



Data Clustering using
Self Organizing Feature Map

By

Mr. Sanhapon Thadapradit

Submitted in Partial Fulfillment of
the Requirements for the Degree of
Master of Science
in Computer Science
Assumption University

December, 2000

Data Clustering using Self Organizing Feature Map

by

Mr. Sanhapon Thadapradit

The seal of Assumption University of Thailand is a circular emblem. It features a central shield with four quadrants: top-left (blue with a white lily), top-right (white with a blue sailboat), bottom-left (white with a blue star), and bottom-right (red with a white cross and the letters 'DS'). The shield is flanked by golden laurel branches. Above the shield is a golden crown. The circular border contains the text 'ASSUMPTION UNIVERSITY OF THAILAND' at the top and 'BROTHERS OF THE HOLY GABRIEL' at the bottom. Below the shield, it says 'SINCE 1969'.

**Submitted in Partial Fulfillment of
the Requirements for the Degree of
Master of Science
in Computer Science
Assumption University**

December, 2000

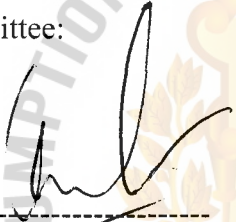
The Faculty of Science and Technology

Thesis Approval


| | |
|----------------|---|
| Thesis Title | Data Clustering Using Self Organizing Feature Map |
| By | Mr. Sanhapon Thadapradit |
| Thesis Advisor | Asst.Prof.Dr. Tang Van To |
| Academic Year | 1/2000 |


The Department of Computer Science, Faculty of Science and Technology of Assumption University has approved this final report of the **twelve** credits course. **SC7000 Master Thesis**, submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science.

Approval Committee:



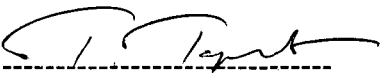
(Asst.Prof.Dr. Tang Van To)
Advisor

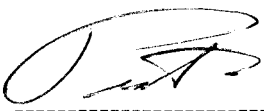
(Dr. Thitipong Tanprasert)
Committee

(Dr. Jirapun Daengdej)
Committee

(Asst.Prof.Dr. Surapong Auwatanamongkol)
Representative of Ministry of
University Affairs

Faculty Approval:



(Dr. Thitipong Tanprasert)
Director

(Asst.Prof.Dr. Pratit Santiprabhob)
Dean

December / 2000

TABLE OF CONTENTS

| | |
|---|-----|
| ACKNOWLEDGEMENT | i |
| ABSTRACT | ii |
| LIST OF FIGURES | iii |
| LIST OF TABLES | iv |
| CHAPTER 1: INTRODUCTION | 1 |
| 1.1 Motivation of problem | 1 |
| 1.2 Statement of the problem | 2 |
| 1.3 Objectives and scopes of the study | 3 |
| CHAPTER 2: LITERATURE REVIEW | 4 |
| CHAPTER 3: THEORY BACKGROUND | 6 |
| 3.1 Knowledge Discovery from Database | 6 |
| 3.1.1 Requirements of Knowledge Discovery | 6 |
| 3.1.2 The most commonly used Techniques in KD | 8 |
| 3.2 Learning Techniques | 9 |
| 3.3 Similarity Measures | 10 |
| 3.4 Clustering and Similarity Measures | 12 |
| 3.5 Clustering algorithms | 13 |
| 3.5.1 Sequential algorithms | 13 |
| 3.5.2 Hierarchical clustering algorithms | 15 |
| 3.5.3 Clustering algorithms based on | 18 |
| cost function optimization | |
| 3.5.4 Others clustering algorithms | 19 |
| 3.5.5 Clustering validation | 20 |

| | |
|--|----|
| 3.6 Winner-Take-All Learning | 23 |
| 3.7 Self Organizing Feature Map | 27 |
| 3.7.1 Initialization of Weights | 29 |
| 3.7.2 The Main Characteristics of SOFM | 30 |
| 3.7.3 The Learning Algorithm | 30 |
| 3.8 Self Organizing Semantic Map | 32 |
| 3.9 Multiple Linear Regression | 35 |
| CHAPTER 4: PROPOSED METHODOLOGY | 37 |
| 4.1 Classification using Kohonen Network | 37 |
| 4.2 The processes to classify using Kohonen Network | 38 |
| 4.3 Establishing model for each cluster using MRE | 40 |
| 4.4 Predict the GPA from available data | 41 |
| CHAPTER 5: EXPERIMENTS AND DISCUSSION | 42 |
| 5.1 Soybean data clustering | 42 |
| 5.2 Windowing Assumption University's student data clustering | 45 |
| 5.3 Randomly selected Assumption University's student data clustering | 49 |
| 5.3.1 Factors of cluster | 55 |
| 5.3.2 The effect of English and Mathematics to GPA | 55 |
| CHAPTER 6: CONCLUSION AND RECOMMENDATION | 57 |
| REFERENCES | 58 |
| APPENDIX A: EXPERIMENT 1 | 60 |
| APPENDIX B: EXPERIMENT 2 | 64 |
| APPENDIX C: EXPERIMENT 3 | 75 |

ACKNOWLEDGEMENT

I would like to thank you my family who always cheer and encourage me to continue the study in this master degree.

I would like to thank you my advisor, Asst. Prof. Dr. Tang Van To, for his help. He always presents me his kindness, gives me the inestimable suggestions when I faced the trouble problems. This may be my last chance to say sorry to him for my mistakes during my research.

I would like to acknowledge my committees, Dr. Thitipong Tanprasert, Dr. Jirapun Daengdej and Asst. Prof. Dr. Surapong Auwatanamongkol., the external committee from ministry of university affairs, who spend their valuable time on my thesis presentation and give me the worthy comments and suggestions.

Many thanks to Nulek, P'Ta, P'Nok, P' Poon and all my friends for their help and their friendship at Assumption University.

ABSTRACT

From the data kept in the powerful database tools, we find the hidden features or characteristics of these data that we can not find from functions in the existing database tools.

This research proposes the framework to classify the student's academic records of students for unveiling how Mathematics and English background affects to the achievement of students in Assumption University.

Kohonen network was used to find the map of activated output neurons and to find the clusters from this map. After that we built the models from clusters by multiple regression to predict the value of GPA based on independent attributes. We compared the predicted GPA, which was calculated by model of appropriate cluster (local model) to the model from a whole data (global model). It is shown that the local models predict more accurate than the global model.

We also tested the clustering methodology with Soybean data from UCI database repository, the clustering gave an appropriate result.

The model from a whole data shows that both English and Mathematics strongly influence to the study performance of Assumption University's students and English has more significant influence than Mathematics. Only English or Mathematics affects to the model of some clusters. Some cluster does not depend on English or Mathematics.

LIST OF FIGURES

| | | |
|-------------|---|----|
| Figure 3-1 | Supervised learning model | 9 |
| Figure 3-2 | Unsupervised learning model | 10 |
| Figure 3-3 | Measures of similarity for clustering data using Euclidean distance | 11 |
| Figure 3-4 | Measures of similarity for clustering data using cosine | 11 |
| Figure 3-5 | The clustering hierarchy based on <i>GAS</i> | 16 |
| Figure 3-6 | Learning layer | 23 |
| Figure 3-7 | Weight vector adaptation | 25 |
| Figure 3-8 | Self-Organizing Feature Map Model | 27 |
| Figure 3-9 | Example of topological neighborhood $N(t)$ | 28 |
| Figure 3-10 | Self-Organizing feature map for two input nodes, arrays of neurons, and uniformly distributed excitation | 31 |
| Figure 3-11 | Table of attributes for the set of $P=16$ animal objects | 33 |
| Figure 3-12 | Strongest response due to the symbol part only excitation | 34 |
| Figure 3-13 | Strongest response domains | 35 |
| Figure 4-1 | The process to establish model using neuron network | 38 |
| Figure 4-2 | Model from each cluster | 40 |
| Figure 5-1 | Clusters from the strongest response of all input | 44 |
| Figure 5-2 | The activated neuron of each input | 44 |

LIST OF TABLES

| | | |
|------------|---|----|
| Table 5-1 | The color used to paint activated neurons | 43 |
| Table 5-2 | The criteria to select sample data (windowing data) | 45 |
| Table 5-3 | Subject list used in the experiment | 46 |
| Table 5-4 | Numerization of nonnumeric data | 46 |
| Table 5-5 | The colors used to paint activated neurons versus GPA of the second semester | 47 |
| Table 5-6 | Characteristic of each cluster | 47 |
| Table 5-7 | Cluster in each group | 48 |
| Table 5-8 | The number of data in each cluster compare to the number of data in matching sample data group | 48 |
| Table 5-9 | The criterion of selecting sample data | 50 |
| Table 5-10 | Subject list used in the experiment | 51 |
| Table 5-11 | Range of GPA, color, and cluster in experiment | 51 |
| Table 5-12 | Models with testing t-test hypothesis | 52 |
| Table 5-13 | Predicted value of GPA2s with testing t-test hypothesis | 53 |
| Table 5-14 | Means square error from predicted value of GPA2 | 54 |

CHAPTER 1

INTRODUCTION

1.1 Motivation of study

At the present time, the amount of data kept in the database has a trend to be very large. It is kept in the powerful storage, managed and maintained. Normally, we only maintain this database with add, delete, append operations and make queries on them to get the desired data. Using database tools and technology advancement can help us decrease time processing, do backup easily, manage the data efficiently. We have rich data but poor information, in fact we could use data more powerful by extracting the implicit information, for example, to predict the future trends of data, or to find the meaning or characteristic of this data that we have never known from traditional transaction.

Knowledge discovery from database (KDD) also called data mining is a mechanism that can help to answer above questions. It finds relationship and global patterns that are existed in a large database, but they are hidden among the vast amount of data. These relationships present a valuable knowledge about the data.

The self-organizing feature map (SOFM), one of the major unsupervised learning methods in the area of artificial neural networks, can be used for knowledge discovery from database mechanism. Using self-organizing feature map, the spatially patterns or clusters will be created; we use these patterns to analyze the whole data to find useful information.

Sometimes the whole data set should not be analyzed at once. If there are several distinct clusters, we should consider either analyzing each cluster separately or completing the analysis that takes account of the different groups

In this study, we classify the data into cluster by SOFM, and use Multiple Regression to find how attributes affect to GPA for each cluster as well as to forecast the academic achievement of undergraduate students based on history data such as basic subjects (Mathematics and English), the number of credits earned and GPA in some recent semesters. From a set of inputs, they can self-adjust to produce consistent response. A network is trained so that set of inputs produces a desired set of outputs or spatially patterns. These are the clusters that are used to model for prediction.

1.2 Statement of the problem

From the huge size of data kept in the database, how can we extract the hidden information? How can we know the characteristic of the complicated data? How can we get the benefit from them?

A number of techniques from different fields have been proposed and used with varying success for knowledge discovery. Each kind has its own merits and demerits. By comparing their pros and cons, it is not possible to judge any of the methods as the best. This will need to be studied further.

This study deals with knowledge extraction that uses Kohonen network and clustering methodology to classify data and then applies statistic method to establish the models of the clusters.

The expected contribution is how to use the Kohonen network to classify the academic records of students and how Mathematics and English affect to the

academic achievement of the Assumption University students. Then, once the academic records of the student are clustered, modeled to show how the last academic records affect to the performance of students in the next semester for each cluster. Finally we use these models as a predictor to forecast the study performance for students based on their last performances.

1.3 Objectives and scopes of the study

This work is aimed to demonstrate an approach on how artificial neural network technology can help to address the problem of knowledge discovery from database or data mining apply to statistic method. The main objectives of the study are:

1. To study the developments in knowledge discovery.
2. To study and apply SOFM to knowledge discovery.
3. To discover knowledge in terms of pattern information or clusters.
4. To create model from data by using unsupervised learning and statistic method.

CHAPTER 2

LITERATURE REVIEW

There are a lot of works that deal with KDD but, in our research, we emphasize on Self-Organizing Feature Map approach to set up the context of our research.

Sergios Theodoridis, in his book, “Pattern Recognition” [15], provided several classification techniques, a measurement of the similarity and dissimilarity, clustering algorithms and cluster validation methods.

Ting-Peng Liang, “Research in Integrating Learning Capabilities into Information Systems” [16], reviewed major learning paradigms, examined the role of learning in intelligent information systems, and discussed potential research issues in integrating learning capabilities in information system design.

Ming-Syan Chen, “Data Mining: An Overview from a Database Perspective” [9], provided a survey on the data mining techniques, classification of the available data mining techniques, and a comparative study of such techniques.

Teuvo Kohonen published a book “Self-Organization and Associative memory” [6]. He gave a detail of SOM, associative memory and the classical learning systems.

Teuvo Kohonen published his paper “The Self-Organizing Map” [7]. The paper presents the basic architectures of SOM, algorithms and results in SOM. The Self-Organizing Semantic Maps have been also demonstrated in the paper.

Sushil Acharya, in his Ph.D., Thesis, “Knowledge Discovery and Self-Organizing System” [14], studied in SOM and applied 3-D graph technique to find the cluster boundary. He also analysed the effect on network training of data operations like delete, append and modification on the case database. The KD methodology has been tested on large standard database. Comparisons have been made with other techniques on knowledge based on the same or similar large database.

Murthy Ranjit, in his Master thesis, “Knowledge Discovery from Database using Self Organizing Maps” [10], SOM was used to map the student response data from 8 dimension to 2 dimension. The author also tested the network with other standard data, Breast Cancer Identifications Database, from UCI Repository. He analyzed the effect of network by performing the different network parameters.

Janon Ong and Syed Sibte Raza Abidi, in their paper, “Data Mining Using Self-Organizing Kohonen maps” [5], used Kohonen network to project high-dimensional data to two-dimension and then apply K-Means to find the cluster from the planar map. They also used U-Matrix method as a visualization technique for demarcating the trained SOM into distinct cluster of similar data elements.

CHAPTER 3

THEORY BACKGROUND

3.1 Knowledge Discovery from Database

Knowledge discovery (KD) or data mining is the process of discovering interesting, non-trivial patterns in data [3]. For the large size of data, there are hidden value information that are difficult to recognize. Normal transactions on database such as add/delete/append/query data help us comfortable to manage the data, convenient to visualize the data, and keep the data effectively. Queries allow us to know only the data, for example, it can deliver the employees who has month salary higher than 10,000 baht, but it could not derive automatically the knowledge that most of these people has high professional education.

In a huge database, there is a lot of hidden information. These can be useful for us to make a decision, to interpret the meaning of data, to find special characteristic, to get rid of noise data, to eliminate overfitting and to predict the trend of data. The knowledge discovered can be used to generate the model or structure of data.

3.1.1 Requirements of Knowledge Discovery

In order to conduct an effective KD, firstly one needs to examine what kind of features the KD system is expected to have and what kind of challenges one may face in the development of KD techniques. Some characteristics of KD [9] are:

a) Handling of different types of data

Because there are many kinds of databases and data used in different applications, it is expected that a KD system should be able to perform effectively KD on different kinds of data. Different data mining systems should be constructed for knowledge mining on different kinds of database, such as systems dedicated to knowledge mining in relational databases, transaction database, spatial database, multimedia database...

b) Efficiency and scalability of data mining algorithms

To effectively extract information from a huge amount of data in database, the knowledge discovery algorithms must be efficient and scalable.

c) Usefulness, certainty, and expressiveness of KD result

The discovered knowledge should accurately portray the contents of the database and be useful for certain applications. The imperfection and incompleteness must be expressed by measures of uncertainty, in the form of approximate rules or quantitative rules.

d) Different kinds of knowledge can be discovered from a large amount of data

This requires us to express both the data mining requests and the discovered knowledge friendly in high-level languages or graphical user interfaces so that the data mining task can be specified by nonexperts and the discovered knowledge can be understandable and directly usable by users.

e) Interactive mining knowledge at multiple abstraction levels

Since it is difficult to predict what exactly could be discovered from a database, a high-level KD query should be treated as a probe that may disclose some interesting traces for further exploration.

f) Mining information from different sources of data

g) Protection of privacy and data security

It is important to study when KD may lead to an invasion of privacy and what security measures can be developed for preventing the disclosure of sensitive information.

3.1.2 The most commonly used techniques in KD

a). Artificial neural networks: Non-linear predictive models that learn through training and resemble biological neural networks.

b). Decision tree: Tree-shaped structures that represent sets of decision generate rules for the classification of a data set.

c). Genetic algorithms: Optimization techniques that use process such as genetic combination, mutation, and natural selection in a design based on the concepts of evolution.

d). Nearest neighbor method: A technique that classifies each record in a data set based on a combination of the classes of the k records most similar to it in a historical data set. Sometimes it is called the k-nearest neighbor technique.

e). Rule induction: The extraction of useful if-then rules from data based on the statistical significance.

In this study, we concentrate on Self-Organizing Feature Map (SOFM), one method in Artificial Neural Network. It is unsupervised learning method, the method will learn the implicit relationships in the database without specifying any the prior knowledge which is sometimes difficult to know and understand. SOFM will be described in detail latter.

3.2 Learning Techniques

In order to understand the characteristics of data we need to figure out the model or structure inside the data, or we must have method to learn from the data and to present the outcome (a model or structure of the data). Learning methodology can be divided into two classes: Supervised Learning and Unsupervised Learning.

a) Supervised learning: Supervise leaning is the learning method that has a teacher or prior knowledge. When the input data is put to the system, a teacher or prior knowledge will check the correctness of input. If the model is satisfied we get the output, otherwise the system need to be modified (see Figure 3-1).

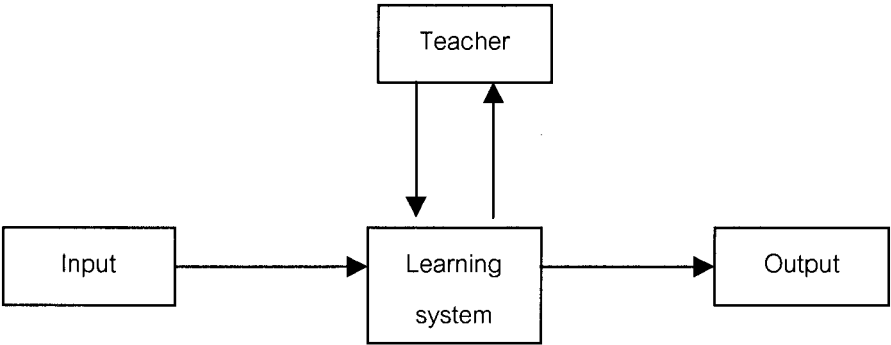


Figure 3-1: Supervised learning model

b) Unsupervised learning: In unsupervised learning or self-organized learning system, there is no external teacher to oversee the learning process. The data will be learned to catch their characteristics without prior knowledge. Clustering/classification is an unsupervised learning method; it clusters data into similar groups without external knowledge.

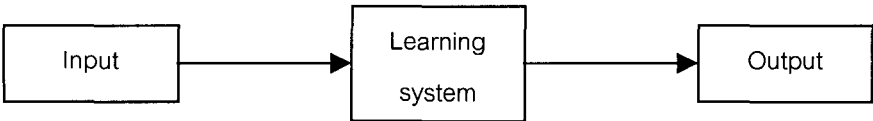


Figure 3-2: Unsupervised learning model

c) Supervised learning vs. unsupervised learning: Supervised learning is far more popular than unsupervised learning. Most learning methods used including ID3 [11], and neural networks are supervised learning methods. The mechanism of supervised learning is very similar to that of traditional discriminate analysis or statistical regression analysis used extensively in business problem solving [16]. Meanwhile, for unsupervised learning, there is no need of external knowledge. In several systems prior knowledge are not known and it is difficult to recognize. In this case, unsupervised learning method can play an important role.

3.3 Similarity measures

To define a cluster, we need to establish a basis for assigning patterns to the domain of a particular cluster. In this work we emphasize on the Euclidean distance between two patterns.

$$\|x - x_i\| = [(x - x_i) \cdot (x - x_i)]^{1/2} \quad (3.1)$$

This rule of similarity is simple; the smaller of distance, the closer of patterns.

Using Eq.3.1, the distances between all pairs of points are computed. A parameter t can be chosen to discriminate clusters, as the maximum distance between patterns within the same cluster [4].

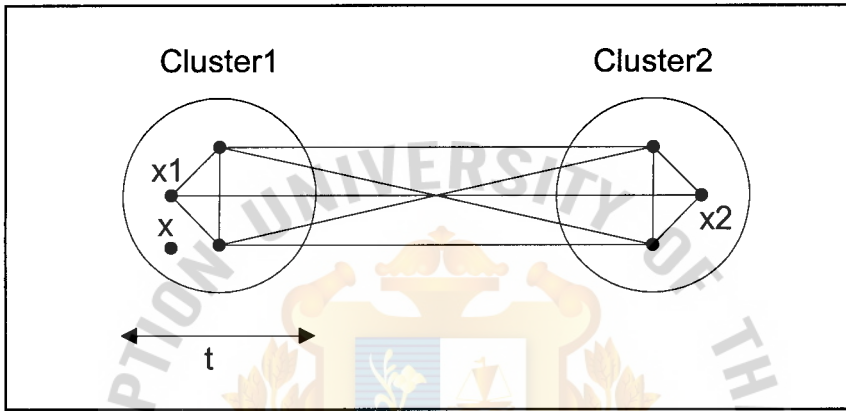


Figure 3-3: Measures of similarity for clustering data using Euclidean distance [4]

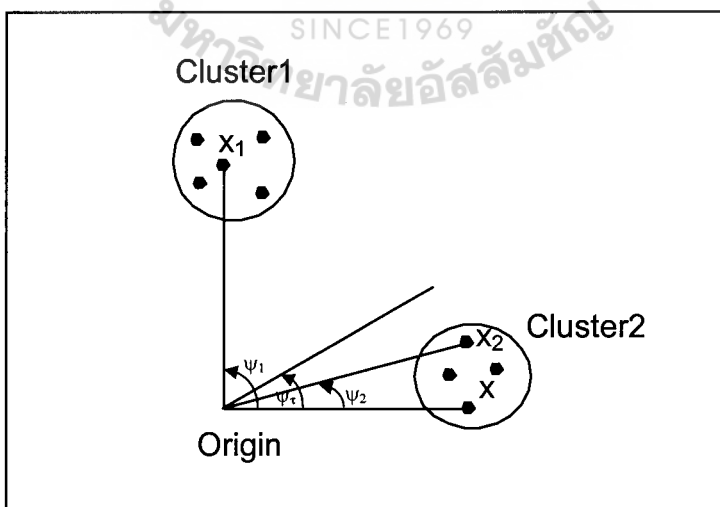


Figure 3-4: Measures of similarity for clustering data using cosine

Another similarity rule is the cosine of the angle between x and x_i

$$\cos \psi_i = x^t x_i / \|x\| \cdot \|x_i\| \quad (3.2)$$

If data vectors are normalized, then $\cos \psi_i = x^t x_i$. For $\cos \psi_2 < \cos \psi_1$, pattern x is more similar to x_2 than to x_1 . It would thus be natural to put x to the same cluster of x_2 instead to the same cluster of x_1 . To facilitate this decision, the threshold angle ψ_t can be chosen as the maximum angular distance (consine) between patterns in the same cluster.

3.4 Clustering and similarity measures

Clustering is one of the most primitive mental activities of humans, used to handle the huge amount of information they receive everyday. Processing every piece of information as a single entity would be impossible. Thus, humans tend to categorize entities in to clusters. Each cluster is then characterized by the common attributes of the entities it contains.

The basic step that an expert must follow in order to develop a clustering task is the following:

1. Feature selection
2. Proximity measurement
3. Clustering criterion selection
4. Clustering algorithm selection and performance
5. Validation of the results
6. Interpretation of the results

In a number of cases, a step known as clustering tendency should be involved. This includes various tests that indicates whether the available data posses a clustering structure.

Obviously different choices of features, proximity measures, clustering criteria and clustering algorithms may lead to totally different clustering results. Moreover, given the time and resource, it is impossible to figure out all patterns of clustering of N vectors into m groups. “Which cluster is correct?” It seems that there is no definite answer. The best thing to do is giving the results to an expert and let the expert decide about the most sensible one. Thus, the final answer to these questions will be influenced by the knowledge of the expert.

3.5 Clustering algorithms

3.5.1 Sequential algorithms

These algorithms produce a single clustering. They are quite straightforward and fast methods. In most of them, all the feature vectors are presented to the algorithm once or a few times. The final result depends on the order of data which is presented.

