2^n} < f, \phi_i > \phi_i.$$

Example In the case of 2 variables, the orthonormal functions are as follows:

$$xy, x(1-y), (1-x)y, (1-x)(1-y).$$

and the representation by orthonormal functions of $x + y - xy$ of two variables (dimension 4) is as follows:

$$f = 1 \cdot xy + 1 \cdot x(1-y) + 1 \cdot (1-x)y + 0 \cdot (1-x)(1-y).$$

The functions obtained by extending the domain of Boolean functions from $\{0, 1\}$ to [0,1], is called continuous Boolean functions, because the functions can satisfy the axioms of Boolean algebra[13].

7.3 The relation between neural networks and multilinear functions

When the domain is [0,1], neural networks are well approximated to multilinear functions with the following:

$$x^k = \begin{cases} x (k \leq a) \\ 0 (k > a), \end{cases}$$

where a is a natural number. When $a = 1$, the above approximation is the linear approximation. The approximation accuracy in [0,1] domain is better than in other domains, because

1. $x^k (k > 1)$ is monotone increasing in [0,1] domain.
2. When k is small, $|x^k - x|$ is less than that in other domains.
3. When x^k is big, x^k is almost 0 in $[0, 1 - \epsilon]$ (ϵ is a small positive number.)

8 Conclusions

Extracting rules(=propositions of classical logic) from trained neural networks is effective for understanding the neural networks. However, the extracted rules approximate the neural networks, and cannot describe a lot of linguistic knowledge contained in the neural networks. The neural networks can be described in detail using the propositions of nonclassical logics.

In order to extract propositions of nonclassical logics from trained neural networks, the relations between nonclassical logics and neural networks should be studied, and it may be necessary to develop new nonclassical logics.

As a preliminary study towards extracting propositions of nonclassical logics from trained neural networks, this paper has studied the relations between nonclassical logics and neural networks.

This paper has explained that neural networks can be regarded as the propositions of intuitionistic logic and intuitionistic modal logic, and the relation between a neural network and the Boolean function nearest to the neural network is represented by intuitionistic modal logic.

There are a lot of open problems on extracting propositions of nonclassical logics from trained neural networks. Therefore, the author hopes that researchers will tackle this field.

References

[1] Amati,G. and Pirri,F.: A Uniform Tableau Method for Intuitionistic Modal Logics I. *Studia Logica* 53:29–60,1994.

[2] Andrews,R., Diederich,J. and Tickle,A.B.: Survey and critique of techniques for extracting rules from trained artificial neural networks, *Knowledge-Based Systems*, Vol.8, No. 6, pp.373-189, 1995.

[3] Bull,R.: A Modal Extension of Intuitionistic Logic, *Notre Dame Journal of Formal Logic*:142–146,1965.

[4] Dalen,D.V.: Intuitionistic Logic, *Handbook of Philosophical Logic III, D. Gabbay and F.Guenthner eds.*, pp.225-339, D.Reidel, 1984.

[5] Dummett,M.: A Propositional Calculus with Denumerable Matrix, *The Journal of Symbolic Logic*, Vol.24, No.2, pp.97-106, 1959.

[6] Fischer Servi, G.: Axiomatizations for Some Intuitionistic Logics. *Rend. Sem. Mat. Univers. Polit* 42:179–194,1994.

[7] H$\acute{a}$jek,P.: *Metamathematics of Fuzzy Logic*, Kluwer, 1998.

[8] Hosoi,T. and H. Ono,H.:Intermediate Propositional Logics(A Survey),*Journal of Tsuda College*, Vol.5, pp.67-82, 1973.

[9] Jacquette,D. (eds.):*A Companion to Philosophical Logic*, Blackwell,2002.

[10] Ono,H. and Komori,Y.: Logics without the contraction rule, *J. Symbolic Logic* 50, pp.169-201, 1985.

[11] Tsukimoto,H. and Morita,C.: The Discovery of Propositions in Noisy Data, *Machine Intelligence 13*, pp.143-167, Oxford University Press, 1994.

[12] Tsukimoto,H.: The discovery of logical propositions in numerical data, *AAAI'94 Workshop on Knowledge Discovery in Databases*, pp.205-216, 1994.

[13] Tsukimoto,H.: On continuously valued logical functions satisfying all axioms of classical logic, *Systems and Computers in Japan*,Vol.25, No.12, pp.33-41, SCRIPTA TECHNICA, INC., 1994.

[14] Tsukimoto,H. and Morita,C.: Efficient algorithms for inductive learning-An application of multi-linear functions to inductive learning, *Machine Intelligence 14*, pp.427-449, Oxford University Press, 1995.

[15] Tsukimoto,H.:Extracting Propositions from Trained Neural Networks. *Proceedings of The 15th International Joint Conference on Artificial Intelligence*, pp.1098-1105, 1997.

[16] Tsukimoto,H.:Symbol pattern integration using multilinear functions, in *Deep Fusion of Computational and Symbolic Processing*, (eds.) Furuhashi,T., Tano,.S., and Jacobsen,H.A. Springer Verlag, pp.41-70,1999.

[17] Tsukimoto,H.: Rule extraction from prediction models, Invited to *The Third Pacific-Asia International Conference on Knowledge Discovery and Data Mining (PAKDD'99)*, pp.34-43, 1999.

[18] Tsukimoto,H: Extracting Rules from Trained Neural Networks, *IEEE Transactions on Neural Networks*, Vo.11, No.2,pp.377-389, 2000.

[19] Tsukimoto,H. and Morita,C.: *Connectionism as Symbolicism:Artificial Neural Networks as Symbols*, Sanshusha, 2001.

[20] Tsukimoto,H.: Logical Regression Analysis:From mathematical formulas to linguistic rules, Invited to *IEEE ICDM02 Workshop Proceedings, The Foundation of Data Mining and Knowledge Discovery(FDM02)*, pp.59-79,2002.

On the Characteristics of Linear Independence in a Contingency Table – Pseudo Statistical Independence –

Shusaku Tsumoto and Shoji Hirano
Department of Medical Informatics,
Shimane University, School of Medicine
89-1 Enya-cho Izumo 693-8501 Japan
E-mail: tsumoto@computer.org

Abstract

A contingency table summarizes the conditional frequencies of two attributes and shows how these two attributes are dependent on each other with the information on a partition of universe generated by these attributes. Thus, this table can be viewed as a relation between two attributes with respect to information granularity. This paper focuses on several characteristics of linear and statistical independence in a contingency table from the viewpoint of granular computing, which shows that statistical independence in a contingency table is a special form of linear dependence. The discussions also show that when a contingency table is viewed as a matrix, called a contingency matrix, its rank is equal to 1.0. Thus, the degree of independence, rank plays a very important role in extracting a probabilistic model from a given contingency table. Furthermore, it is found that in some cases, partial rows or columns will satisfy the condition of statistical independence, which can be viewed as a solving process of Diophatine equations.

1. Introduction

Statistical independence between two attributes is a very important concept in data mining and statistics. The definition $P(A, B) = P(A)P(B)$ show that the joint probability of A and B is the product of both probabilities. This gives several useful formula, such as $P(A|B) = P(A)$, $P(B|A) = P(B)$. In a data mining context, these formulae show that these two attributes may not be correlated with each other. Thus, when A or B is a classification target, the other attribute may not play an important role in its classification.

Although independence is a very important concept, it has not been fully and formally investigated as a relation between two attributes.

In this paper, a statistical independence in a contingency table is focused on from the viewpoint of granular computing.

The first important observation is that a contingency table compares two attributes with respect to information granularity. It is shown from the definition that statistifcal independence in a contingency table is a special form of linear depedence of two attributes. Especially, when the table is viewed as a matrix, the above discussion shows that the rank of the matrix is equal to 1.0. Also, the results also show that partial statistical independence can be observed.

The second important observation is that matrix algebra is a key point of analysis of this table. A contingency table can be viewed as a matrix and several operations and ideas of matrix theory are introduced into the analysis of the contingency table.

The paper is organized as follows: Section 2 discusses the characteristics of contingency tables. Section 3 shows the conditions on statistical independence for a 2×2 table. Section 4 gives those for a $2 \times n$ table. Section 5 extends these results into a multi-way contingency table. Section 6 discusses statistical independence from matrix theory. Section 7 and 8 show pseudo statistical independence. Finally, Section 9 concludes this paper.

2. Contingency Table from Rough Sets

2.1. Rough Sets Notations

In the subsequent sections, the following notations is adopted, which is introduced in [2]. Let U denote a nonempty, finite set called the universe and A denote a nonempty, finite set of attributes, i.e., $a : U \rightarrow V_a$ for $a \in A$, where V_a is called the domain of a, respectively. Then, a decision table is defined as an information system, $A = (U, A \cup \{\mathcal{D}\})$, where $\{\mathcal{D}\}$ is a set of given decision attributes. The atomic formulas over $B \subseteq A \cup \{\mathcal{D}\}$ and V are expressions of the form $[a = v]$, called descrip-

tors over B, where $a \in B$ and $v \in V_a$. The set $F(B,V)$ of formulas over B is the least set containing all atomic formulas over B and closed with respect to disjunction, conjunction and negation. For each $f \in F(B,V)$, f_A denote the meaning of f in A, i.e., the set of all objects in U with property f, defined inductively as follows.

1. If f is of the form $[a = v]$ then, $f_A = \{s \in U | a(s) = v\}$
2. $(f \wedge g)_A = f_A \cap g_A$; $(f \vee g)_A = f_A \vee g_A$; $(\neg f)_A = U - f_a$

By using this framework, classification accuracy and coverage, or true positive rate is defined as follows.

Definition 1
Let R and D denote a formula in $F(B,V)$ and a set of objects whose decision attribute is given as $\lceil$, respectively. Classification accuracy and coverage(true positive rate) for $R \rightarrow \mathcal{D}$ is defined as:

$$\begin{aligned} \alpha_R(D) &= \frac{|R_A \cap D|}{|R_A|} (= P(D|R)), \text{ and} \\ \kappa_R(D) &= \frac{|R_A \cap D|}{|D|} (= P(R|D)), \end{aligned}$$

where $|A|$ denotes the cardinality of a set A, $\alpha_R(D)$ denotes a classification accuracy of R as to classification of $\mathcal{D}$, and $\kappa_R(D)$ denotes a coverage, or a true positive rate of R to $\mathcal{D}$, respectively.

2.2. Two-way Contingency Table

From the viewpoint of information systems, a contingency table summarizes the relation between two attributes with respect to frequencies. This viewpoint has already been discussed in [3, 4]. However, this study focuses on more statistical interpretation of this table.

Definition 2 *Let R_1 and R_2 denote binary attributes in an attribute space A. A contingency table is a table of a set of the meaning of the following formulas: $|[R_1 = 0]_A|$,$|[R_1 = 1]_A|$, $|[R_2 = 0]_A|$,$|[R_1 = 1]_A|$, $|[R_1 = 0 \wedge R_2 = 0]_A|$,$|[R_1 = 0 \wedge R_2 = 1]_A|$, $|[R_1 = 1 \wedge R_2 = 0]_A|$,$|[R_1 = 1 \wedge R_2 = 1]_A|$, $|[R_1 = 0 \vee R_1 = 1]_A| (= |U|)$. This table is arranged into the form shown in Table 1, where: $|[R_1 = 0]_A| = x_{11} + x_{21} = x_{\cdot 1}$, $|[R_1 = 1]_A| = x_{12} + x_{22} = x_{\cdot 2}$, $|[R_2 = 0]_A| = x_{11} + x_{12} = x_{1\cdot}$, $|[R_2 = 1]_A| = x_{21} + x_{22} = x_{2\cdot}$, $|[R_1 = 0 \wedge R_2 = 0]_A| = x_{11}$, $|[R_1 = 0 \wedge R_2 = 1]_A| = x_{21}$, $|[R_1 = 1 \wedge R_2 = 0]_A| = x_{12}$, $|[R_1 = 1 \wedge R_2 = 1]_A| = x_{22}$, $|[R_1 = 0 \vee R_1 = 1]_A| = x_{\cdot 1} + x_{\cdot 2} = x_{\cdot\cdot} (= |U|)$.*

	$R_1 = 0$	$R_1 = 1$	
$R_2 = 0$	x_{11}	x_{12}	$x_{1\cdot}$
$R_2 = 1$	x_{21}	x_{22}	$x_{2\cdot}$
	$x_{\cdot 1}$	$x_{\cdot 2}$	$x_{\cdot\cdot}$ $(= \|U\| = N)$

Table 1. Two way Contingency Table

From this table, accuracy and coverage for $[R_1 = 0] \rightarrow [R_2 = 0]$ are defined as:

$$\begin{aligned} \alpha_{[R_1=0]}([R_2 = 0]) &= \frac{|[R_1 = 0 \wedge R_2 = 0]_A|}{|[R_1 = 0]_A|} = \frac{x_{11}}{x_{\cdot 1}}, \\ &\textit{and} \\ \kappa_{[R_1=0]}([R_2 = 0]) &= \frac{|[R_1 = 0 \wedge R_2 = 0]_A|}{|[R_2 = 0]_A|} = \frac{x_{11}}{x_{1\cdot}}. \end{aligned}$$

2.3. Multi-way Contingency Table

Two-way contingency table can be extended into a contingency table for multinominal attributes.

Definition 3 *Let R_1 and R_2 denote multinominal attributes in an attribute space A which have m and n values. A contingency tables is a table of a set of the meaning of the following formulas: $|[R_1 = A_j]_A|$, $|[R_2 = B_i]_A|$, $|[R_1 = A_j \wedge R_2 = B_i]_A|$, $|[R_1 = A_1 \wedge R_1 = A_2 \wedge \cdots \wedge R_1 = A_m]_A|$, $|[R_2 = B_1 \wedge R_2 = A_2 \wedge \cdots \wedge R_2 = A_n]_A|$ and $|U|$ $(i = 1, 2, 3, \cdots, n$ and $j = 1, 2, 3, \cdots, m)$. This table is arranged into the form shown in Table 1, where: $|[R_1 = A_j]_A| = \sum_{i=1}^{m} x_{1i} = x_{\cdot j}$, $|[R_2 = B_i]_A| = \sum_{j=1}^{n} x_{ji} = x_{i\cdot}$, $|[R_1 = A_j \wedge R_2 = B_i]_A| = x_{ij}$, $|U| = N = x_{\cdot\cdot}$ $(i = 1, 2, 3, \cdots, n$ and $j = 1, 2, 3, \cdots, m)$.*

	A_1	A_2	$\cdots$	A_n	Sum
B_1	x_{11}	x_{12}	$\cdots$	x_{1n}	$x_{1\cdot}$
B_2	x_{21}	x_{22}	$\cdots$	x_{2n}	$x_{2\cdot}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
B_m	x_{m1}	x_{m2}	$\cdots$	x_{mn}	$x_{m\cdot}$
Sum	$x_{\cdot 1}$	$x_{\cdot 2}$	$\cdots$	$x_{\cdot n}$	$x_{\cdot\cdot} = \|U\| = N$

Table 2. Contingency Table ($m \times n$)

3. Statistical Independence in 2×2 Contingency Table

Let us consider a contingency table shown in Table 1. Statistical independence between R_1 and R_2 gives:

$$\begin{aligned} P([R_1=0],[R_2=0]) &= P([R_1=0]) \times P([R_2=0]) \\ P([R_1=0],[R_2=1]) &= P([R_1=0]) \times P([R_2=1]) \\ P([R_1=1],[R_2=0]) &= P([R_1=1]) \times P([R_2=0]) \\ P([R_1=1],[R_2=1]) &= P([R_1=1]) \times P([R_2=1]) \end{aligned}$$

Since each probability is given as a ratio of each cell to N, the above equations are calculated as:

$$\begin{aligned} \frac{x_{11}}{N} &= \frac{x_{11}+x_{12}}{N} \times \frac{x_{11}+x_{21}}{N} \\ \frac{x_{12}}{N} &= \frac{x_{11}+x_{12}}{N} \times \frac{x_{12}+x_{22}}{N} \\ \frac{x_{21}}{N} &= \frac{x_{21}+x_{22}}{N} \times \frac{x_{11}+x_{21}}{N} \\ \frac{x_{22}}{N} &= \frac{x_{21}+x_{22}}{N} \times \frac{x_{12}+x_{22}}{N} \end{aligned}$$

Since $N = \sum_{i,j} x_{ij}$, the following formula will be obtained from these four formulae.

$$x_{11}x_{22} = x_{12}x_{21} \textit{ or } x_{11}x_{22} - x_{12}x_{21} = 0$$

Thus,

Theorem 1 *If two attributes in a contingency table shown in Table 1 are statistical indepedent, the following equation holds:*

$$x_{11}x_{22} - x_{12}x_{21} = 0 \quad (1)$$

□

It is notable that the above equation corresponds to the fact that the determinant of a matrix corresponding to this table is equal to 0. Also, when these four values are not equal to 0, the equation 1 can be transformed into:

$$\frac{x_{11}}{x_{21}} = \frac{x_{12}}{x_{22}}.$$

Let us assume that the above ratio is equal to $C(constant)$. Then, since $x_{11} = Cx_{21}$ and $x_{12} = Cx_{22}$, the following equation is obtained.

$$\frac{x_{11}+x_{12}}{x_{21}+x_{22}} = \frac{C(x_{21}+x_{22})}{x_{21}+x_{22}} = C = \frac{x_{11}}{x_{21}} = \frac{x_{12}}{x_{22}}. \quad (2)$$

This equation also holds when we extend this discussion into a general case. Before getting into it, let us cosndier a 2×3 contingency table.

4. Statistical Independence in 2×3 Contingency Table

Let us consider a 2×3 contingency table shown in Table 3. Statistical independence between R_1 and R_2 gives:

	$R_1=0$	$R_1=1$	$R_1=2$	
$R_2=0$	x_{11}	x_{12}	x_{13}	$x_{1\cdot}$
$R_2=1$	x_{21}	x_{22}	x_{23}	$x_{2\cdot}$
	$x_{\cdot 1}$	$x_{\cdot 2}$	$x_{\cdot\cdot 3}$ $(=\lvert U\rvert = N)$	$x_{\cdot\cdot}$

Table 3. Contingency Table (2×3)

$$\begin{aligned} P([R_1=0],[R_2=0]) &= P([R_1=0]) \times P([R_2=0]) \\ P([R_1=0],[R_2=1]) &= P([R_1=0]) \times P([R_2=1]) \\ P([R_1=0],[R_2=2]) &= P([R_1=0]) \times P([R_2=2]) \\ P([R_1=1],[R_2=0]) &= P([R_1=1]) \times P([R_2=0]) \\ P([R_1=1],[R_2=1]) &= P([R_1=1]) \times P([R_2=1]) \\ P([R_1=1],[R_2=2]) &= P([R_1=1]) \times P([R_2=2]) \end{aligned}$$

Since each probability is given as a ratio of each cell to N, the above equations are calculated as:

$$\frac{x_{11}}{N} = \frac{x_{11}+x_{12}+x_{13}}{N} \times \frac{x_{11}+x_{21}}{N} \quad (3)$$

$$\frac{x_{12}}{N} = \frac{x_{11}+x_{12}+x_{13}}{N} \times \frac{x_{12}+x_{22}}{N} \quad (4)$$

$$\frac{x_{13}}{N} = \frac{x_{11}+x_{12}+x_{13}}{N} \times \frac{x_{13}+x_{23}}{N} \quad (5)$$

$$\frac{x_{21}}{N} = \frac{x_{21}+x_{22}+x_{23}}{N} \times \frac{x_{11}+x_{21}}{N} \quad (6)$$

$$\frac{x_{22}}{N} = \frac{x_{21}+x_{22}+x_{23}}{N} \times \frac{x_{12}+x_{22}}{N} \quad (7)$$

$$\frac{x_{23}}{N} = \frac{x_{21}+x_{22}+x_{23}}{N} \times \frac{x_{13}+x_{23}}{N} \quad (8)$$

From equation (3) and (6),

$$\frac{x_{11}}{x_{21}} = \frac{x_{11}+x_{12}+x_{13}}{x_{21}+x_{22}+x_{23}}$$

In the same way, the following equation will be obtained:

$$\frac{x_{11}}{x_{21}} = \frac{x_{12}}{x_{22}} = \frac{x_{13}}{x_{23}} = \frac{x_{11}+x_{12}+x_{13}}{x_{21}+x_{22}+x_{23}} \quad (9)$$

Thus, we obtain the following theorem:

Theorem 2 *If two attributes in a contingency table shown in Table 3 are statistical indepedent, the following equations hold:*

$$\begin{aligned} x_{11}x_{22} - x_{12}x_{21} &= x_{12}x_{23} - x_{13}x_{22} \\ &= x_{13}x_{21} - x_{11}x_{23} = 0 \end{aligned} \quad (10)$$

□

It is notable that this discussion can be easily extended into a $2xn$ contingency table where $n > 3$. The important equation 9 will be extended into

$$\begin{aligned} \frac{x_{11}}{x_{21}} &= \frac{x_{12}}{x_{22}} = \cdots = \frac{x_{1n}}{x_{2n}} \\ &= \frac{x_{11}+x_{12}+\cdots+x_{1n}}{x_{21}+x_{22}+\cdots+x_{2n}} = \frac{\sum_{k=1}^{n} x_{1k}}{\sum_{k=1}^{n} x_{2k}} \end{aligned} \quad (11)$$

Thus,

Theorem 3 *If two attributes in a contingency table ($2 \times k (k = 2, \cdots, n)$) are statistical indepedent, the following equations hold:*

$$\begin{aligned} x_{11}x_{22} - x_{12}x_{21} &= x_{12}x_{23} - x_{13}x_{22} = \cdots \\ &= x_{1n}x_{21} - x_{11}x_{n3} = 0 \end{aligned} \quad (12)$$

□

It is also notable that this equation is the same as the equation on collinearity of projective geometry [1].

5. Statistical Independence in $m \times n$ Contingency Table

Let us consider a $m \times n$ contingency table shown in Table 2. Statistical independence of R_1 and R_2 gives the following formulae:

$$P([R_1 = A_i, R_2 = B_j]) = P([R_1 = A_i])P([R_2 = B_j]) \\ (i = 1, \cdots, m, j = 1, \cdots, n).$$

According to the definition of the table,

$$\frac{x_{ij}}{N} = \frac{\sum_{k=1}^{n} x_{ik}}{N} \times \frac{\sum_{l=1}^{m} x_{lj}}{N}. \quad (13)$$

Thus, we have obtained:

$$x_{ij} = \frac{\sum_{k=1}^{n} x_{ik} \times \sum_{l=1}^{m} x_{lj}}{N}. \quad (14)$$

Thus, for a fixed j,

$$\frac{x_{i_a j}}{x_{i_b j}} = \frac{\sum_{k=1}^{n} x_{i_a k}}{\sum_{k=1}^{n} x_{i_b k}}$$

In the same way, for a fixed i,

$$\frac{x_{i j_a}}{x_{i j_b}} = \frac{\sum_{l=1}^{m} x_{l j_a}}{\sum_{l=1}^{m} x_{l j_b}}$$

Since this relation will hold for any j, the following equation is obtained:

$$\frac{x_{i_a 1}}{x_{i_b 1}} = \frac{x_{i_a 2}}{x_{i_b 2}} \cdots = \frac{x_{i_a n}}{x_{i_b n}} = \frac{\sum_{k=1}^{n} x_{i_a k}}{\sum_{k=1}^{n} x_{i_b k}}. \quad (15)$$

Since the right hand side of the above equation will be constant, thus all the ratios are constant. Thus,

Theorem 4 *If two attributes in a contingency table shown in Table 2 are statistical indepedent, the following equations hold:*

$$\frac{x_{i_a 1}}{x_{i_b 1}} = \frac{x_{i_a 2}}{x_{i_b 2}} \cdots = \frac{x_{i_a n}}{x_{i_b n}} = const. \quad (16)$$

for all rows: i_a and i_b ($i_a, i_b = 1, 2, \cdots, m$).

□

6. Contingency Matrix

The meaning of the above discussions will become much clearer when we view a contingency table as a matrix.

Definition 4 *A corresponding matrix $C_{T_{a,b}}$ is defined as a matrix the element of which are equal to the value of the corresponding contingency table $T_{a,b}$ of two attributes a and b, except for marginal values.*

Definition 5 *The rank of a table is defined as the rank of its corresponding matrix. The maximum value of the rank is equal to the size of (square) matrix, denoted by r.*

The contingency matrix of Table 2($T(R_1, R_2)$) is defined as $C_{T_{R_1,R_2}}$ as below:

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix}$$

6.1. Independence of 2×2 Contingency Table

The results in Section 3 corresponds to the degree of independence in matrix theory. Let us assume that a contingency table is given as Table 1. Then the corresponding matrix ($C_{T_{R_1,R_2}}$) is given as:

$$\begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix},$$

Then,

Proposition 1 *The determinant of $det(C_{T_{R_1,R_2}})$ is equal to $x_{11}x_{22} - x_{12}x_{21}$,*

Proposition 2 *The rank will be:*

$$rank = \begin{cases} 2, & if\ det(C_{T_{R_1,R_2}}) \neq 0 \\ 1, & if\ det(C_{T_{R_1,R_2}}) = 0 \end{cases}$$

From Theorem 1,

Theorem 5 *If the rank of the corresponding matrix of a $2times2$ contingency table is 1, then two attributes in a given contingency table are statistically independent. Thus,*

$$rank = \begin{cases} 2, & dependent \\ 1, & statistical\ independent \end{cases}$$

This discussion can be extended into $2 \times n$ tables. According to Theorem 3, the following theorem is obtained.

Theorem 6 *If the rank of the corresponding matrix of a $2 \times n$ contigency table is 1, then two attributes in a given contingency table are statistically independent. Thus,*

$$rank = \begin{cases} 2, & dependent \\ 1, & statistical\ independent \end{cases}$$

6.2. Independence of 3×3 Contingency Table

When the number of rows and columns are larger than 3, then the situation is a little changed. It is easy to see that the rank for statistical independence of a $m \times n$ contingency table is equal 1.0 as shown in Theorem 4. Also, when the rank is equal to $\min(m, n)$, two attributes are dependent.

Then, what kind of structure will a contingency matrix have when the rank is larger than 1,0 and smaller than $\min(m, n) - 1$? For illustration, let us consider the following $3times3$ contingecy table.

Example 1 *Let us consider the following corresponding matrix:*

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}.$$

The determinant of A is:

$$\begin{aligned} det(A) &= 1 \times (-1)^{1+1} det \begin{pmatrix} 5 & 6 \\ 8 & 9 \end{pmatrix} \\ &\quad +2 \times (-1)^{1+2} det \begin{pmatrix} 4 & 6 \\ 7 & 9 \end{pmatrix} \\ &\quad +3 \times (-1)^{1+3} det \begin{pmatrix} 4 & 5 \\ 7 & 8 \end{pmatrix} \\ &= 1 \times (-3) + 2 \times 6 + 3 \times (-3) = 0 \end{aligned}$$

Thus, the rank of A is smaller than 2. On the other hand, since $(123) \neq k(456)$ and $(123) \neq k(789)$, the rank of A is not equal to 1.0 Thus, the rank of A is equal to 2.0. Actually, one of three rows can be represented by the other two rows. For example,

$$(4\ 5\ 6) = \frac{1}{2}\{(1\ 2\ 3) + (7\ 8\ 9)\}.$$

Therefore, in this case, we can say that two of three pairs of one attribute are dependent to the other attribute, but one pair is statistically independent of the other attribute with respect to the linear combination of two pairs. It is easy to see that this case includes the cases when two pairs are statistically independent of the other attribute, but the table becomes statistically dependent with the other attribute.

