A MULTI-TIER KNOWLEDGE DISCOVERY INFO-STRUCTURE USING ENSEMBLE TECHNIQUES

SAKTHIASEELAN KARTHIGASOO

UNIVERSITI SAINS MALAYSIA
2007

ACKNOWLEDGEMENTS

Completing this thesis has been a great accomplishment to me that would not have been possible without the help, support and mentorship of a few individuals who are my pillar of inspiration throughout the period of this valuable and rewarding journey.

I am deeply grateful to my supervisor, Dr. Cheah Yu-N for his dedication, guidance, advice, ideas, motivation, encouragement and also the humor during my research and the writing of this thesis. Thank you, sir.

I would like to thank Professor Dr. Zaharin Yusoff and my beloved uncle Mr. Senthilathiban Veeriah for their inspiration, intellectual and academic guidance which helped me a lot in making the right decisions in my life and I believe this relationship will always continue among us.

I thank my friends Selvakumar, Shailendra, Chong Yong Han, Janice Ho, Bala and all the rest who know me, for the different perspectives, opinions and making my life much more fully experienced during the period of my research and thesis writing.

I am extremely thankful to my parents: Mr. Karthigasoo Veeriah and Mdm. M.Catherine Maheswary for their supreme love, care, motivation and strength. They were my main driving force that made me pursue my post-graduate degree. Now, this is one of my perfect gifts I will give to them – the joy and happiness seen on their faces always.

Finally, I thank God for his divine grace, mercy and love and for guiding me in the right path for all these years. All glory and splendor goes to Him forever.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

Info-struktur Penemuan Pengetahuan Berbilang Tingkat Menggunakan Teknik Ensembel

ABSTRAK

Fokus utama kami ialah untuk mempelajari keujudan peraturan-peraturan yang ditemui daripada data-data tanpa catatan serta menjana keputusan yang lebih tepat dan muktamad. Ini dilakukan melalui kaedah penghibridan yang merangkumi kedua-dua mekanisma berselia dan tidak berselia. Data tanpa catatan yang sebelum ini tidak mempunyai maklumat klasifikasi sekarang boleh digunakan kerana kajian kami telah menghasilkan wawasan baru dalam bidang penemuan dan pembelajaran pengetahuan.

Metodologi kami untuk Penemuan dan Pembelajaran Pengetahuan terdiri daripada 6 fasa yang penting yang menggunakan pelbagai algorithma untuk menghasilkan keputusan. Fasa dan algorithma yang digunakan ialah seperti berikut: a) Pemprosesan Data melalui Pengisian Min/Mod dan Pelengkapan Kombinatronik, b) Ensembel Pengelompokan menggunakan teknik Menggalak dalam Peta Swaurus Kohonen, c) Pengdiskretasian Data melalui Pentaakulan Boolean dan Entropi/Panjang Penerangan Minima, d) Penjanaan Peraturan melalui Algorithma Genetik, Algorithma Johnson dan Penghampiran Set Kasar, e) Penapisan Peraturan melalui formula Michalski dan teknik Torgo dan f) Pembelajaran menggunakan teknik ensembel dengan 'Bagging' dalam Rangkaian Neural.

Output dari sesuatu fasa akan menjadi input untuk fasa berikutnya. Kesemua 6 fasa tersebut termasuk fungsi dan algorithma masing-masing membentuk suatu integrasi aplikasi-aplikasi berlainan. Seni bina talianpaip yang lengkap membentuk Infrastruktur Penemuan, Penyatuan, dan Pembelajaran Pengetahuan Berbilang Tingkat (MESTAC).

Kami telah menjalankan analisis dan perbandingan dengan dua rangkakerja penemuan pengetahuan serta algoritma yang berbeza untuk menghasilkan model yang terbaik (kombinasi algorithma) yang mampu menghasilkan ramalan dengan ketepatan

yang tinggi. Kami telah memperkenalkan teknik penggalak dalam Peta Swaurus Kohonen untuk menghasilkan keputusan pengelompokan yang lebih baik. Kami juga telah memperkenalkan teknik ensemble ‘bagging’ dalam kombinasi rangkaian neural untuk memantapkan ramalan.

MESTAC mungkin merupakan satu kombinasi fasa-fasa yang kompleks tetapi ia mengandungi 3 kelebihan yang penting dari segi rangkakerja keseluruhannya. MESTAC mudah, cekap dan umum. Mudah di sini membawa makna bahawa MESTAC merupakan suatu infrastruktur modular yang mana setiap fasa merupakan modul yang mempunya fungsi tertentu yang bebas. Kecekapan membawa makna bahawa keputusan yang dihasilkan oleh MESTAC adalah lebih tepat. Umum membawa makna bahawa info-struktur ini boleh digunakan untuk penemuan pengetahuan daripada pelbagai jenis set data, contohnya set data selanjar, bercampur dan yang diskret.