First, we consider the case where all the vectors are presented to the algorithm only once. The number of clusters is not known a priori in this case. In fact, new clusters are created as the algorithm evolves.

Let $d(x,C)$ denote the distance between a feature vector x and a cluster C . This may be defined by taking into account either all vectors of C (using the center) or a representative vector of it. The user-defined parameters required by the algorithmic

scheme are the threshold of the dissimilarity Θ and the maximum allowable number of clusters, q . The basic idea of the algorithm is as following: As each new vector is considered, it is either assigned to an existing cluster or assigned to a newly created cluster, depending on its distance from the already formed ones. Let m be the number of clusters that the algorithm has created up to now. Then the algorithmic scheme may be stated as:

Basic Sequential Algorithmic Scheme (*BSAS*)

1. $m = 1$
2. $C_m = \{ x_1 \}$
3. For $i = 2$ to N
 - 3.1 Find $C_k : d(x_i, C_k) = \min_{1 \leq j \leq m} d(x_i, C_j)$
 - 3.2 If $(d(x_i, C_k) > \Theta)$ and $(m < q)$ then
 - 3.2.1 $m = m + 1$
 - 3.2.2 $C_m = \{ x_i \}$
 - 3.3 Else
 - 3.3.1 $C_k = C_k \cup \{ x_i \}$
 - 3.3.2 Update representative vector for C_k .
- End {for}

It is not difficult to realize that the order in which the vectors are presented to the *BSAS* plays an important role in the clustering results. Different presentation ordering may lead to totally different clustering results, in terms of the number of clusters as well as the clusters themselves.

Another important factor affecting the result of the clustering is the choice of the threshold Θ . This value directly affects the number of clusters formed by *BSAS*. If Θ is too small, unnecessary clusters will be created. On the other hand, if Θ is too large a smaller than appropriate number of clusters will be created.

3.5.2 Hierarchical clustering algorithms

These algorithms are further divided into two following groups:

1. Agglomerative algorithms. These algorithms produce a sequence of clustering of decreasing number of clusters, one at each step. The clustering produced at each step results from the previous one by merging two clusters into one. The main representatives of the agglomerative algorithms are the single and complete link algorithms.

Let $g(C_i, C_j)$ be a similarity function defined for all possible pairs of clusters. This function measures the proximity between C_i and C_j . Let t denote the current level of hierarchy. Then, the general agglomerative scheme may be stated as follows:

Generalized Agglomerative Scheme (GAS)

1. Initialization:

1.1 Choose $R_0 = \{ C_i = \{x_i\}, i = 1, \dots, N \}$ as the initial clustering.

1.2 $t = 0$

2. Repeat:

2.1 $t = t + 1$

2.2 Amount all possible pairs of clusters (C_r, C_s) in R_{t-1} find the one say (C_i, C_j) , such that

2.2.1 $g(C_i, C_j) = \max_{r,s} g(C_r, C_s)$, if g is a similarity function

2.2.2 Defined $C_q = C_i \cup C_j$ and produce the new clustering $R_t = (R_{t-1} - \{C_i, C_j\}) \cup \{C_q\}$

Until all vectors lie in a single cluster.

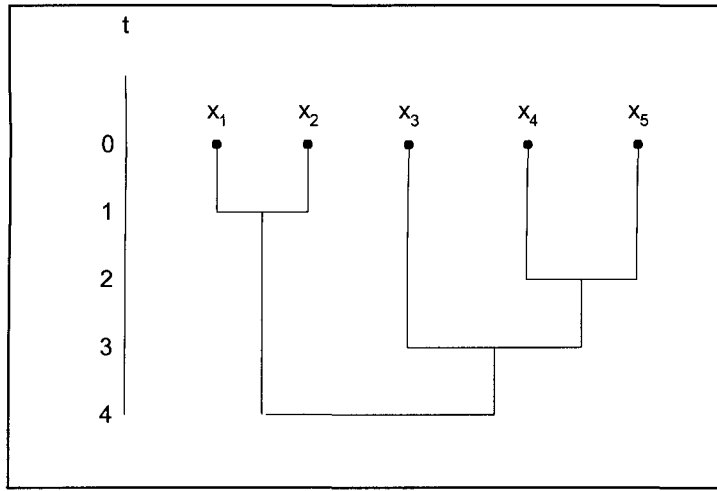


Figure 3-5: The clustering hierarchy based on *GAS*

It is clear that this scheme creates a hierarchy of N clustering, so that each one is nested in all successive clustering (see Figure 3-5). Alternatively, we can say that if two vectors come together into a single cluster at level t of the hierarchy, they will remain in the same cluster for all subsequent clustering.

2. Divisive algorithms. These algorithms act in the opposite direction; that is, they produce a sequence of clustering of increasing m at each step. The clustering produced at each step results from the previous one by splitting single cluster into two. The t^{th} clustering contains $t + 1$ clusters. C_{ij} will denote the j^{th} cluster of the t^{th} clustering R_t , $t = 0, \dots, N-1$, $j = 1, \dots, t+1$. Let $g(C_i, C_j)$ be a similarity function defined for all possible pairs of clusters. The initial clustering R_0 contains the whole set X , that is, $C_{01} = X$. To determine the next clustering, we consider all possible pairs of cluster that form a partition of X . Among them we choose the pair, denoted by (C_{11}, C_{12}) , that minimizes g . These clusters form the next clustering R_1 , that is, $R_1 = \{C_{11}, C_{12}\}$. At the next step we consider all possible pairs of clusters produced by C_{11}

and we choose the one that minimizes g . The same procedure is repeated for C_{12} . Assume that from the two result pairs of clusters, the one originating from C_{11} gives the larger value of g . Let denote this pair as (C_{11}^1, C_{11}^2) . Then the new clustering, R_2 , consists of C_{11}^1, C_{11}^2 , and C_{12} . Relabeling these clusters as C_{21}, C_{22}, C_{23} , respectively, we have $R_2 = \{C_{21}, C_{22}, C_{23}\}$. Carrying on in the same way, we form all subsequent clustering. The general divisive scheme may be stated as follow:

Generalized Divisive Scheme (GDS)

1. Initialization

1.1 Choose $R_0 = \{X\}$ as the initial clustering.

1.2 $t = 0$

2. Repeat

2.1 $t = t + 1$

2.2 for $i = 1$ to t

2.2.1 For all possible pairs of clusters (C_r, C_s) that form a partition of $C_{t-1,i}$, find the pair $(C_{t-1,i}^1, C_{t-1,i}^2)$ that gives the minimum value for g .

2.3 From the t pairs defined in the previous step, choose the one that has minimum g . Suppose that this is $(C_{t-1,j}^1, C_{t-1,j}^2)$.

2.4 The new clustering is

$$R_t = (R_{t-1} - \{C_{t-1,j}\}) \cup \{C_{t-1,j}^1, C_{t-1,j}^2\}$$

2.4 Relabel the clusters of R_t

Until each vector lies in a single distinct cluster.

A different choice of g gives different algorithms. One can easily observe that this divisive scheme needs a lot of computations, even for moderate values of N . This is the main drawback, compared with the agglomerative scheme.

3.5.3 Clustering algorithms based on cost function optimization

This category contains algorithms in which “sensible” is expressed by a cost function, J , in terms of which a clustering is evaluated. This category includes the following subcategories; Crisp clustering algorithms, Probabilistic clustering algorithms, Fuzzy-clustering algorithm and Boundary detection algorithms.

The cost J is a function of the vectors of the data set X and it is parameterized in terms of an unknown parameter vector, θ . For most of the schemes of the family the number of clusters, m , is assume to be known.

Our goal is to estimate the value of θ that characterizes the best clusters underlying X . The parameter vector θ depends strongly on the shape of the clusters such as compact or shell-shaped clusters. A distinct characteristic of most of algorithms in this category, compared with hierarchical clustering algorithms, is that the cluster representatives are computed using all the available vectors of X and not only the vectors which have been assigned to the respective cluster.

For example, K-means algorithm is one of the most popular clustering algorithms in this class. Euclidean distance is adopted to measure the dissimilarity between vector x_i and cluster representative θ_j .

K-means algorithm

1. Choose arbitrary initial estimates $\theta_j(0)$ for $j = 1, \dots, m$.
2. Repeat
 - 2.1 For $i = 1$ to N
 - 2.1.1 Determine the closet representative, θ_j , for x_i
 - 2.1.2 Set $b(i) = j$
- End {for}

2.2 For $j=1$ to m

2.2.1 Parameter updating: Determine θ_j as the mean of the vectors $x_i \in X$ with $b(i) = j$.

End {for}

Until no change in θ_j occurs between two successive iterations.

3.5.4 Other clustering algorithms

This last category contains some special clustering techniques that cannot be assigned to any of the previous categories.

- Minimum spanning tree algorithms. The idea of the algorithm is the following: determine the minimum spanning tree of G and then remove the edges that are unusually large compared with their neighboring edges. These edges are called inconsistent, and it is expected that they connect points from different clusters.
- Branch and bound clustering algorithms. These algorithms provide us with the globally optimal clustering without having to consider all possible clustering for fixed number m of clusters, and for a predefined criterion. However, their computational burden is excessive.
- Genetic clustering algorithms. These algorithms use an initial population of possible clustering and iteratively generate new populations, which, in general, contain better cluster than those of the previous generations, according to a predefined criterion.
- Stochastic relaxation methods. These are methods that guarantee, under certain conditions, convergence in probability to the globally optimum clustering, with respect to a predefined criterion, at the expense of intensive computations.

- Competitive learning algorithms. These are iterative schemes that do not employ cost function. They produce several clustering and they converge to the most sensible one, according to a distance matrix.

3.5.5 Cluster validation

The statistical hypothesis testing is used for clustering validation. The null hypothesis H_0 will be expressed as a statement of randomness concerning the structure of X . The goal is twofold:

- First, we must generate a reference data population under the random hypothesis, that is, a data population that models a random structure.
- Second, we must define an appropriate statistic, whose value are indicated of the structure of a data set, and compare the value that results from our data set X against the value obtained from random population.

There are three different ways to generate the reference population under null hypothesis.

1. **Random position hypothesis.** This hypothesis is appropriate for ratio data. It requires that “All the arrangements of N vectors in a specific region of the 1-dimensional space are equally likely occur”. The random position hypothesis can be used with either external or internal criteria.

-Internal criteria. In this case, the statistic q is defined to measure the degree to which a clustering structure, produced by a clustering algorithm, matches the proximity matrix of the corresponding data set. Let X_i be a set of N vectors generated according to the random position hypothesis and P_i

is the corresponding proximity matrix. We apply the same clustering algorithm to each X_i and to our data set X and let C_i and C be the resulting clustering structures, respectively. For each case, the value of the statistic q is computed. The random hypothesis, H_0 , is then rejected if the value q , resulting from X lies in the critical interval, that is, if q is unusually small or large.

- External criteria. The statistic q is defined so as to measure the degree of correspondence between a predefined structure P imposed on X and the clustering that results after the application of a specific clustering algorithm to X . Then, the value of q corresponding to the clustering C resulting from the data set X is tested against the q_i corresponding to the clustering resulting from the reference population generated under the random position hypothesis. Once more, the random hypothesis is rejected if q is unusually large or small.

2. **Random graph hypothesis.** It is usually adopted when only internal information is available. It is appropriate when ordinal proximity between vectors are used. Let define the ordinal $N \times N$ matrix A as a symmetric matrix with zero diagonal elements and with its upper diagonal elements being integers in the range $[1, N(N-1)/2]$. The entry $A(i, j)$ of A provides only qualitative information about the dissimilarity between the corresponding vectors x_i and x_j . If, for example, $A(2, 3) = 3$ and $A(2, 5) = 5$, we can only conclude that x_3 is more similar to x_2 more than x_5 . Let A_i be an $N \times N$ ordinal proximity matrix with no ties, that is, all entries in the upper diagonal are different from each other. Under the random graph

hypothesis, the reference population consists of such matrix A_i each one generated by inserting randomly the integers in $[1, N(N-1)/2]$, in its upper diagonal entries. Let P be the ordinal proximity matrix associated with the given data set X and C_i be the clustering structure produced by the application of a specific algorithm to P . Finally, let C be the clustering structure produced when the same algorithm is applied to A_i . We may proceed as in the previous case and define a statistic q that measures the agreement between a rank order (proximity) matrix and the corresponding clustering structure. If the value of q , corresponding to P and C , is unusually large or small, the random hypothesis is rejected. It must be emphasized that the random graph hypothesis is not appropriate for ratio-scaled data. For example, the case where the Euclidean distance is in use and $l \leq N-2$ and consider the points $x_1 = 0$, $x_2 = 1$, $x_3 = 3$ on the real line. It is clear that the distance between x_1 and x_3 cannot be smaller than the distance between x_2 and x_3 . That is, the matrix

$$A = \begin{bmatrix} 0 & 2 & 1 \\ 2 & 0 & 3 \\ 1 & 3 & 0 \end{bmatrix} \text{ is not a valid proximity matrix for these ratio scaled data.}$$

3. **Random label hypothesis.** Let us consider all possible partitions, P' , of X into m groups. Each partition may be defined in terms of a mapping g from X to $\{1, \dots, m\}$. The random label hypothesis assumes that all possible mapping is equally likely. The statistic q can be defined so as to measure the degree to which information inherent in the data set X , such as the proximity matrix P , matches a specific partition. The statistic q is then used to test the degree of match between P and an externally imposed partition P , against the q_i corresponding to the random

partitions generated under the random label hypothesis. Once more, H_0 is then rejected if q is unusually large or small.

3.6 Winner-Take-All learning

The network discussed in this section classifies input vectors into a specified number of categories that are detected from the training set $\{x_1, x_2, x_3, \dots, x_n\}$. The training is performed in unsupervised mode and the network undergoes the self-organization process. It is impossible in the training phase to assign network nodes to specific classes in advance and is equally impossible to predict which neuron at the beginning of the training will be activated.

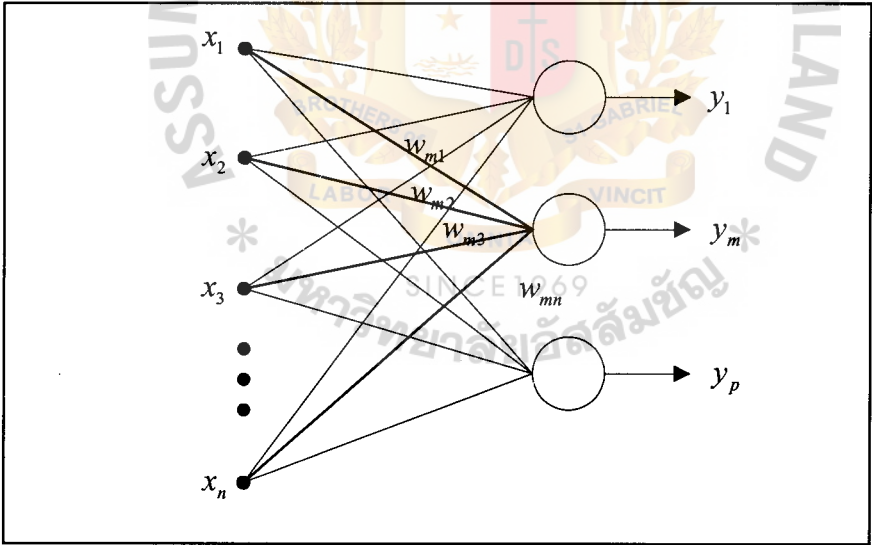


Figure 3-6: Learning layer (adapted weights highlighted)

A Kohonen network is shown as in Figure 3-6. The processing of input data $x = [x_1 \ x_2 \ \dots \ x_n]$ from the training which represents p clusters, follows the customary expression

$$y = \Gamma [wx] \quad (3.3)$$

To analyze network performance, we rearrange the matrix W to the following form:

$$W = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{p1} & w_{p2} & \dots & w_{pn} \end{bmatrix}$$

Components of the row w_m are highlighted in Figure 3-6.

The learning process consists of:

- 1) Initialize p weight vector w_i randomly and normalized.
- 2) Select randomly input vector x and train the network by
 - a) Select the winning neuron: The winning neuron m is the neuron that has the weight vector w_m nearest to x

$$\|x - \hat{w}_m\| = \min_{i=1,2,3,\dots,p} \left\{ \|x - \hat{w}_i\| \right\} \quad (3.4)$$

The left-hand side of Eq.3.4 can be rearranged to the form

$$\|x - \hat{w}_m\|^2 = (x^t x - 2 \hat{w}_m^t x + 1)^{1/2} \quad (3.5)$$

It is obvious that finding the minimum of left-hand side of Eq.3.5 corresponds to finding the maximum among the p scalar products of

$$\hat{w}_m^t x = \max_{i=1,2,3,\dots,p} (\hat{w}_i^t x) \quad (3.6)$$

The left-hand side of Eq.3.6 is called the activation value of the “winning” neuron.

b) After the winning neuron m has been identified and declared as winner, its weights must be adjusted so that the distance in Eq.3.4 is reduced after the current training step.

Thus, in order to reduce $\|x - w_m\|$, w_m is adjusted preferably along the gradient.

$$\nabla w'_m = \alpha (x - w_m) \quad (3.7)$$

Where α is a learning rate selected heuristically, usually between 0.1 and 0.7, the remaining weight vectors w_i for $i \neq m$, are left unaffected. The learning rule in Eq.3.7 in the k 'th step can be rewritten in a more formal way as follows:

$$\hat{w}_m^{k+1} = \hat{w}_m^k + \alpha^k (x - \hat{w}_m^k) \quad (3.8)$$

$$\hat{w}_i^{k+1} = \hat{w}_i^k \quad (3.9)$$

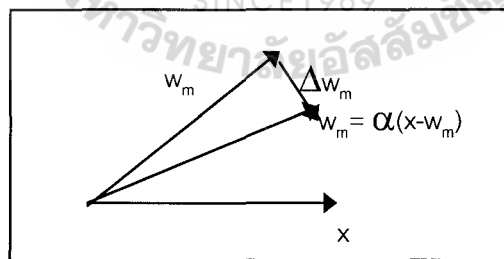


Figure 3-7: Weight vector adaptation

Where α^k is a learning rate at step k and m is the index of the winning neuron. While learning process continues and clusters are developed, the network weights acquired similarity to input data within clusters. Equation 3.9 makes \hat{w}_m^{k+1} nearer to x

than \hat{w}_m^k (see Figure 3-7). This increases the winning chance of the m^{th} neuron, the following inequality should hold.

$$\hat{w}_m^t x < (\hat{w}_m + \Delta \hat{w}_m) x \quad (3.10)$$

or

$$\Delta w_m^t x > 0 \quad (3.11)$$

From Figure 3-7, it is easy to see that

$$x^t x - \hat{w}_m^t x > 0 \quad (3.12)$$

Which is equivalent to

$$\|x^t\| \cdot \|x\| - \|\hat{w}_m^t\| \cdot \|x\| \cos \theta > 0 \quad (3.13)$$

Where θ is the angle between x and \hat{w}_m .

Assume that both x and \hat{w}_m are normalized then Eq.3.14 becomes:

$$1 - \cos \theta > 0 \quad (3.14)$$

Since Eq.3.14 is always true because normally $x \neq \hat{w}_m$, the winner-take-all learning rule produces an update of the weight vector in the proper direction [4].

3.7 Self-Organizing Feature Map (SOFM)

SOFM transforms an incoming signal patterns of arbitrary dimension into a one or two or higher dimensional discrete map, and performs the transformation adaptively in a topologically ordered fashion [13].

It is a sheet-like artificial neural network, each cell becomes specifically tuned to various input signal patterns or classes of patterns through an unsupervised learning process. Only one cell or a local group of cells actively response to the current input. The location of the response neuron tends to become ordered as if some meaningful coordinate system for different input features created over the network [7]. SOFM composed 2 layers, input and output layers, each layer is connected via synaptic weights. Input layer can be high dimension but, for output layer, mostly it should be one or two dimensions because the related features of inputs will be demonstrated on the output layer. In this work, we concentrate on two-dimensional output layer. Figure 3-8. displays a layout of a SOFM. The neurons are arranged in a two-dimensional lattice. This topology ensures that each neuron has a set of its neighbors. Each input is connected to all the neurons in the output layer via synaptic weights.

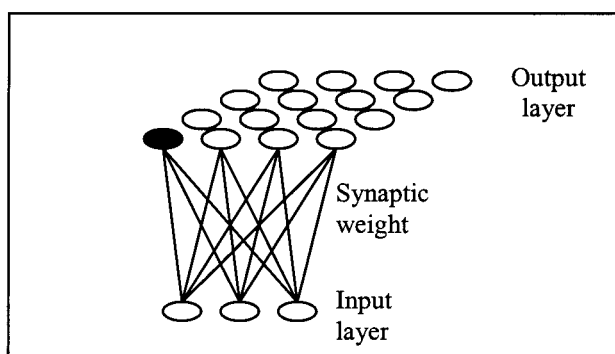


Figure 3-8: Self-Organizing Feature Map Model

SOFM is a competitive learning method, when an input vector is passed through the network, it searches for the winner node. The winner neuron and its neighbors will be activated, They are called “activity bubbles”, and other neurons outside its neighbor are inactive.

To reward the winner and its neighbors, their synaptic weights will be adjusted toward the input vector. The simplest analytical measure for the match of input x with neuron m maybe the inner product $x^T w_m$. However, if the SOFM algorithm is to be used for natural signal patterns relating to metric vector spaces, the better and more convenient matching criterion may be used, based on the Euclidean distances between x and m_i . The minimum distance defines the “winner” m . The algorithm to find m was presented in [4]. The winner node m is defined by

$$\| x - w_m \| = \min_i \{ \| x - w_i \| \}$$

The neighborhood of a winning neuron node is gradual decreased as given in Figure 3-9. The neighborhood can be either hexagonal or rectangle. Figure 3-9 demonstrates hexagonal neighborhood of winner neuron m .

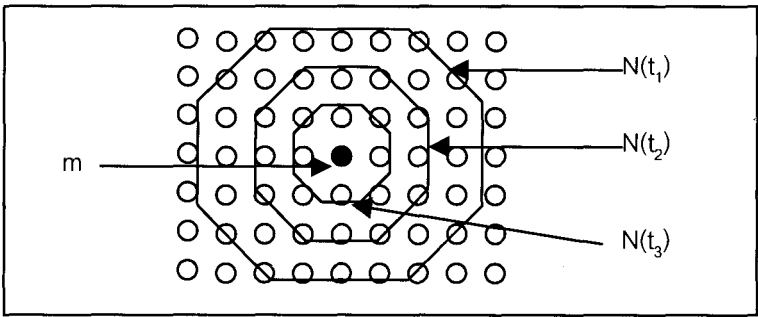


Figure 3-9: Example of topological neighborhood $N(t)$, where $t_1 < t_2 < t_3$

After the winner neuron m is defined, synaptic weights of a neighbor i is adjusted by

$$\Delta w_i(t) = \alpha(d(i,m),t)[x(t) - w_i(t)] \text{ for } i \in N_m(t) \quad (3.16)$$

Where $N_m(t)$ denotes the current spatial neighborhood and α is a positive-valued learning function, $0 < \alpha(d(i,m),t) < 1$. Because α needs to decrease as learning progress, it is often convenient to express it as a decreasing function of time. In addition, α can be expressed as a function of the neighborhood radius. An example of a function yielding good practical results in a series of Konohana's simulations is

$$\alpha(N_i, t) = \alpha(t) \exp[-\|r_i - r_m\| / \sigma^2(t)] \quad (3.17)$$

Where r_m and r_i are the position vectors of the winning cell and of the neighborhood nodes of the winner, respectively, and $\alpha(t)$ and $\sigma(t)$ are suitably decreasing functions of learning time t .

3.7.1 Initialization of weights

The weights are initialized randomly. The learning starts at the above weights and is updated in expression Eq.3.8 and Eq.3.9 with a small positive α value. This makes the weight vector of winning neuron to be closed to the input vectors. A learning parameter rate α is gradually decreased. This allows for the gradual separation of weights according to the input data used for training.

3.7.2 The main characteristics of SOFM

1. The neurons are exposed to a sufficient number of inputs.
2. Only the weights leading to an excited neighborhood of the map are affected.
3. The adjustment is in proportion to the activation received by each neuron within the neighborhood. As a result, the weight adaptation rule tends to enhance the same responses to a sufficiently similar subsequent input.