In other words, the corresponding matrix is a mixure of statistical dependence and independence. We call this case *contextual independent*. From this illustration, the following theorem is obtained:

Theorem 7 *If the rank of the corresponding matrix of a 3×3 contigency table is 1, then two attributes in a given contingency table are statistically independent. Thus,*

$$rank = \begin{cases} 3, & dependent \\ 2, & contextual\ independent \\ 1, & statistical\ independent \end{cases}$$

It is easy to see that this discussion can be extended into $3 \times n$ contingency tables.

6.3. Independence of $m \times n$ Contingency Table

Finally, the relation between rank and independence in a multi-way contingency table is obtained from Theorem 4.

Theorem 8 *Let the corresponding matrix of a given contingency table be a $m \times n$ matrix. If the rank of the corresponding matrix is 1, then two attributes in a given contingency table are statistically independent. If the rank of the corresponding matrix is $\min(m, n)$, then two attributes in a given contingency table are dependent. Otherwise, two attributes are contextual dependent, which means that several conditional probabilities can be represented by a linear combination of conditional probabilities. Thus,*

$$rank = \begin{cases} \min(m, n) & dependent \\ 2, \cdots, & \\ \quad \min(m, n) - 1 & contextual\ independent \\ 1 & statistical\ independent \end{cases}$$

7. Pseudo Statistical Independence: Example

The next step is to investigate the characteristics of linear independence in a contingency matrix. In other words, a $m \times n$ contingency table whose rank is not equal to $\min(m, n)$. Since two-way matrix (2×2) gives a simple equation whose rank is equal to 1 or 2, let us start our discussion from 3×3-matrix, whose rank is equal to 2, first.

7.1. Three-way Contingency Table (Rank: 2)

Let $M(m, n)$ denote a contingency matrix whose row and column are equal to m and n, respectively. Then, a three-way contingency table is defined as:

$$M(3, 3) = \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix}$$

When its rank is equal to 2, it can be assumed that the third row is represented by the first and second row:

$$(x_{31}\ x_{32}\ x_{33}) = p(x_{11}\ x_{12}\ x_{13}) + q(x_{21}\ x_{22}\ x_{23}) \quad (17)$$

Then, we can consider the similar process in Section 5 (13). In other words, we can check the difference defined below.

$$\Delta(i, j) = \frac{x_{ij}}{N} - \frac{\sum_{k=1}^{n} x_{ik}}{N} \times \frac{\sum_{l=1}^{m} x_{lj}}{N}. \quad (18)$$

Then, the following three types of equations are obtained by simple calculation.

$$\begin{aligned}
\Delta(1,j) &= (1+q)\left\{x_{1j}\sum_{k=1}^{3}x_{2k} - x_{2j}\sum_{k=1}^{3}x_{1k}\right\}\\
\Delta(2,j) &= (1+p)\left\{x_{2j}\sum_{k=1}^{3}x_{1k} - x_{1j}\sum_{k=1}^{3}x_{2k}\right\}\\
\Delta(3,j) &= (p-q)\left\{x_{1j}\sum_{k=1}^{3}x_{2k} - x_{2j}\sum_{k=1}^{3}x_{1k}\right\}
\end{aligned}$$

According to Theorem 4, if $M(3,3)$ is not statistically independent, the formula: $x_{1j}\sum_{k=1}^{3}x_{2k} - x_{2j}\sum_{k=1}^{3}x_{1k}$ is not equal to 1.0. Thus, the following theorem is obtained.

Theorem 9 *The third row represened by a linear combination of first and second rows will satisfy the condition of statistical independence if and only if $p = q$.*

We call the above property *pseudo statistical independence*. This means that if the third column satisfies the following constraint:

$$(x_{31}\ x_{32}\ x_{33}) = (x_{11}\ x_{12}\ x_{13}) + (x_{21}\ x_{22}\ x_{23}),$$

the third column will satisfy the condition of statistical independence. In other words, when we merge the first and second row and construct a 2×3 contingency table, it will become statistical independent. For example,

$$D = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 10 & 14 & 18 \end{pmatrix}$$

can be transformed into

$$D' = \begin{pmatrix} 5 & 7 & 9 \\ 10 & 14 & 18 \end{pmatrix},$$

where D' is statistically independent. Conversely, if D' is provided, it can be decomposed into D. It is notable that the decomposition cannot be uniquely determined. It is also notable that the above discussion does not use the information about the columns of a contingency table. Thus, this discussion can be extended into a $3 \times n$ contingency matrix.

7.2. Four-way Contingency Table (Rank: 3)

From four-way tables, the situation becomes more complicated. In the similar way to Subsection 7.1, a four-way contingency table is defined as:

$$M(4,4) = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}$$

When its rank is equal to 3, it can be assumed that the fourth row is represented by the first to third row:

$$\begin{aligned}
(x_{41}\ x_{42}\ x_{43}\ x_{44}) &= p(x_{11}\ x_{12}\ x_{13}\ x_{14})\\
&\quad +q(x_{21}\ x_{22}\ x_{23}\ x_{24})\\
&\quad +r(x_{31}\ x_{32}\ x_{33}\ x_{34}) \quad (19)
\end{aligned}$$

Then, the following three types of equations are obtained by simple calculation.

$$\begin{aligned}
\Delta(1,j) &= (1+q)\left\{x_{1j}\sum_{k=1}^{4}x_{2k} - x_{2j}\sum_{k=1}^{4}x_{1k}\right\}\\
&\quad +(1+r)\left\{x_{1j}\sum_{k=1}^{4}x_{3k} - x_{3j}\sum_{k=1}^{4}x_{1k}\right\}\\
\Delta(2,j) &= (1+p)\left\{x_{2j}\sum_{k=1}^{4}x_{1k} - x_{1j}\sum_{k=1}^{4}x_{2k}\right\}\\
&\quad +(1+r)\left\{x_{2j}\sum_{k=1}^{4}x_{3k} - x_{3j}\sum_{k=1}^{4}x_{2k}\right\}\\
\Delta(3,j) &= (1+p)\left\{x_{2j}\sum_{k=1}^{4}x_{1k} - x_{1j}\sum_{k=1}^{4}x_{2k}\right\}\\
&\quad +(1+q)\left\{x_{1j}\sum_{k=1}^{4}x_{2k} - x_{2j}\sum_{k=1}^{4}x_{1k}\right\}\\
\Delta(4,j) &= (p-q)\left\{x_{1j}\sum_{k=1}^{4}x_{2k} - x_{2j}\sum_{k=1}^{4}x_{1k}\right\}\\
&\quad +(r-p)\left\{x_{3j}\sum_{k=1}^{4}x_{2k} - x_{1j}\sum_{k=1}^{4}x_{1k}\right\}\\
&\quad +(q-r)\left\{x_{2j}\sum_{k=1}^{4}x_{3k} - x_{3j}\sum_{k=1}^{4}x_{2k}\right\}
\end{aligned}$$

Thus, the following theorem is obtained.

Theorem 10 *The fourth row represened by a linear combination of first to third rows (basis) will satisfy the condition of statistical independence if and only if $\Delta(4,j) = 0$.*

Unfortunately, the condition is not simpler than Theorem 9. It is notable $\Delta(4,j) = 0$ is a diophatine equation whose trivial solution is $p = q = r$. That is, the solution space includes not only $p = q = r$, but other solutions. Thus,

Corollary 1 *If $p = q = r$, then the fourth row satisfies the condition of statistical independence.*

The converse is not true.

Example 2 *Let us consider the following matrix:*

$$E = \begin{pmatrix} 1 & 1 & 2 & 2 \\ 2 & 2 & 3 & 3 \\ 4 & 4 & 5 & 5 \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}.$$

The question is when the fourth row represented by the other rows satisfies the condition of statistical independence. Since $x_{1j}\sum_{k=1}^{4}x_{2k} - x_{2j}\sum_{k=1}^{4}x_{1k} = -2$, $x_{1j}\sum_{k=1}^{4}x_{3k} - x_{3j}\sum_{k=1}^{4}x_{1k} = 6$ *and* $x_{2j}\sum_{k=1}^{4}x_{1k} - x_{1j}\sum_{k=1}^{4}x_{2k} = -4$, $\Delta(4,j)$ *is equal to:* $-2(p-q)+6(r-p)-4(q-r) = -8p-2q+10r$.

Thus, the set of solutions is $\{(p,q,r)|10r = 8p+2q\}$, *where* $p = q = r$ *is included.*

It is notable that the characteristics of solutions will be characterized by a diophantine equation $10r = 8p + 2q$ and a contingency table given by a tripule (p, q, r) may be represented by another tripule. For example, $(3, 3, 3)$ gives the same contingency table as $(1, 6, 2)$:

$$\begin{pmatrix} 1 & 1 & 2 & 2 \\ 2 & 2 & 3 & 3 \\ 4 & 4 & 5 & 5 \\ 21 & 21 & 30 & 30 \end{pmatrix}.$$

It will be our future work to investigate the general characteristics of the solution space.

7.3. Four-way Contingency Table (Rank: 2)

When its rank is equal to 2, it can be assumed that the third and fourth rows are represented by the first to third row:

$$\begin{aligned}(x_{41}\ x_{42}\ x_{43}\ x_{44}) &= p(x_{11}\ x_{12}\ x_{13}\ x_{14}) \\ &\quad +q(x_{21}\ x_{22}\ x_{23}\ x_{24}) \quad (20)\\ (x_{31}\ x_{32}\ x_{33}\ x_{34}) &= r(x_{11}\ x_{12}\ x_{13}\ x_{14}) \\ &\quad +s(x_{21}\ x_{22}\ x_{23}\ x_{24}) \quad (21)\end{aligned}$$

$$\begin{aligned}\Delta(1,j) &= (1+q+s)\left\{x_{1j}\sum_{k=1}^{4}x_{2k} - x_{2j}\sum_{k=1}^{4}x_{1k}\right\}\\ \Delta(2,j) &= (1+p+r)\left\{x_{2j}\sum_{k=1}^{4}x_{1k} - x_{1j}\sum_{k=1}^{4}x_{2k}\right\}\\ \Delta(3,j) &= (p-q+ps-qr)\\ &\quad\times\left\{x_{2j}\sum_{k=1}^{4}x_{1k} - x_{1j}\sum_{k=1}^{4}x_{2k}\right\}\\ \Delta(4,j) &= (r-s+qr-ps)\\ &\quad\times\left\{x_{1j}\sum_{k=1}^{4}x_{2k} - x_{2j}\sum_{k=1}^{4}x_{2k}\right\}\end{aligned}$$

Since $p - q + ps - qr = 0$ and $r - s + qr - ps = 0$ gives the only reasonable solution $p = q$ and $r = s$, the following theorem is obtained.

Theorem 11 *The third and fourth rows represened by a linear combination of first and second rows (basis) will satisfy the condition of statistical independence if and only if* $p = q$ *and* $r = w$.

8. Pseudo Statiatical Independence

Now, we will generalize the results shown in Section 7. Let us consider the $n \times m$ contingency table whose r rows (columns) are described by $n - s$ rows (columns). Thus, we assume a corresponding matrix with the following equations.

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix}$$

$$\begin{aligned}&(x_{n-s+p,1}\ x_{n-s+p,2}\ \cdots\ x_{n-s+p,m}) = \\ &\qquad\sum_{i=1}^{n-s}k_{pi}(x_{i1}\ x_{i2}\ \cdots\ x_{im}) \\ &\qquad\qquad (1 \le s \le n-1, 1 \le p \le s) \quad (22)\end{aligned}$$

Then, the following theorem about $\Delta(u, v)$ is obtained.

Theorem 12 *For a contingency table with size* $n \times m$*:*

$$\Delta(u,v) = \begin{cases} \sum_{i=1}^{n-s}(1+\sum_{p=1}^{n-s}k_{pi}) \\ \quad\times\left\{x_{uv}(\sum_{j=1}^{m}x_{ij}) - x_{iv}(\sum_{j=1}^{m}x_{uj})\right\} \\ \quad (1 \le u \le n-s,\ 1 \le v \le m) \\ \sum_{i=1}^{n-s}\sum_{j=1}^{m}\sum_{q=1}^{n-s}x_{q1}x_{ij} \\ \quad\times\{(k_{uq}-k_{ui}) \\ \quad +k_{uq}\sum_{p=1}^{n-s}k_{pi} - k_{ui}\sum_{p=1}^{n-s}k_{pq}\right\} \\ \quad (n-s+1 \le u \le n,\ 1 \le v \le m) \end{cases} \quad (23)$$

Thus, from the above theorem, if and only if $\Delta(u, v) = 0$ for all v, then the u-th row will satisfy the condition of statistically independence. Especially, the following theorem is obtained.

Theorem 13 *If the following equation holds for all* $v(1 \le v \le m)$ *, then the condition of statistical independence will hold for the* u*-th row in a contingency table.*

$$\begin{aligned}&\sum_{i=1}^{n-s}\sum_{j=1}^{m}\sum_{q=1}^{n-s}\{(k_{uq}-k_{ui}) \\ &\qquad +k_{uq}\sum_{p=1}^{n-s}k_{pi} - k_{ui}\sum_{p=1}^{n-s}k_{pq}\right\} = 0 \quad (24)\end{aligned}$$

It is notable that the above equations give diophatine equations which can check whether each row (column) will satisfy the condition of statistical independence. As a corollary,

Corollary 2 *If k_{ui} is equal for all $i = 1, \cdots, n-s$, then the u-th satisfies the condition of statistical independence.*

The converse is not true.

Example 3 *Let us consider the following matrix:*

$$F = \begin{pmatrix} 1 & 1 & 2 \\ 2 & 2 & 3 \\ 4 & 4 & 5 \\ x_{41} & x_{42} & x_{43} \\ x_{51} & x_{52} & x_{53} \end{pmatrix},$$

where the last two rows are represented by the first three columns. That is, the rank of a matrix is equal to 3. Then, according to Theorem 13, the following equations are obtained:

$$(5k_{53} - k_{52} - 4k_{51}) \times \{k_{41} - 2k_{43} + (k_{51} - 2k_{53} - 1\} = 0 \quad (25)$$

$$(5k_{43} - k_{42} - 4k_{41}) \times \{k_{41} - 2k_{43} + (k_{51} - 2k_{53} - 1\} = 0 \quad (26)$$

In case of $k_{41} - 2k_{43} + (k_{51} - 2k_{53} - 1) = 0$, simple calculations give several equations for those coefficients.

$$\begin{aligned} k_{41} + k_{51} &= 2(k_{43} + k_{53}) + 1 \\ k_{42} + k_{52} &= -3(k_{43} + k_{53}) \end{aligned}$$

The solutions of these two equations give examples of pseudo statistical independence. □

9. Conclusion

In this paper, a contingency table is interpreted from the viewpoint of granular computing and statistical independence. From the definition of statistical independence, statistical independence in a contingency table will holds when the equations of collinearity(Equation 14) are satisfied. In other words, statistical independence can be viewed as linear dependence. Then, the correspondence between contingency table and matrix, gives the theorem where the rank of the contingency matrix of a given contingency table is equal to 1 if two attributes are statistical independent. That is, all the rows of contingency table can be described by one row with the coefficient given by a marginal distribution. If the rank is maximum, then two attributes are dependent. Otherwise, some probabilistic structure can be found within attribute -value pairs in a given attribute, which we call contextual independence. Moreover, from the characteristics of statistical independence, a contingency table may be composed of statistical independent and dependent parts, which we call pseudo statistical dependence. In such cases, if we merge several rows or columns, then we will obtain a new contingency table with statistical independence, whose rank of its corresponding matrix is equal to 1.0. Especially, we obtain Diophatine equations for a pseudo statistical dependence. Thus, matrix algebra and elementary number theory are the key methods of the analysis of a contingency table and the degree of independence, where its rank and the structure of linear dependence as Diophatine equations play very important roles in determining the nature of a given table.

References

[1] H. Coxeter, editor. *Projective Geometry*. Springer Verlag, New York, 2nd edition, 1987.

[2] A. Skowron and J. Grzymala-Busse. From rough set theory to evidence theory. In R. Yager, M. Fedrizzi, and J. Kacprzyk, editors, *Advances in the Dempster-Shafer Theory of Evidence*, pages 193–236. John Wiley & Sons, New York, 1994.

[3] Y. Yao and S. Wong. A decision theoretic framework for approximating concepts. *International Journal of Man-machine Studies*, 37:793–809, 1992.

[4] Y. Yao and N. Zhong. An analysis of quantitative measures associated with rules. In N. Zhong and L. Zhou, editors, *Methodologies for Knowledge Discovery and Data Mining, Proceedings of the Third Pacific-Asia Conference on Knowledge Discovery and Data Mining LNAI* **1574**, pages 479–488, Berlin, 1999. Springer.

On the Correspondence between Degree of Dependence and Granularity

Shusaku Tsumoto and Shoji Hirano
Department of Medical Informatics,
Shimane University, School of Medicine,
Enya-cho Izumo City, Shimane 693-8501 Japan
tsumoto@computer.org, hirano@ieee.org

Abstract

This paper gives a relations between the degree of granularity and that of dependence of contingency tables. From the results of determinantal divisors, it seems that the devisors provide information on the degree of dependencies between the matrix of the whole elements and its submatrices and the increase of the degree of granularity may lead to that of dependence. However, this paper shows that a constraint on the sample size of a contingency table is very strong, which leads to the evaluation formula where the increase of degree of granularity gives the decrease of dependency.

1. Introduction

Independence (dependence) is a very important concept in data mining, especially for feature selection. In rough sets[**?**], if two attribute-value pairs, say $[c = 0]$ and $[d = 0]$ are independent, their supporting sets, denoted by C and D do not have a overlapping region ($C \cap D = \phi$), which means that one attribute independent to a given target concept may not appear in the classification rule for the concept.

This idea is also frequently used in other rule discovery methods: let us consider deterministic rules, described as *if-then* rules, which can be viewed as classic propositions ($C \rightarrow D$). From the set-theoretical point of view, a set of examples supporting the conditional part of a deterministic rule, denoted by C, is a subset of a set whose examples belong to the consequence part, denoted by D. That is, the relation $C \subseteq D$ holds and deterministic rules are supported only by positive examples in a dataset[**?**].

When such a subset relation is not satisfied, indeterministic rules can be defined as if-then rules with probabilistic information[**?**]. From the set-theoretical point of view, C is not a subset, but closely overlapped with D. That is, the relations $C \cap D \neq \phi$ and $|C \cap D|/|C| \geq \delta$ will hold in this case.[1] Thus, probabilistic rules are supported by a large number of positive examples and a small number of negative examples.

On the other hand, in a probabilistic context, independence of two attributes means that one attribute (a_1) will not influence the occurrence of the other attribute (a_2), which is formulated as $p(a_2|a_1) = p(a_2)$.

Although independence is a very important concept, it has not been fully and formally investigated as a relation between two attributes. Tsumoto introduces linear algebra into formal analysis of a contigency table [**?**]. The results give the following interesting results. First, a contingency table can be viewed as comparison between two attributes with respect to information granularity. Second, algebra is a key point of analysis of this table. A contingency table can be viewed as a matrix and several operations and ideas of matrix theory are introduced into the analysis of the contingency table. Especially, The degree of independence, rank plays a very important role in extracting a probabilistic model from a given contingency table.

This paper gives a further investigation on the degree of independence of contingency matrix.

Intuitively and empirically, when two attributes has many values, the dependence between these two attributes becomes low. However, from the results of determinantal divisors, it seems that the devisors provide information on the degree of dependencies between the matrix of the whole elements and its submatrices and the increase of the degree of granularity may lead to that of dependence. The key of the resolution of these conflicts is to consider the constraint on the sample size.

In this paper we show that a constraint on the sample size of a contingency table is very strong, which leads to the evaluation formula where the increase of degree of granularity gives the decrease of dependency. The paper is organized as follows: Section 2 shows preliminaries. Section 3

1 The threshold δ is the degree of the closeness of overlapping sets, which will be given by domain experts. For more information, please refer to Section 3.

discusses the former results. Section 4 gives the relations between rank and submatrices of a matrix. Finally, Section 6 concludes this paper.

2. Contingency Table from Rough Sets

2.1. Notations

In the subsequent sections, the following notations is adopted, which is introduced in [**?**]. Let U denote a nonempty, finite set called the universe and A denote a nonempty, finite set of attributes, i.e., $a : U \to V_a$ for $a \in A$, where V_a is called the domain of a, respectively. Then, a decision table is defined as an information system, $A = (U, A \cup \{\mathcal{D}\})$, where $\{\mathcal{D}\}$ is a set of given decision attributes. The atomic formulas over $B \subseteq A \cup \{\mathcal{D}\}$ and V are expressions of the form $[a = v]$, called descriptors over B, where $a \in B$ and $v \in V_a$. The set $F(B, V)$ of formulas over B is the least set containing all atomic formulas over B and closed with respect to disjunction, conjunction and negation. For each $f \in F(B, V)$, f_A denote the meaning of f in A, i.e., the set of all objects in U with property f, defined inductively as follows.

1. If f is of the form $[a = v]$ then, $f_A = \{s \in U | a(s) = v\}$
2. $(f \wedge g)_A = f_A \cap g_A$; $(f \vee g)_A = f_A \vee g_A$; $(\neg f)_A = U - f_a$

2.2. Multi-way Contingency Table

Two-way contingency table can be extended into a contingency table for multinominal attributes.

Definition 1 *Let R_1 and R_2 denote multinominal attributes in an attribute space A which have m and n values. A contingency tables is a table of a set of the meaning of the following formulas: $|[R_1 = A_j]_A|$, $|[R_2 = B_i]_A|$, $|[R_1 = A_j \wedge R_2 = B_i]_A|$, $|[R_1 = A_1 \wedge R_1 = A_2 \wedge \cdots \wedge R_1 = A_m]_A|$, $|[R_2 = B_1 \wedge R_2 = A_2 \wedge \cdots \wedge R_2 = A_n]_A|$ and $|U|$ $(i = 1, 2, 3, \cdots, n$ and $j = 1, 2, 3, \cdots, m)$. This table is arranged into the form shown in Table 1, where: $|[R_1 = A_j]_A| = \sum_{i=1}^{m} x_{1i} = x_{\cdot j}$, $|[R_2 = B_i]_A| = \sum_{j=1}^{n} x_{ji} = x_{i\cdot}$, $|[R_1 = A_j \wedge R_2 = B_i]_A| = x_{ij}$, $|U| = N = x_{\cdot\cdot}$ $(i = 1, 2, 3, \cdots, n$ and $j = 1, 2, 3, \cdots, m)$.*

Example.

Let us consider an information table shown in Table 2. The relationship between b and e can be examined by using the corresponding contingency table as follows. First, the frequencies of four elementary relations are counted, called *marginal distributions*: $[b = 0]$, $[b = 1]$, $[e = 0]$, and $[e = 1]$. Then, the frequencies of four kinds of conjunction are counted: $[b = 0] \wedge [e = 0]$, $[b = 0] \wedge [e = 1]$, $[b = 1] \wedge [e = 0]$, and $[b = 1] \wedge [e = 1]$. Then, the following contingency table is obtained (Table 3). From this table, accuracy and coverage for $[b = 0] \to [e = 0]$ are obtained as $1/(1 + 2) = 1/3$ and $1/(1 + 1) = 1/2$.

	A_1	A_2	$\cdots$	A_n	Sum
B_1	x_{11}	x_{12}	$\cdots$	x_{1n}	$x_{1\cdot}$
B_2	x_{21}	x_{22}	$\cdots$	x_{2n}	$x_{2\cdot}$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
B_m	x_{m1}	x_{m2}	$\cdots$	x_{mn}	$x_{m\cdot}$
Sum	$x_{\cdot 1}$	$x_{\cdot 2}$	$\cdots$	$x_{\cdot n}$	$x_{\cdot\cdot} = \|U\| = N$

Table 1. Contingency Table ($n \times m$)

a	b	c	d	e
1	0	0	0	1
0	0	1	1	1
0	1	2	2	0
1	1	1	2	1
0	0	2	1	0

Table 2. A Small Dataset

	b=0	b=1	
e=0	1	1	2
e=1	2	1	3
	3	2	5

Table 3. Corresponding Contingency Table

One of the important observations from granular computing is that a contingency table shows the relations between two attributes with respect to intersection of their supporting sets. For example, in Table 3, both b and e have two different partitions of the universe and the table gives the relation between b and e with respect to the intersection of supporting sets. It is easy to see that this idea can be extended into $n - way$ contingency tables, which can be viewed as $n \times n$-matrix. When two attributes have different number of equivalence classes, the situation may be a little complicated. But, in this case, due to knowledge about linear algebra, we only have to consider the attribute which has a smaller number of equivalence classes. and the surplus number of equivalence classes of the attributes with larger number of equivalnce classes can be projected into other partitions. In other words, a $n \times m$ matrix or contingency table includes a projection from one attributes to the other one.

3. Rank of Contingency Table (two-way)

3.1. Preliminaries

Definition 2 *A corresponding matrix $C_{T_{a,b}}$ is defined as a matrix the element of which are equal to the value of the corresponding contingency table $T_{a,b}$ of two attributes a and b, except for marginal values.*

Definition 3 *The rank of a table is defined as the rank of its corresponding matrix. The maximum value of the rank is equal to the size of (square) matrix, denoted by r.*

Example.

Let the table given in Table 3 be defined as $T_{b,e}$. Then, $C_{T_{b,e}}$ is:

$$\begin{pmatrix} 1 & 1 \\ 2 & 1 \end{pmatrix}$$

Since the determinant of $C_{T_{b,e}}$ $det(C_{T_{b,e}})$ is not equal to 0, the rank of $C_{T_{b,e}}$ is equal to 2. It is the maximum value ($r = 2$), so b and e are statistically dependent.

3.2. Independence when the table is two-way

From the results in linear algebra, several results are obtained. (The proofs is omitted.) First, it is assumed that a contingency table is given as two-way $m = 2, n = 2$ in Table 1. Then the corresponding matrix ($C_{T_{R_1,R_2}}$) is given as:

$$\begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix},$$

Proposition 1 *The determinant of $det(C_{T_{R_1,R_2}})$ is equal to $|x_{11}x_{22} - x_{12}x_{21}|$.*

Proposition 2 *The rank will be:*

$$rank = \begin{cases} 2, & if\ det(C_{T_{R_1,R_2}}) \neq 0 \\ 1, & if\ det(C_{T_{R_1,R_2}}) = 0 \end{cases}$$

If the rank of $det(C_{T_{b,e}})$ is equal to 1, according to the theorems of the linear algebra, it is obtained that one row or column will be represented by the other column. That is,

Proposition 3 *Let r_1 and r_2 denote the rows of the corresponding matrix of a given two-way table, $C_{T_{b,e}}$. That is,*

$$r_1 = (x_{11}, x_{12}), r_2 = (x_{21}, x_{22})$$

Then, r_1 can be represented by r_2: $r_1 = kr_2$, where k is given as:

$$k = \frac{x_{11}}{x_{21}} = \frac{x_{12}}{x_{22}} = \frac{x_{1\cdot}}{x_{2\cdot}}$$

From this proposition, the following theorem is obtained.

	a=0	a=1	
c=0	0	1	1
c=1	1	1	2
c=2	2	0	2
	3	2	5

Table 4. Contingency Table for a and c

Theorem 1 *If the rank of the corresponding matrix is 1, then two attributes in a given contingency table are statistically independent. Thus,*

$$rank = \begin{cases} 2, & dependent \\ 1, & statistical\ independent \end{cases}$$

4. Rank of Contingency Table (Multi-way)

In the case of a general square matrix, the results in the two-way contingency table can be extended. Especially, it is very important to observe that conventional statistical independence is only supported when the rank of the corresponding is equal to 1. Let us consider the contingency table of c and a in Table 2, which is obtained as follows. Thus, the corresponding matrix of this table is:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix},$$

whose determinant is equal to 0. It is clear that its rank is 2. It is interesting to see that if the case of $[d = 0]$ is removed, then the rank of the corresponding matrix is equal to 1 and two rows are equal. Thus, if the value space of d into $\{1, 2\}$ is restricted, then c and d are statistically independent. This relation is called *contextual independence* [**?**], which is related with conditional independence.

