K-MEANS CLUSTERING USING FEW CORRELATED ATTRIBUTES

THIPPRAPA POPIYATRAKUL

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF SCIENCE
(COMPUTER SCIENCE)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2010

Thesis
entitled

K-MEANS CLUSTERING USING FEW CORRELATED ATTRIBUTES

……
Miss Thipprapa Popiyatrakul
Candidate

……
Lect. Sudsanguan Ngamsuriyaroj,
Ph.D.
Major advisor

……..
Lect. Ananta Srisuphab,
Ph.D.
Co-advisor

…….
Lect. Songsri Tangsripairoj,
Ph.D.
Co-advisor

…….
Prof. Banchong Mahaisavariya,
M.D., Dip.Thai Board of Orthopedics
Dean
Faculty of Graduate Studies
Mahidol University

……
Lect. Sudsanguan Ngamsuriyaroj,
Ph.D.
Program Director
Master of Science Program
in Computer Science
Faculty of Information and
Communication Technology
Mahidol University

Thesis
entitled
K-MEANS CLUSTERING USING FEW CORRELATED ATTRIBUTES

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Science (Computer Science)
on
October 29, 2010

…….……
Miss Thipprapa Popiyatrakul
Candidate

……
Lect. Waraporn Jirapanthong,
Ph.D.
Chair

……
Lect. Sudsanguan Ngamsuriyaroj,
Ph.D.
Member

……..
Lect. Ananta Srisuphab,
Ph.D.
Member

…….
Lect. Songsri Tangsripairoj,
Ph.D.
Member

……
Prof. Banchong Mahaisavariya,
M.D., Dip.Thai Board of Orthopedics
Dean
Faculty of Graduate Studies
Mahidol University

……
Assoc. Prof. Jarernsri L. Mitrpanont,
Ph.D.
Dean
Faculty of Information and
Communication Technology
Mahidol University

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude and deep appreciation to my thesis advisor, Dr. Sudsanguan Ngamsuriyaroj, for her support by contributing her valuable time for guidance, invaluable advices and encouragement throughout this study in addition to spending a lot of time to study for the purpose of giving accurate and helpful advices which greatly sustain me to achieve this thesis.

I also would like to thank Dr. Songsri Tangsripairoj, and Dr. Ananta Srisuphab for serving as the committee of my thesis.

In addition, I would like to thank all of the Faculty of ICT's officers for their kindness and assistance, and for providing good facilities during my study at Mahidol University.

Lastly, I am very grateful to my family, especially my parents for financial support, encouragement and love. Without them, I may not be able to reach the goal of my study life, and I would not be able to come up to this point where pride and proud are found.

Thipprapa Popiyatrakul

K-MEANS CLUSTERING USING CORRELATED ATTRIBUTES

THIPPRAPA POPIYATRAKUL 4937921 ITCS/M

M.Sc.(COMPUTER SCIENCE)

THESIS ADVISORY COMMITTEE: SUDSANGUAN NGAMSURIYAROJ, Ph.D., SONGSRI TANGSRIPAIROJ, Ph.D., ANANTA SRISUPHAB, Ph.D.

ABSTRACT

K-Means clustering is one of the widely used knowledge discovery techniques. One disadvantage of K-means clustering used for a large data set is how to find an initial set of clustered data that are approximately close to the final set of clustered data so that it would not take a lot of time in clustering the final set when compared with the method that selects a data set in a random fashion.

This thesis proposed a new efficient way to do the K-means clustering when only a few correlated attributes are used. For each data set, the correlation among attributes is computed in order to determine which attributes are most related so that they could be the representatives of all attributes. Subsequently, the correlated attributes are selected for computing the clustering, and the results are compared with the clustering outputs obtained from using all attributes in the computation.

We evaluated our work using 10 datasets of UCI Machine Learning Repository including Iris, Ecoli, Yeat, Abalone, White Wine, Page Blocks, Magic Grammar Telescope, Breast Cancer, Waveform, and Letter Recognition. We used the tool called PML (Parallel Machine Learning) developed by IBM for performing the K-means clustering. Our results illustrate that the centroids of correlated attributes are close to the centroids of all attributes while having less computation time.

KEY WORDS: K-MEANS CLUSTERING / INITIAL CENTROID / CORRELATION

50 pages

การทำคลัสเตอร์ด้วยวิธี K-MEANS ที่ใช้ข้อมูลที่สัมพันธ์กัน

K-MEANS CLUSTERING USING CORRELATED ATTRIBUTES

ทิพย์ประภา โพธิ์ปิยตระกูล 4937921 ITCS/M

วท.ม. (วิทยาการคอมพิวเตอร์)

คณะกรรมการที่ปรึกษาวิทยานิพนธ์ : สุดสงวน งามสุริยโรจน์, Ph.D., ทรงศรี ตั้งศรีไพโรจน์, Ph.D., อนันต์ ศรีสุภาพ, Ph.D.

บทคัดย่อ

ในปัจจุบันการทำเหมืองข้อมูลได้รับความนิยมอย่างแพร่หลาย และอัลกอริทึ่มที่นิยมใช้คือ K-Mean clustering แต่อัลกอริทึ่มนี้มีข้อเสียหลายอย่างเช่น การหาจุดศูนย์กลางเริ่มต้นของ Cluster ซึ่งส่วนใหญ่นิยมใช้วิธีการสุ่มเพื่อหาจุดศูนย์กลางเริ่มต้นนั้น ทำให้เกิดปัญหาเช่น เมื่อเลือกโดยวิธีการสุ่มอาจได้จุดศูนย์กลางเริ่มต้นเป็น Outliner เมื่อทำการ Clustering อาจใช้เวลานานกว่าจะได้ผลลัพธ์ที่ต้องการ และปัญหาที่สำคัญอีกประการหนึ่งคือ การทำเหมืองข้อมูลนั้นจะใช้ข้อมูลที่มีขนาดใหญ่ซึ่งอาจเป็นข้อมูลที่สะสมมาเป็นเวลานานดังนั้นกว่าจะได้ผลลัพธ์ที่ต้องการจะทำให้เสียเวลามาก

ดังนั้นเพื่อเป็นการแก้ไขปัญหาข้างต้นที่กล่าวมางานวิจัยนี้จึงได้นำเสนอแนวคิดที่จะลดขนาดข้อมูลลงและยังสามารถหาจุดศูนย์กลางเริ่มต้นที่ใกล้เคียงกับจุดศูนย์กลางเริ่มต้นของผลลัพธ์อีกด้วย โดยใช้คุณลักษณะของข้อมูลที่มีความสัมพันธ์กัน ซึ่งวิธีการนี้จะเริ่มจากการหาค่าความสัมพันธ์กันระหว่าง Attribute ทุกคู่ เพื่อหากลุ่มของ Attribute ที่มีความสัมพันธ์ใกล้ชิดกันมากที่สุดเป็นตัวแทนของข้อมูลทั้งหมด ซึ่งใช้ในการ Clustering

จากวิธีการที่นำเสนอได้ทำการทดสอบกับชุดของข้อมูลทั้งหมด 10 ชุดข้อมูลจาก UCI Machine Learning Repository และใช้โปรแกรม PML (Parallel Machine Learning) ของบริษัท IBM ในการทำ Clustering ซึ่งทำการทดสอบโดยเปรียบเทียบผลลัพธ์ระหว่างการใช้ Attribute ทุกตัวของข้อมูลกับการใช้เพียง Attribute ที่มีความสัมพันธ์ใกล้ชิดกัน ผลลัพธ์ที่ได้นั้นจุดศูนย์กลางเริ่มต้นของการใช้ Attribute ที่มีความสัมพันธ์กันมีความใกล้เคียงกันกับการ Attribute ทั้งหมด ดังนั้นการใช้เพียง Attribute ที่มีความสัมพันธ์ใกล้ชิดกันสามารถหาจุดศูนย์กลางเริ่มต้นของ Cluster ได้

50 หน้า

CONTENTS

CONTENTS(cont.)