3.7.3 The learning algorithm

The learning algorithm given is summarized as follows:

1. Randomly initialize vector weights, α , the number of output neuron.
2. For each input vector do
 - a). Find the winner neuron that d_j between the input vector and its weight vector is minimal. The Euclidean distance is:

$$d_j = \sum_{i=1}^n (x_i(t) - w_{ij}(t))^2 \quad (3.18)$$

for $j = 1, 2, 3, \dots, k$

- b). Update weight vectors of the winner neuron and its neighbors.

End {for}

Numerous technical reports on the applications of the feature map algorithms can be found in [6]. The algorithm has been applied in sensory mapping, robot

control, speech recognition ... Figure 3-10 shows the initial and intermediate phases of self-organization of square feature map for a two-dimensional input vector.

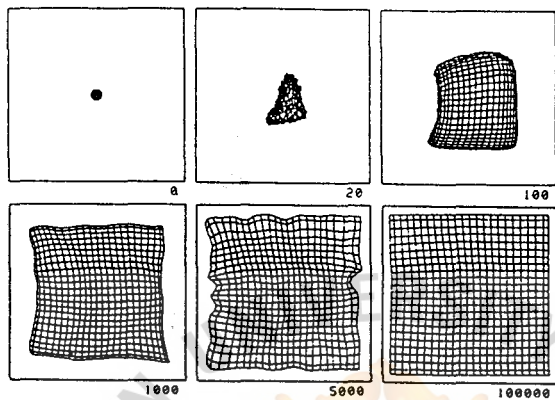


Figure 3-10: Self-Organizing feature map for two input nodes, arrays of neurons, and uniformly distributed excitation

The training input is selected uniformly random independent inputs uniformly distributed over the unity square shown in each of the six boxes. In fact, each of the boxes displays the distribution of weights during training. The initial weights are selected to be random and close to the values of 0.5 and 0.5. The final weights, w_{i1} and w_{i2} of the square array are shown after 100,000 training iterations in the last of the six boxes. The horizontal and vertical coordinates of the plots are the resulting values w_{i1} and w_{i2} , respectively, for an array of 25.25 neurons. Line intersections of the maps specify weight vales for a single i^{th} neuron. Lines between the nodes on the graph merely connect weight points for neurons that are topological nearest neighbors.

3.8. Self-Organizing semantic maps

The semantic maps discussed in this section are implements based on the self-organizing feature map concept described above. This example demonstrates that, in addition to processing quantitative data for feature extraction, the maps make it possible to display abstract data, such as words, arranged in topological neighborhoods. The maps extract semantic relationships that exist within the set of data considered here as a collection of words. Their relative distances on the map containing words positioned according to their meaning or context can reflect the relationships. This indicates that the trained network can possibly detect the logical similarity between words from the statistics of the contexts in which they are used. The context is understood here as an element of a set of attribute values that occur in conjunction with the words. In another approach, the context can be determined by the frequency of neighboring words without regard to the attribute values that are occurring. This discussion is based on experiments and results described by [7] and [12].

Difficulty results when trying to express the human language as neural network input data as opposed to nonsymbolic input data encoding. These data can be represented as continuous or discrete value arranged into a vector variable. The difficulty is due to the fact that for symbolic objects, such as words, no adequate matrix has been developed. In addition, except for a few special words denoting sounds, the meaning of a word is usually disassociated from its encoding into letters or sounds. Moreover, no matrix relation exists whatsoever between the words representing similar objects.

One simple model for contextual representation of a symbol and its attribute enabling the topological mapping on the semantic map is to concatenate the data vector as follows: $x^T = [x_s \ x_a]$

Where x_s and x_a are the symbol and the attribute parts of the data vector. Assuming the local representation of the symbols, such that the i^{th} symbol in the set of symbols is assigned a p-dimensional symbol vector x_{si} whose i^{th} component is equal to a fixed value of c, and other components in x_{si} are zero. For example $\hat{x}_{s1} = [c \ 0 \dots 0]^t$

| | | dove | hen | duck | goose | owl | hawk | eagle | fox | dog | wolf | cat | tiger | lion | horse | zebra | cow |
|----------|----------|------|-----|------|-------|-----|------|-------|-----|-----|------|-----|-------|------|-------|-------|-----|
| is | small | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| | medium | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| | big | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| has | 2 legs | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 4 legs | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | hair | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | hooves | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| | mane | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| | feathers | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| likes to | hunt | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| | run | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| | fly | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | swim | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Figure 3-11: Table of attributes for the set of P = 16 animal objects

The attributes are present or absent for each p symbols. The presence, or absence, of a particular attribute is indicated by a binary entry 1 or 0, respectively. Figure 3-11 gives an example data vectors for 16 animals (symbol) that are used to demonstrate SOFM. The data for cow is $x = [x_{s16} \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \dots 0]^t$ Where $x_{s16} = [0 \dots c]^t$ and x consists of 29 entries, 16 first entry are used for symbols, the

remains 13 entries are used for attributes. The norm of the difference vector between each pair of symbol vectors (x_{si} and x_{sj} , where i,j) is identical and equal to $\sqrt{2}c$

These data vectors have been used to train a 10.10 planar array of neurons. The initial weights have been selected of random values so that no initial ordering was imposed. After about 2000 training input presentations, the neighborhood sensitivity has been ascertained. The neurons' responses became consistently stronger or weaker in certain regions. This has been learned due to either of the excitations $x = [x_{si} \ 0]^t$, or $x = [0 \ x_{ai}]^t$. The neurons with the strongest response due to the symbol vector excitation only (set 0 to all attributes) are shown in Figure 3-12. They are labeled using the symbol that elicits the particular response. The neurons represented by dots indicate their nondominant responses on the map of symbols. Figure 3-13 showing the strongest response domains. Each neuron of this map is marked with the stimulus eliciting its strongest response.

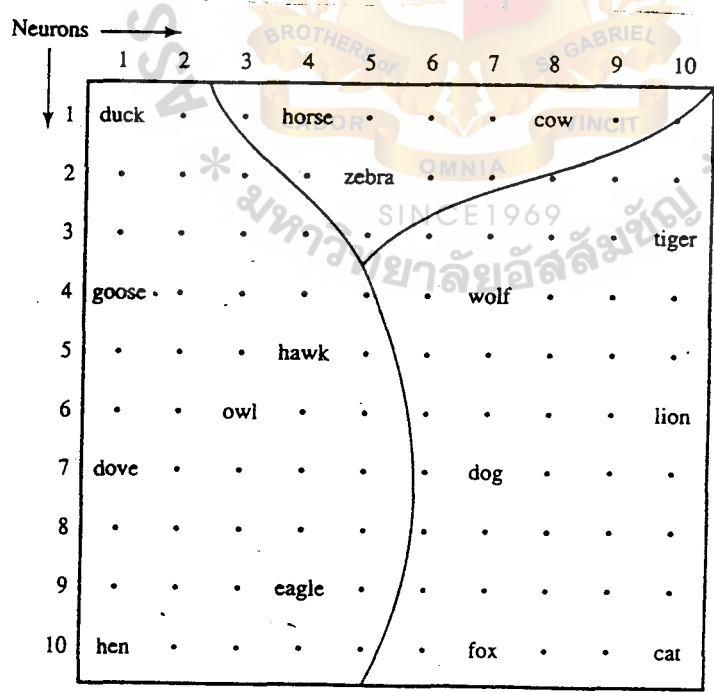


Figure 3-12: Strongest responses due to the symbol part only excitation

| | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|------|------|-------|-------|
| duck | duck | horse | horse | zebra | zebra | cow | cow | cow | cow |
| duck | duck | horse | zebra | zebra | zebra | cow | cow | tiger | tiger |
| goose | goose | goose | zebra | zebra | zebra | wolf | wolf | tiger | tiger |
| goose | goose | hawk | hawk | hawk | wolf | wolf | wolf | tiger | tiger |
| goose | owl | hawk | hawk | hawk | wolf | wolf | wolf | lion | lion |
| dove | owl | owl | hawk | hawk | dog | dog | dog | lion | lion |
| dove | dove | owl | owl | owl | dog | dog | dog | dog | lion |
| dove | dove | eagle | eagle | eagle | dog | dog | dog | dog | cat |
| hen | hen | eagle | eagle | eagle | fox | fox | fox | cat | cat |
| hen | hen | eagle | eagle | eagle | fox | fox | fox | cat | cat |

Figure 3-13: Strongest response domains

3.9 Multiple linear regression

Regression analysis is a method can be used to measure the statistical relationship that exists between two or more variables.

In regression analysis, an estimating, or predicting, equation is developed to describe the pattern or functional nature of the relationship that exists among variables. As the name implies, an analyst prepares an estimating (or regression) equation to make estimates of values of one variable from given values of the others.

The dependent variable is the variable to be estimated; it is customarily plotted on the vertical or y-axis of a chart and is therefore identified by the symbol y .

Independent variables (one in simple regression, two or more in multiple regression) are variables that presumably exert influences on or explains variations in the dependent variable. The format of multiple regression equation is

$$y = b_0 + b_1x_1 + b_2x_2 + \dots + b_nx_n$$

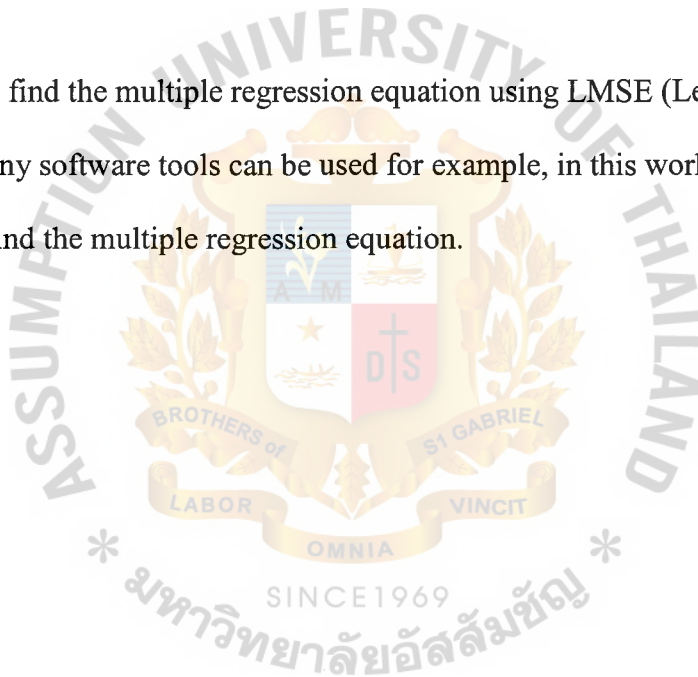
Where y = the estimated value of the dependent variable

b_0 = the y-intercept

x_i = i^{th} independent variable

b_i = slope associated with x_i

In order to find the multiple regression equation using LMSE (Least Mean Square Error), many software tools can be used for example, in this work, Minitab is used as a tool to find the multiple regression equation.



CHAPTER 4

PROPOSED METHODOLOGY

In chapter 3, we review the previous works of Self-Organizing Feature Map area. Most of works recommends that the way to define clear-cut boundaries of the clusters should be developed further. This because the result obtained from Kohonen Network is the planar map, then the researchers defined clusters by their methodologies which produce ambiguous boundaries of the clusters, therefore, we cannot find which is the best method to classify data. The different data is suited for different approach to classify. The best way to classify is let the domain expert use his commonsense to draw the clusters. Thus in our research we propose the new methodology to classify data by using SOM as describing in the following sections.

4.1 Classification using Kohonen Network

As mentioned in chapter 1, in this study we try to use the Kohonen network to cluster the data of students' academic records. Then using LMSE method to build a model for each cluster, the estimated equations are used to predict the GPA of student based on his/her old academic records.

In this work, we use Kohonen network to find clusters. The input data must be prepared and normalized. The output from network is two dimensional map, which is no explicit clusters are shown so we have to find the clusters from this map. We group the nearby activated neurons in the map which have GPA of the second semester in

the same interval into the same cluster. After that we find model of each cluster by using Multiple Regression Equation. The scheme is given in Figure 4-1.

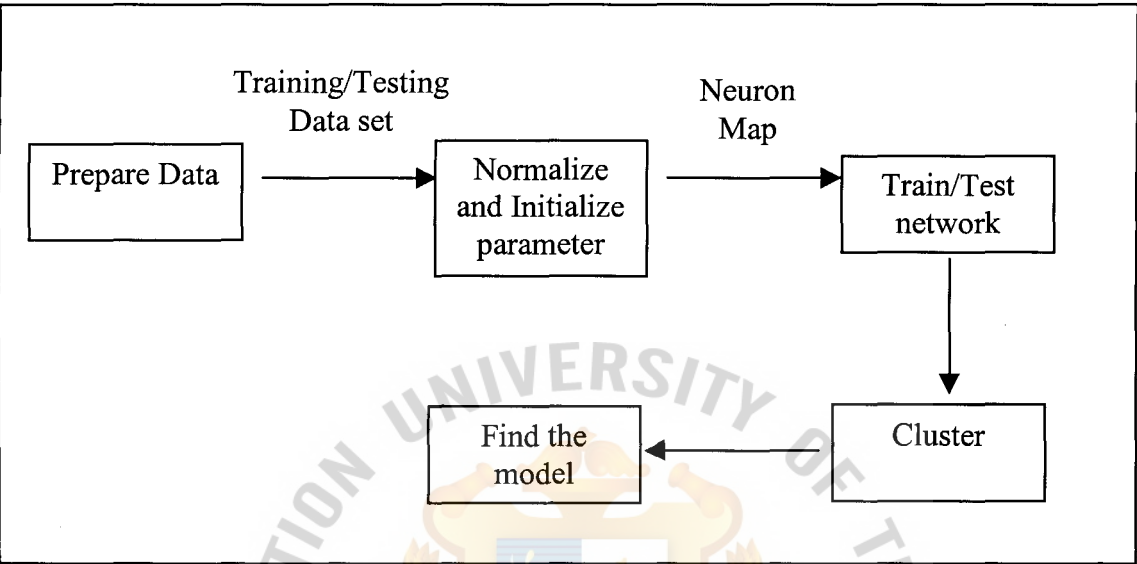


Figure 4-1: The process to establish model using neuron network

4.2 The processes to classify using Kohonen Network

- Data preparation.

It is obvious that only some fields or attributes of database are meaningful in the KD process. The selection of suitable fields is difficult, normally they are selected in heuristic way or trial and test way.

- Select training and testing data set.

Training data set is the set of data that is used to train the network. In our framework, after network is trained, weight is kept in file. We use testing data set to test the performance as well as to evaluate the efficiency of the network. The training/testing data set should reflect the entire data. This can be done by randomly selection.

- Data normalization and parameter initialization.

Data vectors are normalized to make them be unit input vector for the network.

Normally the learning parameter rate α is initialized at the value between 0.7 and 0.9 and it will be decreased as the growing epoch. Learning rate converge the weight adaptation. The number of output neurons is sufficient to make clearly the cluster boundary. The initial size of neighborhood of the center-activated neuron should cover all of the neurons, for the neurons at the boundary the neighbor are covered half of map.

The number of training epochs should be sufficient to adjust weights in network training phase but if it is too large then overffiting will be occurred. It makes network too strength if the new-presented input is a bit different from the training data, the network can not recognize it. The initial vector weights are selected randomly in the range from 0.00 to 1.00.

- Training and testing the network

SOFM algorithm is used to find the winning node and update vector weights. Neighbor size and learning parameter rate α decreases in each epoch. When neighbor size decrease to zero, only the winning node weight is updated. The learning parameter rate α decreases to 0.01 gradually. At the end of training phase, the weight vectors are saved. The positions of activated neurons are calculated based on the weights. All output nodes are classified to input vector that makes the strongest response.

- Clustering.

From the output layer, the nearby activated neurons which have same GPA of the second semester range are grouped to same cluster. The range of GPA of the second semester is selected by trial and test way. If we narrow range, there are more clusters, but if we use range too wide, we get too less clusters.

4.3 Establishing model for each cluster using multiple regression

When clusters and their members are determined, we can find Multiple Regression Equation (MRE) of each cluster. In this work we find both MRE for each cluster and MRE for the whole input data. The data in each cluster is used to establish models by MRE. When we get clusters, we know which data belongs to which cluster. We use the normalized data for training phase but, instead, we use the raw data to set up the MRE.

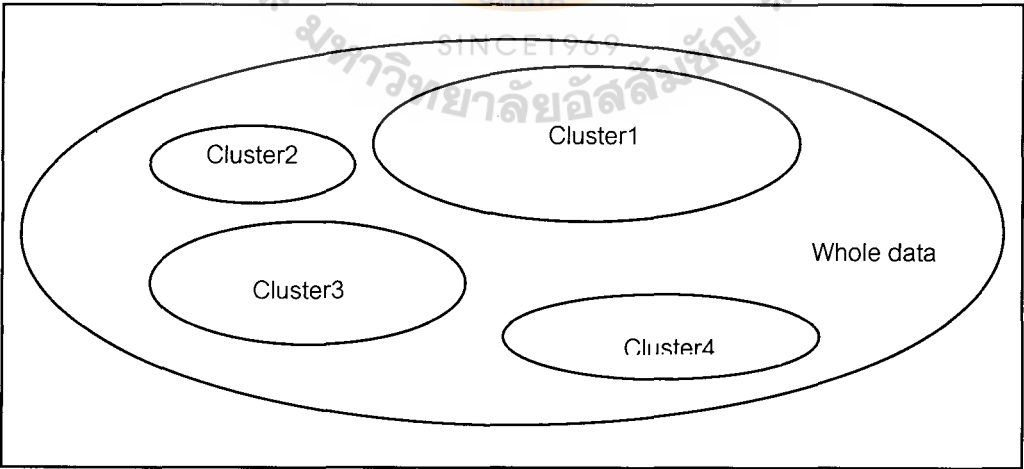


Figure 4-2: Model from each cluster

4.4 Predict the GPA2 from available data

When a new data is presented to forecast the GPA of the second semester we find which cluster it belongs to. Its GPA of the second semester is set to be zero and normalized as the input to network in testing phase. The model (MRE) of cluster that data belongs to is used to predict GPA of the second semester. Similar to the way to find model, we use raw data apply to equation to predict.



CHAPTER 5

EXPERIMENTS AND DICUSSION

In this chapter we discuss the experiments that we have done. We divided the experiments into 3 groups.

First, Soybean data from UCI Database Repository at University of California Irvine is used as a sample data. Michalski [8] and many well-known researchers have used this data in widely area. These data are given in appendix A.

Second, we use Assumption University students' data to find the clusters. This data have been windowing, so we know characteristic of the data in advance.

Third, we select Assumption University students' data randomly to find clusters and build the model from each cluster. We also randomly select the testing data to test models.

5.1 Soybean data clustering

In this experiment, we use soybean data that we get from UCI database repository. The soybean data has 47 instances of 35 attributes. They are divided into 4 clusters by Michalski as:

Cluster X1: D1-D10

Cluster X2: D11-D20

Cluster X3: D21-D30

Cluster X4: D31-D47

Data are trained and classified to clusters by our proposed methodology with 15x15 neuron output map, initial neighbor size = 15, $\alpha = 0.9$, and 50,000 iterations. We do not need to normalize the data because the data has been normalized already. Figure 5.1 shows the output map, in which the activated output neurons are painted with the color as given in Table 5-1. There are 5 clusters as follows.

- Cluster1: D1-D10
- Cluster2: D11-D20
- Cluster3: D21-D30
- Cluster4: D31, D32, D34, D35, D37, D41, D42, D44, D46, D47
- Cluster5: D33, D36, D38, D39, D40, D43, D45

Table 5-1: The color used to paint activated neurons

| Group | Color |
|------------|-------|
| Cluster 1 | Blue |
| Cluster 2 | Green |
| *Cluster 3 | Red |
| Cluster 4 | Black |
| Cluster 5 | Black |

The result obtained here is almost the same as Michalski ’s results except the cluster X4 of Michalski is separated into two clusters.

We see that cluster 4 and 5 are subsets of cluster X4 in Michalski [8]. Pure Kohonen network is also used to find the output map and drawn clusters by hand are given in Figure 5-2.

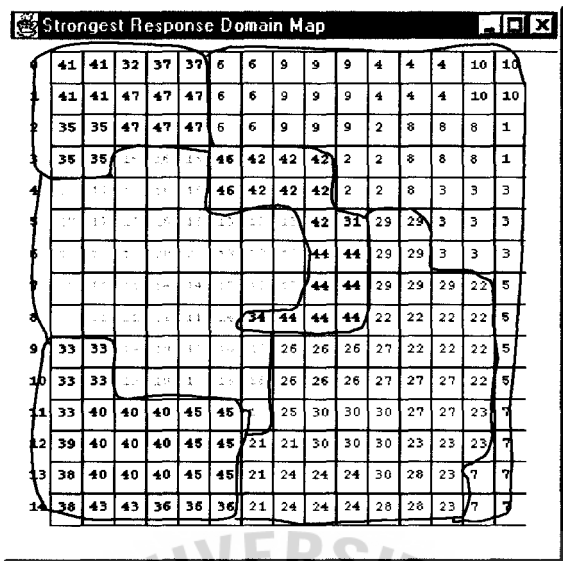


Figure 5-1: Clusters from the strongest response of all input

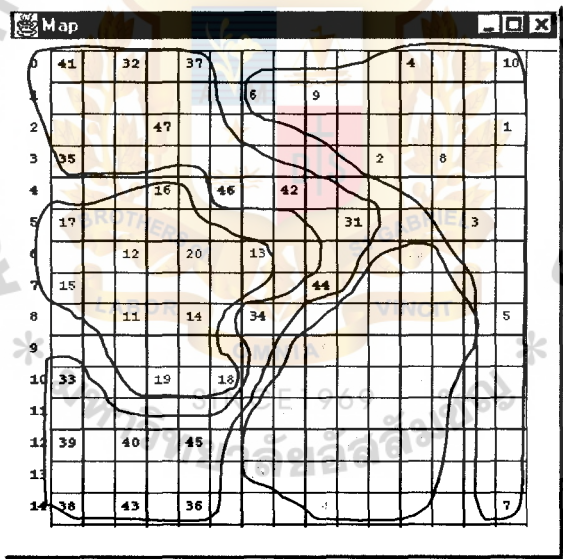


Figure 5-2: The activated neuron

In order to predict the attribute for the new presented data, we need to find the cluster that it belongs to. For the pure Kohonen network (Figure 5-2.), it is quite difficult to determine the cluster of new data, when activated neuron of this new data fall in the position that does not belong to any clusters. Meanwhile, with our proposed

approach, it is easy to find the cluster because every position in output map must belong to one cluster, thus no matter which neuron is activated by the new data, we know which cluster it belongs to.

5.2 Clustering selected Assumption University students’ data

In this experiment, we select the 350 sample data that we use as a training data set which is distributed by several ranges of G2 (GPA of the second semester), English and Mathematics as shown in Table 5-2. Thus we know the characteristic of the data in advance. These groups of sample data are mixed up so that there is no order in the training data set. The criteria used to select the sample data are shown in Table 5-2. In this experiment, only English, Mathematics and GPA of the second semester are representative of student data. The data information and numerization of nonnumeric data are shown in Table 5-3 and Table 5-4 respectively. Appendix B.1 shows the data that is used in this experiment. Activated output neurons are painted with color as in Table 5-5. Appendix B.2 shows the map and clusters from this experiment.