However, another type of weak independence is observed: let us consider the contingency table of a and c. The table is obtained as Table 4:
Its corresponding matrix is:

$$C_{T_{a,c}} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 2 & 0 \end{pmatrix},$$

Since the corresponding matrix is not square, the determinant is not defined. But it is easy to see that the rank of this matrix is two. In this case, even any attribute-value pair removed from the table will not generate statistical independence. Finally, the relation between rank and independence in a multi-way contingency table is obtained.

Theorem 2 *Let the corresponding matrix of a given contingency table be a square $n \times n$ matrix. If the rank of the cor-*

responding matrix is 1, then two attributes in a given contingency table are statistically independent. If the rank of the corresponding matrix is n , then two attributes in a given contingency table are dependent. Otherwise, two attributes are contextual dependent, which means that several conditional probabilities can be represented by a linear combination of conditional probabilities. Thus,

$$rank = \begin{cases} n & dependent \\ 2, \cdots, n-1 & contextual\ independent \\ 1 & statistical\ independent \end{cases}$$

□

This theorem can be generalized into $m \times n$ matrix. If the corresponding matrix of a given contingency table is not square and of the form $m \times n$, then its rank is at most $\min(m, n)$.

For example, since $C_{T_{a,c}}$ is 3×2, the rank is at most 2. Actually, from the calculation of subdeterminants shown in the next section, this matrix has a rank of 2.

Theorem 3 *Let the corresponding matrix of a given contingency table be a $m \times n$ matrix. The rank of this matrix is less than $\min(m, n)$. If the rank of the corresponding matrix is 1, then two attributes in a given contingency table are statistically independent. If the rank of the corresponding matrix is n , then two attributes in a given contingency table are dependent. Otherwise, two attributes are contextual dependent, which means that several conditional probabilities can be represented by a linear combination of conditional probabilities. Thus,*

$$rank = \begin{cases} \min(m, n) & dependent \\ 2, \cdots, & \\ \quad \min(m, n) - 1 & contextual\ independent \\ 1 & statistical\ independent \end{cases}$$

□

In the cases of $m \neq n$, we need a discussion on submatrix and subderminant in the next section.

5. Rank and Degree of Dependence

5.1. Submatrix and Subdeterminant

The next interest is the structure of a corresponding matrix with $1 \leq rank \leq n-1$. First, let us define a submatrix (a subtable) and subdeterminant.

Definition 4 *Let A denote a corresponding matrix of a given contigency table ($m \times n$). A corresponding submatrix $A^{i_1 i_2 \cdots i_r}_{j_1 j_2 \cdots j_s}$ is defined as a matrix which is given by an intersection of r rows and s columns of A ($i_1 < i_2 < \cdots < i_r, j_1 < j_2 < \cdots < j_r$).*

Definition 5 *A subdeterminant of A is defined as a determinant of a submatrix $A^{i_1 i_2 \cdots i_r}_{j_1 j_2 \cdots j_s}$, which is denoted by $det(A^{i_1 i_2 \cdots i_r}_{j_1 j_2 \cdots j_s})$.*

Let us consider the contingency table given as Table 1. Then, a subtable for $A^{i_1 i_2 \cdots i_r}_{j_1 j_2 \cdots j_s}$ is given as Table 5.

	A_{j_1}	A_{j_2}	$\cdots$	A_{j_r}	Sum
B_{i_1}	$x_{i_1 j_1}$	$x_{i_1 j_2}$	$\cdots$	$x_{i_1 j_r}$	$x_{i_1 \cdot}$
B_{i_2}	$x_{i_2 j_1}$	$x_{i_2 j_2}$	$\cdots$	$x_{i_2 j_r}$	$x_{i_2 \cdot}$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
B_{i_r}	$x_{i_r j_1}$	$x_{i_r j_2}$	$\cdots$	$x_{i_r j_n}$	$x_{i_r \cdot}$
Sum	$x_{\cdot 1}$	$x_{\cdot 2}$	$\cdots$	$x_{\cdot n}$	$x_{\cdot\cdot} = \lvert U \rvert = N$

Table 5. A subtable ($r \times s$)

5.2. Rank and Subdeterminant

Let δ_{ij} denote a co-factor of a_{ij} in a square corresponding matrix of A. Then,

$$\Delta_{ij} = (-1)^{i+j} det(A^{1,2,\cdots,i-1,i+1,\cdots,n}_{1,2,\cdots,j-1,j+1,\cdots,n}).$$

It is notable that a co-factor is a special type of submatrix, where only ith-row and j-column are removed from a original matrix. By the use of co-factors, the determinant of A is defined as:

$$det(A) = \sum_{j=1}^{n} a_{ij} \Delta_{ij},$$

which is called *Laplace expansion*.

From this representation, if $det(A)$ is not equal to 0, then $\Delta_{ij} \neq 0$ for $\{a_{i1}, a_{i2}, \cdots, a_{in}\}$ which are not equal to 0. Thus, the following proposition is obtained.

Proposition 4 *If $det(A)$ is not equal to 0 if at least one co-factor of $a_{ij} (\neq 0)$, Δ_{ij} is not equal to 0.*

It is notable that the above definition of a determinant gives the relation between a original matrix A and submatrices (co-factors). Since cofactors gives a square matrix of size $n-1$, the above proposition gives the relation between a matrix of size n and submatrices of size $n-1$. In the same way, we can discuss the relation between a corresponding matrix of size n and submatrices of size $r (1 \leq r < n-1)$.

5.3. Rank and Submatrix

Let us assume that corresponding matrix and submatrix are square ($n \times n$ and $r \times r$, respectively).

Theorem 4 *If the rank of a corresponding matrix of size $n \times n$ is equal to r, at least the determinant of one submatrix of size $r \times r$ is not equal to 0. That is, there exists a submatrix $A^{i_1 i_2 \cdots i_r}_{j_1 j_2 \cdots j_r}$, which satisfies $det(A^{i_1 i_2 \cdots i_r}_{j_1 j_2 \cdots j_r}) \neq 0$*

Corollary 1 *If the rank of a corresponding matrix of size $n \times n$ is equal to r, all the determinants of the submatrices whose number of columns and rows are larger than $r+1 (\leq n)$ are equal to 0.* □

Example. Let us consider the corresponding matrix mentioned in the above section, $C_{T_{a,c}}$. The submatrices of this matrix are:

$$\begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 2 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

Since all the subderminants are not equal to 0, the rank of this corresponding matrix is equal to 2.

Example 1 *Let us consider the following corresponding matrix:*

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}.$$

The determinant of A is:

$$\begin{aligned} det(A) &= 1 \times (-1)^{1+1} det \begin{pmatrix} 5 & 6 \\ 8 & 9 \end{pmatrix} \\ &\quad +2 \times (-1)^{1+2} det \begin{pmatrix} 4 & 6 \\ 7 & 9 \end{pmatrix} \\ &\quad +3 \times (-1)^{1+3} det \begin{pmatrix} 4 & 5 \\ 7 & 8 \end{pmatrix} \\ &= 1 \times (-3) + 2 \times 6 + 3 \times (-3) = 0 \end{aligned}$$

Thus, the rank of A is smaller than 2.

All the subdeterminants of A are:

$$\begin{aligned} &det \begin{pmatrix} 5 & 6 \\ 8 & 9 \end{pmatrix} = -3, \quad det \begin{pmatrix} 4 & 6 \\ 7 & 9 \end{pmatrix} = -6, \\ &det \begin{pmatrix} 4 & 5 \\ 7 & 8 \end{pmatrix} = -3, \quad det \begin{pmatrix} 1 & 2 \\ 7 & 8 \end{pmatrix} = -6, \\ &det \begin{pmatrix} 1 & 3 \\ 7 & 9 \end{pmatrix} = -12, \quad det \begin{pmatrix} 2 & 3 \\ 8 & 9 \end{pmatrix} = -6, \\ &det \begin{pmatrix} 1 & 2 \\ 4 & 5 \end{pmatrix} = -3, \quad det \begin{pmatrix} 1 & 3 \\ 4 & 6 \end{pmatrix} = -6, \\ &det \begin{pmatrix} 2 & 3 \\ 5 & 6 \end{pmatrix} = -3. \end{aligned}$$

Since all the subdeterminants of A are not equal to 0, the rank of A is equal to 2. Actually, since

$$\begin{pmatrix} 4 & 5 & 6 \end{pmatrix} = \frac{1}{2}\{\begin{pmatrix} 1 & 2 & 3 \end{pmatrix} + \begin{pmatrix} 7 & 8 & 9 \end{pmatrix}\},$$

and $(7 \quad 8 \quad 9)$ cannot be represented by $k(1 \quad 2 \quad 3)(k : integer)$, the rank of this matrix is equal to 2.

Thus, one attribute-value pair is statistically dependent on other two pairs, statistically independent of the other attribute. In other words, if two pairs are fixed, the remaining one attribute-value pair will be statistically independently determined.

5.4. Determinantal Divisors

From the subdeterminants of all the submatrices of size 2, all the subdeterminants of a corresponding matrix has the greatest common divisor, equal to 3.

From the recursive definition of the determinants, it is show that the subdeterminants of size $r+1$ will have the greatest common divisor of the subdeterminants of size r as a divisor. Thus,

Theorem 5 *Let $d_k(A)$ denote the greatest common divisor of all the subdeterminants of size k, $det(A^{i_1 i_2 \cdots i_k}_{j_1 j_2 \cdots j_r})$. $d_1(A), d_2(A), \cdots, d_n(A)$ are called determinantal divisors. From the definition of Laplace expansion,*

$$d_k(A) | d_{k+1}(A).$$

□

In the example of the above subsection, $d_1(A) = 1$, $d_2(A) = 3$ and $d_3(A) = 0$.

Example 2 *Let us cosider $C_{T_{a,c}}$ as an example. $d_1(C_{T_{a,c}}) = 1$ and $d_2(C_{T_{a,c}}) = 1$.*

Example 3 *Let us consider the following corresponding matrix:*

$$B = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & \mathbf{11} & 9 \end{pmatrix}.$$

Calculation gives: $d_1(B) = 1$, $d_2(B) = 3$ and $d_3(B) = 18$.

It is notable that a simple change of a corresponding matrix gives a significant change to the determinant, which suggests a change of structure in dependence/independence.

The relation between $d_k(A)$ gives a interesting constraint.

Proposition 5 *Since $d_k(A) | d_{k+1}(A)$, the sequence of the devisiors is monotonically increasing one:*

$$d_1(A) \leq d_2(A) \cdots \leq d_r(A),$$

where r denotes the rank of A.

The sequence of B illustrates this: $1 < 3 < 18$.

Let us define a ratio of $d_k(A)$ to $d_{k-1}(A)$, called *elementary divisors*, where C denotes a corresponding matrix and $k \leq rank A$:

$$e_k(C) = \frac{d_k(C)}{d_{k-1}(C)} (d_0(C) = 0).$$

The elementary divisors may give the increase of dependency between two attributes. For example, $e_1(B) = 1$, $e_2(B) = 3$, and $e_3(B) = 6$. Thus, a transition from 2×2

to 3×3 have a higher impact on the dependency of two attributes.

It is trivial to see that $det(B) = e_1e_2e_3$, which can be viewed as a decomposition of the determinant of a corresponindg matrix.

5.5. Divisors and Degree of Dependence

Since the determinant can be viewed as the degree of dependence, this result is very important. If values of all the subdeterminants (size r) are very small (nearly equal to 0) and $d_r(A) \simeq 1$, then the values of the subdeterminants (size $r+1$) are very small. This property may hold until the r reaches the rank of the corresponding matrix. Thus, the sequence of the divisiors of a corresponding matrix gives a hidden structure of a contingency table.

Also, this results show that $d_1(A)$ and $d_2(A)$ are very important to estimate the rank of a corresponding matrix. Since $d_1(A)$ is only given by the greatest common divisor of all the elements of A, $d_2(A)$ are much more important components. This also intuitively suggests that the subdeterminants of A with size 2 are principal components of a corresponding matrix from the viewpoint of statistical dependence.

Recall that statistical independence of two attributes is equivalent to a corresponding matrix with rank being 1. A matrix with rank being 2 gives a context-dependent independence, which means three values of two attributes are independent, but two values of two attributes are dependent.

5.6. Subdeterminants and Degree of Dependence

Since the determinants give the degree of dependence, the degree of dependence can be evaluted by the values of subdeterminants.

For the above examples (A), since

$$det \begin{pmatrix} 1 & 3 \\ 7 & 9 \end{pmatrix} = -12$$

gives the maximum value, the first and the third attribute-value pairs for two attributes are dependent each other.

On the other hand, concerning B, since

$$det \begin{pmatrix} 2 & 3 \\ 11 & 9 \end{pmatrix} = -15$$

gives the maximum value, the second and the third attribute-value pairs for two attributes are dependent each other.

This discussion can be extended into the dependency between attribute-value pairs and a corresponding attribute. Let us consider 3×2 submatrices of A, which removes one column of the matrix.

$$A_1 = \begin{pmatrix} 2 & 3 \\ 5 & 6 \\ 8 & 9 \end{pmatrix}, \ A_2 = \begin{pmatrix} 1 & 3 \\ 4 & 6 \\ 7 & 9 \end{pmatrix}, \ A_3 = \begin{pmatrix} 1 & 2 \\ 4 & 5 \\ 7 & 8 \end{pmatrix}.$$

From the discussions in the above subsections, a set of the subderminants of 2×2 submatrices of A_j, denoted by D_{A_j} are obtained as:

$$\begin{aligned} D_{A_1} &= \{-3, -6, -3\} \\ D_{A_2} &= \{-12, -6, -6\} \\ D_{A_3} &= \{-3, -6, -3\} \end{aligned}$$

Thus, the first and three attribute value pairs are more dependent than second value pairs, concerning the classification of attributes for the rows.

5.7. Elementary Divisors and Elementary Transformation

Let us define the following three elementary (row/column)transformations of a coreesponding matrix:

1. Exchange two rows (columns), i_0 and j_0 ($P(i_0, j_0)$).
2. Multiply -1 to a row (column) i_0 ($T(i_0; -1)$).
3. Multiply t to a row (column) j_0 (i_0) and add it to a row i_0 (j_0). ($W(i_0, j_0, t)$).

Then, three transformations have several interesting characterics.

Proposition 6 *Matrices corresponding to three elementary transformations are regular.*

Proposition 7 *Three elementary transformations do not change the rank of a corresponding matrix.*

Proposition 8 *Let $\tilde{A}$ denote a matrix transformed by finite steps of three operations. Then,*

$$rank\tilde{A} = rankA, \quad d_r(\tilde{A}) = d_r(A),$$

where r denotes the rank of matrix A.

Then, from the results of linear algebra, the following interesting result is obtained.

Theorem 6 *With the finite steps of elementary transformations, a given corresponding matrix is transformed into*

$$\tilde{A} = \left(\begin{array}{ccc|c} e_1 & & & \\ & e_2 & & \\ & & \ddots & O \\ & & & e_r \\ \hline & O & & O \end{array} \right),$$

where $e_j = \frac{d_j(A)}{d_{j-1}(A)} (d_0(A) = 1)$ and r denotes the rank of a corresponding matrix. Then, the determinant is decomposed into the product of e_j.

$$d_r(\tilde{A}) = d_r(A) = e_1e_2 \cdots e_r.$$

□

6. Degree of Granularity and Dependence

From Theorem 6, it seems that the increase of the degree of granularity gives that of the dependence between two attributes.

However, our empirical observations are different from the above intuitive analysis. Thus, there should be a strong constraint which suppress the above effects on the degree of granularity.

Let us assume that the determinant of a give contingency matrix gives the degree of the dependence of the matrix. Then, from the results of linear algebra, we obtain the following theorem.

Theorem 7 *Let A denote a $n \times n$ contingency matrix, which includes N samples. If the rank of A is equal to n, then there exists a matrix B $(n \times n)$ which satisfies*

$$BA = \begin{pmatrix} \rho_1 & & & \\ & \rho_2 & & O \\ & & \ddots & \\ & O & & \rho_n \end{pmatrix} = P,$$

where $\rho_1 + \rho_2 + \cdots + \rho_n = N$.

It is notable that the value of determinants of P is larger than A:

$$detA \leq detP$$

□

Example 4 *Let us consider B as an example (Example 3). Let C denote the orthogonal matrix for transformaiton of B. Since the cardinality of B is equal to 48, the diagonal matrix which gives the maximum determinant is equal to:*

$$\begin{pmatrix} 16 & 0 & 0 \\ 0 & 16 & 0 \\ 0 & 0 & 16 \end{pmatrix}.$$

On the other hand, the determinant of B is equal to 18. Thus, $detB = 18 < 16^3 = 4096$. Then, C is obtained from the following equation.

$$C \times \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 11 & 9 \end{pmatrix} = \begin{pmatrix} 16 & 0 & 0 \\ 0 & 16 & 0 \\ 0 & 0 & 16 \end{pmatrix}.$$

Thus,

$$C = \begin{pmatrix} -56/3 & 40/3 & -8/3 \\ 16/3 & -32/3 & 16/3 \\ 8 & 8/3 & -8/3 \end{pmatrix}$$

It is notable that the determinant of C is equal to $2048/9$. Also, since $detB = 18$, we do not have any diagonal matrix whose determinant is equal to 18 and the sum of all the elements is equal to 48. □

It is easy to see that the tranformed matrix P has a very nice property to calculate the determinant.

Proposition 9 *The determinant of the transformed matrix P is equal to the multiplication of ρ_1 to ρ_n. That is,*

$$detP = \rho_1 \rho_2 \cdots \rho_n$$

□

Then, the following constraint will be have the special meaning:

$$\rho_1 + \rho_2 + \cdots + \rho_n = N, \tag{1}$$

because the following inequality holds in general:

$$\frac{\rho_1 + \rho_2 + \cdots + \rho_n}{n} \geq \sqrt[n]{\rho_1 \rho_2 \cdots \rho_n}, \tag{2}$$

where the equality holds when $\rho_1 = \rho_2 = \cdots = \rho_n$. Since the above inequality can be transformed into:

$$\rho_1 \rho_2 \cdots \rho_n \leq \left(\frac{\rho_1 + \rho_2 + \cdots + \rho_n}{n} \right)^n,$$

the following inequality is obtained:

$$detP = \rho_1 \rho_2 \cdots \rho_n \leq \left(\frac{\rho_1 + \rho_2 + \cdots + \rho_n}{n} \right)^n, \tag{3}$$

where the equality holds when $\rho_1 = \rho_2 = \cdots = \rho_n$. From the theorem 7 and equation 1, the following theorem is obtained.

Theorem 8 *When a contingency matrix A holds $AB = P$, where P is a diagonal matrix, the following inequality holds:*

$$detA \leq \left(\frac{N}{n} \right)^n,$$

Proof.

$$\begin{aligned} detA &= det(PB^{-1}) \\ &\leq detP \\ &= \rho_1 \rho_2 \cdots \rho_n \\ &\leq \left(\frac{\rho_1 + \rho_2 + \cdots + \rho_n}{n} \right)^n = \left(\frac{N}{n} \right)^n, \end{aligned} \tag{4}$$

where the former equality holdes when $detB^{-1} = detB = 1$ and the latter equality holds when $\rho_1 = \rho_2 = \cdots = \rho_n = \frac{N}{n}$.

Example 5 *Let us consider the following contingency matrices D and E:*

$$D = \begin{pmatrix} 1 & 2 & 3 & 0 \\ 4 & 5 & 6 & 0 \\ 7 & \mathbf{11} & 9 & 0 \\ 0 & 0 & 0 & \mathbf{1} \end{pmatrix}$$

$$E = \begin{pmatrix} 1 & 2 & 3 & 0 \\ 4 & 5 & 6 & 0 \\ 7 & \mathbf{10} & 9 & 0 \\ 0 & 0 & 0 & \mathbf{1} \end{pmatrix}$$

The numbers of examples of D and E are 49 and 48, respectively, which can be comparable to that of B. Then, from Theorem 8,

$$\begin{aligned} detD = 18 &< (49/4)^4 = \frac{5764801}{256} \sim 22518 \\ detE = 12 &< (48/4)^4 = 20736 \end{aligned}$$

Thus, the maximum value of the determinant of A is at most $\left(\frac{N}{n}\right)^n$. Since N is constant for the given matrix A, the degree of dependence will decrease very rapidly when n becomes very large. That is,

$$detA \sim n^{-n}.$$

Thus,

Corollary 2 *The determinant of A will converge into 0 when n increases into infinity.*

$$\lim_{n \to \infty} detA = 0.$$

□

This results suggest that when the degree of granularity becomes higher, the degree of dependence will become lower, due to the constraints on the sample size.

However, it is notable that N/n is very important. If N is very large, the rapid decrease will be observed N is close to n. Even N is 48 as shown in Example 5, $n = 3, 4$ may give a strong dependency between two attributes. For the behavior of $(N/n)^n$, we can apply the technique of real analysis, which will our future work.

7. Conclusion

In this paper, a contingency table is interpreted from the viewpoint of granular computing and statistical independence. Matrix algebra is a key point of the analysis of a contingency table and the degree of independence, rank plays a very important role in extracting a probabilistic model. From the correspondence between contingency table and matrix, the following results are obtained: First, the value of determinants gives the degree of of dependency between attribute-value pairs for a set of submatrices with the same size. Second, from the characteristics of the determinants, the larger rank a correponding matrix has, the higher the two attributes are dependent. This results is shown by a monotonicity of a sequence of determinantal divisors. Third, elementary divisors give a decomposition of the determinant of a corresponding matrix. Finally, the constraint on the sample size of a contingency table is very strong, which leads to the evaluation formula where the increase of degree of granularity gives the decrease of dependency.

Acknowledgement

This work was supported by the Grant-in-Aid for Scientific Research (13131208) on Priority Areas (No.759) "Implementation of Active Mining in the Era of Information Flood" by the Ministry of Education, Science, Culture, Sports, Science and Technology of Japan.

References

[1] C. Butz. Exploiting contextual independencies in web search and user profiling. In *Proceedings of World Congress on Computational Intelligence (WCCI'2002) (CD-ROM)*, 2002.

[2] Z. Pawlak. *Rough Sets*. Kluwer Academic Publishers, Dordrecht, 1991.

[3] A. Skowron and J. Grzymala-Busse. From rough set theory to evidence theory. In R. Yager, M. Fedrizzi, and J. Kacprzyk, editors, *Advances in the Dempster-Shafer Theory of Evidence*, pages 193–236. John Wiley & Sons, New York, 1994.

[4] S. Tsumoto. Knowledge discovery in clinical databases and evaluation of discovered knowledge in outpatient clinic. *Information Sciences*, (124):125–137, 2000.

[5] S. Tsumoto. Statistical independence as linear independence. In A. Skowron and M. Szczuka, editors, *Electronic Notes in Theoretical Computer Science*, volume 82. Elsevier, 2003.

[6] S. Tsumoto and H. Tanaka. Automated discovery of medical expert system rules from clinical databases based on rough sets. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining 96*, pages 63–69, Palo Alto, 1996. AAAI Press.

Data Reconstruction through a Fisher Game

R. C. Venkatesan
Systems Research Corporation
Aundh, Pune 411007, India
Email: ravi@systemsresearchcorp.com

Abstract

A principled formulation for reconstructing pdf's from discrete data comprising of random sequences, based on an invariance preserving extension of the Extreme Physical Information (EPI) theory, is presented. The Fisher information is employed as a measure of uncertainty in the measurement-response model, that represents an information theoretic game. A quantum mechanical connotation is ascribed to the reconstruction process, which is cast as a variational principle. Numerical results with Gaussian mixture models formed from random sequences demonstrate the efficacy of the Fisher game formulation.

1. Introduction

Extreme Physical Information (EPI) [1, 2] is a self contained theory to elicit physical laws from a system/process (Nature) based on a *measurement-response* framework. EPI may be construed as being a zero-sum-game (Fisher game) between an ("intelligent") observer (inhabiting a *measurement space*) and a system under observation (characterized by a *demon,* reminiscent of the Maxwell *demon*, residing in a *system space*).

EPI and its variants have been utilized to solve an impressive array of problems that include quantum electrodynamics, statistical physics, carcinogenesis, econophysics/financial mathematics, optics, cosmology (references in [2]), and, fuzzy clustering [3, 4]. The measure of uncertainty in EPI is the Fisher information (FI) measure (FIM).

A superficially similar technique is the Minimum Fisher Information (MFI) theory [5]. MFI replaces the Shannon entropy in the Maximum Entropy (MaxEnt) formulation of Jaynes [6] with the FIM. Both EPI and MFI yield MaxEnt-like results for equilibrium distributions.

Measurement-response models have been gaining much prominence of late in the design of experiments in knowledge acquisition [7]. A thorough review of the theory of measurements and experimentation may be obtained in the works of Fadeev [8] and van Trees [9].

EPI originates from the *I-theorem* $dI^{Fisher}(t)/dt \le 0 \Rightarrow I^{Fisher}_{Equilibrium}$. The *I-theorem* states that the Fisher information reaches its minimum value (at equilibrium) with increasing time. As a consequence, it is implied that the FIM is a monotonic measure of disorder. This is the information theoretic equivalent of Risken's H-theorem $dH_B(t)/dt \ge 0$ which is a statement of the second law of thermodynamics [1, 2]. Here, H_B is the Boltzmann entropy.

Conceptual analogs exist between EPI and Brillouin's theory of measurement based on the "Szilard engine", exemplified by the relation between the change in the Shannon and Boltzmann entropies: $\delta I^{Shannon} \le \delta H_B$ [10].

The Fisher game is characterized by the following competing actions by the observer and *demon*, respectively: the observer wishes to increase its knowledge of the system by maximizing the FIM in the *measurement space*. Simultaneously, the *demon* seeks to minimize the FIM available to the observer.

The *payoff of the competitive game* results *in a variational principle* that *defines the physical law that generates the observations made by the observer, as a consequence of the response of the system to the measurements*.

The derivative term in the FIM facilitates optimization problems to be replaced by systems of differential equations. The form of the FIM permits construction of a Lagrangian which yields a Schrödinger-like model as the Euler-Lagrange equation (E-Le). This enables statistical processes to be ascribed quantum mechanical (QM) connotations.

Recent studies of statistical processes employing QM theory have been conducted by Horn and Gottlieb [11] ("crisp"/hard data clustering), Venkatesan [3], Lemm *et. al*. [12] (reconstruction of oscillatory potentials from Gaussian Mixture Models (GMM's)), and, Bogdanov [13] (statistical data analysis for mutually complementary experiments in conjugate spaces).