LIST OF TABLES

LIST OF TABLES (cont.)

LIST OF FIGURES

CHAPTER I
INTRODUCTION

A number of researchers have currently paid high attention to data mining so as to use it for analyzing business requirements and developing potential solutions. Most business applications involve commercial sales planning, product campaign to customer, product arrangement in the shop, and assistance in executives' decision-making. Consequently, many people and organizations are interested in applying data mining to help improve their efficiency and productivity.

In addition, most organizations have collected their data for several years. Such data normally contains some interesting patterns. As a result, data mining applications in any agency or organization increasingly become popular as a technique to find interesting patterns from the data compiled for a period of time. The process will give the utmost benefits in the decision making of an organization on various issues including self-improvement for better responses to different problems.

Due to the fact that data mining usually processes on the enormous amount of data collected for many years, an interesting data pattern may not be able to obtain in time for decision making on vital issues which require a fast and effective technique in order to compete with other competitors. Another problem found is the limit of computer memory size which cannot support a large amount of data. Thus, some organizations have to spend a great deal of money to acquire highly efficient computers for use in data mining processing. With limited budget, we are motivated to find an efficient and accurate way to do data mining as well as apply it to all organizations.

Data mining algorithms widely used are association rule mining, classification and clustering techniques. Each algorithm has different performance. Association rule mining is used to find the relationship of data for analyzing or predicting various situations. One example of association rule mining techniques is market basket analysis which analyzes customers' selections to purchase goods and

make rules for each product's relationship. For classification techniques, rules are created based on the existing data and used for predicting the trend of data that have not happened yet. In this regard, the data is divided into two parts. The first part is a training set which is used to generate a classification model, whereas the second one is a testing set which is used to test the classification model for finding out whether the model is accurate or not. One of the popular classification algorithms is the decision tree. For clustering techniques, data of similar characteristics are grouped into the same cluster. There are two types of clustering algorithms: hierarchical clustering such as agglomerative clustering, and partition clustering such as k-means clustering in which data are divided into predefined clusters and they are not overlapped. However, some disadvantages of k-means clustering are the undefined number of clusters, and the sampling of an initial centroid. If a poor starting point or an outlier is chose, the processing time could take longer. Thus, if the selected initial centroid is closed to the final cluster, the time spent on grouping all clusters would be reduced.

In addition, high performance computing can be applied to help the processing of data mining since parallel processing uses a number of computers to work on a small piece of task in parallel, and the large amount of data divided and stored in different computers can also solve the problem of limited memory size.

1.1 Motivation

Data mining is to mine interesting data having the considerable amount, and most of them are historical data that have been kept for several years. If these data are brought up to be analyzed, many interesting patterns would be found.

The k-means clustering has been a widely used data mining technique. However, one of the problems found in k-means clustering is to set an initial centroid, and the accuracy of clustering. Moreover, parallel processing approaches cannot be directly applied to the k-means clustering. Thus, we aim to propose a technique that can help improve the efficiency of setting an initial centroid and the accuracy of clustering outputs.

1.2 Objectives of the Thesis

This thesis has the following objectives.

1. To find a new efficient approach to help improve the k-means clustering while maintaining the same level of precision and accuracy.
2. To improve the performance of the k-means clustering.
3. To propose a new k-means clustering that uses only the attributes that are most closely related.
4. To perform the clustering of data when only a partial set of attributes are used.

1.3 Scope of the Thesis

This thesis has the following scope.

1. Reducing the number of attributes in order to decrease the amount of data to be processed by the k-means clustering.
2. Evaluating the accuracy of the proposed approach by comparing the centroid of all clusters when only related attributes are used and when all the attributes are used.
3. The data used in testing are obtained from the UCI machine learning repository.

1.4 Organization of the Thesis

This thesis consists of six chapters listed below.

- **Chapter I: Introduction** presents the statement of the problem as well as the objectives and the scope of this thesis.
- **Chapter II: Background** explains the basic knowledge related to this research which includes data mining concepts, the PML (Parallel Machine Learning) tool from IBM, and the UCI Machine Learning Repository.
- **Chapter III: Related Work** describes the related work to this research.

- **Chapter IV: Proposed Work** proposes the new approach of the k-means clustering
- **Chapter V: Experimental Results** present the experiments and describes the results of the experiments conducted.
- **Chapter VI: Conclusions** make a conclusion and gives the suggestions for future work.

CHAPTER II
BACKGROUND

2.1 Data Mining Concepts

Data mining is to search for patterns of relationships among the large amount of data. It helps us understand the characteristics of data and identify factors that affect some characteristics which give the prediction of the new trend in the near future. Typically, data mining consists of the following processes as shown in Figure 2.1 below.