Table 5-2: The criteria to select sample data (windowing data)

| Group | English | Mathematics | G2 | Amount |
|-------|-------------------------|--------------------------|------------|--------|
| 1 | 0-1 (U1-U2) | 0-1 (U-F) | [0.0, 2.0] | 50 |
| 2 | 2-5 (S2-C) | 2-3 (D-C ⁺) | [2.0, 2.5] | 100 |
| 3 | 5-7 (C-B ⁻) | 4-7 (C-B ⁻) | [2.5, 3.0] | 100 |
| 4 | 8-9 (B-B ⁺) | 7-8 (B-B ⁺) | [3.0, 3.5] | 50 |
| 5 | 8-11 (B-A) | 9-10 (A ⁻ -A) | [3.5, 4.0] | 50 |

Table 5-3: Subject list used in the experiment

| Subject | Code | Subject Name |
|-------------|--------|-------------------|
| English | BG0001 | Basic English1 |
| | BG0002 | Basic English2 |
| | BG1200 | English1 |
| Mathematics | BG0200 | Re-Math |
| | BG1200 | Math for Business |

Table 5-4: Numerization of nonnumeric data

| Subject | Grade | Mapping Value |
|-------------------|-------|---------------|
| Basic English1 | U | 0 |
| | S | 1 |
| Basic English2 | U | 1 |
| | S | 2 |
| English1 | F | 2 |
| | WP | 2 |
| | D | 3 |
| | C- | 4 |
| | C | 5 |
| | C+ | 6 |
| | B- | 7 |
| | B | 8 |
| | B+ | 9 |
| | A- | 10 |
| Re-Math | A | 11 |
| | WP | 0 |
| | U | 0 |
| Math for Business | S | 1 |
| | F | 1 |
| | WP | 1 |
| | D | 2 |
| | C- | 3 |
| | C | 4 |
| | C+ | 5 |
| | B- | 6 |
| | B | 7 |
| | B+ | 8 |
| | A- | 9 |
| | A | 10 |

Table 5-5: The colors used to paint the activated neurons versus G2

| Range of G2 | Color |
|-------------|-------|
| [0.0, 2.0) | Blue |
| [2.0, 2.5) | Green |
| [2.5, 3.0) | Cyan |
| [3.0, 3.5) | Black |
| [3.5, 4.0) | Red |

We have trained the network with 30x30 output neuron map, initial neighborhood size =15, $\alpha =0.9$ and 50,000 iterations. We get 15 clusters and we will discuss each cluster in detail as of following.

Table 5-6: Characteristic of each cluster

| Group of clusters | English | Mathematics | G2 |
|-------------------|---------|-------------|--------------|
| A | [8, 11] | [7, 10] | [3.00, 4.00] |
| B | 8 | 7 | [3.00, 3.50] |
| C | [5, 7] | [4, 7] | [2.00, 3.00] |
| D | [2, 5] | [2, 3] | [2.00, 2.50] |
| E | [0, 2] | [0, 2] | [0.00, 2.00] |

If we take a close look at clusters that we get and the group of sample data, we see that the clusters from subset of each sample data. That is

Table 5-7: Cluster in each group

| Group | Cluster |
|-------|--------------------------|
| 1 | A (10,11,12,13,14,15,16) |
| 2 | B (8,9) |
| 3 | C (5,6,7) |
| 4 | D (4) |
| 5 | E (1,2,3) |

The number of data in each cluster compare to the number of data in matching sample data group is shown in Table 5-8.

Table 5-8: The number of data in each cluster compare to the number of data in matching sample data group

| Group of clusters | Cluster | # of data in cluster | Total data in group of clusters | Sample group | # of data in group |
|-------------------|---------|----------------------|---------------------------------|--------------|--------------------|
| E | 10 | 21 | 58 | 1 | 50 |
| | 11 | 25 | | | |
| | 12 | 3 | | | |
| | 13 | 6 | | | |
| | 14 | 1 | | | |
| | 15 | 1 | | | |
| | 16 | 1 | | | |
| D | 8 | 59 | 84 | 2 | 100 |
| | 9 | 25 | | | |
| C | 5 | 37 | 100 | 3 | 100 |
| | 6 | 32 | | | |
| | 7 | 31 | | | |
| B | 4 | 58 | 58 | 4 | 50 |
| A | 1 | 17 | 50 | 5 | 50 |
| | 2 | 17 | | | |
| | 3 | 16 | | | |

Thus we can say that our clustering methodology can classify data into clusters so that each of them has its own characteristic. Each cluster owns the specific

characteristic of each sample data group, for example, in sample data group 1, there are 7 subgroups (E) and each subgroup shows subset of characteristic of sample data group 1.

Form Table 5-8 it is easy to see that some data members fell to wrong cluster, for example, the total number of members in E (cluster 10 to 16) is 58, that is greater than the number of members of sample data group 1 that is 50.

There are 32 data out of 350 data fell into the wrong cluster, that is 9.143%.

This error may be the result of selecting attributes. We use only English, Mathematics, and GPA of the second semester to find the cluster, meanwhile other attributes may affect to the performance of students.

Beside this, we can notice the students that have same English and same Mathematics may have a little bit difference in GPA of the second semester. This because GPA strongly depends on the level of English and Mathematics, but it also depends on other attributes that we not include. Moreover, we assume that GPA depends linearly to the levels of English and Mathematics, this assumption maybe not completely exact.

There are other attributes may effect to GPA of the second semester, for example, the number of credits the student complete in each semester, GPA in the first semester, or others.

5.3 Randomly selected Assumption University's student data clustering

In this experiment, we select sample data from database with different ranges of GPA. The criterion of selecting data is shown in Table 5-9. English, Mathematics, the number of credits, and GPA of the first and second semester, are used in sample

data. The subjects related to English and Mathematics used in this experiment is shown in Table 5-10 and the numerization of nonnumeric data is similar to the last experiment and is shown in Table 5-4. Appendix C.1 shows the data that is used in this experiment. Appendix C.2 shows the map and clusters from this experiment.

We extract 100 testing data from the selected data. Thus the remainder 300 sample data are used to train and find the models.

The training has been performed on a random independent input from training data set. The initial weights are selected randomly between 0 and 1, $\alpha = 0.9$. The output is the array of 30x30 neurons. The final weight is saved after 50,000 training iterations.

Table 5-9: The criterion for selecting sample data

| Group | Criterion (G2) | Amount |
|-------|----------------|--------|
| 1 | [0.00, 2.00) | 133 |
| 2 | [2.00, 300) | 133 |
| 3 * | [3.00, 4.00] | * 134 |

The activated neurons are painted based on color in Table 5-11. We group the adjacent neurons in the map that has similar color into the same cluster. There are 13 clusters from the experiment. Some activated neuron cannot group into any cluster because it does not adjacent to other clusters. Some group of the neurons cannot form the cluster because they do not have enough data members. We do not establish the model for these small clusters and consider them as isolated data.

Table 5-10: Subject list used in the experiment

| Subject | Code | Subject Name |
|-------------|--------|-------------------|
| English | BG0001 | Basic English1 |
| | BG0002 | Basic English2 |
| | BG1200 | English1 |
| Mathematics | BG0200 | Re-Math |
| | BG1200 | Math for Business |

Table 5-11: Range of GPA, Color, and Cluster in experiment

| Range (GPA) | Color | Cluster |
|--------------|-------|-------------|
| [0.00, 2.00) | blue | 1,2,3 |
| [2.00, 3.00) | green | 4,5,6,7,8,9 |
| [3.00, 4.00] | red | 10,11,12 |

Table 5-11 summarizes the clusters, GPA and colors in this experiment. We use the data members in each cluster to build the model. Some clusters cannot build the model if they have less than 6 members. The cluster1 and cluster10 has largest members (69 members). The cluster5, 6,9,11,and 12 has smallest members (6 members).

After clustering data, we use multiple regression to build the model for each cluster as shown in Table 5-12. In this work, we test these models with t-test hypothesis testing with 90% confidence intervals in 2 tails critical value.

Table 5-12: Models with testing t-test hypothesis (E, M, C1, C2, G1, G2 stand for English, Mathematics, and the number of credits and GPA of the first and second semester)

| Cluster | Model |
|------------|---|
| Whole Data | $G2 = - 0.367 + 0.141E + 0.0749M - 0.0386C1 + 0.538G1 + 0.0535C2$ |
| 1 | $G2 = 0.699 + 0.0348E + 0.0211M - 0.0170C1 + 0.161G1 + 0.0366C2$ |
| 2 | $G2 = 2.30 - 0.248G1$ |
| 3 | $G2 = 0.590 - 0.0625M + 0.4281$ |
| 4 | $G2 = 1.53 - 0.200E + 0.0619C1 + 0.0553C2$ |
| 5 | $G2 = 1.88 - 0.0627E + 0.0545C2$ |
| 6 | $G2 = 1.69 + 0.0263C1 + 0.176G1$ |
| 7 | $G2 = 2.51 - 0.262E$ |
| 8 | $G2 = 1.60 + 0.0729M + 0.0423C1$ |
| 9 | $G2 = 0.45 + 0.0779C1 + 0.238G1$ |
| 10 | $G2 = 0.643 + 0.458G1 + 0.0758C2$ |
| 11 | $G2 = 0.724 - 1.19E + 0.307C1 + 0.150C2$ |
| 12 | $G2 = - 3.62 + 0.407C1 + 0.183C2$ |
| 13 | $G2 = 2.50 + 0.0624C2$ |

To evaluate the models, we use the testing data set, that we extract from the selecting data in the data preparation process, to test the model. G2 (GPA of the second semester) is the value that we want to predict. In order to predict G2, first we set G2 to zero, normalize the data. The data will be classified into clusters in the output map. We use the model of the cluster, that the testing data belongs to, to predict the value of G2.

There are 13 different local models for 13 different clusters.

Table 5-13 shows the predicted values of G2 from the global model and local model, real G2 of each data and its cluster label.

Table 5-14 shows that means square error of local model less than means square error of global model. It means that predicting by appropriate local model is more accurate than predicting by global model.

Table 5-13: Predicted value of G2s with testing t-test hypothesis

| Data | G2* | G2** | G2 | Cluster |
|------|-------|-------|-------|---------|
| 1 | 1.676 | 1.428 | 1.667 | 1 |
| 2 | 1.834 | 2.277 | 1.400 | 8 |
| 3 | 2.684 | 2.736 | 1.833 | 5 |
| 4 | 1.727 | 1.504 | 1.600 | 1 |
| 5 | 1.661 | 1.368 | 1.500 | 1 |
| 6 | 2.444 | 2.736 | 1.833 | 5 |
| 7 | 1.648 | 1.424 | 1.500 | 1 |
| 8 | 2.157 | 2.380 | 1.750 | 8 |
| 9 | 2.459 | 2.736 | 1.833 | 5 |
| 10 | 1.911 | 1.549 | 1.400 | 1 |
| 11 | 2.035 | 2.307 | 1.800 | 8 |
| 12 | 2.405 | 2.872 | 2.800 | 8 |
| 13 | 2.291 | 2.736 | 1.500 | 5 |
| 14 | 2.476 | 2.872 | 1.833 | 8 |
| 15 | 2.303 | 1.726 | 1.833 | 3 |
| 16 | 2.084 | 2.572 | 1.800 | 5 |
| 17 | 1.900 | 1.351 | 1.670 | 3 |
| 18 | 2.097 | 1.598 | 1.800 | 3 |
| 19 | 1.614 | 1.396 | 1.900 | 1 |
| 20 | 1.973 | 1.639 | 1.250 | 1 |
| 21 | 2.027 | 1.516 | 2.000 | 1 |
| 22 | 1.745 | 2.308 | 2.000 | 5 |
| 23 | 2.264 | 1.649 | 2.400 | 1 |
| 24 | 1.944 | 2.409 | 2.000 | 5 |
| 25 | 2.346 | 2.350 | 2.400 | 8 |
| 26 | 2.545 | 2.914 | 2.400 | 8 |
| 27 | 1.916 | 2.635 | 2.000 | 5 |
| 28 | 2.765 | 2.014 | 2.429 | 1 |
| 29 | 2.684 | 2.736 | 2.333 | 5 |
| 30 | 1.288 | 3.062 | 2.333 | 13 |
| 31 | 2.459 | 2.736 | 2.167 | 5 |
| 32 | 2.584 | 2.872 | 2.400 | 8 |
| 33 | 2.143 | 2.307 | 2.200 | 8 |
| 34 | 2.524 | 2.572 | 2.400 | 5 |
| 35 | 2.085 | 2.350 | 2.250 | 8 |
| 36 | 2.149 | 2.308 | 2.000 | 5 |
| 37 | 3.071 | 1.954 | 2.400 | 1 |
| 38 | 1.745 | 2.308 | 2.000 | 5 |
| 39 | 2.346 | 2.350 | 2.400 | 8 |
| 40 | 2.086 | 2.736 | 2.000 | 5 |
| 41 | 2.138 | 2.572 | 2.800 | 5 |
| 42 | 2.298 | 2.736 | 2.667 | 5 |
| 43 | 2.819 | 2.914 | 2.833 | 8 |
| 44 | 1.925 | 2.572 | 2.600 | 5 |
| 45 | 2.335 | 1.830 | 2.833 | 1 |
| 46 | 2.488 | 1.812 | 2.833 | 3 |
| 47 | 2.252 | 3.154 | 2.600 | 10 |
| 48 | 2.684 | 2.533 | 2.667 | 8 |
| 49 | 2.335 | 1.830 | 3.200 | 1 |
| 50 | 2.604 | 1.904 | 2.667 | 1 |
| 51 | 2.459 | 2.736 | 2.500 | 5 |
| 52 | 1.982 | 2.307 | 2.750 | 8 |
| 53 | 2.066 | 1.749 | 2.667 | 1 |
| 54 | 1.637 | 1.559 | 2.600 | 1 |
| 55 | 3.031 | 2.350 | 2.667 | 8 |
| 56 | 2.297 | 1.813 | 2.500 | 1 |
| 57 | 2.362 | 2.572 | 2.600 | 5 |
| 58 | 2.594 | 3.623 | 2.500 | 13 |
| 59 | 3.070 | 2.736 | 2.833 | 5 |
| 60 | 2.684 | 2.736 | 2.500 | 5 |
| 61 | 3.188 | 2.899 | 3.286 | 5 |

| | | | | |
|-----|-------|-------|-------|----|
| 62 | 3.070 | 2.736 | 3.167 | 5 |
| 63 | 2.953 | 2.736 | 3.333 | 5 |
| 64 | 3.092 | 1.966 | 3.200 | 1 |
| 65 | 2.300 | 2.315 | 3.000 | 8 |
| 66 | 2.679 | 3.623 | 3.166 | 13 |
| 67 | 3.393 | 2.736 | 3.000 | 5 |
| 68 | 1.733 | 1.402 | 2.000 | 1 |
| 69 | 3.485 | 2.790 | 3.368 | 5 |
| 70 | 2.819 | 2.914 | 3.166 | 8 |
| 71 | 2.579 | 1.707 | 2.000 | 1 |
| 72 | 3.271 | 2.020 | 3.000 | 1 |
| 73 | 2.942 | 3.436 | 3.200 | 13 |
| 74 | 2.835 | 1.974 | 1.900 | 1 |
| 75 | 3.070 | 2.736 | 3.000 | 5 |
| 76 | 3.497 | 2.914 | 1.900 | 8 |
| 77 | 3.592 | 2.899 | 3.143 | 5 |
| 78 | 2.684 | 2.736 | 3.000 | 5 |
| 79 | 3.070 | 2.736 | 3.167 | 5 |
| 80 | 1.745 | 2.308 | 3.333 | 5 |
| 81 | 3.841 | 3.090 | 2.500 | 8 |
| 82 | 3.807 | 3.133 | 2.800 | 8 |
| 83 | 3.122 | 3.623 | 1.900 | 13 |
| 84 | 3.715 | 3.090 | 3.526 | 8 |
| 85 | 2.977 | 1.950 | 1.600 | 1 |
| 86 | 3.393 | 2.736 | 3.667 | 5 |
| 87 | 3.552 | 3.133 | 3.830 | 8 |
| 88 | 3.841 | 3.090 | 3.830 | 8 |
| 89 | 2.349 | 1.675 | 1.800 | 1 |
| 90 | 2.734 | 1.848 | 2.000 | 1 |
| 91 | 3.807 | 2.143 | 2.400 | 1 |
| 92 | 3.327 | 2.003 | 1.720 | 1 |
| 93 | 3.233 | 2.003 | 2.500 | 1 |
| 94 | 3.580 | 2.105 | 2.100 | 1 |
| 95 | 3.807 | 2.143 | 2.510 | 1 |
| 96 | 3.807 | 2.143 | 2.330 | 1 |
| 97 | 2.819 | 1.873 | 2.240 | 1 |
| 98 | 2.791 | 1.865 | 2.300 | 1 |
| 99 | 3.214 | 1.988 | 2.780 | 1 |
| 100 | 3.722 | 2.117 | 2.910 | 1 |

Note that: G2* is predicted value from model of the whole data.

G2** is predicted value from the local model of its cluster.

Table 5-14: Means square error from predicted value of G2
with testing t-test hypothesis

| Type of model | MSE |
|-----------------------|---------|
| Model from whole data | 0.06237 |
| Model from clusters | 0.05967 |

5.3.1 Factors of cluster

Now we analyzed what the factor in our methodology that affect to the cluster because number of clusters and number of members in each cluster influences to the model.

Range of G2 used to cluster the data affects to the number of clusters. If the range is small, the number of clusters increases and the members in each cluster decreases. Having more clusters to build the models, it makes the more accuracy but some clusters may not have enough members to build the model.

We check whether the difference value (x) between weight vectors and input data makes neuron activated strongest greater than some value or not. We define this value as a threshold value. If the threshold value increases the number of cluster decreases and the number of members in cluster increases. This threshold value clarifies the boundary of clusters because the border neurons of cluster will have less value of x than the center of cluster.

5.3.2 The Effect of English and Mathematics to GPA

Form global model in Table 5-12, it is easy to see that both English and Mathematics has significance to the study of student in Assumption University. However in some local model, English and Mathematics does not have significance because their values are almost the same.

From model of the whole data, the coefficient of English is greater than Mathematics, it means that English affect to G2 more than Mathematics. This is also

true for the model of cluster 1. Models of some clusters do not depend on English or Mathematics.



CHAPTER 6

CONCLUSION AND RECOMMENDATION

In this work we studied Self-Organizing Feature Map and use it to find the hidden relationship of data from database. We studied how to use Kohonen network find the cluster. The first year academic records of students in Assumption University have been used in the thesis. Data are trained and mapped to 2-dimensional output neuron map. After that we use the proposed methodology to find the cluster from Kohonen map and use Multiple Regression to find the equation (model) from the clusters.

The comparison of the correctness between local models and global model demonstrates that the appropriated selecting local model can make a predicting more accuracy than use the global model (Table 5-14)

From the model of whole data in experiment 3, we can make a conclusion that English and Mathematics ability has significant influence to the study of the student. Furthermore, English has stronger influence than Mathematics because coefficient of English is greater than coefficient of Mathematics.

With the proposed approach, it is easier to build the clusters and find out the cluster boundary than pure Kohonen network. When the new data is presented, it is easy to find which cluster it belongs to.

Obviously, data analysis as well as Knowledge discovery depends strongly to how to selected the sample data. In order to find out what are the main factors affect to the performance of AU students may be need to be studied further.

The number of data selected and the appropriated size of output neuron map play an important role in building the models are recommended of further study.

REFERENCES

[1]. An Introduction to Data mining.

<http://www3.shore.net/~kht/text/dmwhite/dmwhite.shtml>

[2]. Donald H. Sanders. STATISTICS A First Course, Fifth Edition, United States of America. McGraw-Hill, inc. 1995.

[3]. Frawley, W.J., Piatetsky-Shapiro, G. and Matheus, C.J., “Knowledge Discovery in Database: An Overview. In Knowledge Discovery in Database”, AAAI/MIT Press, Vol. 19, pp.1-27, 1991.

[4]. Jacek M. Zurada, Introduction to Artificial Neural Systems, Singapore: West Publishing Company, 1992.

[5]. Janon Ong and Syed Sibte Raza Abidi, “Data Mining for Ranged Association Rules”, IC-AI’99 International Conference, 1999.

[6]. Kohonen, T., Self-Organization and Associative Memory, Berlin: Springer-Verlag, 1984.

[7]. Kohonen, T., “The Self-Organizing Map”, Proceeding of the IEEE 78(9), pp. 1464-1480, 1990.

[8]. Michalski, R.S. and Chilausky, R.L. “Learning by Being Told and Learning from Examples: An Experimental Comparison of the Two Methods of Knowledge Acquisition in the Context of developing an Expert System for Soybean Disease Diagnosis”, International Journal of Policy Analysis and Information System, Vol. 4, No 2, 1980.

[9]. Ming-Syan Chen, Jiawei Han, and Philip S. Yu. “Data Mining: An Overview from a Database Perspective”, IEEE Transaction on Knowledge and Data Engineering, Vol. 8, No 6, pp. 866 – 881, Dec 1996.

- [10]. Murthy Ranjit, Knowledge Discovery from Databases using Self Organized Maps, AIT M.Eng Thesis, AIT Library, No. CS-95-3, 1995.
- [11]. Quinlan, J.R, Introduction of decision trees, Machine Learning, 1, 1(1986), 81-106.
- [12]. Ritter, H., and T. Kohonen, “Self-Organizing Semantic Map”, Biolog. Cybern. 61, pp. 241-254, 1989.
- [13]. Simon HayKin, Neural Networks a Comprehensive Foundation, Second Edition. New Jersey, Prentice-Hall, Inc, 1999.
- [14]. Sushil Acharya, KNOWLEDGE DISCOVERY AND SELF-ORGANIZING SYSTEMS”, AIT Ph.D. Eng Thesis, AIT Library, No. CS-97-5, 1997.
- [15]. Theodoridis Sergios, PATTERN RECOGNITION, Academic Press, USA, 1998.
- [16]. Ting-Peng Liang, “Special Section: Research Integrating Learning Capabilities into Information Systems”, JMIS Vol9, No4, pp 5-15, spring 1993.

APPENDIX A

EXPERIMENT 1

A.1 SOYBEAN DATA ATTRIBUTES

| Attribute No. | Attribute Name | Attribute Values |
|---------------|------------------|---|
| 1 | Date: | April, May, June, July, August, September, October,?. |
| 2 | Plant-stand: | Normal,lt-normal,?. |
| 3 | Precipitation: | lt-norm,norm,gt-norm,?. |
| 4 | Temperature: | lt-norm,norm,gt-norm,?. |
| 5 | Hail: | yes,no,?. |
| 6 | Crop-history: | Diff-1st-year,same-1st-yr,same-1st-two-yrs, same-1st-sev-yrs,?. |
| 7 | Area-damaged: | Scattered,low-areas,upper-areas,whole-field,?. |
| 8 | Severity: | Minor,pot-severe,severe,?. |
| 9 | Seed-treatment: | None,fungicide,other,?. |
| 10 | Germination: | 90-100%,80-89%,lt-80%,?. |
| 11 | Plant-growth: | Norm,abnorm,?. |
| 12 | Leaves: | Norm,abnorm. |
| 13 | Leafspots-halo: | Absent,yellow-halos,no-yellow-halos,?. |
| 14 | Leafspots-marg: | w-s-marg,no-w-s-marg,dna,?. |
| 15 | Leafspot-size: | lt-1/8,gt-1/8,dna,?. |
| 16 | Leaf-shread: | Absent,present,?. |
| 17 | Leaf-malf: | Absent,present,?. |
| 18 | Leaf-mild: | Absent,upper-surf,lower-surf,?. |
| 19 | Stem: | Norm,abnorm,?. |
| 20 | Lodging: | Yes,no,?. |
| 21 | Stem-cankers: | Absent,below-soil,above-soil,above-sec-nde,?. |
| 22 | Canker-lesion: | DNA,brown,dk-brown-blk,tan,?. |
| 23 | Fruiting-bodies: | Absent,present,?. |
| 24 | External decay: | Absent,firm-and-dry,watery,?. |
| 25 | Mycelium: | Absent,present,?. |
| 26 | Int-discolor: | None,brown,black,?. |
| 27 | Sclerotia: | Absent,present,?. |
| 28 | Fruit-pods: | Norm,diseased,few-present,dna,?. |
| 29 | Fruit spots: | Absent,colored,brown-w/blk-specks,distort,dna,?. |
| 30 | Seed: | Norm,abnorm,?. |
| 31 | Mold-growth: | Absent,present,?. |
| 32 | Seed-discolor: | Absent,present,?. |
| 33 | Seed-size: | Norm,lt-norm,?. |
| 34 | Shriveling: | Absent,present,?. |
| 35 | Roots: | norm,rotted,galls-cysts,?. |

A.2 SOYBEAN DATA ATTRIBUTES AND REPRESENTATION

| Attribute No. | Attribute Value | Representation |
|---------------|---|-----------------------|
| 1 | April, May, June, July, August, September, October,?. | 0,1, 2, 3, 4, 5, 6, 7 |
| 2 | Normal,lt-normal,?. | 0, 1, 2 |
| 3 | lt-norm,norm,gt-norm,?. | 0, 1, 2 |
| 4 | lt-norm,norm,gt-norm,?. | 0, 1, 2 |
| 5 | yes,no,?. | 0, 1, 2 |
| 6 | Diff-lst-year,same-lst-yr,same-lst-two-yrs, same-lst-sev-yrs,?. | 0,1, 2, 3, 4 |
| 7 | Scattered,low-areas,upper-areas,whole-field,?. | 0,1, 2, 3, 4 |
| 8 | Minor,pot-severe,severe,?. | 0,1, 2, 3 |
| 9 | None,fungicide,other,?. | 0,1, 2, 3 |
| 10 | 90-100%,80-89%,lt-80%,?. | 0,1, 2, 3 |
| 11 | Norm,abnorm,?. | 0,1, 2 |
| 12 | Norm,abnorm. | 0,1, 2 |
| 13 | Absent,yellow-halos,no-yellow-halos,?. | 0,1, 2, 3, 4 |
| 14 | w-s-marg,no-w-s-marg,dna,?. | 0,1, 2, 3 |
| 15 | lt-1/8,gt-1/8,dna,?. | 0,1, 2, 3 |
| 16 | Absent,present,?. | 0,1, 2 |
| 17 | Absent,present,?. | 0,1, 2 |
| 18 | Absent,upper-surf,lower-surf,?. | 0,1, 2, 3 |
| 19 | Norm,abnorm,?. | 0,1, 2 |
| 20 | Yes,no,?. | 0,1, 2 |
| 21 | Absent,below-soil,above-soil,above-second,?. | 0,1, 2, 3 |
| 22 | DNA,brown,dk-brown-blk,tan,?. | 0,1, 2, 3, 4 |
| 23 | Absent,present,?. | 0,1, 2 |
| 24 | Absent,firm-and-dry,watery,?. | 0,1, 2, 3 |
| 25 | Absent,present,?. | 0,1, 2 |
| 26 | None,brown,black,?. | 0,1, 2, 3 |
| 27 | Absent,present,?. | 0,1, 2 |
| 28 | Norm,diseased,few-present,dna,?. | 0,1, 2, 3 |
| 29 | Absent,colored,brown-w/blk-specks,distort,dna,?. | 0,1, 2, 3, 4, 5 |
| 30 | Norm,abnorm,?. | 0,1, 2 |
| 31 | Absent,present,?. | 0,1, 2 |
| 32 | Absent,present,?. | 0,1, 2 |
| 33 | Norm,lt-norm,?. | 0,1, 2 |
| 34 | Absent,present,?. | 0,1, 2 |
| 35 | norm,rotted,galls-cysts,?. | 0,1, 2, 3 |

Note: Unknown value “?” is denoted by 0.