MESTAC telah menunjukkan keupayaannya sebagai suatu rangkakerja tersaur dengan menggunakan set data kanser buah dada. Keputusan positif daripada kajian ini menunjukkan bahawa kaedah ini berkesan dan boleh digunapakai sebagai satu kaedah penemuan dan pembelajaran pengetahuan yang baru.

A Multi-tier Knowledge Discovery Info-structure using Ensemble Techniques

ABSTRACT

Our terminal focus is to learn rules instances that have been discovered from unannotated data and generate results with high accuracy. This is done via a hybridized methodology which features both supervised and unsupervised techniques. Unannotated data without prior classification information could now be useful as our research has brought new insight to knowledge discovery and learning altogether.

Our Methodology for Knowledge Discovery and Learning (MKDL) consists of 6 important phases that used different algorithms to produce the outcome. The phases and algorithms used are as follows: a) Data Preprocessing using Mean/Mode Fill and Combinatorial Completion, b) Clustering Ensemble using Boosting technique within Kohonen Self Organizing Map, c) Data Discretization using Boolean Reasoning and Entropy/Minimum Description Length, d) Rule Generation using Genetic Algorithm, Johnson Algorithm and Rough Sets Approximation, e) Rule Filtering using Michalski's formula and Torgo's technique and f) Learning using the ensemble technique with Bagging within Neural Networks.

An output from one phase will be an input to the next phase. All the 6 phases combined with its functions and algorithm form an integration of different application. This complete architecture forms the Multi-tier Knowledge Discovery, Amalgamation and Learning Info-structure (MESTAC).

We performed comparison and analysis with 2 knowledge discovery frameworks and different algorithms to come up with the best model (combination of algorithms) that result in high accuracy in prediction. We introduced a boosting ensemble technique into Kohonen Self Organizing Map to produce better clustering results. We also introduced

bagging ensemble technique to a combination of neural network algorithm to produce precision in prediction.

MESTAC may seem to be a complex combination of phases but there are 3 important advantages in terms of its overall methodology. MESTAC is simple, efficient and generic. Simplicity here indicates that MESTAC is a highly modular info-structure, where each phase is an independent functional-specific module. Efficiency here indicates that the final outcome of the info-structure is more accurate. Genericity here indicates that the info-structure can be used to discover knowledge for different types of data-sets such as continuous, mixed and discrete data-sets.

MESTAC has demonstrated to be a feasible method using a well-known breast cancer dataset. The positive results from the empirical study indicate that the methodology is sound and is indeed applicable to be a new knowledge discovery and learning methodology.

Chapter 1

Introduction

There are two sides to every question.

- –Protagoras (485 BC - 421 BC)

Formatted: Bulleted + Level: 1 + Aligned at: 36 pt + Tab after: 54 pt + Indent at: 54 pt

If a man empties his purse into his head, no one can take it from him. An investment in knowledge always pays the best interest.

- Benjamin Franklin (1706 - 1790)

1.1 The Data Overflow

These days, the trademark of the ongoing information and knowledge revolution is the generation and accumulation of very large amount of data. These data is sourced from a variety of places namely, manufacturing, commercial transactions, scientific explorations, telecommunication networks, space science, medical research, manufacturing lines services among others. Here the largeLarge-scale deployment of various hardware and software technologies – i.e. fast communication, powerful microprocessorsmicroprocessor and servers, more accurate predictions, high-capacity databases, data mining, data warehousing, knowledge discovery – has led to the explosive growth in the quantity and , variety of data and quality of knowledge. This phenomenonoccurrence is often referred to as *information overload*, which emphasizeshighlights the discontinuity between quantitative data and human-comprehensible knowledge.

The reality of the information revolution is that current technologies for data acquisition, storage and retrieval have far outstripped methodologies for analysis and

knowledge extraction or discovery and thereafter learning the knowledge for future predictions. This motivates the association of *value* with the frequently massive amount of data stored in enterprise data centers, against which to balance off the expense and effort associated with data acquisition, amalgamation and long-term storage. The curious scenario which is being created is an environment that is *data-rich*. It could be data-rich but it would be *knowledge-poor*.

The research presented in this thesis is motivated by this fundamental disagreement between data, information and knowledge. We explore the various methods for the acquisition of data, the extraction of comprehensible and useful knowledge and thereafter for the learning of this knowledge. These are done from raw unannotated data from which little or no background information exists. The capacity to acquire such knowledge of this nature which is usually encoded as symbolic rules, associations or patterns is essentially an attempt to extract value from the data overflow. This would essentially be useful in a broad range of services and relationship-oriented applications namely manufacturing, transactional and service-oriented institutions.

1.2 Data Mining and Knowledge Discovery

Research into automated or computer-assisted mechanisms with which to derive useful information and knowledge from the data overflow is usually referred to as knowledge discovery from databases (KDD) or data mining. KDD is a syncretism domain influenced by more well-established research area, i.e. machine learning, expert systems, pattern recognition, statistical analysis, artificial neural network, and high performance computing. A functional KDD solution should provide a non-trivial ~~process~~procedure of identifying valid, novel, potentially useful and ultimately understandable patterns in data. This will enhance the real-world value of a database, which is in its original state likely to contain poor quality data – i.e. noisy, redundant, missing values and inaccurate. The output of

KDD solutions is expected to be concisebrief, human readable and insightful; hence the necessity to address issues of knowledge representation, uncertainty modeling, output evaluation, dynamic data environments and even system integration with the underlying data sources and context-providing application domain.