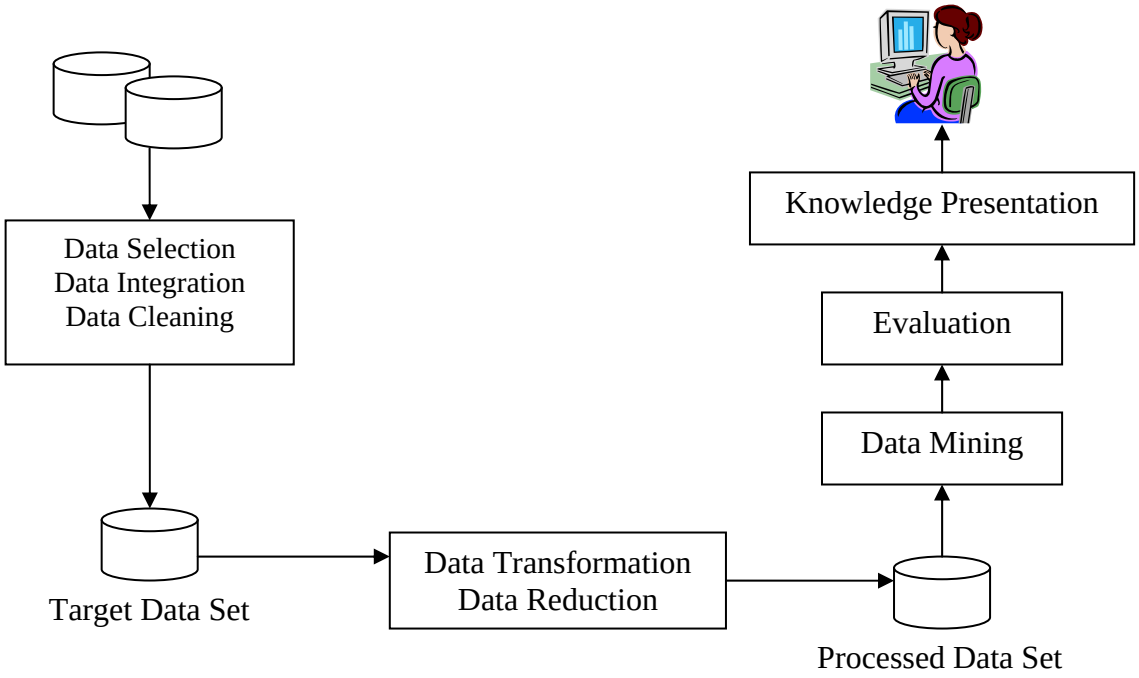

Figure 2.1: Stage of Knowledge Discovery in Database

Figure 2.1 shows the stages of knowledge discovery which include data preprocessing, data mining, analysis and evaluation, and the application of the outputs. Each process can be described in detail as follows.

Data preprocessing stage consists of the following processes.

- **Data selection** is to select the necessary and appropriate data for further analysis.
- **Data integration** is to integrate data from various database sources, and transformed them into the same format.

Multiple data sources → Unified Structure

- **Data cleaning** is to clean dirty data such as repetitious or incomplete data since parts of data are missing or inconsistent. These data must be cleaned before being the input to any data mining module; otherwise, the outcome may contain some errors or inaccuracy.
- **Data transformation** is to transform the data format into an appropriate one so that it can be used in a data mining algorithm such as normalization.
- **Data reduction** is to reduce the number of attributes and records since most data have the tremendous size. The reduced data set should be the representative of the original data.

There are different data mining techniques, and each technique uses different data format for finding interesting patterns. Most widely used techniques include association rules, classification, and clustering. Each technique is briefly described as follows.

2.1.1 Association Rules

This mining technique can be practically applied to many types of data. The main idea of this technique is to find the patterns of data relationship from the large amount of data to use for analysis or for prediction of situations. The association rules algorithm has been derived from the behavior of customers when buying goods, and it is called “Market basket analysis”. The objective is to find interesting relations in a set of items from the transactions of the database. The process of deriving association rules has two main steps described below.

- Finding frequent item sets that have frequency values higher than or equal to the minimum support.

- Deriving an association rule from frequent item sets obtained from the previous step. The association rule is accepted when it has confidence values higher than or equal to the minimum confidence.

Most researches on association rules propose efficient techniques for finding the relationship of each item set. In addition, they focus on the first step because it typically takes a long time to compute and lots of I/O operations are preformed. Several issues have been discussed including the followings.

- To design the appropriate data structure that is efficient in terms of the speed of deriving association rules and the minimized memory space required.
- To design a counting technique that gives better efficiency in finding important data.
- To reduce the database size so that the data can be stored in the memory for fast and efficient reading as well as for decreasing the number of data reading from disk.

One example of patterns found using association rule mining is the situation when a customer purchases one product item, he or she always buys another product item at the same time. This pattern may help the shop owner to arrange these product items next to each other in the shop. For instance, the following relation shows that when the milk is bought, there is 100% chance that the bread is bought too. In addition, there is 80% chance that both milk and bread are bought simultaneously.

buys(x, Milk) → buys(x, Bread) [80%, 100%]

2.1.2 Classification

This technique creates the model of data management for predicting or categorizing data into the predefined group, so-called Supervised Learning. It generates rules based on the existing data and uses them to predict the trend of data which have not happened yet. The existing data is divided into two parts listed below

- **Training set** is the data used to generate the model of data for the system to learn.
- **Testing set** is the data used to verify the model created from the training set whether the model is correct or not. If not, the model will be modified until

it is correct. Subsequently, the revised model is used to classify the unclassified data in order to predict the new coming data.

Figure 2.2 shows the processes of classification techniques that can be represented by various models such as the If-Then rules, the decision tree, the neural network, and others. The decision tree technique is widely used for classification.

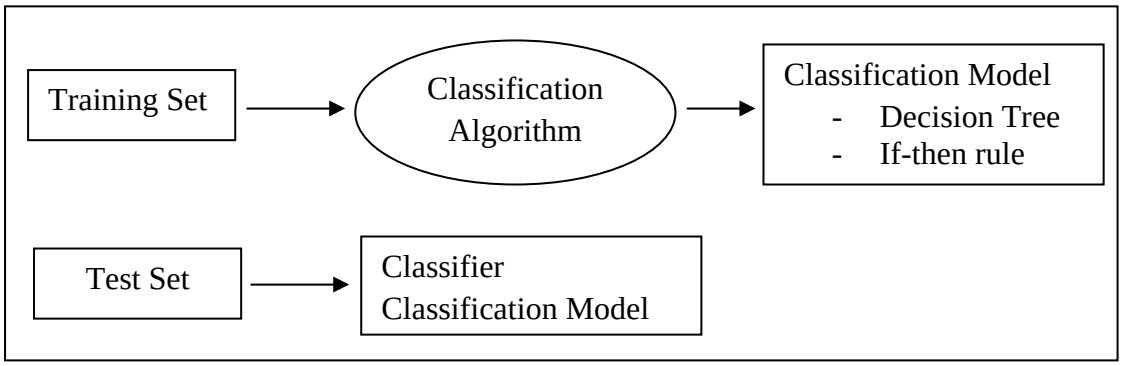

Figure 2.2: Processes of Classification Techniques

Table 2.1 illustrates an example of the stock quality data which includes the growth of income, the ability to control budget, the fluctuation of incomes and returns (vary), the capability of the executive board, and the company's financial status (finance). All of these are used to indicate whether the stock has the quality worth to invest or not.

Table 2.1: Training Set of Stock Quality Data

Stock	Income	Budget	Vary	Board	Finance	Quality
A1	Good	good	no	good	good	S
B4	Good	bad	no	bad	good	B
C3	Bad	good	yes	good	good	C
C54	Bad	bad	no	bad	bad	F

Figure 2.3 shows an example of a decision tree that illustrates how the analyzed stocks are evaluated to fit in which groups and how they are correlated with the five groups of stocks listed below.

- Stocks of poor quality (Grade-F stocks). The company has its business run at a loss, with the large amount of debt or uncertain returns, or the profit is on and

off, or the executives of the company cannot be trusted. These stocks do not need to be evaluated since they should not be bought.