A.3 SOYBEAN DATA AND MICHALSKI CLASSIFICATION

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D1 | 4 | 0 | 2 | 1 | 1 | 1 | 0 | 1 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 3 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D2 | 5 | 0 | 2 | 1 | 0 | 3 | 1 | 1 | 1 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 3 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D3 | 3 | 0 | 2 | 1 | 0 | 2 | 0 | 2 | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 3 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D4 | 6 | 0 | 2 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 3 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D5 | 4 | 0 | 2 | 1 | 0 | 3 | 0 | 2 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 3 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D6 | 5 | 0 | 2 | 1 | 0 | 2 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 3 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D7 | 3 | 0 | 2 | 1 | 0 | 2 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 3 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D8 | 3 | 0 | 2 | 1 | 0 | 1 | 0 | 2 | 1 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 3 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D9 | 6 | 0 | 2 | 1 | 0 | 3 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 3 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D10 | 6 | 0 | 2 | 1 | 0 | 1 | 0 | 1 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 3 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D11 | 6 | 0 | 0 | 2 | 1 | 0 | 2 | 1 | 0 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D12 | 4 | 0 | 0 | 1 | 0 | 2 | 3 | 1 | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D13 | 5 | 0 | 0 | 2 | 0 | 3 | 2 | 1 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D14 | 6 | 0 | 0 | 1 | 1 | 3 | 3 | 1 | 1 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D15 | 3 | 0 | 0 | 2 | 1 | 0 | 2 | 1 | 0 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D16 | 4 | 0 | 0 | 1 | 1 | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D17 | 3 | 0 | 0 | 1 | 0 | 1 | 2 | 1 | 0 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D18 | 5 | 0 | 0 | 2 | 1 | 2 | 2 | 1 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D19 | 6 | 0 | 0 | 2 | 0 | 1 | 3 | 1 | 1 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D20 | 5 | 0 | 0 | 2 | 1 | 3 | 3 | 1 | 1 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D21 | 0 | 1 | 2 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D22 | 2 | 1 | 2 | 0 | 0 | 3 | 1 | 2 | 0 | 1 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D23 | 2 | 1 | 2 | 0 | 0 | 2 | 1 | 1 | 0 | 2 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D24 | 0 | 1 | 2 | 0 | 0 | 0 | 1 | 1 | 1 | 2 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D25 | 0 | 1 | 2 | 0 | 0 | 2 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D26 | 4 | 0 | 2 | 0 | 1 | 0 | 1 | 2 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D27 | 2 | 1 | 2 | 0 | 0 | 3 | 1 | 2 | 0 | 2 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D28 | 0 | 1 | 2 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D29 | 3 | 0 | 2 | 0 | 1 | 3 | 1 | 2 | 0 | 1 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D30 | 0 | 1 | 2 | 0 | 0 | 1 | 1 | 2 | 1 | 2 | 1 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| D31 | 2 | 1 | 2 | 1 | 1 | 3 | 1 | 2 | 1 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 2 | 2 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D32 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D33 | 3 | 1 | 2 | 0 | 0 | 1 | 1 | 2 | 1 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D34 | 2 | 1 | 2 | 1 | 1 | 1 | 1 | 2 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D35 | 1 | 1 | 2 | 0 | 0 | 3 | 1 | 1 | 1 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D36 | 1 | 1 | 2 | 1 | 0 | 0 | 1 | 2 | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D37 | 0 | 1 | 2 | 1 | 0 | 3 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D38 | 2 | 1 | 2 | 0 | 0 | 1 | 1 | 2 | 0 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D39 | 3 | 1 | 2 | 0 | 0 | 2 | 1 | 2 | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D40 | 3 | 1 | 1 | 0 | 0 | 2 | 1 | 2 | 1 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D41 | 0 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D42 | 1 | 1 | 2 | 1 | 1 | 3 | 1 | 2 | 0 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 1 | 2 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D43 | 1 | 1 | 2 | 0 | 0 | 0 | 1 | 2 | 1 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D44 | 1 | 1 | 2 | 1 | 1 | 2 | 3 | 1 | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 2 | 2 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D45 | 2 | 1 | 1 | 0 | 0 | 3 | 1 | 2 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D46 | 0 | 1 | 1 | 1 | 1 | 2 | 1 | 2 | 1 | 0 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 1 | 2 | 2 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |
| D47 | 0 | 1 | 2 | 1 | 0 | 3 | 1 | 1 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 1 |

Note: Michalski classification

1. D1-D10
2. D11-D20
3. D21-D30
4. D31-D47

APPENDIX B

EXPERIMENT 2

B.1 AU STUDENTS' DATA SELECTED FOR EXPERIMENT 2

| Data | E | M | G2 |
|------|---|----|------|
| 1 | 5 | 4 | 2.83 |
| 2 | 5 | 7 | 2.80 |
| 3 | 3 | 2 | 2.25 |
| 4 | 5 | 7 | 2.83 |
| 5 | 1 | 1 | 1.67 |
| 6 | 5 | 7 | 2.50 |
| 7 | 5 | 7 | 2.83 |
| 8 | 8 | 10 | 3.60 |
| 9 | 5 | 2 | 2.40 |
| 10 | 5 | 7 | 2.50 |
| 11 | 8 | 7 | 3.20 |
| 12 | 5 | 4 | 2.67 |
| 13 | 5 | 7 | 2.50 |
| 14 | 5 | 2 | 2.33 |
| 15 | 5 | 4 | 2.80 |
| 16 | 5 | 4 | 2.50 |
| 17 | 5 | 7 | 2.60 |
| 18 | 5 | 4 | 2.60 |
| 19 | 5 | 2 | 2.33 |
| 20 | 5 | 7 | 2.50 |
| 21 | 5 | 7 | 2.60 |
| 22 | 8 | 7 | 3.17 |
| 23 | 5 | 7 | 2.60 |
| 24 | 1 | 1 | 1.50 |
| 25 | 5 | 7 | 2.83 |
| 26 | 5 | 7 | 2.50 |
| 27 | 5 | 7 | 2.50 |
| 28 | 5 | 4 | 2.60 |
| 29 | 5 | 4 | 2.85 |
| 30 | 5 | 7 | 2.50 |
| 31 | 5 | 2 | 2.00 |
| 32 | 5 | 7 | 2.67 |
| 33 | 5 | 4 | 2.50 |
| 34 | 5 | 7 | 2.67 |
| 35 | 5 | 4 | 2.83 |
| 36 | 8 | 10 | 3.50 |
| 37 | 5 | 4 | 2.67 |
| 38 | 5 | 7 | 2.83 |
| 39 | 8 | 7 | 3.33 |

| Data | E | M | G |
|------|---|----|------|
| 40 | 5 | 4 | 2.83 |
| 41 | 5 | 4 | 2.83 |
| 42 | 5 | 7 | 2.67 |
| 43 | 5 | 7 | 2.75 |
| 44 | 1 | 1 | 1.50 |
| 45 | 5 | 4 | 2.50 |
| 46 | 5 | 7 | 2.50 |
| 47 | 1 | 1 | 1.67 |
| 48 | 5 | 2 | 2.00 |
| 49 | 5 | 7 | 2.83 |
| 50 | 8 | 10 | 3.67 |
| 51 | 1 | 0 | 1.00 |
| 52 | 1 | 1 | 1.00 |
| 53 | 5 | 2 | 2.00 |
| 54 | 5 | 2 | 2.00 |
| 55 | 5 | 7 | 2.83 |
| 56 | 5 | 7 | 2.67 |
| 57 | 5 | 2 | 2.20 |
| 58 | 1 | 1 | 1.50 |
| 59 | 8 | 7 | 3.00 |
| 60 | 5 | 7 | 2.67 |
| 61 | 5 | 7 | 2.50 |
| 62 | 5 | 7 | 2.50 |
| 63 | 5 | 4 | 2.80 |
| 64 | 5 | 7 | 2.67 |
| 65 | 5 | 7 | 2.67 |
| 66 | 5 | 7 | 2.83 |
| 67 | 5 | 7 | 2.83 |
| 68 | 5 | 4 | 2.50 |
| 69 | 5 | 4 | 2.67 |
| 70 | 1 | 1 | 1.33 |
| 71 | 5 | 7 | 2.60 |
| 72 | 8 | 7 | 3.33 |
| 73 | 5 | 7 | 2.80 |
| 74 | 5 | 7 | 2.67 |
| 75 | 8 | 7 | 3.17 |
| 76 | 5 | 7 | 2.60 |
| 77 | 8 | 10 | 3.83 |
| 78 | 5 | 7 | 2.50 |

| Data | E | M | G2 |
|------|---|----|------|
| 79 | 8 | 10 | 3.67 |
| 80 | 5 | 7 | 2.67 |
| 81 | 5 | 2 | 2.20 |
| 82 | 5 | 4 | 2.50 |
| 83 | 5 | 7 | 2.83 |
| 84 | 8 | 7 | 3.00 |
| 85 | 5 | 7 | 2.80 |
| 86 | 5 | 7 | 2.67 |
| 87 | 5 | 4 | 2.67 |
| 88 | 5 | 7 | 2.60 |
| 89 | 8 | 7 | 3.00 |
| 90 | 8 | 10 | 3.86 |
| 91 | 5 | 7 | 2.80 |
| 92 | 5 | 4 | 2.80 |
| 93 | 5 | 7 | 2.50 |
| 94 | 5 | 4 | 2.83 |
| 95 | 5 | 4 | 2.50 |
| 96 | 8 | 10 | 3.57 |
| 97 | 5 | 7 | 2.80 |
| 98 | 5 | 7 | 2.50 |
| 99 | 8 | 7 | 3.17 |
| 100 | 8 | 10 | 3.50 |
| 101 | 5 | 7 | 2.83 |
| 102 | 5 | 7 | 2.50 |
| 103 | 5 | 4 | 2.83 |
| 104 | 5 | 7 | 2.83 |
| 105 | 5 | 7 | 2.83 |
| 106 | 5 | 4 | 2.75 |
| 107 | 5 | 2 | 2.20 |
| 108 | 5 | 7 | 2.50 |
| 109 | 5 | 7 | 2.60 |
| 110 | 1 | 1 | 1.67 |
| 111 | 8 | 10 | 3.67 |
| 112 | 8 | 10 | 3.50 |
| 113 | 5 | 7 | 2.67 |
| 114 | 5 | 4 | 2.50 |
| 115 | 1 | 1 | 1.33 |
| 116 | 8 | 7 | 3.00 |
| 117 | 8 | 10 | 3.83 |
| 118 | 5 | 7 | 2.50 |
| 119 | 5 | 7 | 2.80 |
| 120 | 5 | 4 | 2.50 |
| 121 | 5 | 7 | 2.80 |
| 122 | 5 | 7 | 2.83 |
| 123 | 8 | 10 | 3.50 |
| 124 | 1 | 1 | 1.86 |
| 125 | 8 | 7 | 3.33 |
| 126 | 5 | 7 | 2.67 |
| 127 | 1 | 1 | 1.50 |

| Data | E | M | G2 |
|------|----|----|------|
| 128 | 1 | 1 | 1.67 |
| 129 | 8 | 10 | 3.50 |
| 130 | 8 | 7 | 3.40 |
| 131 | 5 | 7 | 2.83 |
| 132 | 5 | 7 | 2.80 |
| 133 | 5 | 4 | 2.67 |
| 134 | 8 | 10 | 3.67 |
| 135 | 5 | 7 | 2.83 |
| 136 | 8 | 10 | 3.50 |
| 137 | 2 | 2 | 2.20 |
| 138 | 8 | 10 | 3.67 |
| 139 | 5 | 7 | 2.83 |
| 140 | 5 | 7 | 2.83 |
| 141 | 5 | 4 | 2.67 |
| 142 | 5 | 7 | 2.83 |
| 143 | 5 | 7 | 2.80 |
| 144 | 5 | 7 | 2.50 |
| 145 | 8 | 7 | 3.00 |
| 146 | 5 | 4 | 2.83 |
| 147 | 5 | 7 | 2.83 |
| 148 | 5 | 7 | 2.50 |
| 149 | 5 | 7 | 2.67 |
| 150 | 5 | 7 | 2.83 |
| 151 | 5 | 2 | 2.00 |
| 152 | 5 | 7 | 2.60 |
| 153 | 5 | 4 | 2.60 |
| 154 | 8 | 7 | 3.33 |
| 155 | 1 | 1 | 1.86 |
| 156 | 5 | 4 | 2.50 |
| 157 | 8 | 10 | 3.67 |
| 158 | 8 | 7 | 3.33 |
| 159 | 1 | 1 | 1.67 |
| 160 | 1 | 1 | 1.33 |
| 161 | 5 | 2 | 2.17 |
| 162 | 0 | 1 | 1.67 |
| 163 | 2 | 2 | 2.40 |
| 164 | 11 | 10 | 3.83 |
| 165 | 5 | 7 | 2.67 |
| 166 | 8 | 10 | 3.50 |
| 167 | 8 | 10 | 3.80 |
| 168 | 8 | 7 | 3.40 |
| 169 | 1 | 1 | 1.33 |
| 170 | 1 | 0 | 1.50 |
| 171 | 1 | 1 | 1.71 |
| 7172 | 5 | 2 | 2.00 |
| 173 | 8 | 10 | 3.83 |
| 174 | 8 | 7 | 3.17 |
| 175 | 8 | 7 | 3.33 |
| 176 | 8 | 10 | 3.50 |

| Data | E | M | G2 |
|------|---|----|------|
| 177 | 8 | 10 | 3.67 |
| 178 | 8 | 10 | 3.67 |
| 179 | 8 | 10 | 3.83 |
| 180 | 8 | 10 | 3.67 |
| 181 | 8 | 10 | 3.83 |
| 182 | 8 | 10 | 3.50 |
| 183 | 8 | 10 | 3.50 |
| 184 | 8 | 10 | 3.67 |
| 185 | 8 | 10 | 3.67 |
| 186 | 8 | 10 | 3.83 |
| 187 | 8 | 10 | 3.80 |
| 188 | 8 | 10 | 3.50 |
| 189 | 8 | 7 | 3.00 |
| 190 | 8 | 7 | 3.17 |
| 191 | 8 | 10 | 3.50 |
| 192 | 1 | 1 | 1.00 |
| 193 | 1 | 1 | 1.67 |
| 194 | 2 | 2 | 2.40 |
| 195 | 1 | 1 | 1.67 |
| 196 | 8 | 10 | 3.67 |
| 197 | 8 | 10 | 3.80 |
| 198 | 8 | 7 | 3.40 |
| 199 | 1 | 1 | 1.20 |
| 200 | 8 | 10 | 3.67 |
| 201 | 8 | 10 | 3.83 |
| 202 | 5 | 2 | 2.25 |
| 203 | 2 | 2 | 2.00 |
| 204 | 8 | 7 | 3.33 |
| 205 | 5 | 2 | 2.25 |
| 206 | 8 | 10 | 3.53 |
| 207 | 8 | 7 | 3.17 |
| 208 | 8 | 10 | 3.83 |
| 209 | 8 | 7 | 3.00 |
| 210 | 8 | 10 | 3.50 |
| 211 | 8 | 7 | 3.17 |
| 212 | 8 | 7 | 3.00 |
| 213 | 8 | 10 | 3.83 |
| 214 | 1 | 0 | 1.00 |
| 215 | 5 | 2 | 2.00 |
| 216 | 2 | 2 | 2.00 |
| 217 | 8 | 10 | 3.67 |
| 218 | 1 | 1 | 1.33 |
| 219 | 8 | 10 | 3.67 |
| 220 | 8 | 10 | 3.83 |
| 221 | 5 | 2 | 2.29 |
| 222 | 8 | 10 | 3.67 |
| 223 | 2 | 2 | 2.17 |
| 224 | 2 | 2 | 2.00 |
| 225 | 0 | 1 | 1.50 |

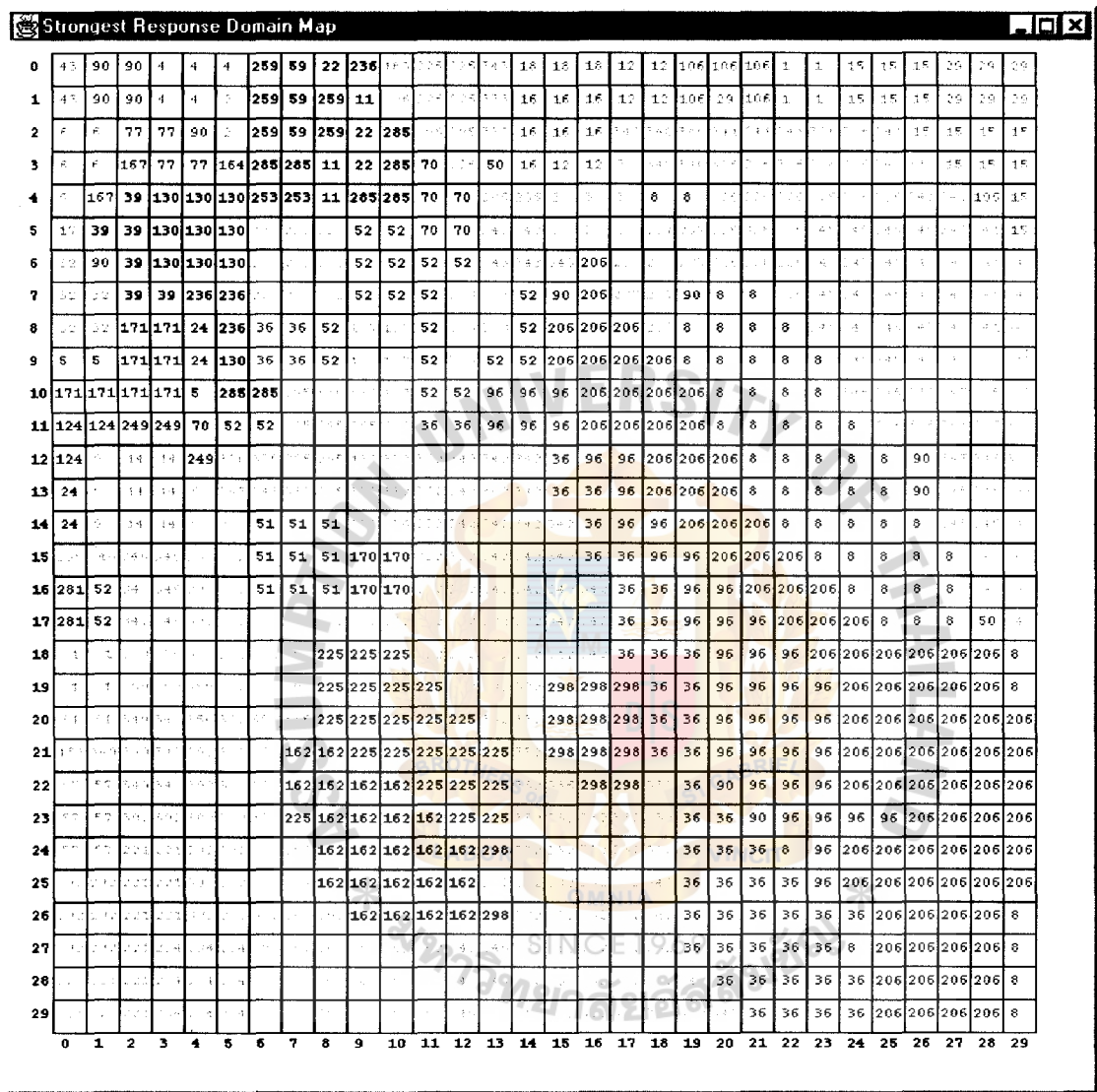
| Data | E | M | G2 |
|------|---|----|------|
| 226 | 2 | 2 | 2.33 |
| 227 | 8 | 10 | 3.83 |
| 228 | 2 | 2 | 2.17 |
| 229 | 8 | 10 | 3.50 |
| 230 | 2 | 2 | 2.40 |
| 231 | 8 | 10 | 3.83 |
| 232 | 5 | 2 | 2.00 |
| 233 | 2 | 2 | 2.33 |
| 234 | 5 | 2 | 2.31 |
| 235 | 2 | 2 | 2.00 |
| 236 | 8 | 7 | 3.43 |
| 237 | 5 | 2 | 2.00 |
| 238 | 8 | 7 | 3.00 |
| 239 | 5 | 2 | 2.17 |
| 240 | 2 | 2 | 2.33 |
| 241 | 2 | 2 | 2.17 |
| 242 | 8 | 7 | 3.00 |
| 243 | 8 | 7 | 3.17 |
| 244 | 2 | 2 | 2.00 |
| 245 | 5 | 2 | 2.33 |
| 246 | 2 | 2 | 2.00 |
| 247 | 1 | 1 | 1.50 |
| 248 | 2 | 2 | 2.00 |
| 249 | 1 | 1 | 1.60 |
| 250 | 2 | 2 | 2.40 |
| 251 | 8 | 7 | 3.17 |
| 252 | 2 | 2 | 2.20 |
| 253 | 8 | 7 | 3.37 |
| 254 | 5 | 2 | 2.17 |
| 255 | 8 | 7 | 3.33 |
| 256 | 8 | 7 | 3.00 |
| 257 | 8 | 7 | 3.40 |
| 258 | 8 | 7 | 3.17 |
| 259 | 8 | 7 | 3.05 |
| 260 | 8 | 7 | 3.17 |
| 261 | 8 | 7 | 3.33 |
| 262 | 8 | 7 | 3.00 |
| 263 | 8 | 7 | 3.17 |
| 264 | 8 | 7 | 3.00 |
| 265 | 8 | 7 | 3.00 |
| 266 | 8 | 7 | 3.33 |
| 267 | 8 | 7 | 3.37 |
| 268 | 8 | 7 | 3.33 |
| 269 | 8 | 7 | 3.00 |
| 270 | 5 | 2 | 2.00 |
| 271 | 2 | 2 | 2.00 |
| 272 | 1 | 1 | 1.00 |
| 273 | 1 | 1 | 1.33 |
| 274 | 1 | 0 | 1.50 |

| Data | E | M | G2 |
|------|---|---|------|
| 275 | 2 | 2 | 2.00 |
| 276 | 5 | 2 | 2.40 |
| 277 | 2 | 2 | 2.00 |
| 278 | 5 | 2 | 2.00 |
| 279 | 2 | 2 | 2.00 |
| 280 | 1 | 0 | 1.00 |
| 281 | 1 | 0 | 0.00 |
| 282 | 2 | 2 | 2.00 |
| 283 | 1 | 1 | 1.50 |
| 284 | 1 | 1 | 1.00 |
| 285 | 8 | 7 | 3.27 |
| 286 | 3 | 2 | 2.00 |
| 287 | 3 | 2 | 2.15 |
| 288 | 5 | 2 | 2.17 |
| 289 | 0 | 1 | 1.50 |
| 290 | 3 | 2 | 2.00 |
| 291 | 8 | 7 | 3.40 |
| 292 | 1 | 1 | 1.00 |
| 293 | 8 | 7 | 3.37 |
| 294 | 1 | 1 | 1.67 |
| 295 | 2 | 2 | 2.25 |
| 296 | 2 | 2 | 2.00 |
| 297 | 1 | 1 | 1.67 |
| 298 | 0 | 1 | 1.33 |
| 299 | 1 | 1 | 1.50 |
| 300 | 1 | 1 | 1.00 |
| 301 | 1 | 1 | 1.50 |
| 302 | 2 | 3 | 2.40 |
| 303 | 1 | 1 | 1.50 |
| 304 | 2 | 2 | 2.17 |
| 305 | 2 | 2 | 2.17 |
| 306 | 2 | 2 | 2.33 |
| 307 | 1 | 1 | 1.50 |
| 308 | 1 | 1 | 1.67 |
| 309 | 1 | 0 | 1.00 |
| 310 | 1 | 1 | 1.50 |
| 311 | 1 | 1 | 1.00 |
| 312 | 2 | 2 | 2.25 |
| 313 | 2 | 2 | 2.20 |
| 314 | 2 | 2 | 2.40 |
| 315 | 5 | 2 | 2.00 |
| 316 | 5 | 2 | 2.20 |
| 317 | 2 | 2 | 2.00 |
| 318 | 5 | 2 | 2.25 |
| 319 | 2 | 2 | 2.33 |
| 320 | 5 | 2 | 2.33 |
| 321 | 5 | 2 | 2.00 |
| 322 | 2 | 2 | 2.00 |
| 323 | 2 | 2 | 2.20 |