Data mining is conventionally understood to pertain to the application of analysis and discovery formulas and algorithms on large datasets, so as to uncover useful regularities (Fayyad et al., 1996b; Witten & Frank 1999). This domain can therefore be considered to address the core activity within the broader KDD framework. It is important to emphasize the necessary assumption of implicit regularities, patterns or trends within the targeted dataset. This allows generalization in some compact form of propositional rules, decision trees and artificial neural network (Witten & Frank, 1999). This will also allow data mining to be conceptualized as a paradigm in more established methods (e.g. statistical-based). Effective data mining enables discovery and characterization of strong regularities which can be employedused as a concise human-comprehensible generalization of the data-set and which. This would be useful in a decision-support of predictive capacity that is accurate enough.

The data itself Data can be visualized as a *m x n pattern matrix*, with. Here *m* is the number of datum features or attributes (i.e. the data-set dimensionality) and *n* denotes the number of elements (i.e. data-set size)). Individual datum. A dataset might also have an additional attribute, i.e. the classification – as would be applied by a human domain expert which. These are *a priori* labels or categories subset within the data-set. In KDD, this is also called the *decision attribute*.

Knowledge of data-set classification allows for the application of various a variety of well-established supervised learning methodologies. On the other hand, real world KDD scenarios or situation usually deal with un-annotated data-sets for which the classification is unknown, or *in extremis*. KDD is also used when there is no available classification

Formatted: Indent: First line: 36 pt

schema. This latter situation would arise from the scarcity of domain expertise, which. This would be typical for previously unobserved phenomenonoccurrence or when the data-set is relatively complex.

The presented methodology assumes the genericity of unannotated data-sets, hence the necessity for an internalized determination of the class attribute from intrinsic properties of the data elements themselves. The basic strategy would be to employ variousutilize a range of data clustering methodologies to discover the underlying conceptualtheoretical geometry of the data-set, thereby allowing for the subsequent determination of previously unseen regularities and dependencies. This integrationcombination of unsupervised and supervised methods is necessary for any practical knowledge extraction framework and is a significant componentfactor of our research.

In this thesis, two main areas that we will be offering our contributions are at the knowledge discovery level (which is a pipeline of algorithm) and the learning level.

1.3 Research Focus

In this section we highlight the focus of research undertaken in this thesis within the context of existing work on data mining, knowledge discovery and learning, and data mining in particular. We begin by providing a problem statement, followed by the proposed problem solution and lastly an accountsolutions. Finally a description of the contributions made through our research in rule generationfrom the rules generated from un-annotated data and the learning phase that comes after it.

1.3.1 Problem Statement

Most of the research in KDD centers on the extraction of patterns or regularities from annotated datasets where the classification or decision label for each object is known in

advance. Problem may arise when there are no decision labels or attributes where it will be difficult to make a correct decision or conclusion for a particular set a condition. The data-defined nature of the extracted knowledge and thereafter the learning of this knowledge also raise issues of thelike quality, accuracy and comprehensibility, particularly in scenarios featuring previously unobserved phenomena. These factors provide the motivation for our investigation into real-world databases and subsequently our modus operandi of using ensemble techniques into a knowledge discovery and learning methodology using which at the same time addresses:-

1) ***Data uncertainty and imprecision:*** This refers to the inevitable accumulation of errors, inconsistencies and omissions (missing values) during data acquisition in the real world. These discrepancies will introduce uncertainty and imprecision into the knowledge discovery process and this will also jeopardize the prediction accuracy of the learning process. It is therefore important that every phase that is involved in a knowledge discovery process plays a very important role. Here its breaks down to the data accuracy at every phase.
2) ***HeterogeneousDiverse*** ***data types:*** This refers to continuous and nominal numeric data as well as discrete data occurring in the same dataset. This is reflective of real world analogue data and is opposed to the fundamentally discretized nature of symbolic rule representation, which is in turn dependent on the distinct clusters (clustering accuracy) and class constructions. These should be a mechanism that can be fed with different types of dataset and still manage to generate a desired result.
3) ***Inaccurate clustering:*** Many clustering algorithms only seem to be used to cluster data which is annotated. What about clustering data which is unannotated? This is an issue that needs to be addressed. A better method is also required so

that the clustering can be more accurate. Many clustering methods are available but the process of generating a more accurate classification still needs refinement.