- Stocks of intermediate quality (Grade-C stocks). The company has the certain amount of debts while the incomes and the returns rarely increase, or slowly grow for not more than 5% per year.
- Stocks of fairly good quality (Grade-B stocks). The company does not have lots of debts while the incomes steadily grow with future returns have the growth rate of 5-15% per year.
- Stocks of good quality (Grade-A stocks). The company has only a few debts or no debts. The incomes constantly increase and the future returns are expected to rise at 15% or over per year.
- Stocks of excellent quality (Super stocks). The company has only a few or no debts. The incomes are steady and constantly grow. The returns increase in the range of 20-30% per year. The executives are proficient, diligent and honest. The business has good promising and high bargaining power on suppliers. The business is in an industry that is not competitive mainly on prices, and customers can bear some burdens.

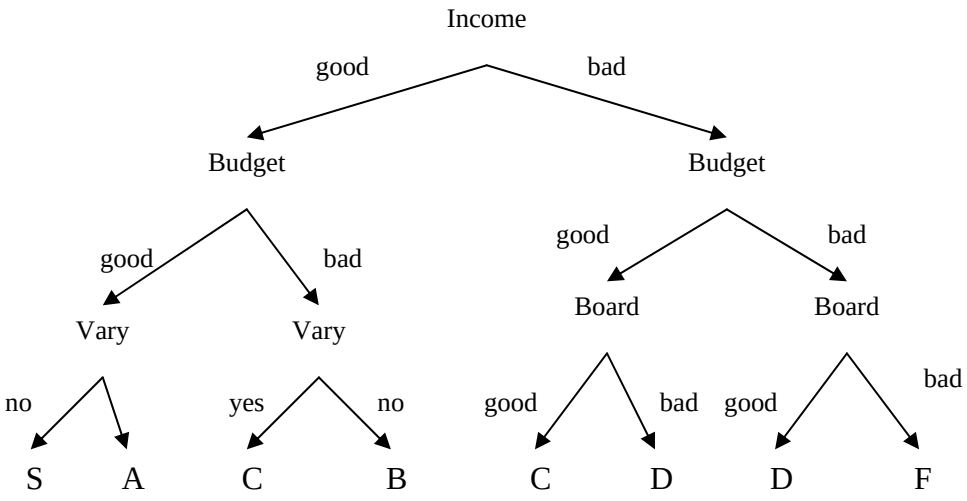

Figure 2.3: Sample Decision Tree for Stock Quality Data

2.1.3 Clustering

Clustering is a technique to group the data of similar characteristics into the same cluster. Thus, data in the same cluster usually should have the same or similar characteristics. Moreover, two clusters should have different characteristics and it would be easy for data analysis. For example, grouping of different symptoms may indicate different diseases.

For data clustering, the dissimilarity value of each data item is measured, and data with little differences in the dissimilarity values will be put into the same cluster. On the contrary, data in different clusters will have high differences in the dissimilarity value. Clustering can be applied in many research areas including visualization, pattern recognition, computer graphics, and intelligent systems.

The calculation of dissimilarity values d(i,j) can be done in various ways as listed below.

- Euclidean distance = $\sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + ... + (x_{in} - x_{jm})^2}$
- Manhattan distance = $|x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + ... + |x_{in} - x_{jn}|$
- Minkowski distance = $\sqrt[p]{|x_{i1} - x_{j1}|^p + |x_{i2} - x_{j2}|^p + ... + |x_{in} - x_{jn}|^p}$

Table 2.2 shows a set of sample data used for computing the dissimilarity values and the followings are the sample calculation.

Table 2.2: Sample Data for Dissimilarity Value Calculation

	w	x	y	z
i	22	1	42	10
j	20	0	36	8

The dissimilarity values are calculated based on the Euclidean distance below.

d(i,j) = $\sqrt{(22-20)^2 + (1-0)^2 + (42-36)^2 + (10-8)^2}$ = $\sqrt{45}$ =6.71

The dissimilarity values are calculated based on the Manhattan distance below.

d(i,j) = $|22-20| + |1-0| + |42-36| + |10-8|$ = 2+1+6+2 =11.0

The dissimilarity values are calculated based on the Minkowski distance for p=3 below.

$$d(i,j) = \sqrt[3]{(22-20)^3 + (1-0)^3 + (42-36)^3 + (10-8)^3} = \sqrt[3]{233} = 6.15$$

Clustering has two types as described below.

- **Hierarchical clustering**

A hierarchical tree or dendrogram is generated and data is separated into sub-trees as illustrated in Figure 2.4 below.

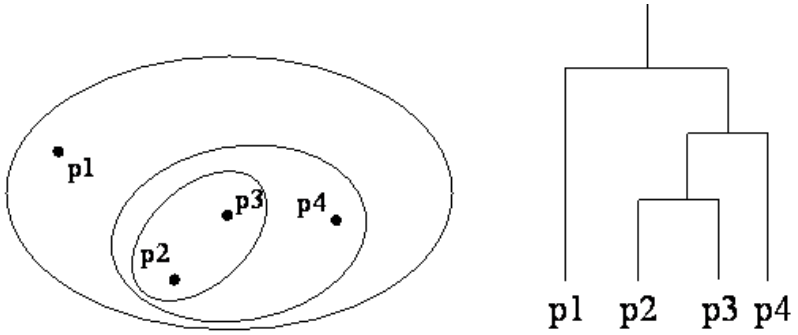

Figure 2.4: Hierarchical Clustering vs. Dendrogram

- **Partitional clustering**

Data are grouped into the predefined clusters without overlapping as illustrated in Figure 2.5, while data lying outside a cluster is called outlier. Well-known clustering algorithms include k-means, k-medoids, k-harmonic means, and estimation maximization (EM).

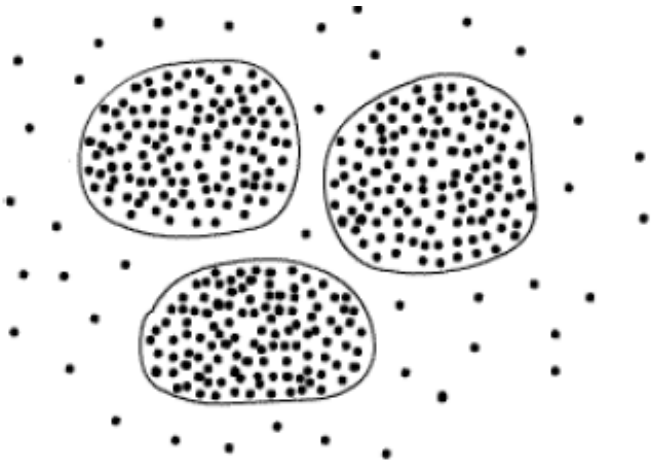

Figure 2.5: Partitional Clustering

2.1.4 K-means Clustering

K-means clustering is the technique that groups similar data into the same cluster. The algorithm works as shown in Figure 2.6.

Input: K: The number of clusters required; D: All data
Output: A group of clusters
Method:
(1) Select an initial centroid by sampling some data from D as the beginning set of data to perform clustering.
(2) Repeat
  a. Calculate the distance between a point of data and each centroid, and select the data that is closest to one of the centroid before putting it into the same cluster.
  b. Calculate the mean of all data inside a cluster.
(3) Repeat until the cluster is not changed.