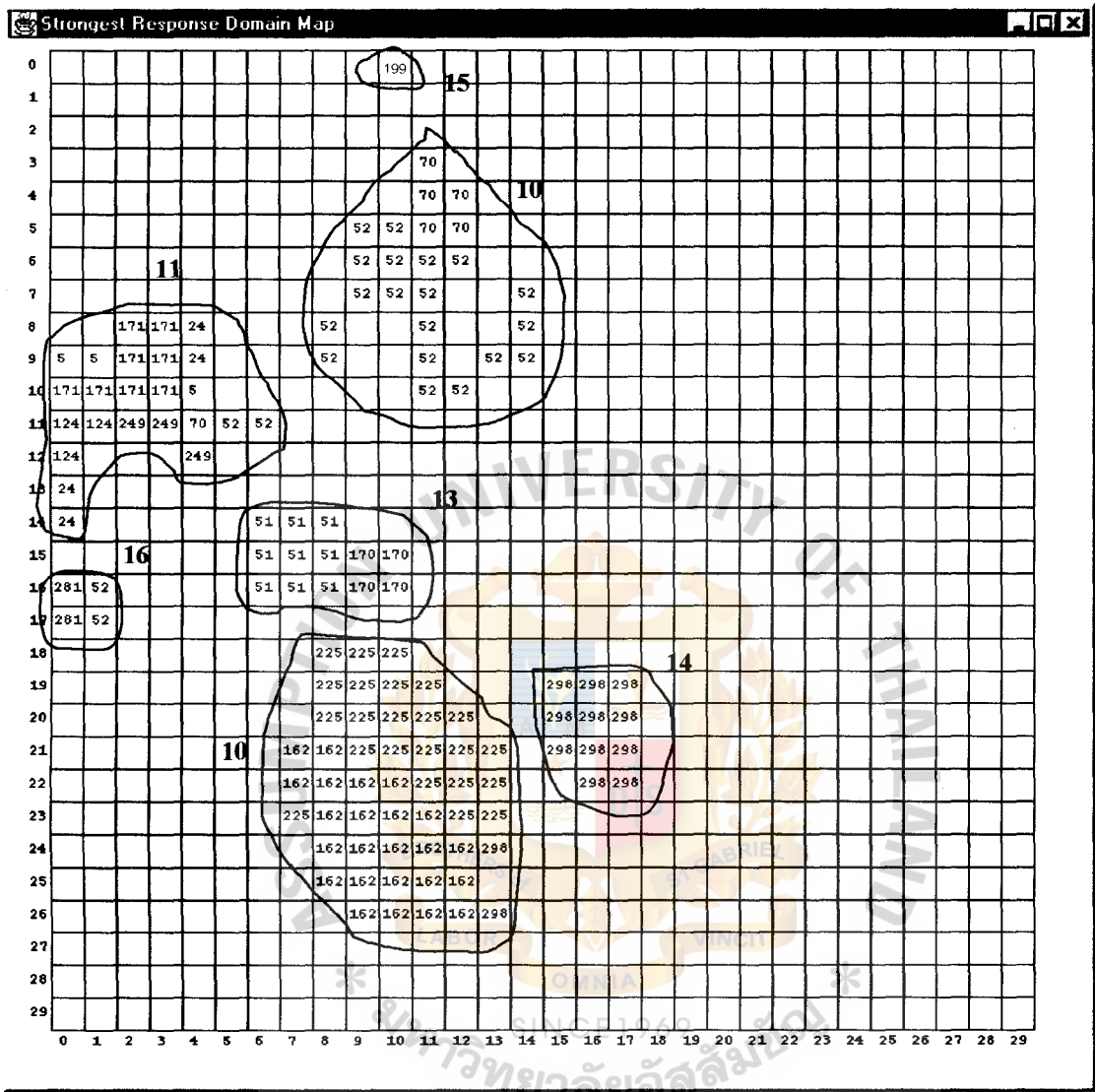
| Data | E | M | G2 |
|------|---|---|------|
| 324 | 2 | 2 | 2.40 |
| 325 | 5 | 2 | 2.00 |
| 326 | 2 | 3 | 2.00 |
| 327 | 5 | 2 | 2.00 |
| 328 | 5 | 2 | 2.40 |
| 329 | 2 | 2 | 2.25 |
| 330 | 2 | 2 | 2.21 |
| 331 | 2 | 3 | 2.40 |
| 332 | 2 | 3 | 2.33 |
| 333 | 2 | 2 | 2.05 |
| 334 | 2 | 2 | 2.05 |
| 335 | 2 | 3 | 2.33 |
| 336 | 2 | 3 | 2.42 |
| 337 | 2 | 2 | 2.25 |
| 338 | 2 | 2 | 2.33 |
| 339 | 2 | 2 | 2.00 |
| 340 | 2 | 3 | 2.17 |
| 341 | 2 | 3 | 2.13 |
| 342 | 2 | 3 | 2.30 |
| 343 | 2 | 2 | 2.08 |
| 344 | 2 | 2 | 2.40 |
| 345 | 2 | 3 | 2.10 |
| 346 | 2 | 3 | 2.38 |
| 347 | 2 | 2 | 2.08 |
| 348 | 2 | 2 | 2.17 |
| 349 | 5 | 2 | 2.05 |
| 350 | 5 | 3 | 2.15 |

Note that: E, M, G2 stand for English,
Mathematics and GPA of the second
semester

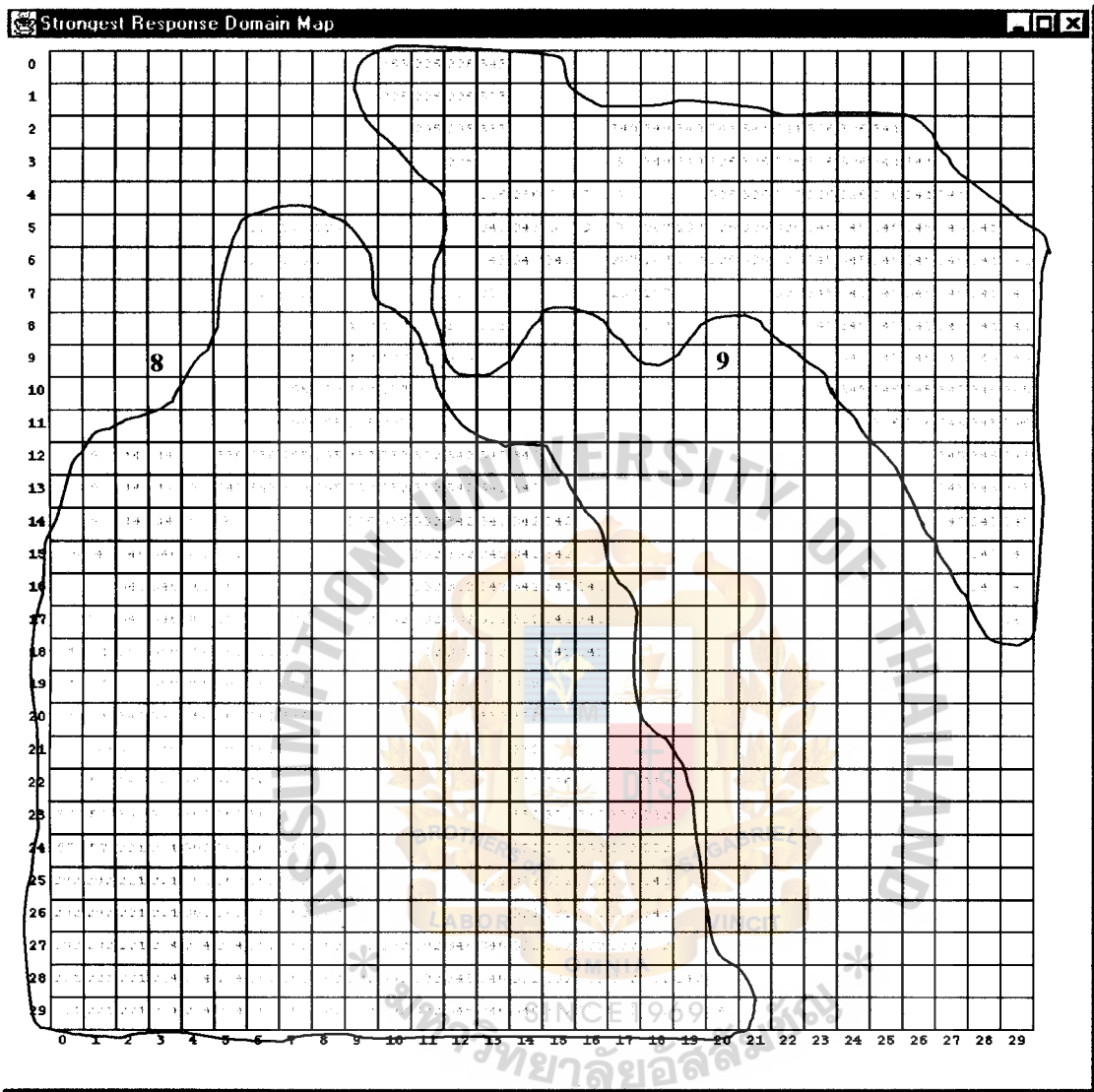
B.2 CLUSTERS VERSUS THE STRONGEST RESPONSE OF ALL INPUT



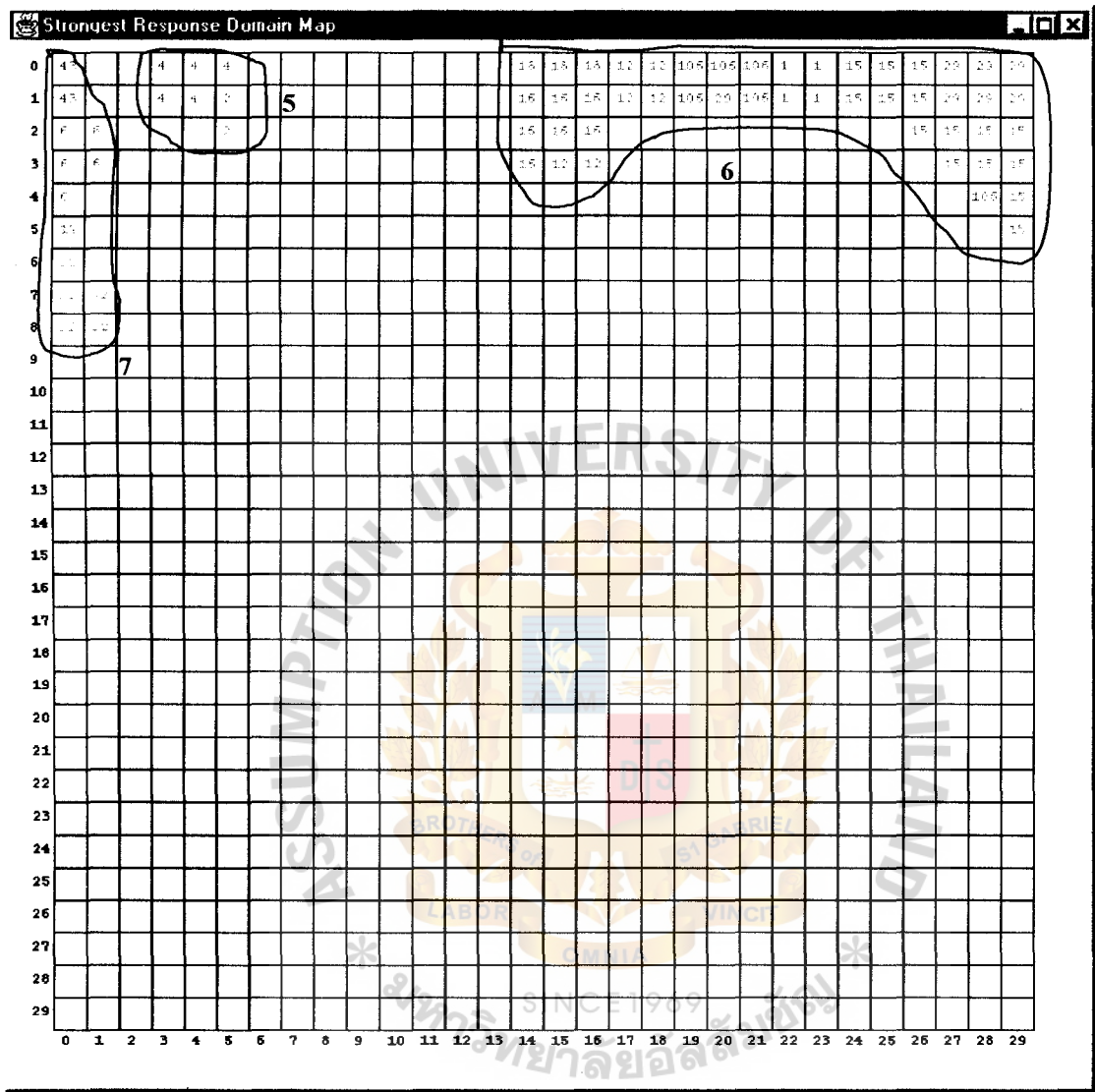
B.2.1 CLUSTERS VERSUS THE STRONGEST RESPONSE WITH G2 = [0.0, 2.0)



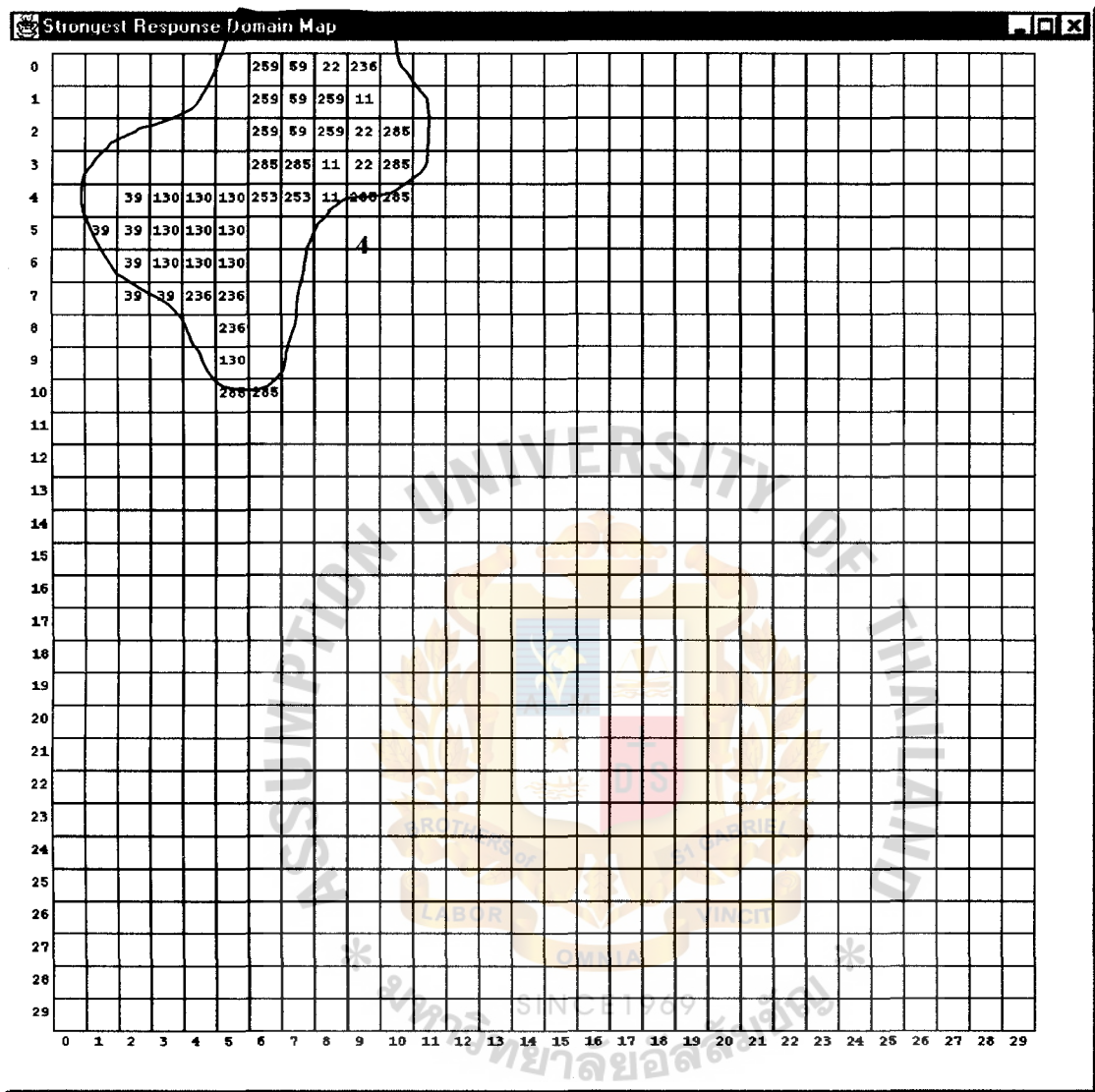
B.2.2 CLUSTERS VERSUS THE STRONGEST RESPONSE WITH G2 = [2.0, 2.5)



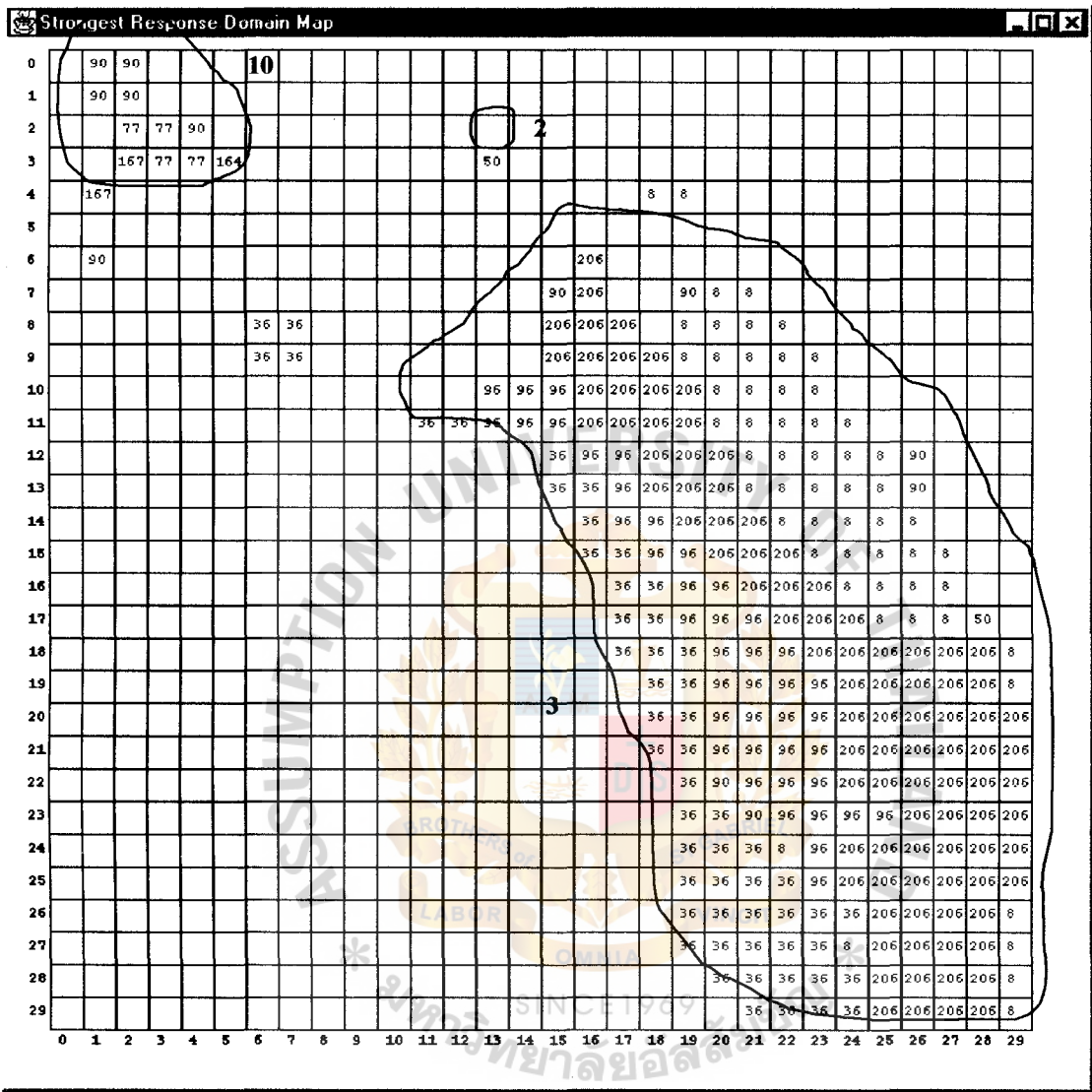
B.2.3 CLUSTERS VERSUS THE STRONGEST RESPONSE WITH G2 = [2.5, 3.0)