4) ***Suboptimum discretization methods:*** Many methods are available to discretize continuous data to be used for further processing but the choices are often unclear. Using the correct discretization method is important for knowledge discovery. Therefore comparing different discretization results using the same dataset will surely carry a greater value. This way we can make a choice to select the correct discretization results to be inputted for the next phase.
5) ***Generation of incomprehensible rules:*** Rules that are generated may be of little value due to excessive number of attributes. This is due to the ineffective removal of unimportant or insignificant attributes. Derived rule should ideally be comprehensible and also concise – having antecedents with a small number of descriptors. Having said this, using the right reducts algorithm to generate rules is indeed vital before the rules are generated.
6) ***Insufficient evaluation and refinement of rule quality:*** Current rule evaluation criteria may not reflect the rules' usefulness from a real-world standpoint nor take into account human reasoning factors (Tsumoto, 1998) other than simple predictive accuracy. We need to use a filtering method that could allow us to take into account only rules that are of high quality.
7) Inaccuracy in prediction: Current prediction techniques are still inaccurate. Predictions that are not accurate can cause many setbacks, both isolated ones and big ones due to a chain of reaction. A better method to learn discovered knowledge and later use these learned knowledge to predict new cases which have no conclusions would be advantageous.

1.3.2 Research Objectives

The objective of the research undertaken is to extract concise rules and thereafter to generate accurate prediction for decision support from un-annotated data. The most important goal is the inclusion of the ensemble techniques for better accuracy within the phases. This thesis outlines a knowledge discovery (extraction) and learning methodology to address the above-stated problem issues. Our proposed solution can be conceptualized as a functional methodology to transform data to knowledge, and then to extend this transformation to include the learning of knowledge. A discussion of the featured methodology and architectural inter-connectivity is given in Chapter 3. It should be emphasized that the presented ***Methodology for Knowledge Discovery and Learning (MKDL)*** is intended to be fundamentally *open* in nature, thereby allowing for integration of various mechanisms so long as certain data transformation and knowledge conceptualization are satisfied. MKDL can therefore be described as a generic specification for knowledge discovery and learning framework with certain internalized tasks, i.e. data preprocessing, data clustering via boosting ensemble technique, data discretization, rule generation, rule filtering and learning via bagging ensemble technique – the sequential execution of which enables effective conceptualization of real-world data.

At the core MKDL is an innovative synergy of six (6) mechanisms which makes possible the semi-automated discovery of knowledge from unannotated data. The mechanisms, which the framework capitalizes upon, are briefly explained in the following points.

- *Data Preprocessing:* Missing values from the dataset are filled using statistical methods. We explored 2 different methods to compare which method could give better results from this preprocessing process in terms of accuracy and logic reasoning.

- *Data Clustering Ensemble:* A finite set of clusters is identified to describe the data by modeling the similarities or dissimilarities of the object space, thus construction the classes or categories necessary for rule generation. We used a novel approach for this purpose called clustering ensemble and we would like to highlight the novelty of using this method in our thesis.
- *Data Discretization:* This involves the transformation of attributes with continuous or real values to those with a finite number of discrete intervals or bins. Although the process will coarsen the representation of the actual data, but interestingly, it has the potential to reduce overfitting and improves the predictive capability of the derived rules. Again, two different methods were explored to make a comparison to identify which methods produce better discretization results.
- *Rule Generation:* This involves the formation of non-deterministic classification or decision rules from annotated data. The main objective would be to find an accurate, generalized and compact description of the data in the form of *if-then* rules. Here, a rule generation approach was employed which leverages on the theory of rough set (Pawlak, 1982), a relatively recent mathematical approach to reasoning about imprecision in data.
- *Rule Filtering:* The rules that have been generated will be filtered out for high quality. Rules that are of low quality (lower than the threshold) will be removed. This will ensure that the knowledge discovered from the data is compact.
- *Learning using Ensemble:* Rules instances which are filtered are trained using neural network ensemble methods. We use a novel combination of neural networks (or neural network ensemble called bagging technique) to train these rules instance where we can later predict new cases with higher accuracy. Our

main concern at this phase is to generate the desired result to be as close or similar to the actual result.

MKDL can be therefore considered as an amalgamation of various (both unsupervised and supervised) machine learning techniques and methods. Most important of all is the incorporation of the "ensembling techniques" into MKDL.

1.3.3 Contributions

The following are our 4 research contributions and objectives:-

1. ***An effective clustering ensemble technique:*** A simple but novel and effective heuristic was formulated to generate more accurate clustering compared to those used in other knowledge discovery framework. We used Arcing-x4, a boosting method (ensemble technique) within Self Organizing Map (SOM), to generate more accurate clustering results from the cleansed data of the previous phase. This method is known as clustering ensemble which produces more accurate results.
2. ***Comparing different discretization techniques:*** For this contribution, two methods were explored, analyzed and compared: (1) Boolean Reasoning based on *equation* (Nguyen and Skowron, 1995), and (2) Entropy-MDL based on *entropy* (McEliece, 1977) reduction and principles of the *Minimum Description Length*, or MDL (Rissanen, 1986) criterion.