Figure 2.6: K-means Clustering Algorithm

K-means clustering is one of widely used clustering techniques since it is easy to understand and develop, uses linear time in computation, and has relatively inexpensive cost. However, it has some disadvantages described below.

- Selection of an initial centroid has a significant impact on the clusterin calculation since it may need to calculate so many times before the desired results are obtained.
- The number of clusters has to be defined before calculation and it may not be appropriate for a specific set of data. Thus, the calculation of the number of K is unpredictable.
- Outliers are another factor affecting the results since they may cause the mean of clusters to be inaccurate. In addition, if the initial centroid is chosen as an outlier, the calculation will do so many rounds.
- The calculation will stop when there is no change in members inside a cluster, or no change in the centroid. However, there is sometimes an endless change of clusters.

2.1.5 Correlation Analysis

The correlation coefficient is a statistical value that tells how close that two attributes are related. It can be computed based on Pearson's Product moment coefficient using the equation below.

$$r_{A,B} = \frac{\sum (A - \overline{A})(B - \overline{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum (AB) - n(\overline{A}\overline{B})}{(n-1)\sigma_A \sigma_B}$$

where

n is the number of tuple.

$\overline{A}$ is the mean of attribute A

$\overline{B}$ is the mean of attribute B

σ_A is the standard deviation of attribute A

σ_B is the standard deviation of attribute B

Figure 2.7: Correlation Algorithm

The correlation coefficient is determined using the following table.

Table 2.3: The correlation coefficient is determined

$r_{A,B} > 0$	A and B are positively correlated
$r_{A,B} = 0$	A and B are independent
$r_{A,B} < 0$	A and B are negatively correlated

2.2 Parallel Machine Learning (PML)

Parallel machine learning is a program used to test the proposed technique. It was developed by IBM and provides tools for the execution of data mining and machine learning algorithm in multiple processor environments or on multiple threaded machines. The PML toolbox is comprised of two main component: an API for running the users' own machine learning algorithms and several pre-programmed algorithms that serve as both as examples and for purposes of comparison. The pre-programmed algorithm includes a parallel version of the Support Vector Machine

(SVM) classifier, linear regression, k-means, fuzzy k-means, kernel k-means, PCA, and kernel PCA.

2.3 UCI Machine Learning

UCI Machine Learning Repository is a collection of databases, domain theories and data generators used by the machine learning community for the empirical analysis of machine learning algorithms. The archive was created in 1987 by David Aha and his fellow graduate students at UC Irvine. Since that time, it has been widely used by students, educator and researchers all over the world as a primary source of machine learning data sets.

CHAPTER III
RELATED WORK

This chapter discusses some research work related to ours. Some works address the problem of setting initial centroids, or find out how to handle the clustering efficiently. The five related papers are described in summary below.

3.1 A New Algorithm to Get the Initial Centroid [1]

To enhance the k-means algorithm's efficiency, choosing good initial centroids is one way to reduce the calculation time because selecting initial centroids that are closed to the final cluster would help decrease the rounds of computation. In this paper, the k-means algorithm's efficiency is improved by calculating points of data to be included into initial centroids. The experimental results are compared with those finding initial centroids via the sampling technique. Specifically, the technique defines U as a set of data among n data points, and starts the calculation as described below.

1) Compute the distance between each point of all data inside U using the Euclidean distance technique.

2) Find two data points having the shortest distance between them, and put them into the data set A1, and delete these points from U.

3) Calculate the distance between each data point in U as well as in A1.

4) Find any data that is closest to A1, put it into A1, and delete it from U.

5) Repeat the step 4 until the number of data in A1 is equal to the threshold value as predefined by the user.

6) Repeat the step 2 to find another data set like A1 until we get k data sets.

This paper uses the data set from the UCI repository. They are the data sets of wine, iris, balance scale and car. The results are compared with those of calculation using sampling initial centroids. The outputs are shown in Table 3.1 below.

Table 3.1 illustrates our computation of initial centroids when compared with the standard k-means which uses the sampling technique. The experiments are run 10 times and the accuracy of clustering is averaged. Unfortunately, the average accuracy of the proposed work is not high as it is ranged from 60-80 %.

Table 3.1: Experimental Results of Calculating Initial Centroids [1]

data / algorithms		wine		iris		balance scale		car	
		Initial centroids	Accruracy	Initial centroids	Accuracy	Initial centroids	Accuracy	Initial centroids	Accuracy
k-means algorithm randomly select initial centroids	First	111,235,608	0.6416	13,50,58	0.666667	64,66, 116	0.651685	196,357,432	0.703057
	Second	109, 113, 363	0. 5968	11,50,122	0.666667	26,73, 111	0.421348	132,417,299	0.652838
	Third	106,281,324	0.4816	47,51, 77	0.653333	81,134,151	0.421348	106,276, 419	0.626638
	Fourth	107,134,483	0.6480	10,55,108	0.573333	75,81,159	0.421348	130, 255,288	0.703057
	Fifth	159,499,530	0.6592	51,52, 20	0.573333	90,116,139	0.702247	69,103,313	0.652838
	Sixth	123,457,573	0.6624	56,78,106	0.666667	43,75,87	0.421348	10,67, 423	0.703057
	Seventh	137,359,414	0.6112	53,99,106	0.573333	11,88,118	0.61236	161,304,424	0.652838
	Eighth	81,141, 149	0.6496	53,134,139	0.653333	4,86,155	0.421348	100, 178,314	0.703057
	Ninth	176,266,543	0.5216	7,58, 77	0.573333	80,86, 92	0.61236	202, 206,311	0.703057
	Tenth	174,155,608	0.7392	57,63,73	0.573333	86,118,151	0.61236	75,249,429	0.703057
	Mean value	---	0.62112	---	0.617333	---	0.529775	---	0.68035
Improved algorithm		---	0.685393	---	0.886667	---	0.6656	---	0.812227

In summary, the k-means algorithm is a widely used technique but having the problem of the centroid selection since it may make the process of clustering take the considerable time for calculation.

3.2 Cluster Center Initialization Algorithm for K-means Clustering [2]

In this paper, a technique for calculating the initial cluster centroids is proposed to help improve the calculation efficiency since the sampling of initial centroids typically make more rounds of calculation. Nevertheless, if initial centroids are closed to the final cluster, the rounds of computation will be reduced and the accuracy is enhanced.

Thus, the proper selection of initial centroids will result in better accurate clustering. In addition, if outliers are chosen as an initial centroid, the clustering will take longer time. In this paper, an algorithm to select a good initial centroid has been

proposed. The main idea is based on the fact that each attribute can be used in finding good initial cluster centroids. The details of the algorithm, called *CCIA Algorithm* (Cluster Center Initialization Algorithm), are given in Figure 3.1.

For the evaluation of the CCIA algorithm's efficiency, the UCI repository data are used for testing. They are Fossil, Wine recognition, Iris, Ruspini, and Letter image recognition. The results are shown in Table 3.2.