Invariance under groups of infinitesimal transformations constitutes an important aspect in exploratory data analysis. Groups of infinitesimal

transformations are usually associated with continuous models [14].

Models that evaluate symmetry properties of difference equations have been developed within a rigorous framework [15]. These models, however, assume the existence of a continuous model by evaluating symmetries at the grid point values that are isomorphic to those possessed by the continuum being approximated.

Many scenarios encountered in data mining, pattern recognition, machine learning and allied disciplines involve discrete data. Continuum models, if extant, often represent prohibitively restrictive scenarios.

Motivated by the seminal work of T. D. Lee [16], a Discrete (lattice) Variational Complex (DVC), that evaluates symmetries and conservation laws for continuum-free models, has been formulated [17, 18]. This has been accomplished by projecting the celebrated Poincaré lemma, onto discrete (lattice) spaces.

This paper suggests a principled methodology to reconstruct probability densities from *arbitrary* time independent discrete data, employing an invariant EPI (IEPI) model, formulated with the aid of the DVC. The theory of IEPI has been exemplified by Venkatesan [3].

Data reconstruction (either time–independent or temporal) is a fundamental problem in exploratory data analysis, which finds applications in areas as diverse as quantum statistics [12, 13] to data mining [19, 20]. To exemplify the robustness and efficacy of the Fisher game, *pdf's* are reconstructed from GMM's formed from random sequences. It is demonstrated that the reconstructed *pdf*'s possess a high degree of fidelity.

A plethora of techniques have been employed in different disciplines to reconstruct *pdf*'s from empirical data. Some of the prominent reconstruction methods are splines, support vector machines, pruning decision trees, neural networks, regularization techniques [21], projection pursuit mappings, and non-parametric Bayesian methods [12], FIM models based on the theory of Maximum Likelihood Estimation (MLE) [13], amongst others. Note that representative works of the above techniques that are not referenced in this paper may be found in Ref. [12].

It is thus imperative to justify the choice of the EPI/IEPI approach to solve the inverse problem in statistics, by highlighting its qualitative distinctions. In doing so, we provide a brief overview of the physical and epistemological features of the Fisher game.

EPI facilitates the design of an experiment for knowledge acquisition, based on a measurement-response framework, by "embedding" the observer into the process of information transfer and acquisition*. This results in a self-consistent variational principle that includes both the FIM made available to the observer, and, the response of the demon in system space to measurements made by the observer.*

EPI theory and terminologies may be summarized as follows: consider an observer inhabiting a *measurement space* who carries out independent measurements on a phenomenon/process in a *system space*. The objective of the measurements is to elicit an unknown physical law that is manifested by the process in *system space*. *The source effect* constitutes the physical properties and constraints of the process in *system space*, that manifest the unknown probability law which is to be determined.

Measurements are carried through a medium/carrier depending upon the physical process being studied. Examples of carriers are photons, acoustic signals, etc… EPI requires a thorough comprehension of *the information flow route and carriers*. This comprises a two-stage process. First the transmission of the measurements from *measurement space* to *system space.* This results in the perturbation of the *source effect* in *system space*.

The next stage involves the role of the *information carrier* in relaying the effect of the perturbation of the *source effect* in *system space* (as a response to the measurements) to the *measurement space*. Note that the *information carriers* serve a dual purpose of perturbing the system space, and, relaying its state to the observer.

In EPI, t*he message* that conveys to the observer the state of the decoupled *system space* is manifested by the *bound information*. It is important to note that the term *bound information* has been borrowed from Brillouin's theory of measurements [10]. The EPI and Szilard-Brillouin versions of the *bound information* sharply differ qualitatively. Within the framework of the Szilard-Brillouin theory, the *bound information* is referred to within the context of the thermodynamic (Boltzmann) entropy of the system under measurement.

Within the context of EPI, the bound information, which is representative of the source effect, is the manifestation of the FIM in system space. The constraints and properties of the process under observation are contained in the bound information. The EPI the bound information may possess connotations and properties that transcend a thermodynamic description, depending upon the nature of the problem being analyzed.

The final stage in EPI is the *measurements of the channel*, which involves collection and estimation of the observed data by the observer in *measurement space* (treated in Section 2 of this paper). The final stage of the EPI process is a consequence of the FIM in the *measurement space* as a consequence of the perturbation of the phenomenon in *system space*.

The objective of the above sequence of events is to minimize the FIM in the *measurement space*. Qualitatively this criterion coincides with a minimum *payoff* for the observer in the Fisher game.

The above discussion exemplifies the utility of EPI to observe, measure, and estimate data in a systematic and

efficient manner. This ensures the role of EPI/IEPI as a valuable tool in knowledge acquisition and exploratory data analysis.

The present Section is concluded with an overview of the objectives and results presented in this paper. Data reconstruction is an ill-posed problem [21], requiring *a-priori* information/assumptions.

Prior attempts to reconstruct data based on QM theory involved well defined potentials (described by analytical expressions), and initially assuming that the observational data are low order moments of the observables [6, 22].

Within the terminology of QM, *observables* refers to quantities that may be observed/measured. Examples of observables are position coordinates, momentum, wave functions, amongst others. *Note that observational data do not constitute observables.*

Reconstructed wave functions (state vectors) and the corresponding *pdf*'s inherently contain noise, which distorts the reconstructed quantities from the true (unknown) values. Thus, even *pdf*'s reconstructed from a purely Gaussian data exhibit highly oscillatory behavior caused by high frequency noise. *The reconstruction noise inhabits a space orthogonal to the true state vector.*

This paper provides a principled strategy to achieve fidelity of the reconstructed *pdf*'s through the IEPI formulation, resulting in a discrete/lattice variational principle. The reconstructed wave functions (and associated *pdf*'s) are expressed in terms of an ortho-normal series expansion.

EPI/IEPI employ a particular form of the FIM, known as the Fisher channel capacity (FCC, described in Section 2 of this paper). The FCC is the trace of the FI matrix, and represents the information made available to the observer.

A significant derivation/concept described in this paper is a form of the FIM/FCC solely expressed in terms of the basis function coefficients of the series expansion. The state vectors are estimated by Chebyshev-Hermite (Hermite-Gauss) expansions. These are series expansions weighted by a Gaussian term. The objective of most contemporary studies in employing QM theory to the inverse problem in statistics is to maximizing the likelihood (or log-likelihood) [12, 13]. ***Minimizing the FCC is a unique feature of this paper.***

The ansatz describing the state estimators (wave functions/probability amplitudes-square root of the pdf, defined in Section 2) and the values of the basis function coefficients are obtained so as to self-consistently satisfy the Fisher game corollary [2] at each lattice observable. Specifically, the ansatz and basis function coefficients permit the demon to make the closing move in the Fisher game, by minimizing the FCC. This corresponds to a state of maximum uncertainty, and, is in keeping with the demon's strategy of minimizing the information made available to the observer.

Finally, we briefly justify our choice of Chebyshev-Hermite (C-H) expansions as state estimators over competing density estimators [23] (such as the Rosenblatt-Parzen windows, histogram, etc) that are efficient in their own right.

One of the fundamental concepts in QM is the QM harmonic oscillator (QMHO) [24]. The QMHO may possess a number of energy levels. C-H expansions constitute the fundamental solutions of the QMHO. The lowest energy level of the QMHO is the ground state, which corresponds to Gaussian *pdf*'s (*zeroth-order* C-H expansions).

The noise in the reconstructed state vectors is regularized/mollified by adding higher order high frequency terms of the C-H expansion. These correspond to higher order energy states (excited states) of the QMHO. In effect, the reconstruction process in this paper emulates a QMHO at each discrete lattice observable (Cartesian coordinates in this case).

2. The measurement model and the Fisher channel capacity

Consider a *measurement space* inhabited by an ("intelligent") observer who initiates *independent measurements* on a *system space* inhabited by the process under observation. The system's response perturbs the probability density in the measurement space setting off the EPI process. The *measurement space* is characterized by intrinsic *N-vectors* obeying the ***measurement estimation relation***: $y_n = \theta_n + x_n., n = 1,..., N$.

The closed system is defined by y_n (the imperfect measurement/observed data), θ_n (the parameter to be estimated, for example the mean/expectation of a datum), and x_n (a fluctuation which may be treated as a random noise). Employing the *Mach principle* [1, 2, 25], the total conditional probability for independent data is

$$p(y|\theta) = p(y - \theta) = p(x) = \prod_{n=1}^{N} p_{\boldsymbol{n}}(y_n - \theta_n) = \prod_{n=1}^{N} p_{\boldsymbol{n}}(x_n).$$

The *Mach principle* for translational families implies isotropy of the *measurement space*. Specifically, the dependency of the systems response to a measurement is independent of the position of the observer in *measurement space*. We now define the probability amplitude (wave function) as $\Psi^2(x) = p(x)$.

One of the primary tenets of EPI is the gathering of *iid* data by the observer. As a consequence of this, the FIM that is employed is the trace of the FI matrix [1, 2]. This is known as the FCC.

EPI may be applied to two separate measurement-response scenarios that facilitate the collection of independent data by the observer. The first case (Type A)

involves "N" independent measurements on a single datum, measuring its θ_n at each repetition of the experiment. Here, $\Psi^2(x) = p(x) = \frac{1}{N}\sum_n \Psi^2(x_n) = \frac{1}{N}\sum_n \Psi_n^2$. This corresponds to the case of maximum ignorance (in the quantum mechanical sense), where the θ_n are equi-probable for each measurement. The joint *pdf* is $p(x;\theta) = p(x) \bullet p(\theta) = \frac{1}{N}\sum_n \Psi^2(x_n) = \frac{1}{N}\sum_n \Psi_n^2$.

Here, $p(\theta_n) = 1/N, n = 1,...,N$ for "N" measurements. Within the continuum limit, the FCC for the Type A scenario is $I^{FCC}\left[\Psi(x)\right] = 4N\int_{-\infty}^{+\infty} dx\left(d\Psi(x)/dx\right)^2$.

The other data gathering scenario consists of a single measurement on "N" separate datum to obtain "N" different θ_n . This is referred to as a Type B scenario, which yields $I^{FCC}\left[\Psi(x)\right] = 4\sum_{n=1}^{N}\int_{-\infty}^{+\infty} dx\left(d\Psi(x)/dx\right)^2$ as the FCC.

The Type A data collection scenario is analogous to a person (the observer) sampling a liquid in a container through "N" different intakes using a single straw. On the other hand, the Type B may be described as being analogous to the observer sampling single intakes of a liquid in a container through "N" different straws, separated an arbitrary (but small) distance from each other. Here, the straw is analogous to the FCC.

Note that the FCC for both Type A and Type B scenarios described assume uniform efficiency of the measurements, and do not take into account efficiency of the measurement device/probe.

Reconstruction of *pdf*'s from discrete observational data may be posed within both the Type A and Type B scenario. Within the Type A framework, the discrete particles may be treated as a single ensemble ("macro particle").

The "macro particle" possesses a FCC that is "N" times that of a single particle. The reconstruction problem, however, fits naturally within the Type B framework. Herein, each datum is subjected to a single measurement.

It should be noted that a relation between the Type A and Type B models might sometimes be necessitated by the requirements of the given problem (see Section 4). It is noteworthy to mention that both data gathering models are statistically equivalent.

Given y_n , an estimate $\hat{\theta}(y_n)$ of θ is obtained. The Cramer-Rao bound (C-Rb) is $e^2 I(\theta) \geq 1$, where $e^2 = \left\langle \left(\hat{\theta}(y) - \theta\right)^2 \right\rangle$ is the mean square error (MSE)/variance of the estimator. Note that $\langle \bullet \rangle$ denotes the expectation value.

The C-Rb governs the accuracy with which a measurement can be made. Thus, it is a measure of the inability of the observer to acquire knowledge. Such an inability can be quantified as the numerical *uncertainty* in knowledge of a parameter, as in an "*uncertainty principle*". The celebrated Heisenberg uncertainty principle [24] is one such example.

3. EPI within the DVC

The constraints and properties of the process in *system space* constitute the *bound information* $J[\Psi(x)]$ [1,2]. EPI and MFI are defined as a set of axioms:

Table 1: EPI and MFI axioms

MODEL	AXIOM 1	AXIOM 2
EPI	$\frac{\delta}{\delta\Psi(x)}\left\{I^{FCC}\left[\Psi(x)\right] - J\left[\Psi(x)\right]\right\} = \delta K^{EPI}\left[\Psi(x)\right] = 0.$	$I^{FCC}[\bullet] - \kappa J[\bullet] =$ $0 < \kappa \leq 1.$
MFI	$\frac{\delta}{\delta\Psi(x)}\left\{I^{FCC}\left[\Psi(x)\right] + \underbrace{-\left\langle \sum_{n=0}^{N} \lambda_n A_n(x) \right\rangle}_{data}\right\} = 0$	

Simultaneous solution of EPI Axioms 1 & 2 yields physical laws and the efficiency parameter κ. This is accomplished by the simultaneous determination of the bound information and the probability amplitude. EPI Axiom 2, a consequence of the *I-theorem*, describes the efficiency of the information transfer $J[\bullet] \rightarrow I[\bullet]$.

Conjugate measurement and system spaces related through a unitary transform (UT) constitute a sufficient condition for both the existence of EPI solutions, and, maximally efficient information transfer. IEPI relates EPI solutions to variational symmetries through an invariance preserving amalgamation of EPI and MFI.

By definition [14], the symmetries contained by a given Lagrangian are referred to as variational symmetries. Variational symmetries form a subset of the symmetries of the E-Le, obtained by variational extremization of a given Lagrangian.

The MFI data constraint terms are specified to manifest the EPI *bound information* in *measurement space*. The *system space* is treated as momentum space. The present study considers the case where the number of observational data points is equal to the number of constraint terms in the IEPI variational principle. Given observational data $d=\{d_1,...,d_N\}$, which are expectation values of the discrete observables such that $d_n=\langle A(x_n)\rangle$. Here, $A(x_n)$ forms the *pseudo-potential* in the time independent Schrödinger-like model (TISM) formed by solution of the EPI process. The grid size for the observables is taken to be non-uniform: $\Delta_n = x_{n+1}-x_n \neq x_{n+2}-x_{n+1}=\Delta_{n+1}$.

The arbitrary nature of the grid spacing for the observables permits specifying, as a matter of convenience, the distance between the discrete observables Δ_n to be equal to the distance between the discrete observational data $\delta_n = d_{n+1}-d_n$ (not to be confused with the lattice variational derivative in (2), below). Precaution need be taken in order that the values of the discrete observables x_n do NOT coincide with the values of the corresponding observational data d_n.

The discrete Cartesian coordinate is $x_n = n\Delta_n$, where n is the lattice index of the discrete observables. C*anonical quantization* [26] yields the commutation relation $\left[x_n,\mu_m=-i\hbar\partial/\partial x_n=-i\hbar\{S-id\}/\Delta_n\right]=i\hbar$, $\hbar$ is Planck's constant, and, μ_m and m are the lattice coordinate and index of the *system space*. Non-differentiability of the lattice space requires discrete derivatives to be expressed as shift maps [17, 18] $\partial/\partial x_n=\{S-id\}/\Delta_n$, id is the identity operator. The necessity of shift maps stems from the fact that the DVC does not obey the Leibniz rule of differentiation in lattice space. The IEPI Lagrangian functional is:

$$K_D^{IEPI}=\sum_{n=1}^{N}\left[I_D^{FCC}(\Psi_n)-\left\langle\lambda_0-\sum_{n=1}^{N}\lambda_n A_n(x_n)\right\rangle\right]$$
$$=\sum_{n=1}^{N}\left[I_D^{FCC}(\Psi_n)-J\left[\Psi_n=\Psi(x_n)\right]\right],\text{where,} \quad (1)$$
$$UJ\left[\Psi_n\right]=J\left[\Phi_m=\Phi(\mu_m)\right],U=e^{i\varepsilon G_{UT}}$$

Here $I_D^{FCC}(\Psi_n)$ is the discrete FIM, and Φ_m is the discrete conjugate amplitude. Also, U is the unitary operator, ε the group parameter of infinitesimal UT, and, $G_{UT}=-i\hbar\partial/\partial x_n=-i\hbar(S-1)/\Delta_n$ is the unitary group. The FCC in mutually conjugate spaces is related as $I_D^{FCC}(\Psi_n)=4\sum_{n=1}^{N}\left[\frac{\{S-id\}\Psi_n}{\Delta_n}\right]^2=\frac{4}{\hbar^2}\sum_m\left[\mu_m^2\Phi_m^2\right]=\frac{4}{\hbar^2}\left\langle\mu_m^2\right\rangle, Ux_nU^*=\mu_m,$ and, the adjoint UT is $U^*=e^{-i\varepsilon G_{UT}}$. The IEPI lattice variational, and, zero-condition are respectively:

$$\frac{\delta K_D^{IEPI}}{\delta_n}=\sum_l S^{-l}\left(\frac{\partial K_D^{IEPI}}{\partial\Psi_{n+l}}\right),and,$$
$$K_D^{IEPI}=4\sum_{n=1}^{N}\left[I_D^{FCC}(\Psi_n)-J\left[\Psi_n\right]\right]. \quad (2)$$

It may be noted that groups of infinitesimal unitary transformations provide an elegant alternative to the application of Fourier transforms (followed by the enforcement of the Parseval theorem) [1, 2], when studying mutually conjugate spaces. The self-adjoint [14, 17] E-Le's are the lattice 1-D TISM:

$$\sum_{n=1}^{N}\left\{\frac{-4(\Psi_{n+1}-2\Psi_n+\Psi_{n-1})}{\Delta_n^2}+\lambda_n A_n(x_n)\Psi_n+\lambda_{on}\Psi_n\right\}. \quad (3)$$

Note that in 1-D, the wave functions may be approximated as being real quantities (without an imaginary component).

A growing trend in contemporary physics is to select models possessing the simplest symmetry structure and not the simplest model. This is the symmetry version of Occam's principle [27]. All forms of the TISM constitute Ricatti equations, with the scaling invariance $\Psi_n\partial/\partial\Psi_n$ constituting the simplest and most prominent symmetry group.

The form of the FIM in (2) does not yield the scaling invariance possessed by (3). To rectify this discrepancy, an equivalent IEPI Lagrangian is reconstructed from (3) via *homotopy* [14, 17] (the inverse problem of the calculus of variations):

$$K_D^{IEPI}=\int_0^1 d\beta P[\beta\Psi]\Psi_n=$$
$$\sum_{n=1}^{N}\left\{\Psi_n\left[\frac{-4\left[\Psi_{n+1}-2\Psi_n+\Psi_{n-1}\right]}{\Delta_n^2}\right]+\left(\lambda_{on}-\lambda_n A_n(x_n)\right)\Psi_n^2\right\} \quad (4)$$

The *effective FIM/FCC* is of the form $\tilde{I}_D^{FCC} = -4\sum_{n=1}^{N} \frac{\Psi_n\left[\{\Psi_{n+1} - 2\Psi_n + \Psi_{n-1}\}\right]}{\Delta_n^2}$.

Note that the form of $\tilde{I}_D^F$ is not unique. Further, the choice of an optimal form for a discrete model is a topic of much contemporary interest [16, 17]. The discrete form of the effective FCC has been chosen to resemble the continuous FIM subjected to a single integration-by-parts, followed by discretization. Expanding (3) yields the system:

$$\begin{aligned}
&\frac{-4\left(\Psi_2 - 2\Psi_1 + \Psi_o\right)}{\Delta_1^2} + \lambda_1 A_1(x_1)\Psi_1 = \lambda_{o1}\Psi_1,\\
&\frac{-4\left(\Psi_3 - 2\Psi_2 + \Psi_1\right)}{\Delta_2^2} + \lambda_2 A_2(x_2)\Psi_2 = \lambda_{o2}\Psi_2,\\
&\bullet\\
&\frac{-4\left(\Psi_{N+1} - 2\Psi_N + \Psi_{N-1}\right)}{\Delta_N^2} + \lambda_N A_N(x_N)\Psi_N = \lambda_{oN}\Psi_N.
\end{aligned} \tag{5}$$

The discrete system (3) and (5) are manifestations of the eigensystem $H\Psi_n = \lambda_{on}\Psi_n; n = 1,...,N$. The Hamiltonian operator is $H = \frac{-4\left(\Psi_n - 2\Psi_n + \Psi_{n-1}\right)}{\Delta_n^2} + \lambda_n A_n(x_n)\Psi_n$. ***The Lagrange multipliers (LM's)*** $\lambda_{on} = \hbar\omega\left(k_n + \frac{1}{2}\right)$; $k_n = 0,1,...; n = 1,...,N$ ***denote the energy eigenvalues of a QMHO.*** Here ω is the natural frequency of the oscillator.

Data reconstruction involves unknown forms of the potential and hence the Hamiltonian H. The energy eigenvalues are obtained in the direct problem by solving the eigenvalue problem $\left|H - \lambda_o\right| = 0$. Here, $\lambda_o = \sum_{n=1}^{N} \lambda_{on}$ is the total energy, described by the normalization LM.

Here, $V(x_n) = \lambda_n A_n(x_n)\Psi_n^2; n = 1,...,N$ denotes the *pseudo-potential*, which is inferred from the observational data $\{d_n\}_{n=1}^N$, by specifying $\langle A_n = A_n(x_n)\rangle = d_n$. Without loss of generality, ***"zero-mean" operators*** of the form $(A_n - d_n)$ are introduced into (3) [22]. The potential is transformed as $V_n^* = V_n - \sum_{n=1}^{N} \lambda_n d_n \Rightarrow \langle V_n^*\rangle = 0$. Thus, (3) yields:

$$\tilde{I}_D^{FCC} = 4\langle E^*\rangle = 4\sum_{n=1}^{N} (\lambda_{on} - \lambda_n d_n)\Psi_n^2. \tag{6}$$

Here, (6) represents a *fiduciary conversion* of the total energy into the kinetic energy. *More importantly, it explicitly introduces the FCC into the TISM. In QM, the virial theorem states that the expectation (average) of the potential energy equals the expectation of the kinetic energy. Within the context of this paper, the data FIM is the expectation of the kinetic energy and the data terms and the concomitant constraints constitute the potential energy.* From canonical quantization, the *virial theorem* is ($\hbar^2/2m_e = 1$, m_e the particle mass):

$$\begin{aligned}
&\frac{4\sum_m \mu_m^2 \Phi_m^2}{\hbar^2} = \frac{\langle \mu_m^2\rangle}{2m_e} = \left[x_n \frac{\partial V(x_n)}{\partial x_n}\right] = \left\langle \frac{x_n[S - id]V(x_n)}{\Delta_n}\right\rangle\\
&= \langle n[S - id]V(x_n)\rangle = \sum_{n=1}^{N} n\lambda_n\delta_n = \tilde{I}_D^{FCC} = I_D^{HO}.
\end{aligned} \tag{7}$$

The above relation serves as an invaluable tool within the framework of the present model. Specifically, the *virial theorem* for a QMHO with eigenstate k_n is:

$$\sum_{n=1}^{N} n\lambda_n\delta_n = I_D^{HO} = \sum_{n=1}^{N} \frac{\hbar\omega}{2}\left(k_n + \frac{1}{2}\right). \tag{8}$$

The number of terms of the C-H expansion equals the eigenstate of the QMHO that is *emulated* at each lattice observable. This result will be further elucidated upon and utilized in the following Sections. This Section is concluded by expressing the FI Legendre transform structure (LTS) [28] $\partial I^F / \partial\langle A_n\rangle = \lambda_n$ within the DVC framework as:

$$\frac{\partial x_n}{\partial\langle A_n\rangle}\frac{\partial \tilde{I}_D^{FCC}(x_n)}{\partial x_n} = \frac{\tilde{I}_D^{FCC}(x_{n+1}) - \tilde{I}_D^{FCC}(x_n)}{(d_{n+1} - d_n) = \delta_n} = \lambda_n. \tag{9}$$

The LTS plays an important role in assigning a thermodynamic representation to the FIM. Note that the independence of the lattice observables permits the FI LTS to be evaluated piecewise along the lattice (as was the case with the FCC, (6), and, (7)).

4. The inverse problem in statistics

The wave function/probability amplitude is ascribed a form given by the following *ansatz*:

$$\Psi(x)=\sum_{n=1}^{N}\Psi(x_n)=\sum_{n=1}^{N}\sum_{i=0}^{s-1}c_i\phi_i(x_n)=\sum_{n=1}^{N}\sum_{i=0}^{s-1}c_i\phi_i(x_n);$$

$$c_o=\sqrt{1-c_1^2-...-c_{s-1}^2} \qquad (10)$$

Here, the eigenvectors $\phi_i(\bullet)'s$ are taken as real, which may be evaluated using a variety of density estimators such as orthonormal series expansions (C-H polynomials), histograms, etc. Note that $c_i, i=1,...,s-1$ are independent. *The zeroth-order* basis coefficient c_o in the series expansion corresponds to Gaussian *pdf*, and relates to the basis function coefficients for higher order harmonics through the normalization condition in (10).

The eigenvectors of the orthonormal series corresponding to the QMHO. Here c_o corresponds to the ground state, and $c_i, i=1,...,s-1$ denote deviations from ground state. ***Note that the ground state is characterized by the eigenvalue*** $\lambda_{oG}=\hbar\omega/2$. Substituting (2) into (1) and evoking the discrete ortho-normality condition $\sum_{n,m}\phi_i(x_n)\phi_i(x_m)=\delta_{nm}/\Delta_n$ [29, 30], we obtain:

$$\tilde{I}_D^{FCC}=8\sum_{n=1}^{N}\frac{\Psi_n^2}{\Delta_n^3}=8\sum_{n=1}^{N}\sum_{i=0}^{s-1}\frac{c_i^2}{\Delta_n^3}. \qquad (11)$$

It is our objective to re-parameterize the wave functions $\Psi(x_n)$ in terms of the basis function coefficients***. It is important to note that owing to the independence of the lattice observables (Cartesian coordinates), the FCC, (6) and (7) may be evaluated piecewise along the lattice space. The sum total of the FCC may be obtained from the additive property of the FIM/FCC:*** $I=\sum_n I_n=\sum_n \tilde{I}_n^{FCC}$.

The EPI measurement estimation framework does not readily permit a re-parameterization of the FCC in terms of the basis function coefficients. The definition of the Fisher information matrix (FIM) for *iid* data to the multi-parameter case [31] is:

$$I_{Coeff}^{FCC}=\sum_{n=1}^{N}\sum_{i=0}^{s-1}\frac{1}{P(x_n|c)}\left(\frac{\partial P(x_n|c)}{\partial c_i}\right)^2, \qquad (12)$$

$c=(c_o,c_1,...,c_{s-1})$.