3. ***Extension of knowledge discovery framework to include rule learning via ensemble technique:*** This thesis demonstrates the applicability of including rule learning via ensemble technique after the rules have been generated and filtered. This is novel. Many knowledge discovery frameworks stop after the rule generation phase or the rule filtering phase.

   A novel community (ensemble) of ANN algorithms or classifiers was used which involved Multilayer Perceptron (MLP), Generalized Feedforward Network (GFN), Modular Neural Network (MNN) and Radial Basis Function Networks (RBF). The generated rules were then subjected to the neural network ensemble's bagging technique to produce higher accuracy in the prediction and make decision support more reliable and trustworthy.

4. ***Multi-tier Knowledge Discovery, Amalgamation and Learning Info-Structure* (MESTAC)*:*** The realization of a generic, fully automated knowledge discovery system is still far from reach. However this thesis shows that the combinative and linear application of particular data mining and learning mechanisms as in the MESTAC is a promising step in the development of more comprehensive and generic systems for knowledge discovery and the learning of these knowledge from real-world databases. We have tested MESTAC on medical dataset collected from patients with breast cancer.

   After researching on different algorithms, different knowledge discovery frameworks, better learning methods, data mining strategies and so forth, we came up with our own methodology designed with six different phases performing different functions towards one goal which was better accuracy in prediction from unannotated data. Results were promising as each phase generated a better or more optimum result which proved our hypothesis very valuable at the end of the day.

1.4 Thesis Outline

The remainder of this thesis is organized in the following manner:-

Chapter 2 – *(Literature Review):* Reviews various mechanisms for data preprocessing, clustering ensemble, data discretization, rule generation, rule filtering and learning which support the main objective of this thesis. Apart from reviewing the various algorithms for our six phases, we also reviewed two very important knowledge discovery frameworks.

Chapter 3 – *(A Methodology for Knowledge Discovery and Learning):* Presents the motivation behind the conceptual MKDL and, subsequently, we present details on the multi-mechanism pipeline known as ***MESTAC.*** We also discuss the advantages of MESTAC in terms of the main contributions and also comparing them to the former knowledge discovery frameworks.

Chapter 4 – *(Implementation):* In this chapter we present the training and testing strategy in the evaluation of the knowledge together with the mechanism details and experimental methodology. We discussed the consideration for the implementation and the fundamental requirements of MESTAC. We argued on the design approach that we chose and explained why we used the particular approach. The architectural overview of MESTAC which describes the main phases is also presented.

Chapter 5 – *(Case Study and Results):* Here we present experimental evidence on the capability of our proposed knowledge discovery and learning framework. The featured knowledge discovery and learning solution is applied on a well-studied data-set - i.e. breast cancer prognosis.

This thesis concludes in **Chapter 6** with a summary of the research undertaken and results obtained, and followed by the identification of interesting directions for future research.

Chapter 2

Literature Review

There are two sides to every question.

- Protagoras (485 BC - 421 BC)

We don't know a millionth of one percent about anything.

- Thomas Edison (1847 - 1931)

2.1 Introduction

In general data mining (sometimes also called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information and knowledge. This in turn can be used in decision making. Data mining and knowledge discovery is the process of finding correlations or patterns among dozens of fields or attributes in large relational databases using a combination of various methods. This includes using statistics, artificial intelligence and machine learning among others.

In this chapter of literature review, we introduce different data mining and knowledge discovery mechanisms which are popularly used in most data mining and machine learning exercises in the following methodology for knowledge discovery and learning for unannotated datasets. We will also emphasize on the functions of each data mining mechanisms. We will then explain what we could do with all these different data mining mechanisms. We will then explain how current knowledge discovery frameworks (Risvik, 1997) and (Abidi and Hoe, 2002) employ these mechanisms. Once this is done we

will discuss and evaluated these existing frameworks and finally conclude by indicating what we have decided to do from our in-depth literature review.

2.2 Data Preprocessing

The main function of data preprocessing is to prepare the data (raw data) to be processed or fed into another processing phase – e.g. prediction, clustering, and discretization among others. Data preprocessing is performed because real world data are generally incomplete, noisy and/or inconsistent.

In order to address the problems associated with real world data, four main data preprocessing methods are commonly employed:- data cleaning, integration, transformation and reduction.

2.2.1 Data Cleaning

Essentially, data cleaning is carried out for the following purposes:

***1) To fill in missing values (attribute or class value)*:** There are few possible approaches to do this as stated below:-

- Ignore the tuple: This is usually done when class label is missing. A tuple is defined in the same way as a list, except that the whole set of elements is enclosed in parentheses instead of square brackets, e.g. tuple = ("a", "b", "c", "d", "e").
- Use the attribute mean (or majority nominal value) to fill in the missing values: This is done by substituting missing values of numerical attributes with the mean value for all observed entries for that attribute. For string attributes, missing values can be substituted by the “mode” value, which is the most frequently occurring value among the observed entries for that attribute.