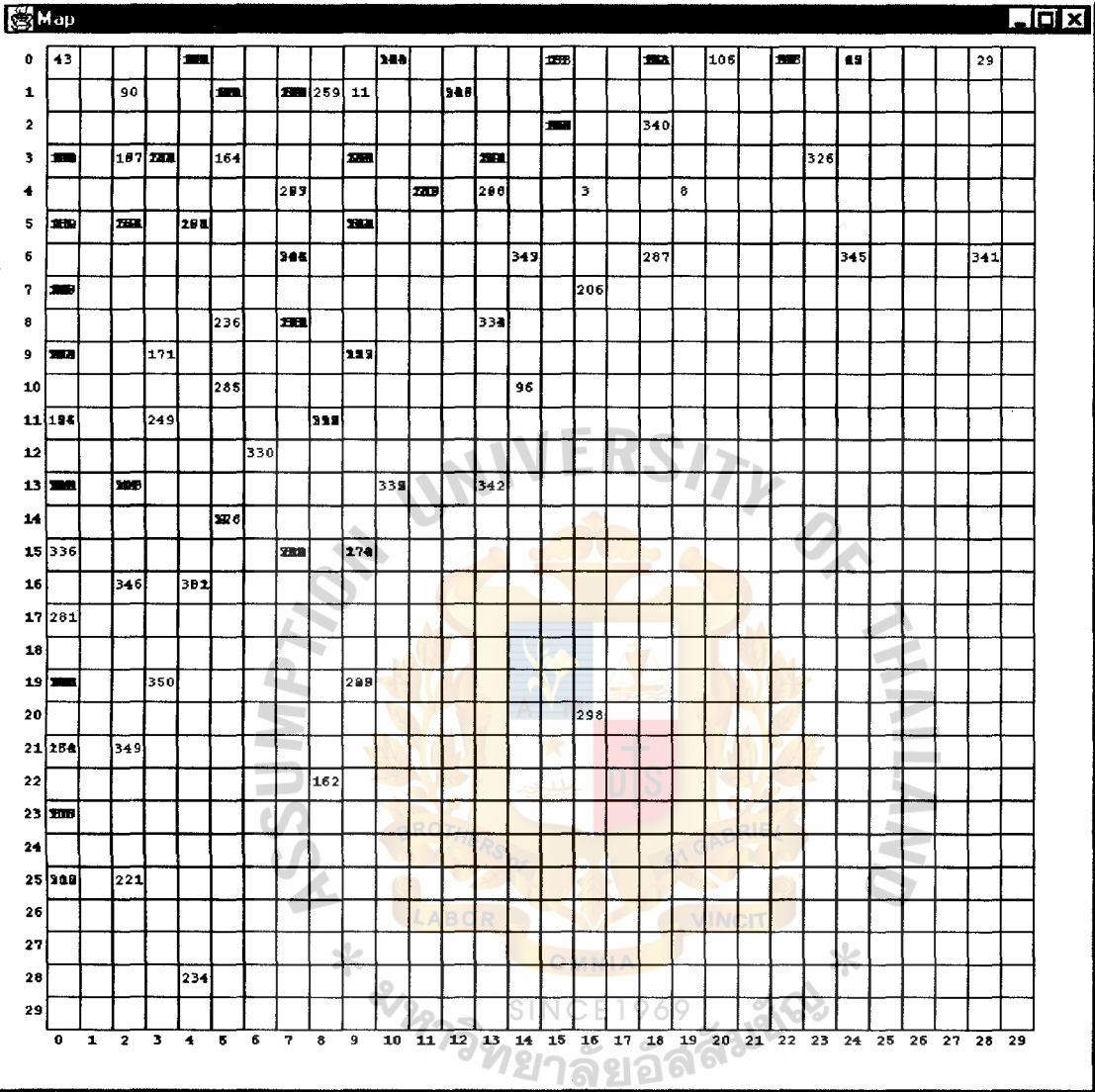
B.2.4 CLUSTERS VERSUS THE STRONGEST RESPONSE WITH G2 = [3.0, 3.5)



B.2.5 CLUSTERS VERSUS THE STRONGEST RESPONSE WITH G2 = [3.5, 4.0]



B.3 THE ACTIVATED NEURON OF EACH INPUT



APPENDIX C

EXPERIMENT 3

C.1 AU STUDENTS' DATA USED IN EXPERIMENT 3

| Data | E | M | C1 | G1 | C2 | G2 |
|------|---|----|----|-------|----|-------|
| 1 | 5 | 0 | 15 | 2.200 | 16 | 1.750 |
| 2 | 5 | 1 | 13 | 2.846 | 12 | 1.750 |
| 3 | 1 | 4 | 7 | 2.286 | 9 | 1.333 |
| 4 | 5 | 4 | 19 | 2.737 | 18 | 1.833 |
| 5 | 3 | 4 | 19 | 2.421 | 12 | 1.250 |
| 6 | 5 | 2 | 19 | 2.053 | 15 | 1.600 |
| 7 | 5 | 1 | 16 | 2.688 | 15 | 1.200 |
| 8 | 2 | 7 | 16 | 3.250 | 18 | 1.833 |
| 9 | 3 | 4 | 19 | 2.684 | 12 | 1.250 |
| 10 | 3 | 1 | 16 | 2.688 | 15 | 1.800 |
| 11 | 2 | 7 | 10 | 3.700 | 18 | 1.667 |
| 12 | 2 | 10 | 10 | 3.100 | 12 | 1.750 |
| 13 | 2 | 2 | 10 | 2.400 | 18 | 1.333 |
| 14 | 1 | 1 | 10 | 2.200 | 15 | 1.800 |
| 15 | 3 | 7 | 15 | 2.400 | 18 | 1.833 |
| 16 | 5 | 4 | 19 | 2.737 | 12 | 1.750 |
| 17 | 5 | 4 | 16 | 2.438 | 15 | 1.600 |
| 18 | 1 | 2 | 9 | 2.333 | 7 | 1.857 |
| 19 | 5 | 4 | 19 | 2.737 | 18 | 1.833 |
| 20 | 5 | 4 | 16 | 2.125 | 6 | 1.000 |
| 21 | 5 | 1 | 13 | 2.385 | 15 | 1.600 |
| 22 | 5 | 7 | 19 | 2.895 | 18 | 1.667 |
| 23 | 2 | 7 | 9 | 3.000 | 12 | 1.250 |
| 24 | 2 | 1 | 6 | 2.500 | 9 | 1.333 |
| 25 | 5 | 7 | 18 | 2.500 | 15 | 1.400 |
| 26 | 1 | 7 | 7 | 3.143 | 12 | 1.500 |
| 27 | 5 | 10 | 19 | 2.263 | 12 | 1.750 |
| 28 | 5 | 1 | 16 | 2.688 | 15 | 1.200 |
| 29 | 1 | 4 | 10 | 2.800 | 6 | 1.000 |
| 30 | 2 | 7 | 10 | 3.400 | 15 | 1.600 |
| 31 | 1 | 4 | 10 | 2.800 | 9 | 1.667 |
| 32 | 5 | 1 | 15 | 2.000 | 15 | 1.800 |
| 33 | 5 | 7 | 19 | 2.737 | 18 | 1.500 |
| 34 | 2 | 7 | 10 | 3.700 | 15 | 1.400 |
| 35 | 5 | 4 | 18 | 2.500 | 15 | 1.600 |
| 36 | 2 | 4 | 10 | 2.800 | 15 | 1.600 |
| 37 | 2 | 4 | 10 | 2.500 | 15 | 1.800 |

| Data | E | M | C1 | G1 | C2 | G2 |
|------|---|----|----|-------|----|-------|
| 38 | 1 | 2 | 9 | 2.000 | 6 | 1.500 |
| 39 | 2 | 7 | 16 | 2.250 | 15 | 1.600 |
| 40 | 5 | 7 | 19 | 3.211 | 9 | 1.667 |
| 41 | 5 | 4 | 19 | 2.684 | 9 | 1.667 |
| 42 | 2 | 4 | 10 | 3.100 | 12 | 1.750 |
| 43 | 2 | 4 | 18 | 1.500 | 12 | 0.750 |
| 44 | 1 | 1 | 4 | 3.250 | 9 | 1.667 |
| 45 | 2 | 7 | 10 | 3.400 | 18 | 1.833 |
| 46 | 2 | 4 | 16 | 2.500 | 15 | 1.600 |
| 47 | 5 | 4 | 19 | 2.263 | 15 | 1.800 |
| 48 | 1 | 4 | 10 | 2.400 | 9 | 1.667 |
| 49 | 5 | 1 | 16 | 2.313 | 12 | 1.500 |
| 50 | 2 | 4 | 10 | 2.500 | 15 | 1.400 |
| 51 | 3 | 1 | 16 | 2.688 | 18 | 1.667 |
| 52 | 2 | 7 | 16 | 3.250 | 15 | 1.600 |
| 53 | 1 | 4 | 10 | 3.100 | 6 | 1.000 |
| 54 | 2 | 4 | 10 | 2.800 | 18 | 1.833 |
| 55 | 5 | 4 | 18 | 2.500 | 18 | 1.500 |
| 56 | 1 | 1 | 6 | 3.000 | 9 | 1.333 |
| 57 | 2 | 4 | 9 | 3.333 | 15 | 1.800 |
| 58 | 1 | 2 | 9 | 2.333 | 3 | 1.000 |
| 59 | 1 | 4 | 3 | 3.000 | 6 | 1.000 |
| 60 | 1 | 7 | 9 | 2.333 | 9 | 1.667 |
| 61 | 1 | 7 | 9 | 2.333 | 9 | 1.667 |
| 62 | 0 | 7 | 9 | 3.667 | 9 | 1.667 |
| 63 | 5 | 2 | 15 | 2.400 | 18 | 1.833 |
| 64 | 5 | 4 | 18 | 2.500 | 15 | 1.400 |
| 65 | 2 | 1 | 6 | 2.500 | 15 | 1.800 |
| 66 | 2 | 1 | 3 | 3.000 | 15 | 1.600 |
| 67 | 2 | 1 | 6 | 2.500 | 18 | 1.833 |
| 68 | 1 | 1 | 3 | 3.000 | 9 | 1.333 |
| 69 | 2 | 10 | 9 | 2.667 | 15 | 1.200 |
| 70 | 2 | 1 | 6 | 3.500 | 18 | 1.833 |
| 71 | 3 | 4 | 19 | 2.368 | 18 | 1.833 |
| 72 | 1 | 4 | 9 | 2.667 | 6 | 1.000 |
| 73 | 2 | 7 | 6 | 3.000 | 18 | 1.833 |
| 74 | 2 | 7 | 16 | 3.250 | 18 | 1.833 |

| Data | E | M | C1 | G1 | C2 | G2 |
|------|---|----|----|-------|----|-------|
| 75 | 2 | 4 | 10 | 2.500 | 12 | 1.250 |
| 76 | 5 | 1 | 16 | 2.688 | 18 | 1.833 |
| 77 | 5 | 1 | 15 | 2.400 | 15 | 1.400 |
| 78 | 5 | 7 | 19 | 2.421 | 18 | 1.500 |
| 79 | 8 | 1 | 16 | 2.875 | 18 | 1.833 |
| 80 | 5 | 4 | 18 | 2.000 | 12 | 1.500 |
| 81 | 5 | 1 | 16 | 3.063 | 18 | 1.833 |
| 82 | 1 | 4 | 10 | 2.700 | 9 | 1.667 |
| 83 | 1 | 1 | 4 | 2.500 | 9 | 1.667 |
| 84 | 2 | 7 | 10 | 3.100 | 18 | 1.833 |
| 85 | 1 | 2 | 7 | 3.143 | 9 | 1.667 |
| 86 | 1 | 4 | 10 | 2.800 | 9 | 1.667 |
| 87 | 2 | 1 | 6 | 3.000 | 15 | 1.800 |
| 88 | 5 | 1 | 15 | 2.200 | 18 | 1.500 |
| 89 | 1 | 1 | 4 | 2.500 | 9 | 1.667 |
| 90 | 1 | 4 | 9 | 2.000 | 9 | 1.667 |
| 91 | 2 | 1 | 6 | 2.500 | 12 | 1.750 |
| 92 | 5 | 0 | 13 | 2.231 | 9 | 1.333 |
| 93 | 5 | 2 | 19 | 2.737 | 12 | 1.500 |
| 94 | 2 | 10 | 6 | 3.000 | 15 | 1.400 |
| 95 | 2 | 7 | 9 | 2.333 | 18 | 0.500 |
| 96 | 2 | 1 | 3 | 2.000 | 12 | 1.500 |
| 97 | 2 | 0 | 10 | 2.700 | 15 | 1.600 |
| 98 | 2 | 2 | 12 | 2.000 | 15 | 1.800 |
| 99 | 2 | 1 | 7 | 2.571 | 9 | 1.333 |
| 100 | 2 | 4 | 13 | 2.692 | 15 | 1.800 |
| 101 | 2 | 4 | 10 | 3.000 | 18 | 2.667 |
| 102 | 2 | 1 | 7 | 3.143 | 18 | 2.667 |
| 103 | 2 | 4 | 9 | 2.333 | 15 | 2.800 |
| 104 | 2 | 4 | 10 | 2.500 | 18 | 2.667 |
| 105 | 2 | 1 | 7 | 3.143 | 18 | 2.333 |
| 106 | 2 | 4 | 10 | 2.800 | 15 | 2.800 |
| 107 | 5 | 4 | 18 | 2.500 | 16 | 2.313 |
| 108 | 5 | 4 | 18 | 2.667 | 16 | 2.688 |
| 109 | 5 | 7 | 18 | 3.167 | 19 | 2.895 |
| 110 | 2 | 4 | 10 | 2.400 | 18 | 2.833 |
| 111 | 2 | 7 | 9 | 3.000 | 18 | 2.833 |
| 112 | 5 | 2 | 19 | 2.421 | 15 | 2.000 |
| 113 | 2 | 1 | 7 | 2.286 | 18 | 2.167 |
| 114 | 5 | 1 | 15 | 2.800 | 18 | 2.667 |
| 115 | 2 | 4 | 9 | 2.667 | 15 | 2.600 |
| 116 | 2 | 4 | 9 | 2.667 | 15 | 2.600 |
| 117 | 8 | 4 | 18 | 3.000 | 15 | 2.400 |
| 118 | 2 | 4 | 9 | 2.667 | 15 | 2.200 |
| 119 | 2 | 1 | 7 | 3.000 | 15 | 2.600 |
| 120 | 2 | 4 | 9 | 2.000 | 19 | 2.895 |
| 121 | 5 | 4 | 19 | 2.684 | 18 | 2.667 |
| 122 | 2 | 4 | 9 | 2.333 | 15 | 2.400 |

| Data | E | M | C1 | G1 | C2 | G2 |
|------|---|----|----|-------|----|-------|
| 123 | 5 | 7 | 18 | 3.000 | 18 | 2.667 |
| 124 | 2 | 4 | 10 | 2.700 | 18 | 2.333 |
| 125 | 2 | 0 | 7 | 3.143 | 12 | 2.750 |
| 126 | 8 | 4 | 19 | 3.053 | 18 | 2.667 |
| 127 | 8 | 4 | 18 | 3.000 | 16 | 2.875 |
| 128 | 2 | 1 | 7 | 3.857 | 18 | 2.167 |
| 129 | 5 | 4 | 19 | 2.842 | 18 | 2.500 |
| 130 | 2 | 4 | 9 | 2.333 | 18 | 2.833 |
| 131 | 2 | 7 | 10 | 3.400 | 18 | 2.833 |
| 132 | 2 | 2 | 10 | 2.200 | 15 | 2.000 |
| 133 | 2 | 7 | 9 | 3.000 | 18 | 2.500 |
| 134 | 2 | 4 | 10 | 2.700 | 18 | 2.500 |
| 135 | 1 | 1 | 4 | 3.250 | 9 | 2.333 |
| 136 | 5 | 7 | 16 | 3.063 | 18 | 2.333 |
| 137 | 5 | 4 | 19 | 2.579 | 18 | 2.500 |
| 138 | 2 | 1 | 6 | 3.000 | 15 | 2.000 |
| 139 | 1 | 4 | 7 | 3.000 | 9 | 2.000 |
| 140 | 2 | 1 | 7 | 3.571 | 18 | 2.833 |
| 141 | 2 | 4 | 10 | 2.800 | 12 | 2.750 |
| 142 | 8 | 0 | 15 | 3.400 | 16 | 2.250 |
| 143 | 5 | 1 | 15 | 2.600 | 16 | 2.313 |
| 144 | 5 | 1 | 16 | 2.688 | 18 | 2.000 |
| 145 | 2 | 4 | 10 | 2.800 | 18 | 2.333 |
| 146 | 2 | 4 | 7 | 2.571 | 15 | 2.000 |
| 147 | 2 | 0 | 7 | 3.000 | 15 | 2.000 |
| 148 | 2 | 4 | 9 | 2.667 | 15 | 2.400 |
| 149 | 2 | 1 | 4 | 3.000 | 18 | 2.167 |
| 150 | 5 | 4 | 19 | 2.684 | 15 | 2.200 |
| 151 | 1 | 7 | 7 | 3.143 | 18 | 2.000 |
| 152 | 2 | 4 | 10 | 2.800 | 15 | 2.200 |
| 153 | 2 | 4 | 10 | 2.800 | 18 | 2.167 |
| 154 | 1 | 1 | 4 | 3.000 | 6 | 2.000 |
| 155 | 2 | 1 | 7 | 3.143 | 18 | 2.167 |
| 156 | 2 | 10 | 10 | 3.400 | 18 | 2.833 |
| 157 | 2 | 7 | 10 | 3.100 | 15 | 2.600 |
| 158 | 2 | 0 | 7 | 3.143 | 12 | 2.500 |
| 159 | 2 | 7 | 9 | 2.667 | 18 | 2.667 |
| 160 | 2 | 1 | 7 | 2.714 | 18 | 2.667 |
| 161 | 2 | 4 | 10 | 2.700 | 18 | 2.333 |
| 162 | 2 | 7 | 10 | 3.000 | 18 | 2.833 |
| 163 | 2 | 7 | 10 | 3.400 | 18 | 2.667 |
| 164 | 5 | 4 | 16 | 3.063 | 15 | 2.400 |
| 165 | 2 | 4 | 10 | 2.100 | 18 | 2.333 |
| 166 | 1 | 1 | 3 | 4.000 | 15 | 2.400 |
| 167 | 2 | 10 | 10 | 3.400 | 18 | 2.333 |
| 168 | 1 | 1 | 4 | 3.250 | 18 | 2.667 |
| 169 | 2 | 4 | 10 | 2.800 | 6 | 2.000 |
| 170 | 1 | 2 | 7 | 2.286 | 18 | 2.333 |

| Data | E | M | C1 | G1 | C2 | G2 |
|------|---|----|----|-------|----|-------|
| 171 | 2 | 4 | 9 | 2.667 | 16 | 2.125 |
| 172 | 5 | 10 | 19 | 3.684 | 18 | 3.667 |
| 173 | 2 | 4 | 10 | 2.500 | 12 | 2.250 |
| 174 | 2 | 7 | 9 | 3.333 | 18 | 2.833 |
| 175 | 5 | 7 | 19 | 3.368 | 18 | 2.833 |
| 176 | 1 | 4 | 7 | 2.714 | 9 | 2.000 |
| 177 | 2 | 7 | 10 | 3.100 | 18 | 2.833 |
| 178 | 2 | 10 | 9 | 3.333 | 21 | 2.857 |
| 179 | 2 | 4 | 9 | 2.667 | 3 | 2.000 |
| 180 | 5 | 7 | 19 | 2.895 | 12 | 2.250 |
| 181 | 2 | 10 | 10 | 4.000 | 18 | 3.833 |
| 182 | 5 | 1 | 15 | 2.000 | 15 | 2.400 |
| 183 | 5 | 10 | 19 | 2.895 | 18 | 3.167 |
| 184 | 1 | 10 | 7 | 4.000 | 15 | 2.800 |
| 185 | 5 | 2 | 18 | 2.333 | 6 | 2.000 |
| 186 | 2 | 1 | 7 | 3.143 | 18 | 2.333 |
| 187 | 8 | 4 | 18 | 3.000 | 15 | 3.200 |
| 188 | 2 | 4 | 10 | 2.200 | 18 | 2.000 |
| 189 | 5 | 7 | 19 | 3.053 | 15 | 2.400 |
| 190 | 2 | 4 | 9 | 3.333 | 15 | 3.400 |
| 191 | 5 | 7 | 18 | 3.500 | 19 | 3.211 |
| 192 | 5 | 10 | 18 | 3.667 | 19 | 3.211 |
| 193 | 2 | 4 | 9 | 2.333 | 18 | 2.833 |
| 194 | 2 | 10 | 9 | 3.333 | 15 | 2.400 |
| 195 | 1 | 4 | 7 | 2.714 | 9 | 2.667 |
| 196 | 2 | 4 | 10 | 3.100 | 18 | 3.000 |
| 197 | 2 | 4 | 9 | 2.667 | 12 | 2.000 |
| 198 | 1 | 4 | 6 | 2.500 | 9 | 2.333 |
| 199 | 2 | 4 | 10 | 2.400 | 18 | 2.167 |
| 200 | 2 | 2 | 9 | 2.333 | 9 | 2.333 |
| 201 | 2 | 10 | 10 | 4.000 | 18 | 3.833 |
| 202 | 2 | 7 | 10 | 3.100 | 18 | 2.667 |
| 203 | 5 | 10 | 18 | 3.333 | 19 | 3.053 |
| 204 | 5 | 7 | 18 | 3.333 | 19 | 3.053 |
| 205 | 2 | 7 | 9 | 3.000 | 18 | 2.000 |
| 206 | 2 | 7 | 10 | 3.100 | 18 | 2.167 |
| 207 | 2 | 10 | 10 | 4.000 | 12 | 3.000 |
| 208 | 2 | 7 | 10 | 3.100 | 18 | 2.333 |
| 209 | 5 | 7 | 19 | 3.053 | 18 | 3.667 |
| 210 | 2 | 7 | 10 | 3.300 | 18 | 2.167 |
| 211 | 5 | 10 | 19 | 3.368 | 18 | 3.333 |
| 212 | 2 | 7 | 9 | 3.333 | 19 | 3.526 |
| 213 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |
| 214 | 2 | 1 | 7 | 2.714 | 18 | 2.167 |
| 215 | 2 | 10 | 10 | 3.400 | 18 | 3.167 |
| 216 | 5 | 10 | 19 | 3.526 | 18 | 3.667 |
| 217 | 2 | 2 | 10 | 2.200 | 18 | 2.167 |
| 218 | 2 | 10 | 9 | 4.000 | 19 | 3.053 |