Note that we have used conventional derivatives with respect to the coefficients instead of shift maps, because basis function coefficients are not regarded as variables in the discrete/lattice theory [17, 18]. Thus, we have the FCC of the form:

$$I_{Coeff}^{FCC}=4\sum_{n=1}^{N}\sum_{i=0}^{s-1}\left(\frac{\partial\Psi(x_n,c)}{\partial c_i}\right)^2. \qquad (13)$$

Substituting (10) into (13), and evoking the discrete ortho-normality condition: $\sum_{n,m}\phi_i(x_n)\phi_i(x_m)=\delta_{nm}/\Delta_n$, we obtain the FCC for the lattice observable x_n to be a diagonal matrix of dimension $(s-1)\bullet(s-1)$ with elements:

$$I_{ii}^{FCC}(x_n)=\frac{4}{\Delta_n}\left[1+c_{ni}^2/c_{no}^2\right], i=1,...,s-1,, \qquad (14)$$

In the asymptotic limit, the variance has elements $e^2(x_n)=\frac{\Delta_n}{4}\left[1-c_i^2\right]; i=1,...,s-1.$ ***The state estimators are solely expressed in terms of the basis coefficients***. *It is imperative to self consistently obtain a value for the coefficient c_o, and, as a result relate the two definitions of the FCC defined by (1) and (5).* Owing to the *iid* nature of the discrete observables, we evoke the additive property of the FIM $I=\sum_n I_n$, and obtain:

$$I_{Coeff}^{FCC}=\sum_{n=1}^{N}Tr\frac{4}{\Delta_n}\begin{bmatrix}1+\frac{c_{n1}^2}{c_{no}^2} & 0 & 0\\ \bullet & \bullet & \bullet\\ 0 & 0 & 1+\frac{c_{n(s-1)}^2}{c_{no}^2}\end{bmatrix} \qquad (15)$$

It is required to obtain the FCC for a single observable x_n on the lattice with the aid of (15). The *virial theorem* (8) for the QMHO at x_n yields:

$$n\lambda_n\delta_n=I_D^{HO}(x_n)=\sum_{\varepsilon\geq 0}\frac{\hbar\omega}{2}\left(k_{n,\varepsilon}+\frac{1}{2}\right). \qquad (16)$$

Here, the eigenstate $k_n=\sum_{\varepsilon\geq 0}k_{n,\varepsilon}$ is the sum of number of energy levels corresponding to the terms in (15). Eq. (16) may be expressed in matrix form as:

$$I_D^{HO}(x_n) = Tr\frac{\hbar\omega}{2}\begin{bmatrix} k_{n,1}+\frac{1}{2} & 0 & 0 \\ \bullet & \bullet & \bullet \\ 0 & 0 & k_{n,s-1}+\frac{1}{2} \end{bmatrix} \quad (17)$$

Comparing (15) and (17) yields $\hbar\omega = \frac{8}{\Delta_n}$ at each discrete observable. The eigenstate (energy level number) $k_{n,\varepsilon}, 0 < \varepsilon \le s-1$ corresponds to the matrix elements $I_{Coeff}^{FCC}(x_n)$ in (15) at each discrete observable. Here, $k_{n,\varepsilon}$ is representative of the number of energy levels above the ground-state (which corresponds to the Gaussian *pdf*), and represent the deviation from Gaussianity by the addition of high frequency smoothening terms.

We thus relate the number of energy levels to the number of terms in the basis function coefficients. Equating the energy levels of the QMHO to the FCC at x_n, we obtain:

$$\frac{4}{\Delta_n}\left(1+\frac{c_{ni}^2}{c_{no}^2}\right) = \frac{\hbar\omega}{2}\left(k_{n,\varepsilon}(=i)+\frac{1}{2}\right), i = 1,...,s-1;. \frac{\hbar\omega}{2} = \frac{4}{\Delta_n} \quad (18)$$

The above procedure elegantly and intuitively relates the basis function coefficients corresponding to the excited energy levels to the ground state coefficient c_{no}^2 as: $c_{n1}^2 = c_{no}^2/2, c_{n2}^2 = 3c_{no}^2/2, ...$

The truncation of the harmonics is dictated by the normalization condition in (10). Since the basis function coefficients are expressed in terms of c_{no}, it is imperative to obtain an expression for the *zeroth-order coefficient*. Note that to obtain c_{no} we consider the purely Gaussian case. Substituting the Gaussian component of (10) into (10), and evoking the discrete ortho-normality condition, we obtain:

$$\tilde{I}_D^{FCC}(x_n) = \frac{8c_{no}^2}{\Delta_n^3} \quad (19)$$

At this stage we convert (12) into the Type A "macro-particle" model for the single parameter c_O. We evoke the argument that for "N" *iid* data samples, given a Type A scenario (see Section 2), the total FIM is "N" times the individual FIM [31]. Thus, (12) becomes:

$$I_{Coeff}^{FCC} = 4N\sum_{n=1}^{N}\left(\frac{\partial\Psi(x_n, c_O)}{\partial c_O}\right)^2 \quad (20)$$

Substituting (10) into (19), and evoking the discrete ortho-normality condition: $\sum_{n,m}\phi_o(x_n)\phi_o(x_m) = \delta_{nm}/\Delta_n$, we obtain:

$$I_{D,Gaussian}^{FCC} = \frac{4N}{\Delta_n}. \quad (21)$$

Equating (19) and (21) yields:

$$c_{no}^2 = \frac{N\Delta_n^2}{2}. \quad (22)$$

Thus, at every lattice observable, we obtain the set of coefficients given by (22) and its multiples for $1 \le n \le N$. ***The ansatz for the state estimator (10) guarantees that the Fisher "game corollary" is always satisfied.*** Taking the example for a single data x_n, the FCC for $i = 0,1$ in (10) is:

$$\tilde{I}_D^{FCC}(x_n) = \frac{8c_{no}^2\left[1+c_{n1}^2/c_{no}^2\right]}{\Delta_n^3}, c_{no}^2 = \left(1-c_{n1}^2-...\right) \quad (23)$$

It is trivial to establish $\frac{d\tilde{I}_D^{FCC}(x_n)}{dc_1} = 0.$ This is the Fisher "game corollary". The result (23) holds good for all orders of expansion of (10).

To obtain the entire set of probability amplitudes, a procedure similar to that presented in [30] for the case of the inverse Schrödinger equation is adopted. It may be noted that the ortho-normality condition may also be expressed in terms of the energy levels as: $\sum_{\varepsilon,\gamma}\phi_i(x_n, \lambda_{on,\varepsilon})\phi_i(x_n, \lambda_{on,\gamma}) = \delta_{\varepsilon\lambda}/\Delta_n$. We obtain a set of recurrence relations from (6) and (3) respectively:

$$\sum_{i=0}^{s-1}\lambda_n d_n c_{ni}^2 = \Delta_n\sum_{i=0}^{s-1}\underbrace{\sum_{\varepsilon\ge 0}\lambda_{on,\varepsilon}\psi^2(\lambda_{on,\varepsilon}, x_n)}_{\lambda_{on}\psi_n^2} - \sum_{i=0}^{s-1}\frac{8c_{ni}^2}{\Delta_n^2}, \quad (24)$$

$$\psi_{n+1}(\lambda_{on+1,\varepsilon}) = \left\{\frac{\Delta_n^2}{4}\left[\lambda_n A_n - \lambda_{on}\right]+2\right\}\psi(\lambda_{on,\varepsilon}, x_n) - \psi_{n-1}(\lambda_{on,\varepsilon}), \quad (25)$$

Two points need be noted. First, ortho-normality of the first term in the RHS of (24) is to be evoked, upon determining the total number of high frequency terms. Next, in (25), $\langle A_n \rangle = \sum_{n=1}^{N} A_n \psi_n^2 = d_n$.

5. Overview of the solution strategy

Procedures for synthetic numerical experiment to test the efficacy of the Fisher game are presented in Figure 1:

---

1. **Algorithm**: 1-D data reconstruction using a Fisher game
2. **Known parameters**: *iid* observational data $\{d_n\}_{n=1}^{N}$ generated from MATLAB® random sequence generator $randn(\bullet)$ corresponding to a GMM with specified mixing weights, means and variances, the distance $\delta_n, i = 1,...,N$ between consecutive observational data points.
3. **Unknown parameters**: $\Delta_n, \lambda_n, \lambda_{on}, \Psi_n$
4. **Outline of solution procedure**:
   - Divide the axis of observables [a,b] into "N+1" different segments, with length $\Delta_n = \delta_n$. Set $\phi(a) = \phi(b) = 0$.
   - Evaluate coefficients c_{no} from (22).
   - Evaluate the higher order basis function coefficients from (18), which exactly/nearly satisfy the normalization condition in (10).
   - Evaluate the Lagrange multipliers at each discrete observable using (9). For initialization purposes, (16) may be employed. Solve the recurrence relations (24) and (25) with the aid of (6) and (16).
   - Utilize a steepest descent algorithm at the end of each complete iteration epoch to correct the values of the LM's $R\{\lambda_n\}_{n=1}^{N} = \lambda_{on} - \sum_{n=1}^{N} \lambda_n \left[\langle \psi_n | A_n | \psi_n \rangle / \langle \psi_n | \psi_n \rangle - d_n \right]$. Iterate till convergence.

Figure 1: High level description of IEPI data reconstruction scheme in 1-D

6. Numerical results and conclusions

A two-component GMM is created from components possessing unit variance, and mixing weights $\pi_1 = 0.7$ and $\pi_2 = 0.3$, using the the MATLAB® random sequence generator $randn(\bullet)$. The means are at the origin and at $x = 3.0$, respectively. A sample of 200 data points representing a two-component GMM is selected.

Figure 2 depicts the reconstructed *pdf* interposed against the real/actual *pdf* (obtained through a Monte Carlo simulation). ***The reconstructed pdf's exhibit a high degree of fidelity***.

The synthetic case study presented in Figure 2 provides an interesting qualitative insight into the IEPI reconstruction strategy. Specifically, the IEPI model reconstructed *pdf*'s that converged to the actual values, utilizing a reduced number of basis function coefficients. This comparison is done vis-à-vis a reconstruction process using the MLE theory [13]. Further, the energy term in the E-Le (6) contributes a regularization-like effect.

Extensive simulation conclusively demonstrates that all FIM models exhibit enhanced reconstruction error for reduced number of basis coefficients, within the ambit of bearing in mind the constraints on the number of basis function coefficients highlighted by the normalization condition in (10). Additional higher-order high frequency terms do not adversely affect the solution, and instead, contribute a smoothening/mollifying effect. Within the framework of the present model, it would require the exact (or near exact) satisfaction of the normalization condition in (10).

Two extensions of the IEPI model are underway. The first is motivated by the fact that the synthetic numerical experiment described in this paper assumes the observational data to manifest the *bound information*. The random sequence generator produces *iid* data. This is tantamount to assuming that the response of the *source effec*t to independent measurements is without correlations. In actuality, this assumption is not generally true, and the observed data often possesses correlations. Correlations prohibit the FCC to be expressed in terms of statistically independent observables. A principled Fisher game formulation guaranteeing statistical independence observables has been developed, utilizing analyses commonly employed in Independent Component Analysis (ICA) [32]. The observed data is subjected to a pre-processing stage of whitening (through application of a linear filter/ PCA) followed by a Givens-Jordan permutation (a unitary transform). The pre-processing stage minimizes the mutual information between the observed data in the *measurement space* y_n, ensuring that the FCC comprises of uncorrelated independent components. The Fisher game is played, and the reconstructed *pdf*'s obtained. A final post-processing stage is involves the application of an inverse linear filter/PCA to obtain the "true" reconstructed *pdf*'s which include the effect of correlations. The second extension is

temporal sequence reconstruction, accomplished using the Ehrenfest theorem [24, 29].

7. References

1. Frieden, B. R. *Physics from Fisher Information*, Cambridge University Press, Cambridge, 1999.
2. Frieden, B. R., *Science from Fisher Information*, Cambridge University Press, Cambridge, 2004.
3. R. C. Venkatesan**,** "Invariant Extreme Physical Information and Fuzzy Clustering*", Proc. SPIE Symposium on Defense & Security, Intelligent Computing: Theory and Applications II*, Priddy, K. L. (ed), Volume 5421, pp. 48-57, Orlando, FL, 2004. (To Appear, *Phys. Rev. E*, 2005).
4. M. Ménard and M. Eboueya, "Extreme Physical Information and Objective Functions in Fuzzy Clustering", *Fuzzy Sets and Systems*, **128**, pp. 285-303, 2002.
5. Huber, P. J. *Robust Statistics*, Wiley, New York, 1981.
6. E. T. Jaynes, "Information Theory and Statistical Mechanics", *Phys. Rev*, **106**, pp. 620, 1957; "Information Theory and Statistical Mechanics-II", *Phys. Rev*, **108**, pp. 171, 1957.
7. L. Paninski, "Information-theoretic design of experiments", *Advances in Neural Information Processing Systems-2003 (NIPS)*, **16**, Thrun, S. (ed), MIT Press, Cambridge, MA, pp. 1319-1326, 2004.
8. Fedorov, V. V. *Theory of Optimal Experiments*, Academic Press, New York, 1972
9. van Trees, H. L., *Detection, Estimation, and Modulation Theory*, Part I, Wiley, New York, 1968.
10. Brillouin, L., *Science and Information Theory*, Academic Press, New York, 1956.
11. D. Horn and A. Gottlieb, "Algorithm for Data Clustering in Pattern Recognition Problems based on Quantum Mechanics", *Phys. Rev. Lett*., **88**, 1, pp. 18702(1-4), 2002.
12. J. C. Lemm, J. Uhlig, and, A. Weiguny, "A Bayesian Approach to Inverse Quantum Statistics", *Phys. Rev. Lett*., **84**, pp. 2008, 2000.
13. Yu. I. Bogdanov, "Fundamental Notions for Classical and Quantum Statistics: A Root Approach", *Optics and Spectroscopy*, **96**, 5, pp. 668-678, 2004.
14. Olver, P. J., *Application of Lie Groups to Differential Equations*, Springer-Verlag, New York, 1993.
15. V. A. Dorodnitsyn, "Symmetry of Finite-Difference Equations", In: Ibragimov N. B. (ed.) *CRC Handbook of Lie Group Analysis of Differential Equations*, vol. 1, CRC Press, Boca Raton, pp. 365-403, 1994.
16. T. D. Lee, "Difference Equations and Conservation Laws", *J. Stat. Phys*., **46**, pp. 843-860, 1987.
17. P. E. Hydon and E. L. Mansfield , "A Variational Complex for Difference Equations", *Found. Comp. Math*., **2**, pp. 187-217, 2004.
18. P. E. Hydon , *Proc. Roy. Soc. Lond. A*, **454**, pp. 2835-2855, 2000.
19. R. Agrawal and R. Srikant, "Privacy-Preserving Data Mining", *Proc. ACM SIGMOD Conference on Management of Data 2000*, pp. 439-450, 2000.
20. C. Faloutsos, H. V. Jagdish, and, N. D. Sidiropoulos, "*Recovering* Information from Summary Data*", Institute of Systems Research, University of Maryland Technical Report: TR-97-7*, College Park, MD, 1997.
21. Tikhonov, A. N. and V. A. Arsenin, *Solutions of Ill-Posed Problems*, W. H. Winston (Wiley), Washington D. C., 1977.
22. M.Casas, A. Plastino, and, A. Puente, "Fisher Information and the Inference of Wave Functions for Systems of Unknown Hamiltonian", *Phys Lett. A*, **248,** pp. 161-166, 1998.
23. Bishop, C. M., *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, 2000.
24. Landau, L. D. and E. M. Lifshitz, *Quantum Mechanics (Non-Relativistic Theory)*, Pergamon Press, Oxford, 1991.
25. Cover, T and J. Thomas, *Elements of Information Theory*, Wiley, New York, 1991.
26. Sakurai, J. J., *Advanced Quantum Mechanics*, Addison-Wesley, Reading, MA, 1967.
27. R. Trejo, V. Kreinovich, and L. Longpré, "Choosing a Physical Model: Why Symmetries?", *Bulletin of the EATCS*, **70**, pp. 159-161, 2000
28. B. R. Frieden, A. Plastino, A. R. Plastino, and, B. H. Soffer, "Fisher-Based Thermodynamics: Its Legendre Transform and Concavity properties", *Phys. Rev. E*, **60**, pp.48-53, 1999.
29. Peebles, P. J. E., *Quantum Mechanics*, Princeton University Press, Princeton, NJ, 1992.
30. Zahhariev, B. N. and A. A. Suzko, *Direct and Inverse problems – Potentials in Quantum Scattering*, Springer-Verlag, Berlin, 1990.
31. Cramer, H. *Mathematical Methods of Statistics*, Princeton University Press, Princeton, NJ, 1946
32. P. Comon, "Independent Component Analysis – A New Concept ?", *Signal Processing*, **36**, pp. 287-314, 1994.

Acknowledgements

Gratitude is felt towards B. R. Frieden, K. Rose, and, Yu. Bogdanov. *This work was supported by MSR contract CSM-DI&M-101107-2003.*

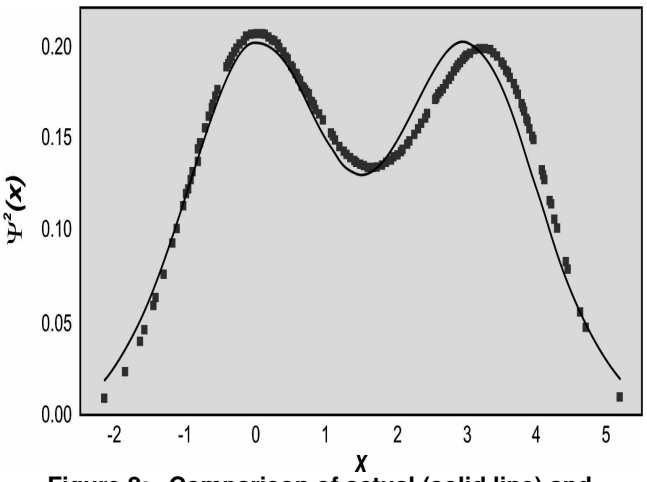


Figure 2: Comparison of actual (solid line) and reconstructed (dotted line representing reconstructed data) *pdf*'s for a two-component GMM

Data Mining Operators

Anita Wasilewska[(1)], E. Menasalvas[(2)]
[(1)] Department of Computer Science,
State University of New York,
Stony Brook, NY, USA .
anita@cs.sunysb.edu
[(2)] Facultad de Informatica.
Universidad Politecnica de Madrid. Spain.
emenasalvas@fi.upm.es

Abstract

We present here an abstract model in which Data Mining algorithms are defined as generalization operators. We use our framework to show that only three generalizations operators: classification, clustering, and association operator are needed to express all Data Mining algorithms for classification, clustering, and association, respectively. Moreover, we formally prove that classification, clustering and association analysis fall into three different generalization categories.

1 Introduction

We build models in order to be able to address formally intuitively expressed notions, or answer intuitively formulated questions. We know, for example, that there are several classification algorithms and hundreds of implemented classifiers. We talk about them, improve them and we compare them usually by the quality of their implementations. A natural questions arise: why very different algorithms are all called classification algorithms? What do they have in common? How do they differ from other algorithms?

We hence need to build models to define what is a classification algorithm, what existing classification algorithms have in common, what is a difference (if any) between a classification algorithm and for example an association analysis algorithm.

In the model we present here each classification algorithm is represented by an operator that is a generalization operator. Moreover, we show that all classification operators belong to one category, distinctive with for example categories of association, or clustering operators.

We usually view Data Mining results and present them to the user in their descriptive form as it is the most natural form of communication. But the Data Mining process is deeply semantical in its nature. The algorithms process records (semantics) finding similarities which are then often presented in a descriptive i.e. syntactic form. Our model is a semantical one. Nevertheless it supports the extraction of syntactical information, at any level of generalization. We will address the semantics-syntax duality in a separate paper.

2 Generalization Model

Data Mining, as it is commonly said, is a process of generalization. In order to model this process we have to define what does it mean that one stage of data mining process is more general then the other. The main idea behind our definition of Generalization Model is that generalization consists in putting objects (records) in sets of objects.

From syntactical point of view generalization consists also of building descriptions (in terms of attribute, values of attributes pairs) of these sets of objects, with some extra parameters, if needed. Our Generalization Model is semantic in nature, but, as we mentioned before, it also incorporates the syntactic information to be extracted, when (and if) needed.

The model presented here generalizes many ideas developed during years of investigations. First they appeared as a part of development of Rough Sets Theory (to include only few recent publications [13],[14], [16], [17], [18]), [5], [14], [7], [6]); then in building Rough Sets inspired foundations of information generalization ([2], [9], [10],[8]).

Definition 2.1 A Generalization Model *is a system*

$$\mathcal{GM} = (U,\ \mathcal{K},\ \mathcal{G},\ \preceq)$$

where

$U \neq \emptyset$ *is the* **universe**,

$\mathcal{K} \neq \emptyset$ *is the set of* **generalization states**,

$\prec\ \subseteq\ \mathcal{K} \times \mathcal{K}$ *is* **a generalization relation**;

We assume that the relation $\preceq$ is transitive.

$\mathcal{G} \neq \emptyset$ *is the set of* **generalizations operators** *such that for every $G \in \mathcal{G}$, for every $K, K' \in \mathcal{K}$,*

$$G(K) = K' \quad if\ and\ only\ if \quad K \preceq K'.$$

We define all components of the model in the following subsections 2.1, 2.2, 2.3, and 2.4.

2.1 Knowledge Generalization System

The *knowledge generalization system* is an extension of the notion of an information system. The information system was introduced in [12] as a database model. The information system represents the relational table with key attribute acting as object attribute and is defined as follows.

Definition 2.2 *Pawlak's* **Information System** *is a system* $I = (U, A, V_A, f)$, *where* $U \neq \emptyset$ *is called a set of* **objects**, $A \neq \emptyset$, $V_A \neq \emptyset$ *are called the set of* **attributes** *and* **values** *of of attributes, respectively,* f *is called an* **information** *function and* $f : U \times A \longrightarrow V_A$

In the data analysis, preprocessing and data mining we start the process with the input data. We assume here that they are represented in a format of information system table. We hence define the lowest level of information generalization as the relational table. The meaning of the intermediate and final results are considered to be of a higher level of generalization. We represent those levels of generalization by a sets of objects of the given (data mining) universe U, as in [2], [9].

This approach follows the granular view of the data mining and is formalized within a notion of knowledge generalization system, defined as follows.

Definition 2.3 A knowledge generalization system *based on the information system* $I = (U, A, V_A, f)$ *is a system*

$$K_I = (\mathcal{P}(U), A, E, V_A, V_E, g)$$

where

E *is a finite set of* **knowledge attributes** *(k-attributes) such that* $A \cap E = \emptyset$.

V_E *is a finite set of* **values of k- attributes**.

g *is a partial function called* **knowledge information function***(k-function)*

$$g : \mathcal{P}(U) \times (A \cup E) \longrightarrow (V_A \cup V_E)$$

such that

(i) $g \mid (\bigcup_{x \in U} \{x\} \times A) = f$

(ii) $\forall_{S \in \mathcal{P}(U)} \forall_{a \in A}((S, a) \in dom(g) \Rightarrow g(S, a) \in V_A)$

(iii) $\forall_{S \in \mathcal{P}(U)} \forall_{e \in E}((S, a) \in dom(g) \Rightarrow g(S, e) \in V_E)$

Any set $S \in \mathcal{P}(U)$ i.e. $S \subseteq U$ is often called **a granule** or **a group** of objects.

Definition 2.4 *The set*

$$Gr_K = \{S \in \mathcal{P}(U) : \exists b \in (E \cup A)((S, b) \in dom(g))\}$$

is called **a granule universe** *of* K_I.

Observe that g is a total function on Gr_K.

2.2 Model Components: Universe and Knowledge States

Any Data Mining process starts with a certain initial set of data. The model of such a process depends on representation of this data and we represent it in a form information system table.

We assume hence that the data mining process we model starts with an initial information system

$$I_0 = (U_0, A_0, V_{A_0}, f_0)$$

and we adopt the **universe** U_0 **as the universe of the model**, i.e.

$$\mathcal{GM} = (U_0, \mathcal{K}, \mathcal{G}, \preceq).$$

Data Mining process consists of transformations the initial I_0 into an initial knowledge generalizations systems K_0 that in turn is being transformed into some knowledge generalizations systems K_I, all of them based on some subsystems I of the input system I_0, what we denote by $I \subseteq I_0$. The formal definition of the notion of subsystem is presented in [11]. These transformations of the initial input data (system I_0) in practice are defined by different Data Mining algorithms, and in our model by appropriate generalization operators. We hence adopt the following definition of the set $\mathcal{K}$ of knowledge states.

$$\mathcal{K} = \{K_I : I \subseteq I_0\}.$$

2.3 Model Components: Generalization Relations

A generalization process starts with the input data I_0 i.e. with the initial knowledge generalization system K_{I_0} with its universe $U = \{\{x\} : x \in U_0\}$, called an **object knowledge generalizations** system. It then produces systems which we call more general, with universes $S \subseteq \mathcal{P}(U_0)$ with more then one element i.e. such that $|S| > 1$. We adopt hence the following definition of generalization relation.

Definition 2.5 *Given set* $\mathcal{K}$ *of knowledge states based on the input data* I_0 *and* $K, K' \in \mathcal{K}$ *i.e.*

$$K = (\mathcal{P}(U_0), A, E, V_A, V_E, g),$$

$$K' = (\mathcal{P}(U_0), A', E', V_{A'}, V_{E'}, g').$$

Let $G_K, G_{K'}$ be granule universes (definition 2.4) of K, K' respectively. We define **a generalization relation**

$$\preceq \subseteq \mathcal{K} \times \mathcal{K}$$

as follows:
$K \preceq K'$ *if and only if the following conditions are satisfied.*

i $|G_{K'}| \leq |G_K|$,

ii $A' \subseteq A$.

If $K \preceq K'$ we say that the system K' **is more or equally general as** K.

Observe that the relation $\preceq$ is reflexive and transitive, but is not antisymmetric, as systems K and K' such that $K \preceq K'$ may have different sets of knowledge attributes and knowledge functions.

Definition 2.6 *Let $\preceq \subseteq \mathcal{K} \times \mathcal{K}$ be relation defined in the definition 2.5.*
A relation

$$\prec_{dm} \subseteq \preceq$$

such that it satisfies additional conditions:

iii $|G_{K'}| < |G_K|$,

iv $\exists S \in G_{K'}(|S| > 1)$

is called **a data mining generalization relation.**

2.4 Model Components: Generalization Operators

Generalization operators by definition, operate on the knowledge states, preserving their generality, as defined by the generalization relation. I.e. a partial function $G : \mathcal{K} \longrightarrow \mathcal{K}$ is called a generalization operator if for any $K, K' \in domainG$

$$G(K) = K' \quad if\ and\ only\ if \quad K \preceq K'.$$

Generalization operators are design to describe the action of different data mining algorithms.

2.5 Data Mining Model

Data Mining process consists of two phases: preprocessing and data mining proper. We concentrate here on the Data Mining phase only and discuss its generalization operators in detail in section 3. The preprocessing operators and preprocessing phase can be expressed within our Generalization Model and are presented in a separate paper [11].

Data Mining Model defined below is a special case of the Generalization Model, with generalization relation being data mining relation as defined in definition 2.6 and in which the generalization operators are defined as follows.

Definition 2.7 *An operator $G \in \mathcal{G}$ is called a* **data mining generalization operator** *if and only if for any $K, K' \in domainG$*

$$G(K) = K' \quad if\ and\ only\ if \quad K \prec_{dm} K'$$

for some data mining generalization relation $\prec_{dm}$ (definition 2.6)

Definition 2.8 *A* **Data Mining Model** *is a system*

$$\mathbf{DM} = (U,\ \mathcal{K},\ \mathcal{G}_{dm},\ \prec_{dm}),$$

where the set $\mathcal{G}_{dm}$ is the set of data mining generalization operators.

The above definition 2.7 defines a class of data mining operators. They are discussed in detail in the next section.

3 Data Mining Generalization Operators

The main idea behind the concept of generalization operator is to capture not only the fact that data mining techniques generalize the data but also to categorize existing methods. We want to do it in as exlusive/inclusive sense as possible. We don't include in our analysis purely statistical methods like regression, etc... This gives us only three data mining generalization operators to consider: classification, clustering, and association.