If a and $\hat{a}$ denote an attribute before and after completion, we have:

$$O_a = \{x \in U \mid a(x) \neq \mathrm{T}\} \tag{2.1}$$

$$O_a^v = \{x \in O_a \mid a(x) \neq v\} \tag{2.2}$$

$$\hat{a}(x) = \begin{cases} a(x) & \text{if } x \in O \\ \dfrac{1}{|O_a|}\sum a(x) & \text{if } x \notin O_a \text{ and } a \text{ is numerical} \\ \arg\max_v \left|O_a^v\right| & \text{if } x \notin O_a \text{ and } a \text{ is not numerical} \end{cases} \tag{2.3}$$

Here, ties for mode values are resolved arbitrarily.

- Use of combinatorial completion: This method expands each missing value for each object into the set of possible values. Therefore, an object here is expanded into several objects covering all the possible combinations of the object's missing values. For example, let us assume a case has missing values for condition attributes a and b, and let |Va| = 3 and |Vb| = 4. The single incomplete case is then expanded into 12 complete cases, covering all possible combinations of values for a and b. This could cause the number of possible combination for cases with multiple missing values to grow very rapidly.
- Predict the missing value by using a learning algorithm: Consider the attribute with the missing value as a dependent (class) variable and run a learning algorithm (usually Bayes or Decision Tree) to predict the missing value.

***2) Identify outlier and smooth out noisy data*:** Possible approaches for this purpose are (1) *binning* – sort the attribute values and partition them into bins, and then smooth them by bin means, bin median, or bin boundaries, (2) *clustering* - group values in clusters and then detect and remove outliers (this can be done

either automatically or manually), and (3) *regression* - smooth by fitting the data into regression functions.

3) ***Correct inconsistent data***: This is done by using domain knowledge or expert decision

2.2.2 Data Integration

Data comes from different sources and may present the following problems:-

1) Same concept but different attribute name: (e.g. ssn; social_security; student_ssn)
2) Same value expressed differently: (e.g. undergraduate; ug)
3) Repeated tuple with different source database

These will cause inconsistencies in the data and lead to data redundancy. Data integration is employed to consolidate different source into one data repository. Usually this is called data warehousing and the process is often referred as schema-reconsolidation. Two methods of performing schema-reconsolidation are by metadata and correlation analysis.

Formatted: Not Highlight

Formatted: Not Highlight

2.2.3 Data Transformation

Normalization is a method used in data transformation. Normalization is done as the range of attribute (features) values differ, thus one feature may overpower the other. Two methods employed in normalization are (1) scaling attribute values to fall within a specified range, and (2) scaling by using mean and standard deviation.

2.2.4 Data Reduction

Data reduction reduces huge dataset to smaller representation which can give a clearer picture of the data representation. Reducing the number of attributes is one method of

performing data reduction. This is done by removing irrelevant attributes. Attributes that do not have a value at a few instances at a specific column is removed from the dataset.

2.2.5 Choosing the Right Data Preprocessing Method

Each data preprocessing technique has its own set of assumptions, strengths and also weakness, which must be considered for the problem at hand. The following are some of the considerations when deciding on the data preprocessing technique:

- Data model: The data preprocessing technique that can be chosen is based on what is to be done and for what purpose. For example, if there are missing attribute values in the dataset, two options that can be performed are (1) data cleaning using filling missing values, and (2) data reduction where the whole attribute will be removed from the dataset.
- Time and space requirements: The algorithmic complexity of the data preprocessing algorithm will be proportional to its overall running time. Algorithmic complexities are determined using the asymptotic complexity measure. This is written in *big-O* notation. This computation is a hypothetical estimate of the amount of computational time which an algorithm will take as the size of its inputs increases. Space requirements refer to the maximum amount of storage that is wanted at any instance during the implementation and execution of data preprocessing. Space requirements normally are also specified using *big-O* notation.
- Results interpretability: It is important that the data preprocessing application is able to provide good descriptions of its results. It should be easily understood by the end-user. Such description can then be understood by the end-user as an insight and new information of the domain of interest.

2.3 Data Clustering

Data clustering is an explorative task that seeks to identify groups of similar objects based on the values of their attributes (Hartigan, 1975; Spath, 1980). Clustering works on the inherent characteristic of the data. It also attempts to discover different groupings and boundaries to divide the data-set into meaningful partitions. The underlying hypothesis of data clustering is that the data is not totally random and that there exist some "hidden" patterns or concepts. This can both be revealed by the clustering effort or form the basis for grouping data-points into higher-level and consolidated groups of data-item. Once this is performed, it is called clusters or classes.

The former property of a group or cluster is known as the *intra-cluster similarity* while the latter is known as the *inter-cluster distance.* Clustering aims to maximize both intra-cluster similarity and inter-cluster distance. A *similarity measure* is essentially a function which, based on the attribute values of given object, computes a real value which indicates the degree of similarity between individual objects or groups. Likewise, a *distance measure* returns a real value indicating the magnitude of dissimilarity or difference between individual objects or groups. Collectively, similarity and distance measure functions used in clustering are called *clustering criterions*. The effectiveness of a clustering technique, i.e. its ability to produce meaningful clusters, depends primarily on the clustering criterion being optimized and the optimization algorithms utilized.