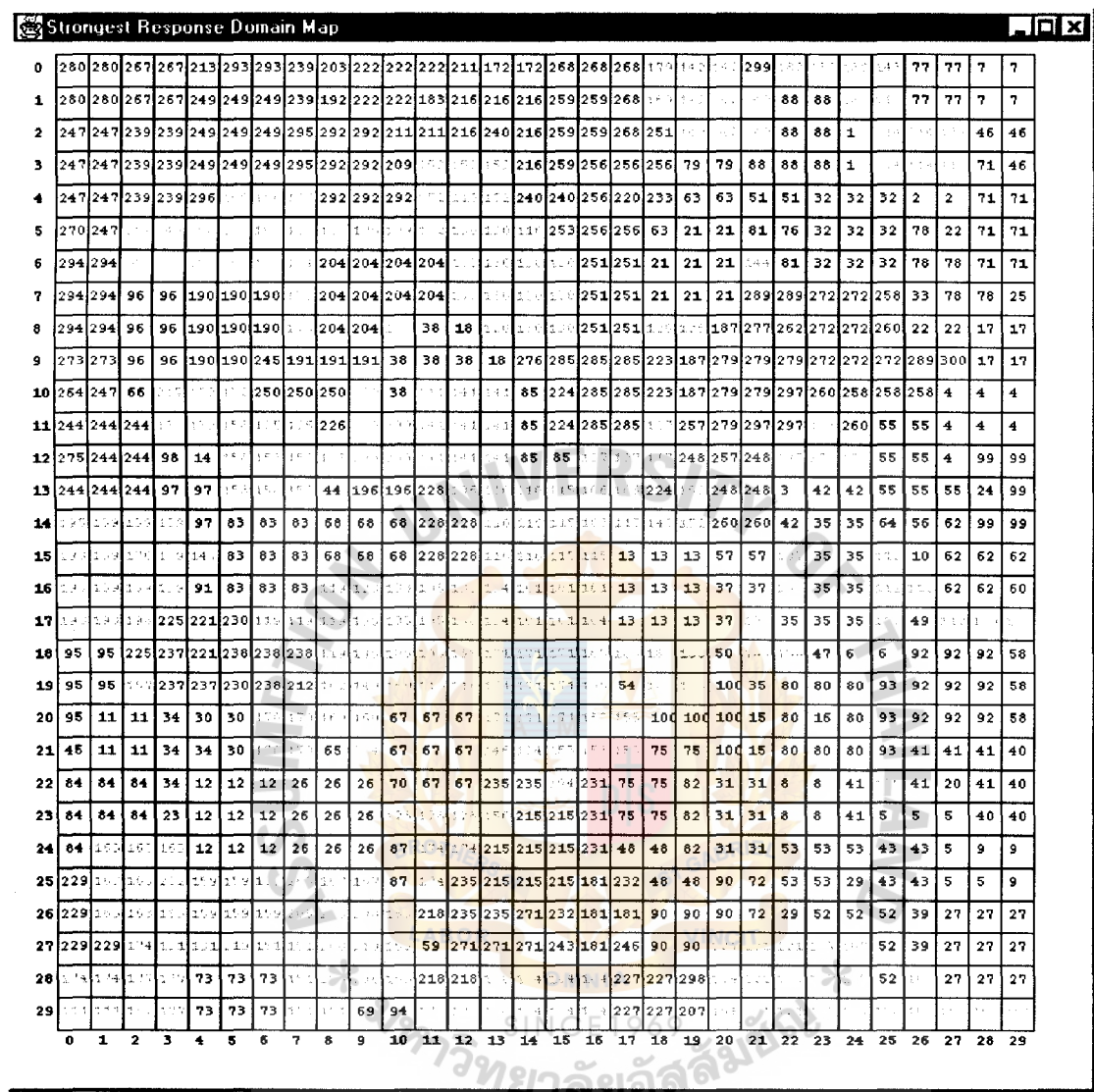
| Data | E | M | C1 | G1 | C2 | G2 |
|------|---|----|----|-------|----|-------|
| 219 | 2 | 7 | 9 | 3.333 | 19 | 2.421 |
| 220 | 5 | 10 | 18 | 3.500 | 18 | 3.833 |
| 221 | 2 | 7 | 10 | 3.100 | 18 | 3.833 |
| 222 | 5 | 10 | 19 | 3.684 | 18 | 3.000 |
| 223 | 8 | 4 | 19 | 3.053 | 15 | 3.000 |
| 224 | 8 | 4 | 19 | 3.368 | 18 | 3.333 |
| 225 | 2 | 7 | 10 | 3.100 | 15 | 3.200 |
| 226 | 1 | 2 | 7 | 1.857 | 9 | 3.000 |
| 227 | 1 | 10 | 7 | 3.571 | 9 | 3.000 |
| 228 | 2 | 4 | 10 | 3.000 | 18 | 3.000 |
| 229 | 2 | 7 | 10 | 3.400 | 18 | 3.167 |
| 230 | 2 | 7 | 10 | 3.400 | 18 | 3.667 |
| 231 | 2 | 10 | 16 | 4.000 | 18 | 3.833 |
| 232 | 2 | 10 | 10 | 4.000 | 18 | 3.667 |
| 233 | 5 | 10 | 18 | 3.667 | 18 | 3.667 |
| 234 | 5 | 10 | 19 | 3.526 | 18 | 3.667 |
| 235 | 2 | 10 | 10 | 3.700 | 21 | 3.714 |
| 236 | 2 | 10 | 10 | 4.000 | 18 | 3.833 |
| 237 | 2 | 7 | 10 | 3.400 | 18 | 3.833 |
| 238 | 2 | 7 | 10 | 3.600 | 18 | 3.667 |
| 239 | 8 | 10 | 19 | 3.684 | 18 | 3.667 |
| 240 | 5 | 10 | 19 | 3.368 | 18 | 3.667 |
| 241 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |
| 242 | 8 | 10 | 19 | 3.684 | 18 | 3.667 |
| 243 | 2 | 10 | 10 | 3.900 | 18 | 3.833 |
| 244 | 8 | 10 | 18 | 3.667 | 18 | 3.667 |
| 245 | 2 | 4 | 10 | 2.800 | 15 | 3.600 |
| 246 | 2 | 10 | 9 | 4.000 | 15 | 3.600 |
| 247 | 8 | 10 | 19 | 3.842 | 21 | 3.857 |
| 248 | 8 | 7 | 19 | 3.684 | 18 | 3.667 |
| 249 | 8 | 10 | 19 | 3.842 | 18 | 3.667 |
| 250 | 1 | 4 | 9 | 2.667 | 9 | 3.667 |
| 251 | 5 | 10 | 18 | 3.667 | 19 | 3.684 |
| 252 | 8 | 10 | 19 | 3.842 | 18 | 3.667 |
| 253 | 5 | 10 | 18 | 3.667 | 19 | 3.789 |
| 254 | 5 | 10 | 19 | 3.684 | 18 | 3.667 |
| 255 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |
| 256 | 5 | 10 | 18 | 3.667 | 18 | 3.833 |
| 257 | 8 | 7 | 19 | 3.526 | 18 | 3.667 |
| 258 | 8 | 7 | 18 | 3.500 | 19 | 3.526 |
| 259 | 5 | 10 | 19 | 3.684 | 18 | 3.833 |
| 260 | 8 | 7 | 18 | 3.667 | 19 | 3.684 |
| 261 | 8 | 10 | 19 | 3.842 | 18 | 3.667 |
| 262 | 8 | 7 | 18 | 3.167 | 18 | 3.667 |
| 263 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |
| 264 | 8 | 10 | 18 | 3.833 | 18 | 3.833 |
| 265 | 8 | 7 | 19 | 3.526 | 18 | 3.667 |
| 266 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |

| Data | E | M | C1 | G1 | C2 | G2 |
|------|---|----|----|-------|----|-------|
| 267 | 8 | 10 | 19 | 3.526 | 18 | 3.833 |
| 268 | 5 | 10 | 18 | 3.667 | 15 | 3.800 |
| 269 | 8 | 10 | 19 | 3.842 | 18 | 3.667 |
| 270 | 8 | 10 | 18 | 3.833 | 19 | 3.684 |
| 271 | 2 | 10 | 9 | 4.000 | 19 | 3.526 |
| 272 | 8 | 7 | 18 | 3.500 | 19 | 3.842 |
| 273 | 8 | 10 | 18 | 3.833 | 19 | 3.842 |
| 274 | 8 | 10 | 18 | 3.833 | 18 | 3.833 |
| 275 | 8 | 10 | 18 | 3.833 | 18 | 3.667 |
| 276 | 8 | 4 | 18 | 3.333 | 18 | 3.667 |
| 277 | 8 | 7 | 18 | 3.667 | 18 | 3.667 |
| 278 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |
| 279 | 8 | 7 | 19 | 3.368 | 18 | 3.833 |
| 280 | 8 | 10 | 19 | 3.368 | 18 | 3.833 |
| 281 | 2 | 10 | 10 | 4.000 | 18 | 3.833 |
| 282 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |
| 283 | 5 | 10 | 19 | 3.684 | 18 | 3.833 |
| 284 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |

| Data | E | M | C1 | G1 | C2 | G2 |
|------|----|----|----|-------|----|-------|
| 285 | 8 | 4 | 18 | 3.000 | 16 | 3.625 |
| 286 | 8 | 10 | 18 | 3.833 | 19 | 3.842 |
| 287 | 2 | 10 | 10 | 4.000 | 18 | 3.667 |
| 288 | 8 | 10 | 19 | 3.842 | 18 | 3.833 |
| 289 | 8 | 7 | 18 | 3.333 | 19 | 3.526 |
| 290 | 8 | 10 | 18 | 3.833 | 19 | 3.842 |
| 291 | 5 | 10 | 19 | 3.368 | 18 | 3.667 |
| 292 | 5 | 7 | 19 | 3.526 | 18 | 3.667 |
| 293 | 8 | 10 | 19 | 3.789 | 18 | 3.833 |
| 294 | 8 | 10 | 18 | 3.667 | 19 | 3.789 |
| 295 | 5 | 7 | 18 | 3.167 | 18 | 3.667 |
| 296 | 8 | 10 | 22 | 3.591 | 21 | 3.714 |
| 297 | 8 | 7 | 19 | 3.526 | 18 | 3.833 |
| 298 | 5 | 10 | 12 | 3.000 | 15 | 3.600 |
| 299 | 11 | 1 | 16 | 4.000 | 18 | 3.667 |
| 300 | 8 | 7 | 18 | 3.167 | 19 | 3.526 |

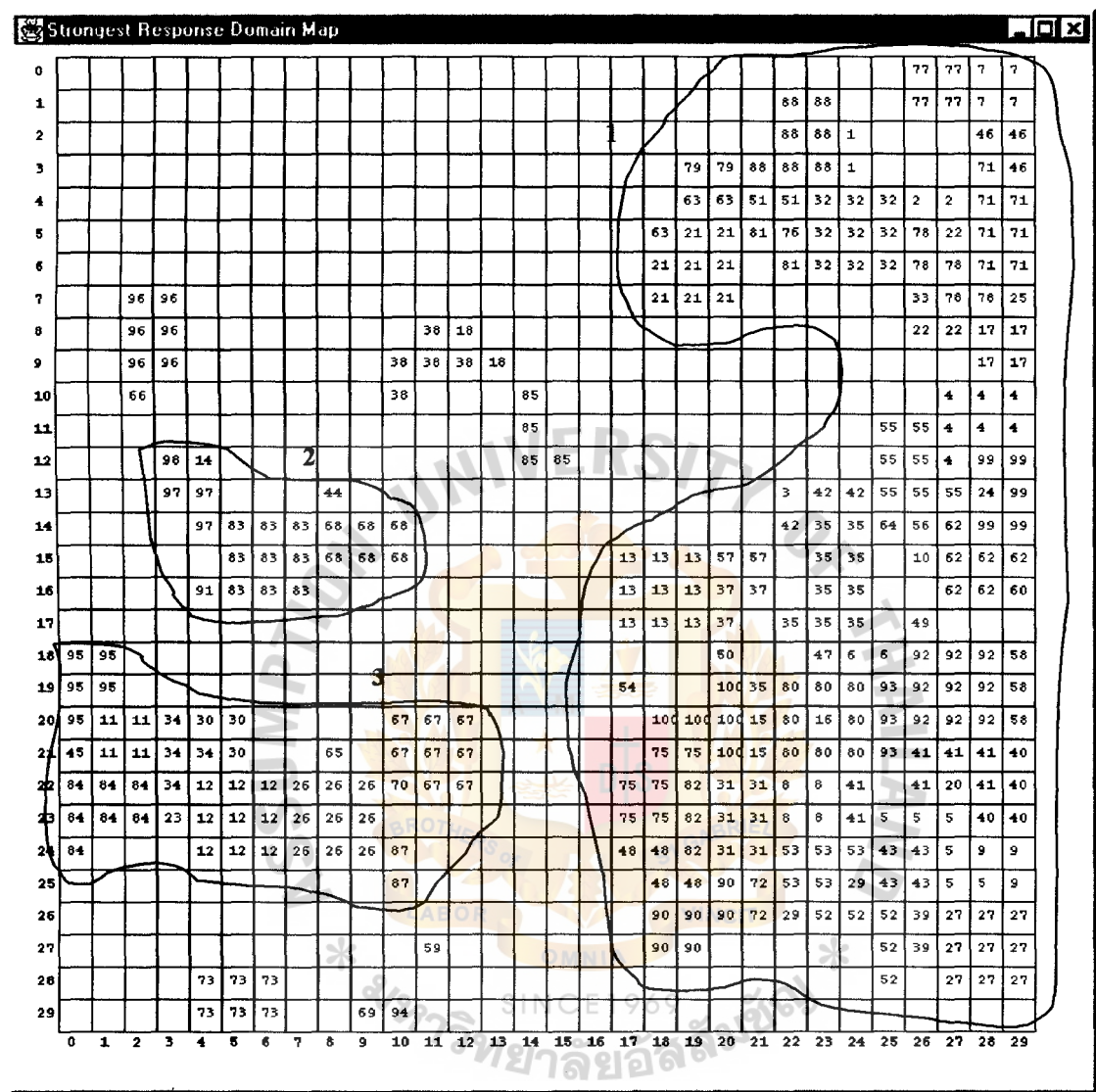
Note that: E, M, C1, C2, G1, G2 stand for English, Mathematics, number of credit, and GPA of the first and second semester.

C.2 CLUSTERS VERSUS THE STRONGEST RESPONSE OF ALL INPUT



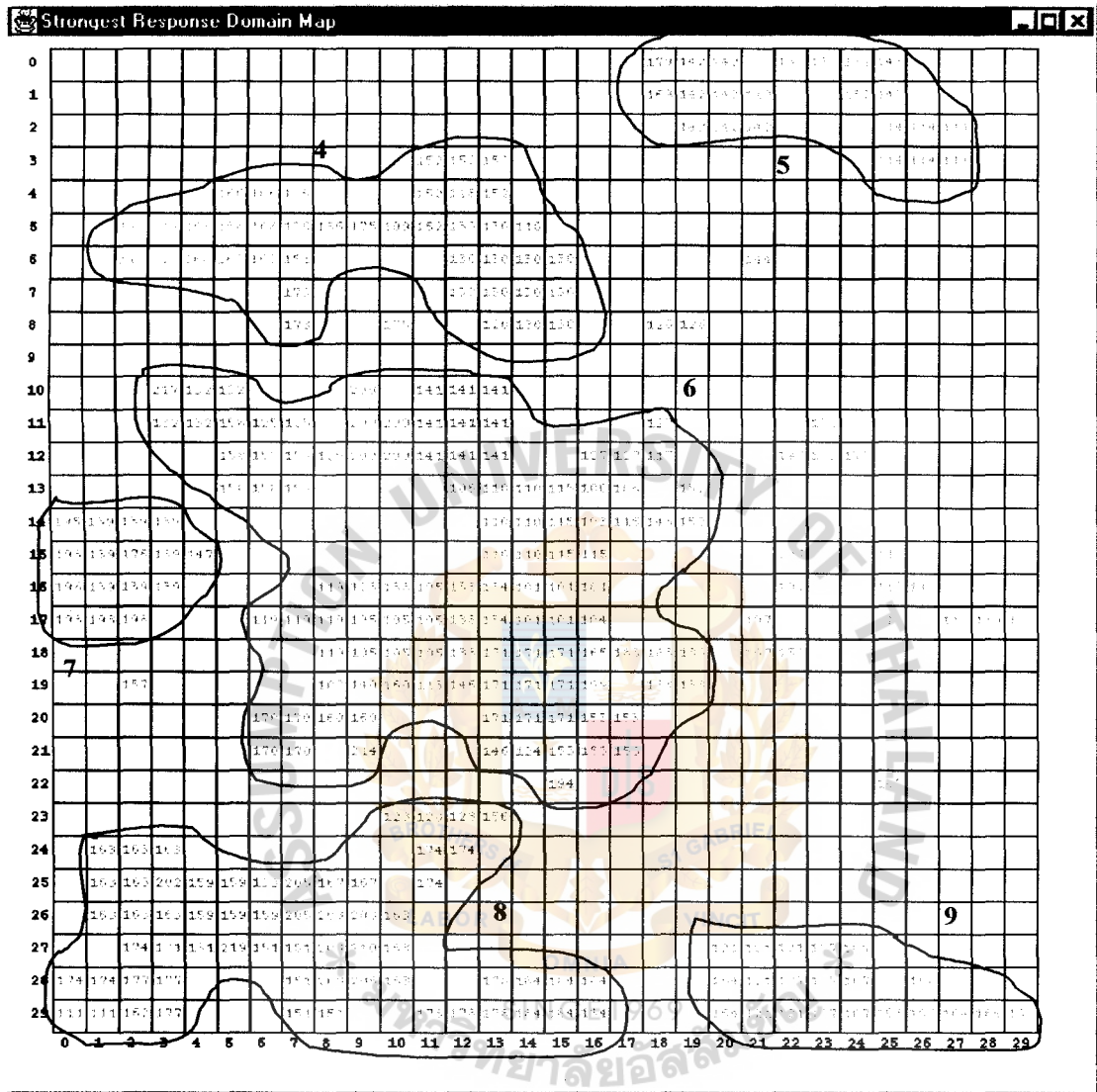
C.2.1 CLUSTERS VERSUS THE STRONGEST RESPONSE OF ALL INPUT

WITH $G2 = [0.0, 2.0)$



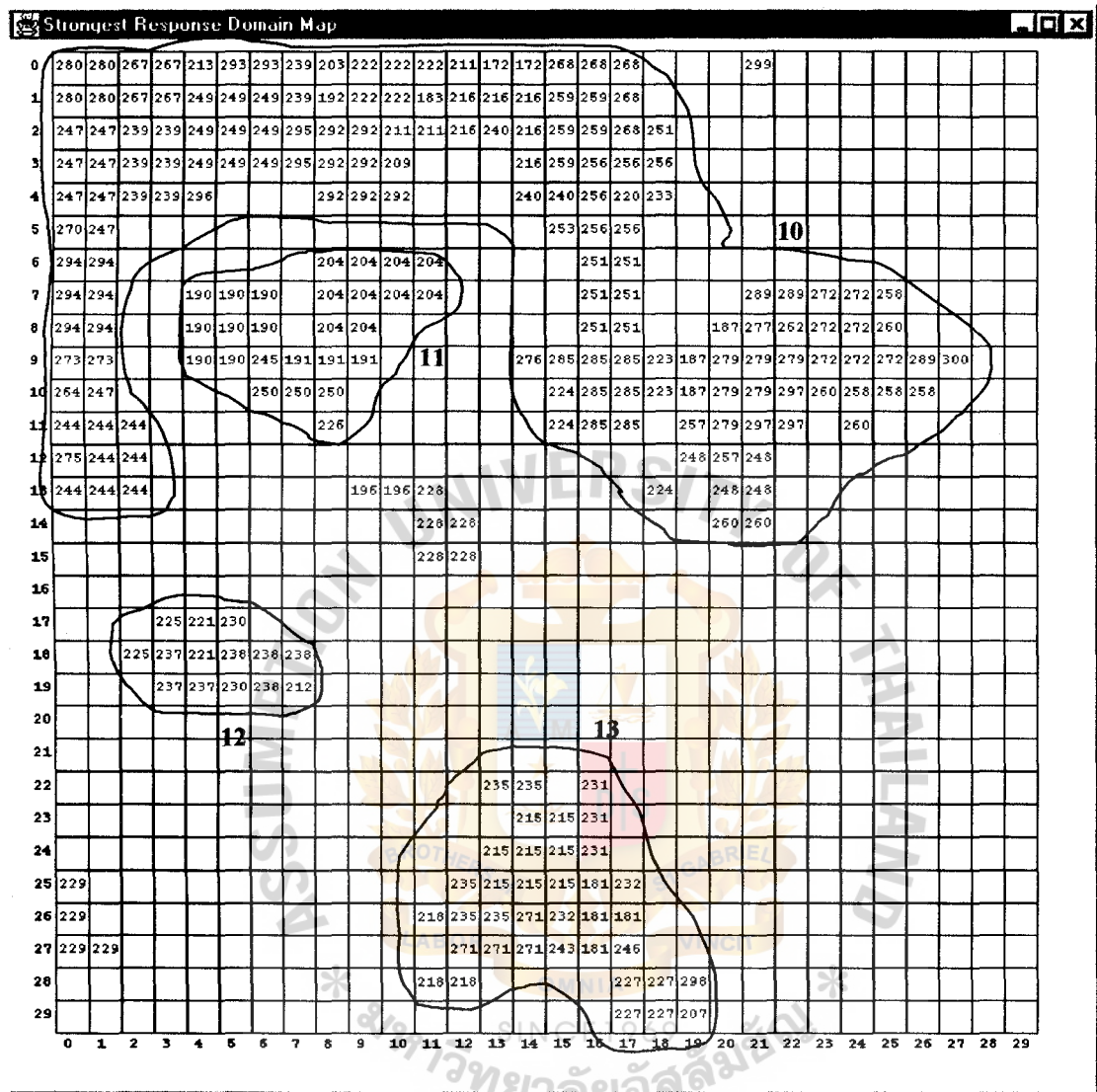
C.2.2 CLUSTERS VERSUS THE STRONGEST RESPONSE OF ALL INPUT

WITH G2 = [2.0, 3.0)



C.2.3 CLUSTERS VERSUS THE STRONGEST RESPONSE OF ALL INPUT

WITH $G2 = [3.0, 4.0]$



C.3 THE ACTIVATED NEURON OF EACH INPUT FOR TRAINING DATA

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 280 | | | 288 | | 293 | | 203 | | 222 | | 211 | 292 | | 268 | | | 179 | | | 299 | | 182 | | 143 | | 77 | | 28 |
| 1 | | | 267 | | | | | 192 | | | | 183 | 216 | | | | | 169 | | 142 | | | | | | | | | |
| 2 | | | | | | 281 | | 295 | | | | | 240 | | 289 | | | | | | | 88 | | | | 114 | | | |
| 3 | 247 | | 240 | | | | | | | 292 | 209 | | | | | | | 256 | | 79 | | | | 1 | | | | | 46 |
| 4 | | | | | 296 | 166 | | | | | | | 118 | | | | | 220 | 233 | 63 | | 51 | | | | | 2 | | |
| 5 | 270 | | | | | | | 136 | | 128 | 109 | 152 | | | | 253 | | | | | | | 76 | | 32 | | 22 | | 71 |
| 6 | | | | 149 | | 168 | | 154 | | | | | | | 198 | | | | | 21 | | 144 | 81 | | | | 78 | | |
| 7 | 294 | | | | | | | | | 204 | | | | | | 251 | | | | | | | | | | | 33 | | 25 |
| 8 | | | | 96 | | 190 | | | | | | | 120 | | | | | 126 | | | 277 | 262 | | 272 | 260 | | | | |
| 9 | 288 | | | | | | 245 | | | 191 | | 38 | | 18 | 276 | | | | | 187 | | | | | | | 289 | 300 | 17 |
| 10 | 294 | | 66 | 217 | | | | 250 | | | | | | | | 285 | | 223 | | | 279 | | | | 258 | | | | |
| 11 | | | | | 132 | | | | 226 | | 200 | | 141 | | 85 | | | | | | 297 | | 173 | | | | | 49 | |
| 12 | 275 | 244 | | 98 | 14 | | 158 | | 125 | | | | | | | | 127 | 117 | | 269 | | 197 | | | | 55 | | | |
| 13 | | | | | 97 | | | | 44 | | 196 | | 106 | | | | | 224 | | 248 | | 3 | 42 | | | | | 24 | 99 |
| 14 | 195 | | | | | | | | | | 228 | | | | 110 | | 103 | 116 | 128 | | | | | | 64 | 56 | | | |
| 15 | | 139 | 176 | | 147 | | 89 | | | 68 | | | | | | | | | | | 57 | | | | | 10 | | 62 | |
| 16 | 198 | | | | 91 | | | | | 138 | | | | | | 101 | | | 13 | | | | | 35 | | 112 | | | 61 |
| 17 | | | | 225 | | | | | 119 | | 166 | | 134 | | | | 104 | | | | 37 | | | | | 49 | | 189 | 180 |
| 18 | | | | | 221 | | | | | | | | | | | 165 | 188 | | 58 | | | 150 | 47 | 6 | | | | | |
| 19 | 95 | | 157 | | 237 | 230 | 238 | 212 | 102 | 140 | 160 | 113 | 146 | | 171 | | 199 | 54 | | | | | | | | | 92 | | 58 |
| 20 | | | | | 30 | | | | | | | | | | | | 153 | | | 100 | | | 80 | 16 | 93 | | | | |
| 21 | 45 | | 11 | | 34 | | 170 | | 65 | 236 | | 67 | | | 186 | | | | | | | 15 | | | | | 41 | | |
| 22 | | | | | | | | | | 70 | | | | | | 194 | | 75 | | | | | | 84 | | 165 | 20 | | 40 |
| 23 | | 84 | | 23 | | 12 | | | 26 | | | 128 | 156 | | | 231 | | | | 86 | | | | | | | | | |
| 24 | | | | | | | | | | 87 | | | | | | 215 | | | 48 | 82 | | | | | | 43 | | | |
| 25 | | | 163 | 202 | | 159 | 133 | | | 167 | | | 235 | | | | | | | | | 72 | | 53 | | | 5 | | 9 |
| 26 | 229 | | | | | | | 205 | | 208 | | 218 | | 271 | 282 | 286 | | | 90 | | | 29 | | | 52 | 39 | | | |
| 27 | | | | 131 | | 219 | | | | 288 | | 59 | | | | 243 | 246 | | | | | | | | | | | 27 | |
| 28 | | 174 | | | | | 161 | | | | | | | | | | | | 298 | | 121 | 199 | 107 | | | 108 | | | |
| 29 | | 111 | 162 | 177 | | 73 | | | | 69 | 94 | | 178 | | | 184 | 227 | | 207 | | | | | | | | | 164 | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |

C.4 THE ACTIVATED NEURON OF TESTING DATA