In the following sections we define the appropriate sets of operators by $\mathcal{G}_{clf}, \mathcal{G}_{clr}$ and $\mathcal{G}_{assoc}$ (definitions 3.5, 3.8, 3.9) and prove the following theorem.

Theorem 3.1 (Main Theorem) *Let $\mathcal{G}_{clf}, \mathcal{G}_{clr}$ and $\mathcal{G}_{assoc}$ be the sets of all classification, clustering, and association operators, respectively. The following conditions hold.*

(1) $\mathcal{G}_{clf} \neq \mathcal{G}_{clr} \neq \mathcal{G}_{assoc}$

(2) $\mathcal{G}_{assoc} \cap \mathcal{G}_{clf} = \emptyset$,

(3) $\mathcal{G}_{assoc} \cap \mathcal{G}_{clr} = \emptyset$.

3.1 Classification Operator

In the classification process we are given a data set (set of records) with a special attribute C, called a class attribute. The values $c_1, c_2, ...c_n$ of the class attribute C are called class labels. The classification process is both semantical

(grouping objects in sets that would fit the classes) and syntactical (finding the descriptions of those sets in order to use them for testing and future classification). In fact all data mining techniques share the same characteristics of semantical-syntactical duality.

The formal definitions of classification data and classification operators are as follows.

Definition 3.1 *Any information system* $I = (U, A \cup \{C\}, V_A \cup V_{\{C\}}, f)$ *with a distinguished class attribute* C *and with the class attribute values* $V_{\{C\}} = \{c_1, c_2, ...c_m\}$,$m \geq 2$ *is called a classification information system, or shortly,* **a classification system** *if and only if the sets*

$$C_n = \{x \in U_0 : \ f(x, C) = c_n\}$$

form a partition of U_0.

The classification information system is called in the Rough Set community and literature ([17], [13], [14], [5], [18]) a decision information system with the **decision attribute** C. We assume here, as it is the case in usual classification problems, that we have only one classification attribute. It is possible, as the Rough Set community does, to consider the decision information systems with any non empty set $C \subset A$ of decision attributes.

Definition 3.2 *Let* $I_0 = (U_0, A, V_A \cup V_{\{C\}}, f)$ *be the initial database with the class attribute* C. *The sets*

$$C_{n,0} = \{x \in U_0 : \ f(x, C) = c_n\}$$

are called the **initial classification classes**.

Definition 3.3 *The corresponding set* $\mathcal{K}_{\mathcal{I}}$ *of knowledge systems based on any subsystem* I *of the initial classification information system* I_0, *as defined in the definition* **??**, *is called the set of* **classification knowledge systems** *if and only if for any* $K \in \mathcal{K}$ *the following additional condition holds.*

$$\forall S \in \mathcal{P}(U)(\exists a \in A(S, a) \in dom(g) \Rightarrow (S, C) \in dom(g)\ ).$$

We denote,

$$\mathcal{K}^{clf}$$

the set of all classification knowledge systems based on a classification system I_0.

Definition 3.4 *For any* $K \in \mathcal{K}^{clf}$ *we the sets*

$$C_{n,K} = \{X \in \mathcal{P}(U_0) : \ g(X, C) = c_n\}$$

are called **group classes** *of* K.

Let $\mathbf{DM} = (U_0,\ \mathcal{K}^{clf},\ \mathcal{G}_{dm},\ \prec_{dm})$ be a **Data Mining Model** based on a classification system I_0.

Definition 3.5 *An operator* $G \in \mathcal{G}_{dm}$ *is called a* **classification operator** *if and only if* G *is a partial function*

$$G : \ \mathcal{K}^{clf} \ \longrightarrow \ \mathcal{K}^{clf},$$

such that the following classification condition holds for any $K \in \mathcal{K}^{clf}$ *such that* $K = G(K')$, $K' \in domG$.

$$\forall X(X \in C_{n,K} \ \Rightarrow \ X \subseteq_K C_{n,0}),$$

where the sets $C_{n,0}, C_{n,K}$ *are the sets from definitions 3.2, 3.4, respectively and* $\subseteq_K$ *is an approximate set inclusion defined in terms of k-attributes of* K.

We denote the set of classification operators by $\mathcal{G}_{clf}$.

Observe that our definition of the classification operators gives us freedom of choosing the level of generality (granularity) with which we want to describe our data mining technique, in this case the classification process.

3.2 Clustering Operator

In intuitive sense the term *clustering* refers to the process of grouping physical or abstract objets into classes of similar objects. It is also called *unsupervised* learning or unsupervised classification. We say that *a cluster* is a collection of data objects that are similar to one another within the collection and are dissimilar to the objects in other clusters ([3]). Clustering analysis constructs hence meaningful partitioning of large sets of objects into smaller components. One of the basic property we consider while building clusters is the measure of similarity and dissimilarity. These measures must be present in our definition of the knowledge generalization system applied to clustering analysis. We define hence a notion of clustering knowledge system as follows.

Definition 3.6 *A knowledge generalization system* $K \in \mathcal{K}$,

$$K = (\mathcal{P}(U), A, E, V_A, V_E, g)$$

is called a **clustering knowledge system** *if and only if* $E \neq \emptyset$ *and there are two knowledge attributes* $s, ds \in E$ *such that for any* $S \in Gr_K$ *(definition 2.4)*

g(S,sm) is a measure of similarity of objects in S,

g(S,ds) is a measure of dissimilarity of objects in S *with other* $X \in Gr_K$.

We put

$$\mathcal{K}^{clr} = \{K \in \mathcal{K} : K \ is \ a \ clustering \ system\}.$$

Definition 3.7 *We denote* $\mathcal{K}^{obj} \subset \mathcal{K}$ *the set of all* **object knowledge generalization** *systems.*

Definition 3.8 *An operator $G \in \mathcal{G}_{dm}$ is called a* **clustering operator** *if and only if G is a partial function*

$$G: \mathcal{K}^{obj} \longrightarrow \mathcal{K}^{clr}$$

and for any $K' \in domG$ such that $G(K') = K$ the granule universe Gr_K (definition 2.4) of K is a **partition** *of U satisfying the following condition:*

$$\forall X, Y \in Gr_K (g(X, sm) = g(Y, sm) \cap g(X, ds) = g(Y, ds)).$$

Elements of the granule universe Gr_K of K are called **clusters** defined (generated) by the operator G and we denote the set of all clustering operators by $\mathcal{G}_{clr}$.

It is possible to define within our framework clustering methods that return not always disjoint clusters (with for example some measures of overlapping).

We can also allow the cluster knowledge system be a classification system. It allows the cluster operator to return not only clusters it has generated, their descriptions (with similarity measures) but their names, if needed.

Finally, our framework allows us to incorporate as well the notion of classification by clustering (for example k-nearest neighbor algorithm) by changing the domain of the cluster operator to allow the use of training examples.

3.3 Association Operator

The association analysis is yet another important subject and will be treated in a separate paper. We can define a special knowledge generalization system system AK, called an **association system**. All frequent $k - associations$ are represented in it (with the support count), as well as all information needed to compute association rules that follow from them.

We put

$$\mathcal{K}^{assoc} = \{K \in \mathcal{K} : K \; is \; an \; association \; system\}.$$

Definition 3.9 *An operator $G \in \mathcal{G}_{dm}$ is called an* **association operator** *if and only if G is a partial function that maps the set of all* **object** *association systems $\mathcal{K}^{oasc}$ into $\mathcal{K}^{assoc}$, i.e.*

$$G: \mathcal{K}^{objassoc} \longrightarrow \mathcal{K}^{assoc}$$

and some specific association conditions hold.

We denote the set of all clustering operators by $\mathcal{G}_{assoc}$.

References

[1] Salvatore Greco, Benedetto Matarazzo, Roman Slowinski, Jerzy Stefanowski. *Importance and Interaction of Conditions in Decision Rules* Proceedings of Third International RSCTC'02 Conference, Malvern, PA, USA, October 2002, pp. 255-262. Springer Lecture Notes in Artificial Intelligence.

[2] M. Hadjimichael, A. Wasilewska. *A Hierarchical Model for Information Generalization.* Proceedings of the 4th Joint Conference on Information Sciences, Rough Sets, Data Mining and Granual Computing (RSDMGrC'98), North Carolona, USA, vol.II, 306–309.

[3] J. Han, M. Kamber. *Data Mining: Concepts and Techniques* Morgan, Kauffman, 2000

[4] T.K. Ho, S.N. Srihati. *Decisions combination in multiple classifier system* IEEE Transactions on Pattern Analysis and Machine Learning, 11:63-90, 1994.

[5] M. Inuiguchi, T. Tanino. *Classification versus Approximation oriented Generalization of Rough Sets* Bulletin of International Rough Set Society, Volume 7, No. 1/2.2003

[6] J. Komorowski. *Modelling Biological Phenomena with Rough Sets Proceedings of Third International Conference RSCTC'02*, Malvern, PA, USA, October 2002, p13. Springer Lecture Notes in Artificial Intelligence.

[7] T.Y. Lin. *Database Mining on Derived Attributes Proceedings of Third International Conference RSCTC'02*, Malvern, PA, USA, October 2002, pp. 14 - 32. Springer Lecture Notes in Artificial Intelligence.

[8] Juan F.Martinez, Ernestina Menasalvas, Anita Wasilewska, Covadonga Fernández, M. Hadjimichael. *Extension of Relational Management System with Data Mining Capabilities* Proceedings of Third International Conference RSCTC'02, Malvern, PA, USA, October 2002, pp. 421- 428. Springer Lecture Notes in Artificial Intelligence.

[9] Ernestina Menasalvas, Anita Wasilewska, Covadonga Fernández *The lattice structure of the KDD process: Mathematical expression of the model and its operators* International Journal of Information Systems) and (Fundamenta Informaticae; special issue 2001, pp. 48 - 62.

[10] Ernestina Menasalvas, Anita Wasilewska, Covadonga Fernández, Juan F. Martinez. *Data Mining- A Semantical Model* Proceedings of 2002 World Congres on Computational Intelligence, Honolulu, Hawai, USA, May 11- 17, 2002, pp. 435 - 441.

[11] Ernestina Menasalvas, Anita Wasilewska. *Data Mining Operators* Proceedings of ICDM'04, The Fourth IEEE International Conference on Data Mining, Brighton, UK, Nov 1-4, 2004 - to appear.

[12] Pawlak, Z. *Information systems - theoretical foundations* Information systems, 6 (1981), pp. 205-218

[13] Pawlak, Z. *Rough Sets- theoretical Aspects Reasoning About Data* Kluwer Academic Publishers 1991

[14] Skowron, A. *Data Filtration: A Rough Set Approach* Proceedings de Rough Sets, Fuzzy Sets and Knowledge Discovery. (1993). Pag. 108-118

[15] A. Wasilewska, Ernestina Menasalvas Ruiz, María C. Fernández-Baizan. *Modelization of rough set functions in the KDD frame* 1st International Conference on Rough Sets and Current Trends in Computing (RSCTC'98) June 22 - 26 1998, Warsaw, Poland.

[16] Wojciech Ziarko, Xue Fei. *VPRSM Approach to WEB Searching* Proceedings of Third International RSCTC'02 Conference, Malvern, PA, USA, October 2002, pp. 514- 522. Springer Lecture Notes in Artificial Intelligence.

[17] Wojciech Ziarko. *Variable Precision Rough Set Model* Journal of Computer and Systen Sciences, Vol.46. No.1, pp. 39-59, 1993.

[18] J.T. Yao, Y.Y. Yao. *Induction of Classification Rules by Granular Computing* Proceedings of Third International RSCTC'02 Conference, Malvern, PA, USA, October 2002, pp. 331-338. Springer Lecture Notes in Artificial Intelligence.

A Three-layered Conceptual Framework of Data Mining

Y.Y. Yao[1], N. Zhong[2] and Y. Zhao[1]
[1] Department of Computer Science, University of Regina
Regina, Saskatchewan, Canada S4S 0A2
E-mail: {yyao, yanzhao}@cs.uregina.ca
[2] Department of Information Engineering, Maebashi Institute of Technology
460-1, Kamisadori-Cho, Maebashi 371, Japan
E-mail: zhong@maebashi-it.ac.jp

Abstract

The study of foundations of data mining may be viewed as a scientific inquiry into the nature of data mining and the scope of data mining methods. There is not enough attention paid to the study of the nature of data mining, or its philosophical foundations. It is evident that the conceptual studies of data mining as a scientific field, instead of as a collection of isolated algorithms, are needed for the further development of the field. A three-layered conceptual framework is thus proposed, consisting of the philosophy layer, the technique layer and the application layer. Each layer focuses on different types of fundamental questions regarding to data mining, and jointly they form a complete characterization of the field. To complement the extensive studies of the technique layer and the application layer, we discuss in detail the main issues of the philosophy layer study.

1 Introduction

With the development and success of data mining, many researchers became interested in a fundamental issue, namely, the foundations of data mining [1, 7, 8, 23]. Although three dedicated international workshops have been held [7, 8, 9], there still do not exist well-accepted and non-controversial answers to many basic questions, such as what is the foundations of data mining? What is the scope of the foundations of data mining? What are the differences, if any, between the existing research and the research on the foundations of data mining? The study of foundations of data mining may be started by answering these questions.

The study of foundations of data mining should be viewed as a scientific inquiry into the *nature* of data mining and the scope of data mining *methods*. This simple view separates two important issues. The study of the nature of data mining concerns the philosophical, theoretical and mathematical foundations of data mining as a subject of study; while the study of data mining methods concerns its technological foundations by focusing on the algorithms and tools. A review of the existing studies show that not enough attention has been paid to the study of the nature of data mining, more specifically, to the philosophical foundations of data mining [23].

The following problem quoted from Salthe [17] about studies of ecosystem is equally applicable to the studies of data mining:

> "The question typically is not what is an ecosystem, but how do we measure certain relationships between populations, how do some variables correlate with other variables, and how can we use this knowledge to extend our domain. The question is not what is mitochondrion, but what processes tend to be restricted to certain region of a cell."[page 3]

In the context of data mining, one is more interested in the algorithms for finding "knowledge", but not what is knowledge and what is the knowledge structure. One is often more interested in a more implementation-oriented view or framework of data mining, rather than a conceptual framework for the understanding of the nature of data mining.

There are many reasons accounting for such unbalanced research efforts. The problems of data mining are first raised by very practical needs for finding useful knowledge. One inevitably focuses on the detailed algorithms and tools, without carefully considering the problem itself. A workable program or software system is more easily acceptable by, and at the same time is more concrete and more easily achievable by, many computer scientists than an in-depth understanding of the problem itself. Furthermore, the fundamental questions regarding the nature of the field, the inherent structure of the field and its related fields, are normally not asked at its formation stage. This is especially true when the initial studies produce useful results [17].

The study of foundations of data mining therefore needs

to adjust the current unbalanced research efforts. We need to focus more on the understanding of the nature of data mining as a field instead of as a collection of algorithms. We need to define precisely the basic notions, concepts, principles, and their interactions in an integrated whole. Results from the studies of cognitive science and education are relevant to such a purpose. Posner suggested that, according to the cognitive science approach, to learn a new field is to build appropriate cognitive structures and to learn to perform computations that will transform what is known into what is not yet known [15]. Reif and Heller showed that knowledge structure of a domain is very relevant to problem solving[16]. In particular, knowledge about a domain, such as mechanics, specifies descriptive concepts and relations described at various levels of abstraction, is organized hierarchically, and is accompanied by explicit guidelines that specify when and how knowledge is to be applied [16]. The knowledge hierarchy is used by Simpson for the study of foundations of mathematics [19]. It follows that the study of foundations of data mining should focus on the basic concepts and knowledge of data mining, as well as their inherent connections, at multi-level of abstractions. Without such an understanding of data mining, one may fail to make further progress.

In order to study the foundations of data mining, we need to move beyond the existing studies. More specifically, we need to introduce a conceptual framework, to be complementary to the existing implementation and process-oriented views. The main objective of this paper is therefore to introduce such a framework.

The rest of the paper is organized as follows. In Section 2, we re-examine the existing studies of data mining. Based on the examination, we can observe several problems and see the needs for the study of foundations of data mining. More specifically, there is a need for a framework, within which to study the basic concepts and principles of data mining, and the conceptual structures and characterization of data mining. For this purpose, in Section 3, a three-layered conceptual framework of data mining is discussed, consisting of the philosophy layer, the technique layer, and the application layer [23]. The relationships among the three layers are discussed. The mains issues of the philosophy layer are discussed in Section 4.

2 Overview of the Existing Studies and the Problems

Data mining, as a relatively new branch of computer science, has received much attention. It is motivated by our desire of obtaining knowledge from huge databases. Many data mining methods, based on the extensions, combinations, and adaptation of machine learning algorithms, statistical methods, relational database concepts, and the other data analysis techniques, have been proposed and studied for knowledge extraction and abstraction.

2.1 Three views of data mining

The existing studies of data mining can be classified roughly under three views.

The function-oriented view

The function-oriented view focuses on the goal or functionality of a data mining system, namely, the discovery of knowledge from data. In a well-accepted definition, data mining is defined as "the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns from data" [2]. Such goal-driven approaches establish a close link between data mining research and real world applications.

The function-oriented approaches put forth efforts on searching, mining and utilizing different patterns embedded in various databases. A pattern is an expression in a language that describes data, and has a representation simpler than the data. For example, frequent itemsets, association rules and correlations, as well as clusters of the data points, are common classes of patterns.

Depending on the data and their properties, one may consider different data mining systems with different functionalities and for different purposes, such as text ming, Web mining, sequential mining, and temporal data mining. Under the function-oriented view, the objectives of data mining can be divided into prediction and description. Prediction involves the use of some variables to predict the values of some other variables, and description focuses on patterns that describe the data [2].

The theory-oriented view

The theory-oriented approaches concentrate on the theoretical studies of data mining, and its relationship to the other disciplines. Many models and processes of data mining have been proposed, critically investigated and examined from the theory-oriented point of view [2, 12, 22, 27].

Conceptually, one can draw a correspondence between scientific research by scientists and data mining by computers [26, 27]. More specifically, they share the same goals and processes. It follows that any theory discovered and used by scientists can be used by data mining systems. Thus, many fields contribute to the theoretical study of data mining. They include statistics, machine learning, databases, pattern recognition, visualization, and many other. There is also a need for the combination of existing theories. For example, some efforts have been made to bring the rough sets and fuzzy logic, utility and measurement theory, concept lattice and knowledge structure, and

other mathematical and logical models into the data mining models.

The procedure/process-oriented view

From the procedure/process-oriented view, data mining deals with "non-trivial" processes consisting of many steps, such as data selection, data preprocessing, data transformation, pattern discovery, pattern evaluation, and result explanations [2, 11, 27, 28]. Furthermore, it should be a dynamically organized process.

Under the process-oriented view, data mining studies have been focused on algorithms and methodologies for mining different types of knowledge, speeding up existing algorithms, and evaluation of discovered knowledge.

The three views jointly provide a complete description of data mining research. The function-oriented view states the goals of data mining, the theory-oriented view establishes the formal foundations, and the process-oriented view deals with how to achieve the goals based on the theoretical foundations.

2.2 Problems and potential solutions

Existing studies of data mining typically focus on a particular aspect, a particular algorithm, or a more specific application problem. To some extent, the three views discussed earlier enable us to see a more complete picture. However, a general conceptual framework treating data mining as a field of study is still not proposed and examined. This requires the study of foundations of data mining so that the fundamental questions of the field itself are asked, examined, explained and formalized. The goal here is to provide a better understanding of the field as a whole, rather than a new or faster algorithm.

The foundations of data mining should not be solely mathematics or logic, or any other individual fundamental disciplines. Considering the different types of databases, the diversity of patterns, the ever changing techniques and algorithms, and the different views, we require a multi-level (or multi-layer) understanding of data mining. By viewing data mining in many layers, one can identify the inherent structure of the fields, and put fundamental questions into their proper perspectives in the conceptual map of data mining.

In forming the foundations of data mining, one need to focuses on its main issues and scope in a wide context. In this aspect, it is necessary to comment on scientific research in general.

Scientific research and data mining have much in common in terms of their goals, tasks, processes and methodologies. Scientific research is affected by the perceptions and the purposes of science. Martella *et al.* summarized the main purposes of science, namely, to describe and predict, to improve or manipulate the world around us, and to explain our world [13]. The results of the scientific research process provide a description of an event or a phenomenon. The knowledge obtained from research this helps us to make predictions about what will happen in the future. Research findings are a useful tool for making an improvement in the subject matter. Research findings also can be used to determine the best or the most effective ways of bringing about desirable changes. Finally, scientists develop models and theories to explain why a phenomenon occurs.

Goals similar to those of scientific research have been discussed by many researchers in data mining. Yao *et al.* compared the research process and data mining process [26, 27]. The comparison led to the introduction of the notion of the explanation-oriented data mining, which focuses on constructing models for the explanation of data mining results [27]. Guergachi also stated that the goal of data mining is what science is and has been all about: discovering and identifying relationships among the observations we gather, making sense out of these observations, developing scientific principles, building universal laws from observations and empirical data [5]. Fayyad *et al.* identified two high-level goals of data mining as prediction and description [2]. Ling *et al.* studied the issue of manipulation and action based on the discovered knowledge [8]. Those studies lay the ground work for the present study.

3 A Three-layered Conceptual Framework

A three-layered conceptual framework is recently proposed by Yao [23], which consists of the philosophy layer, the technique layer, and the application layer. The layered framework represents the understanding, discovery, and utilization of knowledge, and is illustrated in Figure 1.

3.1 The philosophy layer

The philosophy layer investigates the basic issues of knowledge. One attempts to answer the fundamental question, namely, what is knowledge? There are many related issues to this question, such as the representation of knowledge, the expression and communication of knowledge in languages, the relationship between knowledge in the mind, in the external real world, and the classification and organization of knowledge [20]. Philosophical study of data mining serves as a precursor to technology and application, it generates knowledge and the understanding of our world, with or without establish the operational boundaries of knowledge.

3.2 The technique layer

The technique layer is the study of knowledge discovery in machine. One attempts to answer the question, how to discover knowledge? In the context of computer science, there are many issues related to this question, such as the implementation of human knowledge discovery methods by programming languages, which involves coding, storage and retrieval issues in a computer, and the innovation and evolution of techniques and algorithms in intelligent systems. The main streams of research in machine learning, data mining, and knowledge discovery have concentrated on the technique layer. Logical analysis and mathematical modelling are considered to be the foundations of technique layer study of data mining.

3.3 The application layer

The ultimate goal of knowledge discovery is to effectively use discovered knowledge. The question need to be answered is how to utilize the discovered knowledge. The application layer therefore should focus on the notions of "usefulness" and "meaningfulness" of discovered knowledge for the specific domain. These notions can not be discussed in total isolation with applications, as knowledge in general is domain specific.

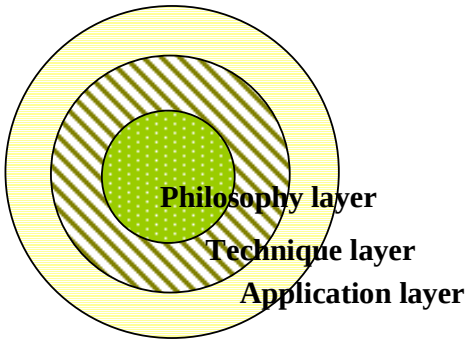


Figure 1: The three-layered conceptual framework of data mining.

3.4 The relationships among the three layers

Two points need to be emphasized about the three-layered conceptual framework.

The three layers are different, relatively independent, and self-contained.

(1.) The philosophical study does not depend on the availability of specific techniques and applications. In other words, no matter knowledge is discovered or not, utilized or not, even if the knowledge structure and expression are recognized or not, it exists. Furthermore, all human knowledge is conceptual and forms an integrated whole [14]. The output of the philosophical study can be expressed as theories, principles, concepts or other knowledge structures. Knowledge structure is built by connecting new bits of information to the old. The study of knowledge at the philosophy layer has important implications for the human society, even if it is not discovered or utilized yet, or it simply provides a general understanding of the real world.

(2.) The technical study can carry out part of the philosophic study results but not all, and it is not constrained by applications. The philosophy layer describes a very general conceptual scheme. The current techniques, including hardware and software, may still be insufficient to bring all of it into reality. On the other hand, the existence of a technique/algorithm does not necessarily imply that discovered knowledge is meaningful and useful. The output of the technical study can be expressed by algorithms, mathematical models, and intelligent systems. The technology can be commercialized. The benefits of technological implementation and innovation tend to move the study of technical layer to be more and more profit-driven.

(3.) The applications of data mining is materialized knowledge in specific domains. They are related to the evaluation of discovered knowledge, the explanation and interpretation of discovered knowledge in a particular domain. The discovered knowledge can be used in many ways. For example, knowledge from transaction databases can be used for designing new products, distributing, and marketing. Comparing to the philosophical and technological studies, the applications have more explicit targets and schedules.

The three layers mutually supports each other and jointly form an integrated whole.

(1.) It is expected that the results from philosophy layer will provide guidelines and set the stage for the technique and application layers. The technology development and innovation can not go far without the conceptual guidance.

(2.) The philosophical study can not developed without the consideration of reality. Technology development may raise new philosophical questions and promote the philosophical study. Technique layer is the bridge between philosophical view of knowledge and the application of knowledge.

(3.) The applications of philosophical and technical outcomes give an impetus for the re-examination of philosophical and technical studies too. The feedbacks from applications provide evidence for the confirmation, re-examination, and modification of philosophical and technical results.

Three layers of the conceptual framework are tightly integrated, namely, they are mutually connected, supported, promoted, facilitated, conditioned and restricted. The division between the three layers is not a clear cut, and may overlap and interweave with each other. Any of them is indispensable in the study of intelligence and intelligent systems. They must be considered together in a common framework through multi-disciplinary studies, rather than in isolation.

The technique layer and application layer have been extensively studied in data mining. In the rest of this paper, we only emphasize on the philosophy layer study of data mining.

4 Main Issues of Philosophy Layer Study

The philosophy layer is the study of knowledge. It deals with many issues, such as concept formation, knowledge representation, evaluation, classification and explanation. We use concepts, a special form of knowledge, as an example to illustrate the basic ideas [24].

4.1 Concept formation and learning

Concepts present a profound development and consciousness of percepts, and enable human to know and understand facts that far outstrip our limited observations [14].

In the process of concept formation and learning, there are two basic issues known as aggregation and characterization [3], as shown in Figure 2. Aggregation aims at the identification of a group of objects so that they form the extension of a concept. Characterization attempts to describe the derived set of objects in order to obtain the intension of the concept [3].

For aggregation, one considers two main processes called differentiation and integration [14]. Differentiation enables us to grasp the differences between elements, so that we can separate one or more elements from the other elements. Integration is the process of generalizing the features of similar elements, then putting together elements into an inseparable whole.

As the final step in concept formation, characterization provides a definition of a concept, condenses the inseparable whole into a brief, retainable statement, tells what distinguishes the units and from what they are being distinguished. This, in Ayn Rand's words, is "to distinguish a concept from all other concepts and thus to keep its units differentiated from all other existents" [14]. A more detailed discussion of concept formation and learning can be found in [24].