There are numerous ways to cluster data. The popular Kohonen Self-Organizing Map is among those highlighted here.

2.3.1 Generalized Clustering Algorithm

Clustering algorithms have mainly adopted two different approaches – referred to as generalized clustering algorithms – to search the space of objects and group them into clusters. These two approaches are:

- Partitional clustering: This approach attempts to directly divide a data set into disjoint sets (clusters) based on some measures of dissimilarity or distance between objects on the data. Every object is then assigned to exactly one cluster.
- Hierarchical clustering: This approach builds a tree-like structure of the given data where the different levels of the tree represent subsets (clusters) of the data at different granularity.

These 2 methods will not be used in our thesis contributions. Therefore we will not go into the details about it.

2.3.2 Neural Network Clustering Algorithm

In 1960, vector quantization problems were studied by mathematicians (Glienn, 1964; Stratonowitch, 1964). In 1973, Von Der Malsburg did the first computer simulation demonstrating self-organization. In 1976, Willshaw and Von Der Malsburg suggested the idea of Self-Organizing Map (SOM). In 1980's work done by Kohonen further developed and studied computational algorithm for SOM.

The Self-Organizing Map (SOM) with its variations is the most popular artificial neural network algorithm in the unsupervised learning category. SOMs work somewhat like K-means clustering but are a little richer. With K-Means, you choose the number of clusters to fit the data into. For a SOM you choose the shape and size of a network of clusters to fit the data into. In a SOM, we call these clusters 'nodes'. Much like for K-

Means clustering, you should choose an initial size based on what you suspect about the number of classes in your data.

Like K-Means, a SOM initially populates its nodes or clusters by randomly sampling the data (or randomly generating points in the data space, depending on the initialization option you choose), and then refines the nodes in a systematic fashion. Unlike K-Means clustering, however, a SOM will not force there to be exactly as many clusters as there are nodes, because it is possible for a node to end up without any associated cluster items when the map is complete. A further difference with K-Means clustering is that the SOM automatically provides some information on the similarity between nodes - i.e., how strongly the certain nodes resemble each other.

2.3.3 Kohonen Self Organizing Map (SOM)

The Kohonen SOM is a multivariate analysis method. Points that have smaller Euclidean Distance between them are grouped into the same cluster ~~(see~~as shown in Figure 2.1 below.

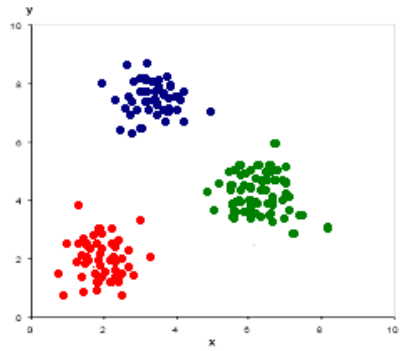
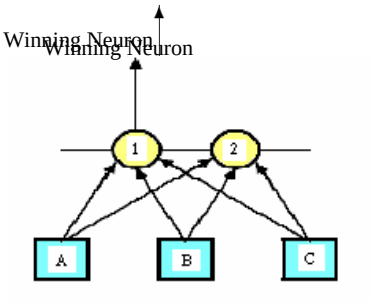


Figure 2.1 Clusters with smaller Euclidean Distance are grouped together to form 3 different clusters.

2.3.4 SOM Architecture

SOM uses Neural Network without hidden layers and with neurons in the output layer competing with each other, so that only one neuron (the winner) can fire at a time.



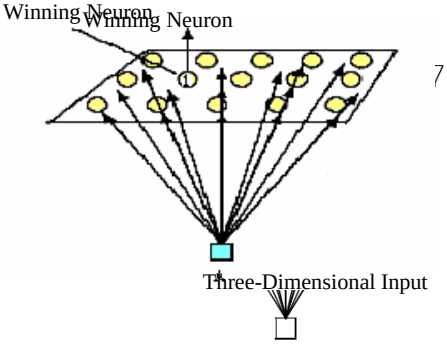


Figure 2.2 One-Dimensional output lattice

Figure 2.3 Two-Dimensional output lattice

- Input layer has *n* nodes. We can represent an input pattern by *n*-dimensional vector $\mathrm{x} = (x_1, ..., x_n) \in \mathrm{R}^n$.
- Each neuron *j* on the output layer is connected to all input nodes, so each neuron has n weights. We represent by *n*-dimensional vector $\mathrm{w}_j = (w_{1j}, ..., w_{nj}) \in \mathrm{R}^n$.
- Usually neurons in the output layer are arranged in a line (one-dimensional lattice) or in a plane (two-dimensional lattice).

- SOM uses unsupervised learning algorithm, which organizes weights w_j in the output lattice so that they "mimic" the characteristic of the input patterns.