Input: D is the total amount of data consisting of A1, A2,…,Am where m is the total number of attributes.
K is the number of clusters.

Output: An initial point of clusters of the k-means algorithm.

Algorithm: It consists of two parts explained below.

(1) For the calculation of the K' cluster centers, each attribute is computed using the mean (μ_j), the standard deviation (σ_j), and the percentile (Z_s) in order to create the initial partition (X_s). Next, the distance between the initial partition and the attribute is calculated using the k-means algorithm for each attribute. The cluster label is inserted for each pattern and the pattern string is generated for each pattern of the attribute. Subsequently, the same pattern string is grouped into the same cluster.

(2) If K' is greater than K, the cluster will be merged so that the number of cluster centers is K using the Merge DBMSDC. The distance between each point of the cluster center of each K' is computed.

Figure 3.1: CCIA Algorithm (Cluster Center Initialization Algorithm)

Table 3.2: Proximity Comparison between CCIA and Random Sampling Algorithms

Table 1
Comparing *CCPI* of data sets

Data set	*CCPI*	
	CCIA	Random
Fossil data	0.0021	0.3537
Iris data	0.0396	0.8909
Wine data	0.1869	0.3557
Ruspini data	0.0361	1.2274
Letter image recognition data	0.0608	0.1572

The results of their experiments show that their initial centers are approximately closed to the required cluster centers than those obtained from the random sampling.

3.3 Refining Initial Points for K-means Clustering [3]

This paper presents a procedure for computing a refined starting condition from a given initial one by estimating the modes of the distribution. It demonstrates the application of this method to the popular K-means clustering algorithm, and shows that the refined initial starting points indeed lead to the improved solution. The solution is the parameterization of each cluster model, and it can be performed by determining the *modes*.

The refinement algorithm initially chooses J small random sub-samples of the data, S_i, $i = 1,\ldots,J$. The sub-samples are clustered using the K-means technique given that the empty clusters at termination will have their initial centers re-assigned and the sub-samples will be re-clustered. The sets CM_i, $i = 1,\ldots,J$ are the clustering solutions over the sub-samples which form the set CM. The set CM is clustered via K-means initialized with CM_i to produce the solution FM_i. The refined initial point is chosen as the FM_i having the minimal deviation over the set CM.

The refinement algorithm operates over the small sub-samples of a given database. By initializing a general clustering algorithm near the nodes, not only are the true clusters often found, but it also allows the clustering algorithm to iterate fewer

times prior to be converged. The refined initial points give the better results than the random initial starting points.

3.4 Enhancing K-Means Algorithm with Initial Cluster Center Derived from Data Partitioning along the Data Axis with the Highest Variance [4]

This paper presents the algorithm to compute the initial cluster centers for K-means clustering. The data is partitioned using a cutting plane that divides a cell into two smaller cells. The plane is perpendicular to the data axis with the highest variance and is designed to reduce the sum squared errors of the two cells.

For the extension, the algorithm can partition a set of data into k cells. In addition, the center of the cells can be used as the good initial cluster center for the K-means algorithm. The steps of the proposed algorithm are as follows:

1. Let cell *c* contain the entire data set.
2. Sort all data in the cell *c* in ascending order on each attribute value and link the data via a linked list for each attribute.
3. Compute the variance of each attribute of cell *c*. Choose an attribute axis with the highest variance as the principal axis for partitioning.
4. Compute the squared Euclidean distances between the adjacent data along the data axis with the highest variance $D_j = d(c_j, c_{j+1})^2$ and compute the sum $dsum_i = \sum_{j=1}^{i} D_j$
5. Compute the centroid distance of cell c: $centroidDist = \frac{\sum_{i=1}^{n} dsum_i}{n}$

where $dsum_i$ is the summation of distances between the adjacent data.

6. Divide cell *c* into two smaller cells. The partition boundary is the plane perpendicular to the principal axis and passes through a point *m* whose $dsum_i$ approximately equals to *centroidDist*. The shorted linked lists of cell *c* are scanned and divided into two for the two smaller cells accordingly.

7. Compute the Delta clustering error for *c* as the total clustering error before the partitioning, minus the total clustering error of its two sub cells, and insert the cell into an empty Max heap with the Delta clustering error as the key.

8. Delete a max cell from the Max heap and assign it as the current cell.

9. For each of the two subcells of *c*, which is not empty, perform the steps 3 to 7 on the subcell.

10. Repeat the steps 8 to 9 until the number of cells reaches *K*.

11. Use the centroids of cells in the Max heap as the initial cluster centers for the *K*-means clustering.

This method has the given data set partitioned into K clusters so that the sum of the total clustering errors for all clusters is minimized as much as possible whereas the inter-distances between clusters are maintained as large as possible. The proposed algorithm is very effective since it can quickly converge to the clustering results. The experiments of proposed algorithm perform better than the random initialization method, and can help decrease the running time of K-means clustering for the large data set. In addition, this method is simpler and easier to implement than CCIA [3].

3.5 K-means Clustering via Principal Component Analysis [5]

This paper presents that the principal components are the continuous solution of the cluster membership indicators in the K-means clustering method. The PCA dimension reduction performs the data clustering according to the K-means objectives, and provides the justification of the PCA-based data reduction.

The paper shows that the cluster centroid subspace can be spanned by the first K-1 principal direction such that the PCA dimension reduction finds the cluster centroid subspace. For experiments, they apply the K-means clustering in the PCA subspace. The data is reduced from the original 1000 dimensions to 40, 20, 10, 6, 5 dimensions, respectively. The results indicated as the clustering accuracy on 10 random samples of each newsgroup combination and size composition are averaged.

However, the differences between the centering data are not good for 10, 6, and 5 dimensions.

CHAPTER IV
PROPOSED WORK

This chapter discusses the detail of the proposed work, and how the performance of our K-means clustering can be evaluated.

4.1 Overview of the Proposed Model

This thesis proposes the model that finds the initial centroids of clusters using few correlated attributes. The overview of the proposed model is shown in Figure 4.1.