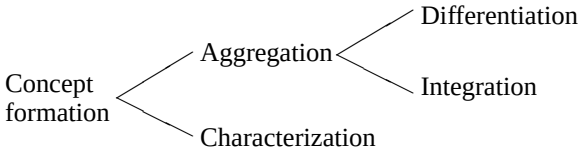


Figure 2: Concept formation and learning.

4.2 Knowledge representation

One needs to define and formulate the knowledge representation clearly and concisely.

A space that can hold knowledge as concepts is called a concept space, namely, it refers to a set or a class of concepts. If we consider the data mining process as a search for concepts in a particular concept space, we need to study different kind of concept spaces first. Inside the concept space, the concept can be represented and discovered. Generally, a concept space S can hold all the concepts, including the ones that can be defined as a formula, and the ones that can not. A definable concept space DS is a sub-space of the concept space S. There are many definable concept spaces in different forms. In most situations, one is only interested in the concepts in a certain form. Consider the class of conjunctive concepts, that formula constructed from atomic formula by only logic connective $\wedge$. A concept space CDS is then referred to as the conjunctively definable space, which is a sub-space of the definable space DS. Similarly, a concept space is referred to as a disjunctively definable space if the atomic formulas are connected by logic disjunctive $\vee$.

The relationship among the above mentioned concept spaces is illustrated in Figure 3. A particular computational model is normally based on one or some philosophical assumptions and may not be able to cover all.

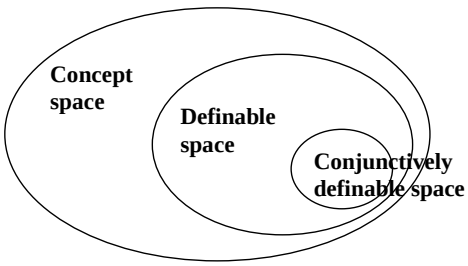


Figure 3: Some concept spaces.

4.3 Knowledge evaluation

Concept formation and knowledge representation do not have to be related directly to quantitative evaluations, although the evaluation of the concepts is an important issue. Many measures have been proposed and studied to quantify the usefulness or interestingness of concepts and concept relations [10, 18, 25]. The results lead to an in-depth understanding of different aspects of knowledge.

Generally, measures can be classified into two categories consisting of objective measures and subjective measures [18]. Objective measures depend on the structure of rules and the data used in the discovery process. Subjective measures depend on the user who examines the rules. While most of the measures are objectively defined by mathematical properties, Yao *et al.* proposed a subjective framework for rule interestingness evaluation based on the user preference [25].

4.4 Knowledge classification and organization

Partitions and coverings are two simple and commonly used knowledge classifications of the universe. A partition of a finite universe is a collection of non-empty, and pairwisely disjoint subsets whose union is the universe. A covering of a finite universe is a collection of non-empty and possibly overlapped subsets whose union is the universe. A partition is a special case of a covering.

Knowledge is organized in a tower (hierarchy) or a partial ordering. Hierarchy means that the base or minimal elements of the ordering are the most fundamental concepts and higher-level concepts depend on lower-level concepts [19]. Partial ordering means that the concepts in the hierarchy are reflexive, anti-symmetric and transitive. Based on the above discussion, we have partition-based hierarchy and covering-based hierarchy. The first-level concept is formed directly from the perceptual data [14]. The higher-level concepts, representing a relatively advanced state of knowledge, are formed by a process of abstracting from abstractions [14].

4.5 Knowledge explanation

Explanation plays a key role in the understanding of knowledge and the knowledge structures. It is the explanation that changes data and information into knowledge.

Explanation-oriented data mining uses the background knowledge to infer features that can possibly explain and interpret knowledge discovered from data. The constructed explanations give some evidence about under what conditions (within background knowledge) the discovered pattern is most likely to happen, or how the background knowledge is related to the pattern.

5 Conclusion

A three-layered conceptual framework of data mining is discussed in this paper, consisting of the philosophy layer, the technique layer and the application layer. The philosophy layer deals with the formation, representation, evaluation, classification and organization, and explanation of knowledge; the technique layer deals with the technique development and innovation; the application layer emphasizes on the application, utility and explanation of mined knowledge.

The layered framework focuses on the data mining questions and issues in different abstract levels, and thus, offers us opportunities and challenges to reconsider many fundamental issues. The framework is aimed at the understanding of the data mining as a field of study, rather than a collection of theories, algorithms, and tools.

References

[1] Chen, Z., The three dimensions of data mining foundation, *Proceedings of IEEE ICDM'02 Workshop on Foundation of Data Mining and Knowledge Discovery*, 119-124, 2002.

[2] Fayyad, U.M., Piatetsky-Shapiro, G. and Smyth, P., From data mining to knowledge discovery: an overview, in: *Advances in Knowledge Discovery and Data Mining*, Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R. (eds.), AAAI/MIT Press, Menlo Park, CA, 1-34, 1996.

[3] Feger, H. and Boeck, P.D., Categories and concepts: introduction to data analysis, in:*Categories and Concepts: Theoretical Views and Inductive Data Analysis*, Mechelen, I.V., Hampton, J., Michalski, R.S. and Theuns, P. (eds.), Acadamic Press Limited, 1993.

[4] Gehrke, J., New research directions in KDD, *SIGKDD Explorations*, **3**, 76-77, 2001.

[5] Guergachi, A.A., Connecting traditional sciences with the OLAP and data mining paradigms, *Proceedings of the SPIE: Data Mining and Knowledge Discovery: Theory, Tools, and Technology*, 5098, 226-234, 2003.

[6] Gunopulos, D. and Rastogi, R., Workshop report: ACM SIGMOD'00 workshop on research issues in data mining and knowledge discovery, *SIGKDD Explorations*, **2**, 83-84, 2000.

[7] Lin, T.Y. and Liau, C.J. (eds.), *Proceedings of the PAKDD'02 Workshop on Fundation of Data Mining, Communications of Institute of Information and Computing Machinery*, **5**, 101-106, 2002.

[8] Lin, T.Y. and Ohsuga, S. (eds.), *Proceedings of IEEE ICDM'02 Workshop on Foundation of Data Mining and Knowledge Discovery*, 2002.

[9] Lin, T.Y., Hu, X.H., Ohsuga, S. and Liau, C.J. (eds.), *Proceedings of IEEE ICDM'03 Workshop on Foundation of New Directions in Data Mining*, 2003.

[10] Lin, T.Y., Yao, Y.Y. and Louie, E., Value added association rules, *Proceedings of PAKDD'02*, 328-333, 2002.

[11] Mannila, H., Methods and problems in data mining, *Proceedings of International Conference on Database Theory*, 41-55, 1997.

[12] Mannila, H., Theoretical frameworks for data mining, *SIGKDD Explorations*, **1**, 30-32, 2000.

[13] Martella, R.C., Nelson, R. and Marchand-Martella, N.E., *Research Methods: Learning to Become a Critical Research Consumer*, Allyn & Bacon, Bosten, 1999.

[14] Peikoff, L., *Objectivism: The Philosophy of Ayn Rand*, Dutton, 1991.

[15] Posner, M.I. (ed.), *Foundations of Cognitive Science*, Preface: learning cognitive science, The MIT Press, Cambridge, Massachusetts, 1989.

[16] Reif, F. and Heller, J.I., Knowledge structure and problem solving in physics, *Educational Psychologist*, **17**, 102-127, 1982.

[17] Salthe, S.N., *Evolving Hierarchical Systems, their Structure and Representation*, Columbia University Press, 1985.

[18] Silberschatz, A. and Tuzhilin, A., What makes patterns interesting in knowledge discovery systems? *IEEE Transactions on Knowledge and Data Engineering*, **8**, 970-974, 1996.

[19] Simpson, S.G., What is foundations of mathematics? 1996. http://www.math.psu.edu/simpson/hierarchy.html, retrieved November 21, 2003.

[20] Sowa, J.F., *Conceptual Structures, Information Processing in Mind and Machine*, Addison-Wesley, Reading, Massachusetts, 1984.

[21] Xie, Y. and Raghavan, V.V., Probabilistic logic-based characterization of knowledge discovery in databases, *Proceedings of IEEE ICDM'02 Workshop on Foundation of Data Mining and Knowledge Discovery*, 107-112, 2002.

[22] Yao, Y.Y., Modeling data mining with granular computing, *Proceedings of the 25th Annual International Computer Software and Applications Conference (COMPSAC 2001)*, 638-643, 2001.

[23] Yao, Y.Y., A step towards the foundations of data mining, *Proceedings of the SPIE: Data Mining and Knowledge Discovery: Theory, Tools, and Technology*, 5098, 254-263, 2003.

[24] Yao, Y.Y., Concept formation and learning: a cognitive informatics perspective, *Proceedings of ICCI'04*, 42-51, 2004

[25] Yao, Y.Y., Chen, Y.H. and Yang X.D., Measurement-theoretic foundation for rules interestingness, *ICDM 2003 Workshop on Foundations of Data Mining*, 221-227, 2003.

[26] Yao, Y.Y. and Zhao, Y. Explanation-oriented data mining, Manuscript, 2004.

[27] Yao, Y.Y., Zhao, Y. and Maguire, R.B., Explanation-oriented association mining using rough set theory, *Proceedings of the 9th International Conference of Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*, 165-172, 2003.

[28] Zhong, N., Liu, C. and Ohsuga, S., Dynamically organizing KDD processes, *International Journal of Pattern Recognition and Artificial Intelligence*, **15**, 451-473.

A novel belief theoretic association rule mining based classifier for handling class label ambiguities

J. Zhang*
Hemispheric Center for Environmental Technology (HCET)
Florida International University, Miami, Florida, USA

S.P. Subasingha, K. Premaratne, M.-L. Shyu, M. Kubat and K.K.R.G.K. Hewawasam†
Department of Electrical and Computer Engineering
University of Miami, Coral Gables, Florida, USA

Abstract

A classification technique based on the induction of the individual class labels from pre-classified training examples whose class labels may possess ambiguities is proposed. We assume the existence of a large number of training instances that have each been assigned an integer class label. At the training stage, an association rule mining technique detects interesting rules; at the classification stage, a classifier applies a belief theoretic method on these rules. The ability of belief theory to accommodate ambiguity makes our algorithm capable of achieving high performance when the training set contains class label ambiguities. It also accounts for highly skewed or 'imbalanced' training sets. These characteristics make it ideally suited for various applications that have recently gained tremendous importance and urgency. For example, in security monitoring and threat classification, different experts are likely to have conflicting opinions about the threat level to be assigned to specific training set instances; moreover, the vast majority of instances are likely not to correspond to a heightened threat level thus giving rise to a highly skewed training set. Experiments on several databases in the UCI data repository demonstrate that, for databases without ambiguities, the proposed classifier achieves performance comparable with available classifiers; for databases with class label ambiguities, it provides higher average classification accuracy and better efficiency.

*This work was conducted while J Zhang was at the University of Miami where his work was supported by NSF Grant IIS-0325260 (ITR Medium).

†Corresponding author is K Premaratne (kamal@miami.edu). The work of SP Subasingha, K Premarnate, M-L Shyu and KKRGK Hewawasam was supported by NSF Grants IIS-0325260 (ITR Medium) and EAR-0323213.

1 Introduction

In classification problems, complete statistical knowledge regarding the conditional density function of each class is rarely available. When no evidence supports one form of the density function or another, often a good solution is to build up a training set of correctly classified feature vectors or samples and to classify each new incoming feature vector using the evidence provided by 'nearby' samples from the training set. For example, in the *voting k nearest neighbor classifier* [9], which we refer to as the *KNN classifier,* an unclassified feature vector is assigned to the class represented by a majority of its k nearest neighbors in the training set. The error rate of the KNN classifier approaches the optimal Bayes error rate as the number of samples N and the number of neighbors k both tend to infinity in such a manner that $k/N \to 0$ [4]. It provides good performance in numerous practical applications and accordingly enjoys significant popularity in the pattern recognition community. Its main drawback is that it implicitly assumes that the k nearest neighbors of an incoming feature vector are contained in a region of relatively small 'volume;' this ensures sufficiently good resolution in the estimates of the different conditional densities.

Numerous modifications of this KNN classifier have been suggested to improve its performance [5, 6]. For example, the belief theoretic KNN method in [5], which we refer to as the *KNN-BF classifier*, not only improves the classification performance over its 'crisp' version, but also handles ambiguities in class labeling in the training set via the inherent ambiguity modeling capability of belief theory [14, 18, 19, 21, 25]. Such a capability is of critical importance in several application scenarios of topical interest.

A case in point is detection and assessment of security threats. For example, in airport terminal security monitoring, data gathered from various heterogeneous sensors (x-

ray scanners, metal detectors, thermal imaging sensors, radiation detectors, chemical sensors, etc.) are used to extract features for classifying targets and potential threat carriers into different threat level classes. For this purpose, a training set, where several domain experts would have assigned an appropriate threat level to each training set instance, needs to be utilized. When a suitable classification algorithm classifies a target as belonging to a heightened threat level, one may assign higher resolution, and perhaps even mobile, sensors for further tracking and evaluation; certain situations might even call for immediate action. The training set employed in such a system, and in similar applications, possesses certain specific characteristics:

(C1) The likelihood of conflicting assignments of threat levels is very high. In other words, ambiguities are unavoidable in training set class labeling.

(C2) The computational burden and storage requirements of the classifier must be tolerable.

(C3) The training set is highly skewed: For example, suppose the threat classes in an airport terminal security monitoring environment are

Class 1: `NotDangerous`; Class 2: `OfConcern`;
Class 3: `Dangerous`; Class 4: `VeryDangerous`.

While most targets entering the terminal gateways fall into the `NotDangerous` classification, only a very small fraction of the targets will belong to the `VeryDangerous` class. Thus, the training set will be skewed in favor of the threat level `NotDangerous`.

Our main purpose was to design a classifier that effectively accommodates these issues.

• To address **(C1)**, we adopt a classifier that is based on belief theoretic notions. Hence, in addition to the improvement in classification performance when class label ambiguities are present, the proposed algorithm can also provide quantitative confidence information regarding the classification decision it makes. We must mention here that, various other mechanisms can also be used to represent ambiguities; indeed, the relationship between belief theory and these other mechanisms can be found in [8, 23, 13, 15]. Our choice of belief theory for the current purpose is motivated mainly by the ease and convenience it offers (see [8] for an interesting discussion on this matter).

• To address **(C2)**, the proposed algorithm is made to operate on rules extracted from an association rule mining (ARM) algorithm. In this respect, our proposed method differs from the KNN-BF classifier in [5]. ARM has been demonstrated to be an extremely powerful tool for extracting interesting and not-so-obvious patterns and associations from large databases [1, 12, 17, 22]. In fact, in [16], a classifier, which we refer to as the *ARM classifier*, was built by first applying a modified ARM method to the training set and then selecting rules with high confidence or support values. However, it does not effectively address **(C1)**.

• We address **(C3)** by adapting a simple and effective ARM strategy that attempts to 'capture' both majority and minority classes.

This paper is organized as follows: Section 2 provides a primer on belief theory; Section 3 presents a new ARM method that effectively accommodates a highly skewed training set; Section 4 contains the proposed classifier; Section 5 is reserved for experimental results; and Section 6 gives the concluding remarks where we identify several research directions that warrant further investigation.

2 Belief Theory: A Primer

Let $\Theta = \{\theta_1, \theta_2, \ldots, \theta_n\}$ be a finite set of mutually exclusive and exhaustive hypotheses about some problem domain. It signifies the corresponding 'scope of expertise' and is referred to as its *frame of discernment (FoD)* [21]. A hypothesis θ_i, also referred to as a *singleton*, represents the lowest level of discernible information. Elements of 2^Θ, the power set of Θ, form all hypotheses of interest. A hypothesis that is not a singleton (e.g., (θ_1, θ_2)) is referred to as a *composite hypothesis*.

2.1 Basic Notions

A *basic probability assignment (BPA)* or *mass* is a function $m : 2^\Theta \to [0, 1]$ that satisfies

$$m(\emptyset) = 0 \text{ and } \sum_{A \subseteq \Theta} m(A) = 1. \tag{1}$$

The quantity $m(A)$ can be interpreted as a measure of the 'support' that one is willing to commit *exactly* to A, and not to any of its subsets. A mass assigned to a composite hypothesis is free to move into its constituent singletons if further evidence warrants it. This is how ambiguity is modeled in belief theory. The set of hypotheses (including composite hypotheses) $\mathcal{F}_\Theta$ that possesses nonzero masses is called the *focal elements* of Θ. The triple $\{\Theta, \mathcal{F}_\Theta, m(\cdot)\}$ is referred to as the corresponding *body of evidence (BoE)*.

Belief theory enables one to conveniently characterize complete ignorance via the *vacuous BPA*

$$m(A) = \begin{cases} 1, & \text{if } A = \Theta; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Intuitively, a portion of belief committed to a hypothesis must also be committed to any hypothesis it implies. In belief theory, one uses the notion of *belief* to quantify this. The belief of $A \subseteq \Theta$ is a function $\mathrm{Bl} : 2^\Theta \mapsto [0, 1]$ where

$$\mathrm{Bl}(A) = \sum_{B \subseteq A} m(B). \tag{3}$$

Then, $\mathrm{Bl}(A)$ can be interpreted as a measure of the total belief committed to A. It can easily be verified that the belief in A and the belief in its negation $\overline{A}$ do not necessarily sum to 1. Hence, in belief theory, the additivity axiom of probability is relaxed.

Consequently, $\mathrm{Bl}(A)$ does not reveal to what extent one believes in $\overline{A}$, i.e., to what extent one doubts A. This latter notion is described by $\mathrm{Bl}(\overline{A})$. Indeed, in belief theory, one uses the notion of *plausibility* to quantify this. The plausibility of $A \subseteq \Theta$ is a function $\mathrm{Pl} : 2^{\Theta} \mapsto [0, 1]$ where

$$\mathrm{Pl}(A) = 1 - \mathrm{Bl}(\overline{A}) = \sum_{B \cap A \neq \emptyset} m(B). \tag{4}$$

Hence, $\mathrm{Pl}(A)$ quantifies the extent to which one fails to doubt A, i.e., to what extent one finds A plausible.

A probability distribution $\mathrm{Pr}(\cdot)$ such that $\mathrm{Bl}(A) \leq \mathrm{Pr}(A) \leq \mathrm{Pl}(A),\ \forall A \subseteq \Theta$, is said to be *compatible* with the underlying BPA $m(\cdot)$. An example of such a probability distribution is the *pignistic probability distribution* $Bp(\cdot)$ defined for each singleton $\theta_i \in \Theta$ as [24]

$$\mathrm{Bp}(\theta_i) = \sum_{\theta_i \in A} m(A) \div |A|, \tag{5}$$

where $|A|$ denotes the cardinality of A.

2.2 Evidence Combination

A strategy whereby the evidence provided by two 'independent' BoEs $\{\Theta, \mathcal{F}_1, m_1(\cdot)\}$ and $\{\Theta, \mathcal{F}_2, m_2(\cdot)\}$ could be 'pooled' to form a single BoE $\{\Theta, \mathcal{F}, m(\cdot)\}$ is provided by the *Dempster's rule of combination (DRC)* which is applicable when the BoEs span identical FoDs [21]. The 'fused' BoE $\{\Theta, \mathcal{F}, m(\cdot)\}$ that the DRC generates is

$$m(C) = \frac{\sum\limits_{A \cap B = C} m_1(A)\, m_2(B)}{1 - \sum\limits_{A \cap B = \emptyset} m_1(A)\, m_2(B)},\ \forall C \neq \emptyset, \tag{6}$$

with $m(\emptyset) = 0$. Here, it is assumed that the two BoEs are *consistent*, i.e., $\sum_{A \cap B = \emptyset} m_1(A)\, m_2(B) \neq 1$. This fusion operation is typically denoted as $m = m_1 \oplus m_2$ and referred to as the *orthogonal sum* of $m_1(\cdot)$ and $m_2(\cdot)$. The DRC is perhaps the most widely used belief theoretic evidence combination function.

3 Partitioned-ARM

ARM extracts rules in a database that satisfy certain minimum support and confidence constraints [1]. In its usual form however, it is not suitable for accommodating the characteristics **(C1-C3)** of the training sets (see Section 1). For example, when dealing with class label ambiguities, it may be possible to construct a classifier based on only those rules that do not possess any class ambiguity. However, such a strategy may potentially ignore a large fraction of training samples; moreover, such an algorithm may inadvertently ignore samples that would have otherwise provided extremely critical evidence that captures more qualitative aspects. The fact that the training set itself is highly skewed brings its own difficulties: unless specifically compensated for, a classifier built on such a training set typically tends to favor the 'majority' classes at the expense of the 'minority' classes. In a threat classification scenario, for instance, such a situation must be avoided at all cost because ignoring the minority class can have devastating consequences.

Although the KNN-BF classifier in [5] is extremely effective against class label ambiguities (i.e., **(C1)**), it is not catered towards addressing **(C2-C3)**. Hence, while we also exploit the advantages of belief theoretic notions in our classifier design, to address computational and storage issues (i.e., **(C2)**), we make the classifier operate on a significantly smaller data set. It is the rule set that an appropriate ARM algorithm would generate that we propose to utilize for this purpose. The effectiveness of such an ARM rule set in classification has been amply demonstrated in [16] and we make use of these notions in this present work. However, if the associated advantages offered by such a strategy regarding reduced computational and storage burdens are to be fully utilized, it is crucial that we effectively address those concerns characteristic of an environment where highly skewed training sets are the norm (i.e., **(C3)**).

In this section, we discuss our solution to alleviate these difficulties; we refer to it as the *partitioned-ARM*.

3.1 Training Set

We denote the training set by $D_{\mathrm{TR}} = \{D_i\},\ i = \overline{1, N_D}$, where D_i indicates a data instance and N_D indicates the cardinality of D_{TR}, i.e., its 'size.' Each data instance D_i is taken to be of the following form: for $i = \overline{1, N_D}$,

$$D_i =< F_i, C_i > \text{ where } F_i =< f_{1i}, f_{2i}, \ldots, f_{N_F i} > . \tag{7}$$

Here, F_i denotes the i-th feature vector extracted from the measurements collected from the data sources. Each feature vector is taken to consist of N_F features; $f_{ji},\ j = \overline{1, N_F}$, denotes the j-th such feature embedded within the i-th data instance in D_{TR}. The class label that has been allocated to this i-th data instance is denoted by C_i.

3.2 Relevant FoDs

Since we intend to utilize belief theoretic notions in our classifier design, it is instructive at this juncture to identify the relevant FoDs.

3.2.1 Feature FoD

The FoD of each feature $f_{ji},\ j = \overline{1, N_F},\ i = \overline{1, N_D}$, is taken to be identical, finite and equal to Θ_f, viz.,

$$\text{FoD}[f_{ji}] = \Theta_f = \{\theta_f^{(1)}, \theta_f^{(2)}, \ldots, \theta_f^{(n_f)}\}, \qquad (8)$$

where n_f denotes the number of possible values a feature may assume. Clearly, the FoD Θ_F of each feature vector $F_i,\ i = \overline{1, N_D}$, is the N_F-fold cross-product of 2^{Θ_f} [7], viz.,

$$\text{FoD}[F_i] = \Theta_F = \underbrace{2^{\Theta_f} \times 2^{\Theta_f} \times \cdots \times 2^{\Theta_f}}_{N_F \text{ times}}. \qquad (9)$$

3.2.2 Class Label FoD

The FoD of each label class $C_i,\ i = \overline{1, N_D}$, is taken to be finite and equal to Θ_C, viz.,

$$\text{FoD}[C_i] = \Theta_C = \{\theta_C^{(1)}, \theta_C^{(2)}, \ldots, \theta_C^{(n_C)}\}, \qquad (10)$$

where n_C denotes the number of different class labels that are to be discerned. For example, in the discussion in Section 1, Θ_C={`NotDangerous`, `OfConcern`, `Dangerous`, `VeryDangerous`}.

3.3 Partitioning the Training Set

To circumvent difficulties associated with 'majority' classes overwhelming 'minority' classes, we propose to apply ARM to a partition of the training set D_{TR}. This partition is constructed according to the class labels the training data instances have been classified, i.e., each class label (irrespective of whether it is modeled as a singleton $\theta_C^{(i)}$ or composite hypothesis generated from Θ_C) corresponds to one partition of the training set D_{TR}. Suppose we enumerate these 'newly' formed class labels as $C^{(k)},\ k = \overline{1, N_C}$, where N_C may take values from $|\Theta_C|$ (this corresponds to only singleton class label classifications) to $|2^{\Theta_C}|$ (this corresponds to class labels spanning all possible subsets of Θ_C). Then, the partitions of D_{TR} can be denoted by $\{P_i\},\ i = \overline{1, N_C}$ where

$$D_{\text{TR}} = \bigcup_{k=1}^{N_C} P^{(k)}, \text{ with } P^{(k_1)} \cap P^{(k_2)} = \emptyset,\ k_1 \neq k_2. \quad (11)$$

Here, for $k = \overline{1, N_C}$, $P^{(k)}$ contains all the data instances that possess $C^{(k)}$ as the corresponding class label.

For example, in the example previously discussed in Section 1, the threat level of each training set data instance was classified into its singletons, viz., `NotDangerous`, `OfConcern`, `Dangerous` or `VeryDangerous`. Suppose, due to differences in opinion of experts whose advise was sought in constructing the training set, some data instances are classified as (`OfConcern,Dangerous`). Then, we would divide D_{TR} into $N_C = 5$ partitions $\{P^{(k)}\},\ k = \overline{1, 5}$, where each would contain data instances that have been identically classified. So, while $\{P^{(1)}, P^{(2)}, P^{(3)}, P^{(4)}\}$ contains data instances having threat level $\{C^{(1)}, C^{(2)}, C^{(3)}, C^{(4)}\} \equiv \{$`NotDangerous`, `OfConcern`, `Dangerous`, `VeryDangerous`$\}$ respectively, $P^{(5)}$ contains data instances having threat level $C^{(5)} \equiv$ (`OfConcern,Dangerous`).

3.4 Partitioned-ARM

A rule generated by an ARM algorithm is an implication of the form $X \rightarrow Y$, where the *antecedent* $X \subseteq \Theta_F$ and the *consequence* $Y \subseteq \Theta_C$. Such a rule $X \rightarrow Y$ holds in the database D_{TR} with *support* s if $s\%$ of the data instances in D_{TR} contain X and are labeled with class Y; the rule $X \rightarrow Y$ holds in the database D_{TR} with *confidence* c if $c\%$ of those data instances in D_{TR} containing X are also labeled with class Y.

We propose to apply an appropriate ARM algorithm (e.g., the Apriori algorithm [2]) on each partition $P^{(k)}$ to generate the rule set $R^{(k)}$ that passes the minimum support constraint $\text{MinSupp}^{(k)}$ specified for that particular partition, viz.,

$$R^{(k)} = \textbf{Apriori}\left(P^{(k)}, \text{MinSupp}^{(k)}\right),\ k = \overline{1, N_C}, \quad (12)$$

where the rule set $R^{(k)}$ is taken to contain $N_C^{(k)}$ rules, viz.,

$$R^{(k)} = \{r_\ell^{(k)}\} \text{ where } r_\ell^{(k)} : F_\ell^{(k)} \rightarrow C^{(k)},\ \ell = \overline{1, N_C^{(k)}}.$$

We refer to this as the *partitioned-ARM*.

In generating the rule set $R^{(k)}$, the support is calculated within $P^{(k)}$ thus ensuring that a balanced number of rules are generated for each class. However, the confidence allocated to each rule $r_\ell^{(k)}$ is calculated within the complete training set D_{TR}. Note that, the rules acquired in this manner are associated with a particular class $C^{(k)}$ of the corresponding partition $P^{(k)}$.