2.3.5 How does SOM Work

The algorithm consists of 3 processes:- competition, cooperation, and adaptation

Competition

Input pattern $x = (x_1, ..., x_n)$ is compared with the weight vector $w_j = (w_{1j}, ..., w_{nj})$ of every neuron in the output layer. The winner is the neuron whose weight w_j is the closest to the input x in terms of Euclidean distance:

$$\begin{aligned} \| x - w_1 \| &= \sqrt{(x_1 - w_{11})^2 + ... + (x_n - w_{n1})^2} \\ &\vdots \\ \| x - w_m \| &= \sqrt{(x_1 - w_{1m})^2 + ... + (x_n - w_{nm})^2} \end{aligned} \quad (2.4)$$

Cooperation

The winner helps its neighbours in the output lattice. Those nodes which are closer to the winner in the lattice get more help, to those which are further.

If the winner is node i, then the amount of help to node j is calculated using the neighbourhood function $h_{ij}(d_{ij})$, where d_{ij} is the distance between i and j in the lattice. A good example of $h_{ij}(d)$ is the Gaussian bell function: $h_{ij}(d) = e^{-\frac{d^2}{2\sigma^2}}$

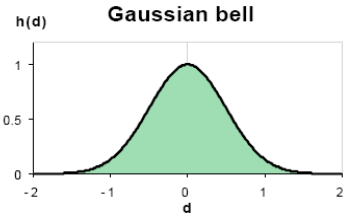


Figure 2.4 Gaussian Bell Function

Do take note that the winner also helps itself more than others for $d_{ii} = 0$

Adaptation

After the input x has been presented to SOM, the weights w_j are adjusted so that they become "closer" to the input. The exact formula for adaptation of weights is:

$$w_j' = w_j + \alpha h_{ij}[x - w_j] \quad (2.5)$$

where α is the learning rate coefficient.

One can see that the amount of change depends on the neighbourhood h_{ij} of the winner. So, the winner helps itself and its neighbours to adapt. Finally the neighbourhood h_{ij} is also function of time, such that the neighborhood shrinks with time, i.e. α decreases with t.

Training Procedure for SOM

1. Initially set all the weights to some random values.

2. Feed a set of data into the network.

3. Find the winner.

4. Adjust the weights of the winner and its neighbours to be more like the input.

5. Repeat from step 2 until the network is stabilized.

2.3.6 Choosing the Right Clustering Technique

Clustering techniques can be evaluated by how they model the input data, their time and space requirements, scalability, input order sensitivity, noise handling mechanism and the interpretability of their results. The following are some of the considerations when deciding on the data clustering technique:

- **Data model:** This is a technique which can be chosen based on the type of data, the inherent pattern that the clustering exercise is meant to capture, the anticipated form of clusters and also the available prior knowledge form the domain. Many clustering algorithms can only work with a particular type of input. K-means, BIRCH (Zhang et. al, 1996) and CURE (Guha et. al, 1998) accept only continuous data as each defines similarity or dissimilarity as actual distances among objects. SOM uses Euclidean distance function which assists to define the similarity or dissimilarity of continuous data. The benefit is that even discrete number can be used with SOM. This occurs because SOM uses neural network algorithm to learn. K-modes (Huang, 1997b) and ROCK (Guha et.al., 1999) only cater for categorical data. SBAC (Li & Biswas, 1998) a hierarchical clustering algorithm uses Goodall similarity measure to cluster datasets which has both continuous and categorical attributes. There are clustering algorithms which also presume some fixed canonical distribution for the input data. One example is, AUTOCLASS (Cheeseman & Stutz, 1996), which is based upon the classical finite mixture model.

This example imposes the subsequent probability density function for different data types: i.e. Bernoulli distributions for nominal attributes and Gaussian distribution for continuous attributes.

- **Time and Space Requirements:** This is a similar consideration to the time and space requirements for data preprocessing ~~(see Section XXX~~ as in Section 2.2.6.
- **Scalability:** Today as scientific, transactional and business activity continues to generate a huge amount of data, very large database will potentially overwhelm even the most powerful computers. Therefore, it is desirable for a clustering technique to be faster and also able to scale well with the dimensions and size of these large quantities of data, and yet produce better results in terms of accuracy.
- **Input Order Sensitivity:** This clause pertains to the change in performance or results of a clustering algorithm when the arrangement of its input is changed. A clustering algorithm should be insensitive to such alteration. However, most available partitional and hierarchical clustering algorithms are sensitive to presentation order of the input to different degrees.
- **Noise Handling Mechanism:** One of the noticeable differences of conventional and current clustering algorithms is that the former tend to ignore the existence of noise or outliers in the input data. It also did not provide any mechanisms to overcome them. More current clustering techniques, have inculcated specific mechanisms to identify and isolate noise of the data and the same time to mitigate their effects on the final results. One example is the use of ensemble technique within clustering methods. This is called clustering ensemble. More details about ensemble technique are discussed in section 2.8.
- **Results Interpretability:** It is particularly advantages that the descriptions are represented in a precise and simple form. This could be like the propositional rules.