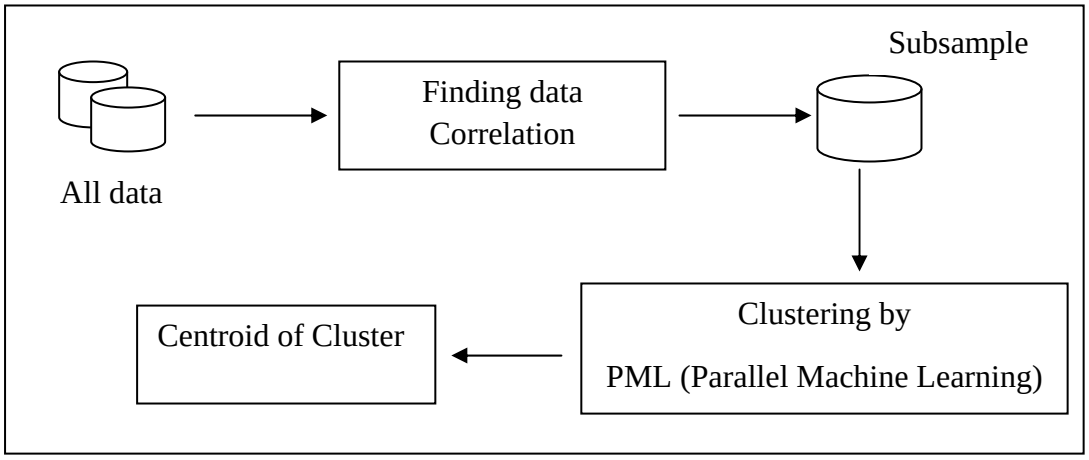

Figure 4.1: Overview of the Proposed Model

The proposed model is divided into two main steps. The first step is to find the correlation among all attributes of a data set so that the most correlated attributes will be selected to represent all the data. The second step is to compute the centroids of each cluster using the few selected attributes via the PML tool developed by IBM. The details of each step are described as follows.

4.2 Finding Data Correlation

Finding a set of the correlated data aims to improve the performance of K-means clustering since the amount of data to be computed has been reduced. The main issue is to find which attributes are closely related. Thus, the correlation values between each pair of the attributes are computed, and plotted on the graph. Later, those attribute pairs having the most similarity are used to classify clusters of similar characteristics into the same group. The process of computing data correlation is given as follows.

Input: Attributes of all data, A1, A2,…,A_m to be considered where m is the total number of attributes.

Process:

(1) Calculate the correlation value (r_{A_i,A_j}) between A_i and A_j using the following formula where i and j are numbers indicating each attribute.

$$r_{A_i,A_j} = \frac{\sum (A_i - \overline{A_i})(A_j - \overline{A_j})}{(n-1)\sigma_{A_i}\sigma_{A_j}} = \frac{\sum (A_i A_j) - n(\overline{A_i}\,\overline{A_j})}{(n-1)\sigma_{A_i}\sigma_{A_j}}$$

(2) Sort the correlation value (r_{A_i,A_j}).

(3) Select a number of attributes which has the highest correlation value (r_{A_i,A_j}). The number of chosen attributes depends on how many of them are related. They will become the representatives used in clustering in the first stage.

(4) Sort the attributes selected from the Step (3).

The diagram showing the process of computing the data correlation is given in Figure 4.2. From our initial experiments using some data sets, we found that the data correlation indicates two correlation types. First, strong correlation is shown among few attributes, and it would be easy to select those attributes for further clustering. An example is shown in Figure 4.3. Second, all attributes are closely correlated. Thus, it would be difficult to select a few attributes to represent all of them. An example shown in Figure 4.4 depicts a flat graph.

4.3 Clustering by Parallel Machine Learning Tool

Parallel Machine Learning Tool or PML is developed by IBM, and it is free to use. The PML software has been installed on a Rocks cluster named as mucluster.mahidol, and it is used to do clustering on the data set having few attributes selected as mentioned earlier. In our experiments, the clustering has been performed on two data sets: one with few correlated attributes and the other with all attributes considered. Subsequently, the results from the two data sets are compared to check for the accuracy of clustering. The command of PML to convert the data into the format of PML is as follows.

dataconverter CSVL datafile.txt datafilename

The command to run the K-Means algorithm is shown below.

mpiexec -np 1 pmlexec ../parameter_files/parameter_file_fuzzy_kmeans.xml datafilename exedata.txt

The results from PML are shown as the text file.