The antecedent $F_\ell^{(k)}$ of rule $r_\ell^{(k)}$ takes the following form:

$$F_\ell^{(k)} = < f_{1\ell}^{(k)}, f_{2\ell}^{(k)}, \ldots, f_{N_F,\ell}^{(k)} >, \qquad (14)$$

where we assume that each feature value $f_{j\ell}^{(k)}$ may only assume the values

$$f_{j\ell} \in \left\{\theta_f^{(1)}, \theta_f^{(2)}, \ldots, \theta_f^{(n_f)}, \Theta_f\right\}, \qquad (15)$$

i.e., the only ambiguity we allow in the feature value is complete ignorance (enabling one to accommodate, for instance, a missing feature value). This is in contrast to the consequence $C^{(k)}$ of rule $r_\ell^{(k)}$ where both partial and complete ambiguity in class label classification is accounted for.

The final rule set $R_{\rm PARM}$ generated is

$$R_{\rm PARM} = \bigcup_{k=1}^{N_C} R^{(k)}. \quad (16)$$

3.5 Rule Pruning

To finally arrive at an appropriate set of rules to be used by the classifier, a pruning algorithm is applied to remove 'redundant' rules that are contained in $R_{\rm PARM}$. First, we need to introduce several useful notions.

Consider any feature, say $f_{j\ell}^{(k)}$, of the antecedent $F_\ell^{(k)}$ of rule $r_\ell^{(k)}$. Then, $f_{j\ell}^{(k)} = \Theta_f$ indicates complete ambiguity regarding the feature value $f_{j\ell}^{(k)}$. It provides no information and hence is of no significance to the rule itself. This, and other relevant notions, are formalized via

Definition 1 (Level of Abstration (LoA)) *For rule* $r_\ell^{(k)}$: $F_\ell^{(k)} \rightarrow C^{(k)},\ \ell = \overline{1, N_C^{(k)}},\ k = \overline{1, N_C}$, *let*

$$Abst\left[F_\ell^{(k)}\right] = \bigcup_j \left\{ f_{j\ell}^{(k)} : f_{j\ell}^{(k)} = \Theta_f \right\}.$$

Then

(i) cardinality of $Abst[F_\ell^{(k)}]$, *i.e.,* $|Abst[F_\ell^{(k)}]|$, *is referred to as the* level of abstraction (LoA) *of rule* $r_\ell^{(k)}$ *(and the corresponding feature* $F_\ell^{(k)}$*);*

(ii) rule $r_{\ell_1}^{(k)}$ *(and the corresponding feature* $F_{\ell_1}^{(k)}$*) is said to be* more abstract *(or* less specific*) than rule* $r_{\ell_2}^{(k)}$ *(and the corresponding feature* $F_{\ell_2}^{(k)}$*) if the LoA of* $r_{\ell_1}^{(k)}$ *is higher than the LoA of* $r_{\ell_2}^{(k)}$*; and*

(iii) rule $r_\ell^{(k)}$ *is said to* cover *the training data instance* $D_i =< F_i, C_i >,\ i = \overline{1, N_D}$, *if*

$$f_{ji} = f_{j\ell}^{(k)},\ \forall j = \overline{1, N_F},\ s.t.\ f_{j\ell}^{(k)} \notin Abst[F_\ell^{(k)}].$$

Hence, if rule $r_{\ell_1}^{(k)}$ is more abstract than rule $r_{\ell_2}^{(k)}$, it means that $r_{\ell_1}^{(k)}$ contains more 'insignificant' features than $r_{\ell_2}^{(k)}$; and if it covers the data instance D_i, it means that the rule 'captures' D_i.

The rule set $R_{\rm PARM}$ generated by partitioned-ARM includes rules at different LoAs. Therefore, each training data instance may be covered by multiple rules at different LoAs, and hence, some rules are redundant and can be safely removed. In essence, we require an effective method that enables one to select the least number of rules that can still cover the complete training set with the least ambiguity.

One selection criterion to achieve this is the following: From among the rules that cover the same training instance, allocate a higher priority to those possessing a higher confidence value and a lower LoA. With this in mind, the first step in the pruning strategy we propose is to sort (implemented via an available sorting algorithm) the rule set $R_{\rm PARM}$ generated from the partitioned-ARM first in descending order of their confidence values; those with the same confidence value are then sorted in ascending order of their LoAs. The sorting scheme we employ differs from that used in [16] where the second level of sorting is determined by the support value. The reason lies in our desire to accommodate class label ambiguities.

Then, starting from the first rule in the sorted rule set, all the data instances that can be covered by each rule are removed from the training set. If the remaining set of training instances is not empty, the rule will be selected to be included in the final rule set; otherwise, it is pruned. This process is terminated when either no rules are left in $R_{\rm PARM}$ or no training instances are left in the training set. At termination, if the training set is not empty, all its remaining data instances are also selected to be included in the final rule set $R_{\rm Pruned}$; these are allocated a confidence value of 1.0. This process, which we refer to as the *RulePruning* algorithm, can be described as follows:

RulePruning($R_{\rm PARM}$,$D_{\rm TR}$){
while (!empty($R_{\rm PARM}$) & !empty($D_{\rm TR}$)){
  r=Top($R_{\rm PARM}$);
  if (!empty($D_{\rm r}$ = **covered**(r,$D_{\rm TR}$))){
    $R_{\rm Pruned} = R_{\rm Pruned} \bigcup r$;
    $R_{\rm PARM} = R_{\rm PARM} \setminus r$;
    $D_{\rm TR} = D_{\rm TR} \setminus D_{\rm r}$}}
if (!empty($D_{\rm TR}$)){
  $R_{\rm Pruned} = R_{\rm Pruned} \bigcup D_{\rm TR}$}}

Here, $D_{\rm TR}$ is the training set, $R_{\rm PARM}$ is the original rule set generated by partitioned-ARM and $R_{\rm Pruned}$ is the rule set selected after pruning; the function **covered**(r,$D_{\rm TR}$) generates the set $D_{\rm r}$ of all training data instances that are covered by rule r.

4 ARM-KNN-BF Classifier

We use the rule set $R_{\rm Pruned}$ generated in Section 3 to design a *belief theoretic association rule mining based classifier* that enables one to accommodate the presence of class label ambiguities. We refer to this as the *ARM-KNN-BF classifier.*

4.1 Rule BPAs

To explain how we incorporate belief theoretic notions, let $F^\sharp =< f_1^\sharp, f_2^\sharp, \ldots, f_{N_F}^\sharp >$ be an incoming feature vector that needs to be classified using the information contained in the rule set. Classifying $F^\sharp$ means assigning to it one class from $C^{(k)},\ k = \overline{1, N_C}$. Our strategy in addressing

this problem is to view each rule $r_\ell^{(k)}$ as a piece of evidence that alters our belief that $F^\sharp$ also belongs to $C^{(k)}$. But, with what 'certainty' can we make this claim? It is reasonable to assume that only some portion of our belief is committed to $C^{(k)}$ whereas the remaining belief should be committed to no other class, i.e., it should be committed to the vacuous BoE Θ_C. The corresponding BPA $m_\ell^{\sharp,(k)}$ then is

$$m_\ell^{\sharp,(k)}(A) = \begin{cases} \alpha_\ell^{(k)}, & \text{if } A = C^{(k)}; \\ 1 - \alpha_\ell^{(k)}, & \text{if } A = \Theta_C, \end{cases} \tag{17}$$

and zero otherwise. Here, $\alpha_\ell^{(k)} \in [0,1]$. Note that, $C^{(k)}$ is allowed to be a composite hypothesis or class label. The following two factors play critical roles in the determination of $\alpha_\ell^{(k)}$.

1. Confidence value $c_\ell^{(k)}$ of rule $r_\ell^{(k)}$: Since $c_\ell^{(k)}$ is a measure of the confidence one places on the rule, it is reasonable to take $\alpha_\ell^{(k)}$ to be proportional to $c_\ell^{(k)}$.

2. Distance between the new feature vector $F^\sharp$ and the antecedent $F_\ell^{(k)}$ of the rule: If $F^\sharp$ is 'far' from $F_\ell^{(k)}$, the class label of the latter can be considered to impact very little on the class of $F^\sharp$; on the other hand, if $F^\sharp$ is 'close' to $F_\ell^{(k)}$, one will be much more inclined to believe that they belong within the same class label. Hence, it is reasonable to postulate that $\alpha_\ell^{(k)}$ should be a decreasing function of $\text{Dist}[F^\sharp, F_\ell^{(k)}]$, a suitable 'distance' function between $F^\sharp$ and $F_\ell^{(k)}$. With these observations in place, we choose $\alpha_\ell^{(k)}$ as

$$\alpha_\ell^{(k)} = \alpha_0 c_\ell^{(k)} e^{\gamma\, \text{Dist}[F^\sharp, F_\ell^{(k)}]}, \tag{18}$$

where $\alpha_0 \in [0,1]$ and $\gamma > 0$ are parameters to be appropriately chosen.

4.2 Distance Function

When computing the distance between an incoming feature vector $F^\sharp$ and the antecedents $F_\ell^{(k)}$ of the rules, it is reasonable to 'trust' more detailed rules over the others. The following distance function captures this and 'penalizes' those rules that are less detailed:

$$\text{Dist}[F^\sharp, F_\ell^{(k)}] = \left\| \begin{bmatrix} d_1 & d_2 & \cdots & d_{N_F} \end{bmatrix}^T \div N_F \right\|_p, \tag{19}$$

where, $[\cdot]^T$ denotes matrix transpose and, for $j = \overline{1, N_F}$,

$$d_j = \begin{cases} |f_j^\sharp - f_{j\ell}^{(k)}|, & \text{for } f_j^\sharp \notin \text{Abst}[F^\sharp] \\ & \quad \text{and } f_{j\ell}^{(k)} \notin \text{Abst}[F_\ell^{(k)}]; \\ 0, & \text{otherwise.} \end{cases} \tag{20}$$

Here, $\|\cdot\|_p$ is any Hölder p-norm.

4.3 Fused BPA

At this juncture, for a given incoming feature vector $F^\sharp$, a BPA $m_\ell^{\sharp,(k)} : 2^{\Theta_C} \mapsto [0,1]$ corresponding to each rule $r_\ell^{(k)}$ has been constructed. Note that $\ell = \overline{1, N_C^{(k)}}$ denotes the number of rules corresponding to class label $C^{(k)}$ and $k = \overline{1, N_C}$ denotes the number of class labels. We can now use DRC to combine these BPAs and arrive at the fused BPA $m^\sharp(\cdot)$. This final BPA, and the corresponding belief and plausibility notions, can now be used to make a 'soft' decision regarding the class label of the incoming feature vector. Alternately, we may use the pignistic probability distribution in (5), or any other decision criterion that is readily available in the literature, to arrive at a 'hard' decision regarding the class label [10].

4.4 Rule Set from k Nearest Neighbors

When the rule set generated is large, we propose to apply the k nearest neighbor principle to choose k rules whose antecedents are nearest (in the sense of the distance function in (19)) to the incoming feature vector $F^\sharp$. This yields only k BPAs, potentially a further significant reduction in computational and storage requirements.

4.5 Rule Refinement

Finally, a rule refinement step is carried out by classifying the training set D_{TR} via the ARM-KNN-BF classifier operating on the rule set R_{Pruned}. Those training instances that were not correctly classified can be considered to provide evidence that is absent in R_{Pruned}. We then supplement R_{Pruned} with these training instances (with a confidence of 1.0) to generate the refined rule set R_{Refined}. This is the rule set that we use with the ARM-KNN-BF classifier.

5 Experimental Results

All average classification accuracy results presented herein correspond to 10-fold random subsampling where each database was split into a training set (70%) and testing set (30%). Classification rules are generated from the training set while the testing set is used to test the performance.

5.1 Databases Without Class Label Ambiguities

For this case, several UCI databases [3] were used to compare the classification accuracy of the proposed ARM-KNN-BF classifier with the KNN [9], C4.5rules [20], KNN-BF [5], and ARM [16] classifiers. The various parameters

and accuracy values for the ARM classifier are borrowed from [16].

The minimum support and confidence values play a vital role in classification accuracy. For the experiments, the support value was kept between 0.01 and 0.10 while the confidence value varied from 0.3 to 0.9. To contain the processing complexity to a tolerable level, it was decided not to use a larger number of neighbors k. For $k = \overline{1, 10}$, the impact of k on classification accuracy was found to be minimal. Hence, all the experiments were run with the identical value of $k = 7$.

Table 1 shows the parameter values used by the ARM-KNN-BF classifier for different databases. Note that, BCancer and TTToe refer to the Breast Cancer and Tic-Tac-Toe databases of [3], respectively.

Table 1. Non-ambiguous databases: Parameters used by ARM-KNN-BF classifier

Database	Support	Confidence
BCancer	0.05	0.3
Car	0.03	0.5
Diabetes	0.03	0.3
Iris	0.08	0.8
Monks	0.06	0.6
TTToe	0.04	0.5
Wine	0.07	0.3

Table 2 compares the average number of rules generated per class by different classifiers. It can be observed

Table 2. Non-ambiguous databases: Average number of rules generated/class

Database	KNN & KNN-BF [5, 9]	ARM [16]	ARM-KNN-BF
BCancer	242	49	76
Car	302	N/A	71
Diabetes	258	57	169
Iris	32	5	18
Monks	200	N/A	48
TTToe	334	8	70
Wine	41	7	30

that the number of rules generated by the ARM classifier is significantly less. However, it cannot accommodate class ambiguities and this is a key requirement that we desire in our classifier. KNN and KNN-BF classifiers are the only other classifiers that satisfy this criterion and compared to these, ARM-KNN-BF classifier generates a significantly lower number of rules. It is worth mentioning that, the way we used the KNN classifier to handle ambiguities is to treat each class label—including those with ambiguity—as a separate class.

Table 3 and Table 4 show, respectively, the classification accuracy and the corresponding standard deviations of each algorithm. For ARM classifier, the best average accuracy

Table 3. Non-ambiguous databases: Classification accuracy

Database	KNN [9]	C4.5 rules [20]	KNN-BF [5]	ARM [16]	ARM-KNN-BF
BCancer	0.97	0.95	0.93	0.96	0.97
Car	0.92	0.93	0.93	N/A	0.93
Diabetes	0.70	0.72	0.71	0.72	0.76
Iris	0.94	0.94	0.96	0.93	0.95
Monks	0.92	0.98	0.97	N/A	0.95
TTToe	0.92	0.98	0.93	1.00	0.99
Wine	0.94	0.91	0.96	0.92	0.96

Table 4. Non-ambiguous databases: Percent standard deviation of classification accuracy

Database	KNN [9]	C4.5rules [20]	KNN-BF [5]	ARM-KNN-BF
BCancer	1.08	0.81	3.03	1.16
Car	1.12	1.51	0.97	1.57
Diabetes	2.13	4.23	2.22	3.99
Iris	2.10	2.74	2.00	2.25
Monks	1.23	0.81	1.74	1.32
TTToe	2.22	2.15	3.03	1.81
Wine	2.50	3.71	2.10	2.83

(i.e., CBA-CAR plus infrequent rules reported in [16]) was used.

Some remarks regarding how these accuracy 'scores' in Table 3 were computed are in order. Although the true class label is 'crisp,' a belief theoretic classifier in general assigns a 'soft' decision to an incoming data instance. Since we are interested in a 'hard' decision, one may employ one of several strategies that are available in the literature to remove the ambiguity associated with a 'soft' decision [10]. We utilized the pignistic probability distribution in (5) [24] for this purpose.

The classification results in Table 3 show that the proposed ARM-KNN-BF classifier compares well with the others. Moreover, it operates on a much smaller rule set than the KNN and KNN-BF classifiers. However, the strength of a belief theoretic classifier is its ability to perform well even in the presence of ambiguities, and the design of such a classifier was exactly what we set out to do.

5.2 Databases With Class Label Ambiguities

Class label ambiguities are actually absent in the available UCI resources. To remedy this situation, it was decided to artificially introduce class label ambiguities to some of

the available UCI databases. Several strategies to achieve this have been discussed in [25]. We believe that the introduction of ambiguities to class labels in a random manner may not reflect reality because ambiguities are typically generated due to inability of an expert to decide between class labels that are 'close' to each other.

Instead, the KNN classifier itself was used to introduce ambiguity into the class label—select k number of neighbors to a particular training instance; if the majority of the class labels of these k neighbors exceed the true label by a pre-specified percentage p, then introduce an ambiguity into the true label. For example, suppose the true label of an instance is $C3$. With $k = 10$, suppose 3 of them belong to class $C2$, 6 belong to class $C3$ and 1 belongs to class $C4$. If $p = 25\%$ (corresponding to 2.5 votes), then we introduce an ambiguity into the class label as $(C2, C3)$. This particular strategy enables us to control the 'level' of ambiguity of the resultant database via changing the value of p. In our experiments, we used $p = 25\%$.

Except for the introduction of ambiguities, the other parameters were basically unchanged from what were used previously. We compared the proposed ARM-KNN-BF classifier with only the KNN and KNN-BF classifiers because of their ability to handle class ambiguities. Table 5 shows the support and confidence values used by the proposed ARM-KNN-BF classifier. As can be observed, the support value is kept quite low (0.10 or less); the corresponding confidence value is fairly high; as before, we used $k = 7$ for all the experiments. Table 6 compares the average number of rules generated per class by different classifiers.

Table 5. Ambiguous databases: Parameters used by ARM-KNN-BF classifier

Database	Support	Confidence
BCancer	0.10	0.8
Car	0.02	0.5
Diabetes	0.10	0.6
Iris	0.06	0.8
Monks	0.06	0.6
TTToe	0.04	0.5
Wine	0.07	0.3

Table 7 shows the classification accuracy of each algorithm. Some remarks regarding how these accuracy 'scores' were computed are in order. Suppose the true class label is $(C1, C2)$. Then, how does one score the assignments $C1$, $(C2, C3)$ or $(C1, C2)$? Clearly, the latter should be considered a 'perfect' classification while the others should be scored less. We believe that a suitable measure for suitably capturing such 'soft' decisions is

$$\text{Score} = \frac{|\text{True label} \cap \text{Assigned label}|}{|\text{True label} \cup \text{Assigned label}|}. \qquad (21)$$

Table 6. Ambiguous databases: Average number of rules generated/class

Database	KNN [9] & KNN-BF [5]	ARM-KNN-BF (% reduction)
BCancer	242	68 (72%)
Car	302	103 (66%)
Diabetes	258	120 (53%)
Iris	32	19 (41%)
Monks	341	62 (82%)
TTToe	333	65 (80%)
Wine	53	40 (25%)

Table 7. Ambiguous databases: Classification accuracy

Database	KNN [9]	KNN-BF [5]	ARM-KNN-BF
BCancer	0.92	0.82	0.94
Car	0.89	0.83	0.88
Diabetes	0.78	0.76	0.79
Iris	0.91	0.94	0.92
Monks	0.86	0.87	0.89
TTToe	0.80	0.82	0.85
Wine	0.87	0.87	0.91

We are not aware of a suitable alternate measure that appears in the literature for this purpose. With (21), the assignment $C1$ would score $1/2$; $(C2, C3)$ would score $1/3$; the only decision that gets a 'perfect' score of 1.0 is the true label, viz., $(C1, C2)$.

These experiments indicate that the proposed ARM-KNN-BF classifier compares quite favorably with the others in handling class label ambiguities. In addition, it utilizes a significantly lower number of rules (see Table 6).

5.3 Discussion

As opposed to mechanisms such as voting (which is exactly what is utilized in KNN), belief theoretic methods (such as KNN-BF and ARM-KNN-BF) enable a much richer information content to be captured and taken into account when making a classification decision. The major factor that sets the proposed ARM-KNN-BF classifier apart from both KNN and KNN-BF is how it interprets the term 'neighbor.' In both KNN and KNN-BF, neighbors are data instance related while in ARM-KNN-BF they are rule related. Recalling that these rules are generated via ARM, this means that a neighbor in ARM-KNN-BF already possesses information regarding the associations among data instances. In other words, such neighbors capture more structural information of data than what can be captured by simple instance neighbors. We believe that these are the reasons for the proposed ARM-KNN-BF classifier's better

performance with a significantly lower number of rules.

6 Concluding Remarks

A novel belief theoretic rule mining based classification algorithm has been introduced in this paper. It addresses the following concerns: **(C1)** ambiguities in class label assignments in the training set samples; **(C2)** computational and storage burdens; and **(C3)** training sets that are highly skewed.

Based on the 'interesting' rules acquired from ARM on the raw training set that is partitioned according to the class label assignments, the belief theory formalism was used to construct the classifier. Each rule is treated as a BoE, and the final classification decision is based upon the fused BoE obtained via the DRC. Only those rules that are 'closer' to the incoming feature are taken into account when constructing this fused BoE.

The ARM-KNN-BF algorithm we have proposed, not only accommodates class label ambiguities, but also operates on a rule set (obtained via ARM) that is significantly smaller than the original training set. The resulting savings in computational and storage requirements can be a significant advantage especially when working with huge training sets. This constitutes the main difference between the ARM-KNN-BF and KNN-BF classifiers.

Accommodating database ambiguities in classification is an important research topic. The scoring system we utilize in (21) is quite novel and we believe it can form the basis on which various classification algorithms may be evaluated against each other.

6.1 Future Research Work

Several other interesting issues have not been addressed in this preliminary work. These include the following:

- As mentioned previously, we believe that this work has important applications in security monitoring and threat classification scenarios. Such applications call for one to err on the side of caution. For example, it is better to over-estimate the true threat level than otherwise; under-estimating a `Dangerous` threat level as `OfConcern` is an error that may have serious consequences. An important research problem is to study what strategies are suitable to reduce the possibility of under-estimating threat level assignments.
- The proposed method, as it stands, can actually accommodate missing features by simply representing such a feature by its corresponding FoD. One should be able to utilize the belief theoretic methodology to handle other types of ambiguities as well. One such approach appears in [11].

References

[1] R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In *Proc. ACM SIGMOD International Conference on Management of Data*, pages 207–216, Washington, D.C., May 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proc. International Conference on Very Large Data Bases (VLDB'94)*, pages 487–499, Santiago de Chile, Chile, Sept. 1994.

[3] C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998.

[4] T. M. Cover and P. E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, Jan. 1967.

[5] T. Denoeux. The k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE Transactions on Systems, Man and Cybernetics*, 25(5):804–813, May 1995.

[6] S. A. Dudani. The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man and Cybernetics*, 6(4):325–327, Apr. 1976.

[7] S. Fabre, A. Appriou, and X. Briottet. Presentation and description of two classification methods using data fusion on sensor management. *Information Fusion*, 2:49–71, 2001.

[8] R. Fagin and J. Y. Halpern. A new approach to updating beliefs. In P. P. Bonissone, M. Henrion, L. N. Kanal, and J. F. Lemmer, editors, *Proc. Conference on Uncertainty in Artificial Intelligence (UAI'91)*, pages 347–374. Elsevier Science, New York, NY, 1991.

[9] E. Fix and J. L. Hodges. Discriminatory analysis: nonparametric discrimination: consistency properties. Technical Report 4, USAF School of Aviation Medicine, Randolph Field, TX, 1951.

[10] S. L. Hegarat-Mascle, I. Bloch, and D. Vidal-Madjar. Introduction of neighborhood information in evidence theory and application to data fusion of radar and optical images with partial cloud cover. *Pattern Recognition*, 31(11):1811–1823, Nov. 1998.

[11] K. K. R. G. K. Hewawasam, K. Premaratne, M.-L. Shyu, and S. P. Subasingha. Rule mining and classification in the presence of feature level and class label ambiguities. In *Intelligent and Unmanned Systems, Intelligent Computing: Theory and Applications III*, Proc. SPIE. Mar./Apr. 2005. Defense and Security Symposium 2005, in review.

[12] T. Karban, J. Rauch, and M. Simunek. SDS-rules and association rules. In *Proc. ACM Symposium on Applied Computing (SAC'04)*, pages 482–489, Nicosia, Cyprus, Mar. 2004.

[13] M. A. Klopotek and S. T. Wierzchon. A new qualitative rought-set approach to modeling belief functions. In L. Polkowski and A. Skowron, editors, *Proc. International Conference on Rough Sets and Current Trends in Computing (RSCTC'98)*, volume 1424 of *Lecture Notes in Computer Science*, pages 346–354. Springer-Verlag, Heidelberg, Germany, 1998.

[14] E. C. Kulasekere, K. Premaratne, D. A. Dewasurendra, M.-L. Shyu, and P. H. Bauer. Conditioning and updating evidence. *International Journal of Approximate Reasoning*, 36(1):75–108, Apr. 2004.

[15] T. Y. Lin. Fuzzy partitions II: Belief functions. A probabilistic view. In L. Polkowski and A. Skowron, editors, *Proc. International Conference on Rough Sets and Current Trends in Computing (RSCTC'98)*, volume 1424 of *Lecture Notes in Computer Science*, pages 381–386. Springer-Verlag, Heidelberg, Germany, 1998.

[16] B. Liu, W. Hsu, and Y. M. Ma. Integrating classification and association rule mining. In *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'98)*, pages 80–86, New York, NY, Aug. 1998.

[17] A. A. Nanavati, K. P. Chitrapura, S. Joshi, and R. Krishnapuram. Mining generalized disjunctive association rules. In *Proc. International Conference on Information and Knowledge Management (CIKM'01)*, pages 482–489, Atlanta, GA, Nov. 2001.

[18] S. Parsons and A. Hunter. A review of uncertainty handling formalisms. In A. Hunter and S. Parsons, editors, *Applications of Uncertainty Formalisms*, volume 1455 of *Lecture Notes in Artificial Intelligence*, pages 8–37. Springer-Verlag, New York, NY, 1998.

[19] K. Premaratne, J. Zhang, and K. K. R. G. K. Hewawasam. Decision-making in distributed sensor networks: A belief-theoretic Bayes-like theorem. In *Proc. IEEE International Midwest Symposium on Circuits and Systems (MWSCAS'04)*, volume II, pages 497–500, Hiroshima, Japan, July 2004.

[20] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Representation and Reasoning Series. Morgan Kaufmann, San Francisco, CA, 1993.

[21] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, NJ, 1976.

[22] M.-L. Shyu, S.-C. Chen, and R. L. Kashyap. Generalized affinity-based association rule mining for multimedia database queries. *Knowledge and Information Systems (KAIS), An International Journal*, 3(3):319–337, Aug. 2001.

[23] A. Skowron and J. Grzymala-Busse. From rough set theory to evidence theory. In R. R. Yager, M. Fedrizzi, and J. Kacprzyk, editors, *Advances in the Dempster-Shafer Theory of Evidence*, pages 193–236. John Wiley and Sons, New York, NY, 1994.

[24] P. Smets. Constructing the pignistic probability function in a context of uncertainty. In M. Henrion, R. D. Shachter, L. N. Kanal, and J. F. Lemmer, editors, *Proc. Conference on Uncertainty in Artificial Intelligence (UAI'89)*, pages 29–40. North Holland, 1989.

[25] P. Vannoorenberghe. On aggregating belief decision trees. *Information Fusion*, 5(3):179–188, Sept. 2004.

Published by the IEEE Computer Society
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1314

IEEE Computer Society Order Number P2142
Library of Congress Number 2004103507
ISBN 0-7695-2142-8