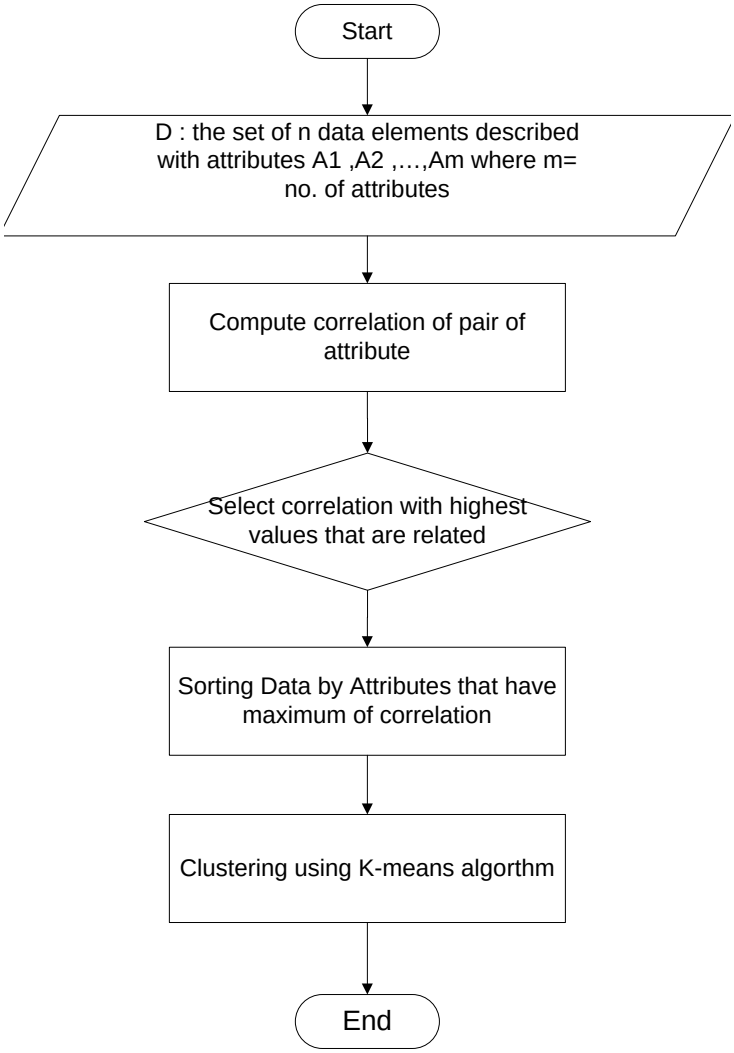

Figure 4.2: Steps of Computing Data Correlation

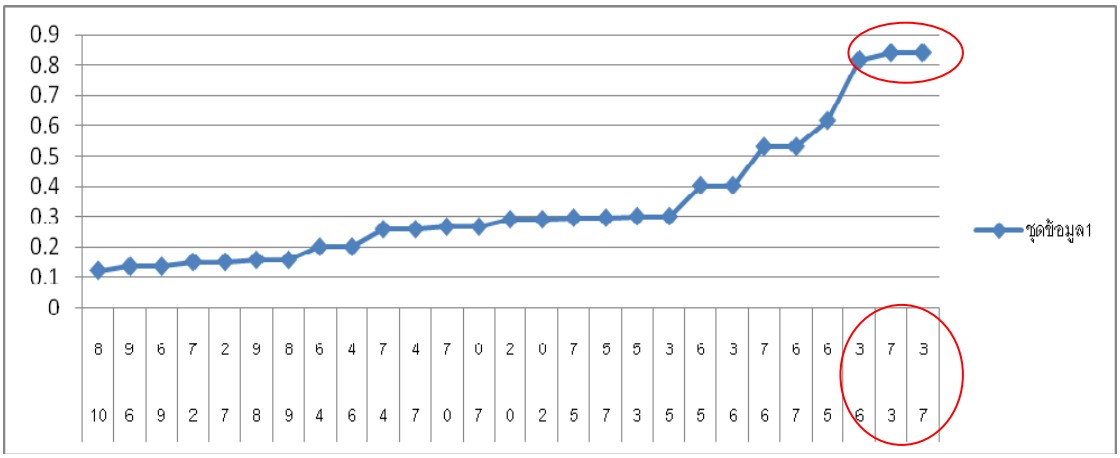

Figure 4.3: Sample of Strong Correlated Attributes

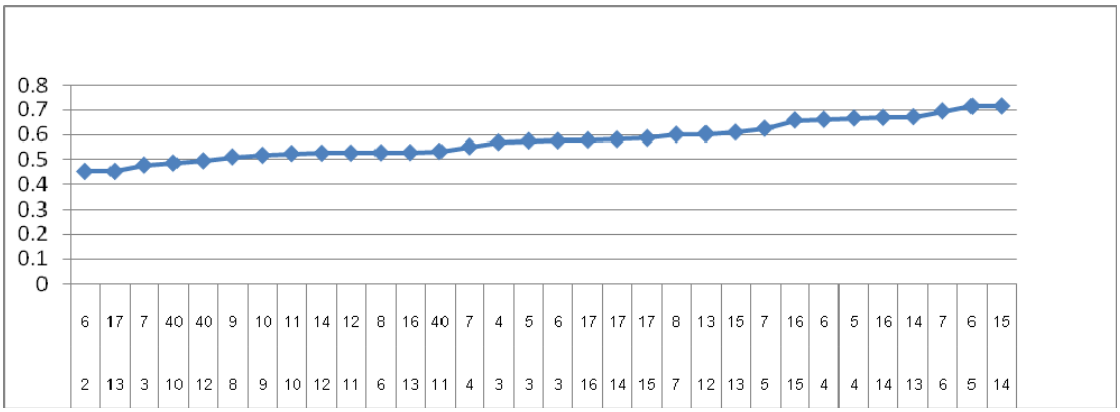

Figure 4.4: Sample of Tightly Closed Correlated Attributes

CHAPTER V
IMPLEMENTATION AND EXPERIMENTS

This chapter presents implementation and experiments of the proposed work. The data sets used in our work are selected from the UCI machine learning repository, and they have different number of attributes and different number of records. The details are as follows.

5.1 System Configuration

In our implementation, we run our work on the Rocks cluster of the Faculty of ICT at Mahidol University. The machine has installed the IBM's parallel data mining tool named Parallel Machine Learning or PML in short. We used the PML software as the tool to run the selected data sets. The specifications of hardware and software used are listed below.

- Hardware Configuration
  For machines on the Rocks cluster, CPU 2.4 GHz, 4GB Memory, ….
  For our machine used for simple calculation,…
- Software Configuration
  Put more details of all the software used here.

5.2 Experimental Steps

The proposed work is performed step by step as follows.

(1) Calculate the correlation value (r_{A_i,A_j}) between any two attributes, A_i and A_j, of the data set. For example, if the data set has eight attributes, the correlation value is computed between each pair of attributes, say, A1 and A2, A2 and A3, A3 and A4, and so on using the equation shown in (1) below.

$$r_{A_i,A_j} = \frac{\sum (A_i - \overline{A_i})(A_j - \overline{A_j})}{(n-1)\sigma_{A_i}\sigma_{A_j}} = \frac{\sum (A_i A_j) - n(\overline{A_i}\,\overline{A_j})}{(n-1)\sigma_{A_i}\sigma_{A_j}} \qquad (1)$$

(2) Sort the correlation values (r_{A_i,A_j}) of all pairs in ascending order as shown as the graph in Figure 5.1. The highest correlation value is determined.

(3) Select only a few attributes which have the highest correlation value. The number of chosen attributes depends on the fact that how many of them are closely related. Significantly, the selected attributes should represent all other attributes. Figure 5.1 shows that the three attributes, 0, 2 and 3 are the most correlated. Thus, these three attributes are selected to represent all data.

(4) Sort the selected attributes according to their correlation values in ascending order.

(5) Do the K-means clustering using the selected attributes.

(6) Compare the centroid of all the computed clusters when using selected attributes and when using all attributes.

5.3 Experimental Data

We select 10 data sets from the UCI machine learning repository for use in our experiments. They have different number of attributes and records as shown in Table 5.1 below.

Table 5.1: Data Sets from UCI Machine Learning repository

No.	Data Set	No. of Attributes	No. of Records
1	Iris	4	150
2	Ecoli	7	336
3	Yeast	8	1,484
4	Abalone	8	4,177
5	Page Blocks	10	5,473
6	White Wine	11	4,898
7	Magic Grammar Telescope	11	19,020
8	Letter Recognition	16	20,000

No.	Data Set	No. of Attributes	No. of Records
9	Breast Cancer	34	198
10	Waveform	40	5,000

The selected data sets have four combinations: few attributes and few records, few attributes and high records, high attributes and few records, and high attributes and high records.

5.4 Experimental Results

In this section, we explain the experimental results on each data set in detail.

5.4.1 Iris Data Set

Iris data set has three classes which refer to each type of Iris plant called Setosa, Versicolour and Virginica. Each class has 50 instances. Thus, the data set has 150 records in total and four attributes which are sepal length, sepal width, petal length and petal width. The correlation value of each pair of the four attributes is calculated and sorted in an ascending order as shown in Figure 5.1 below.

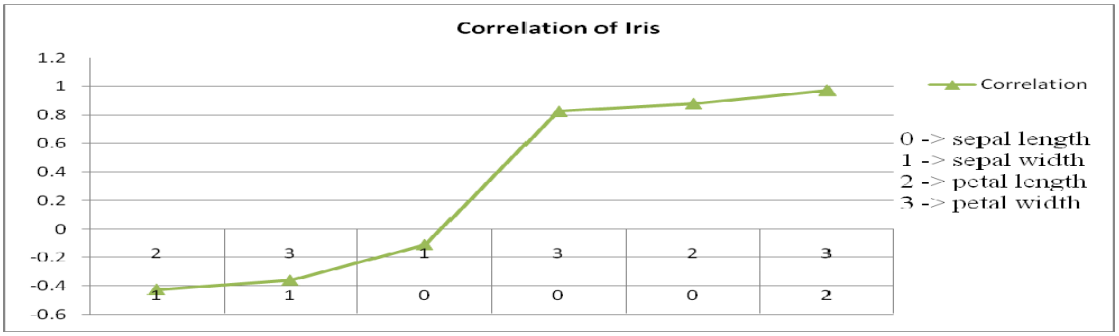

Figure 5.1: Iris Correlation Values

From the graph above, the attribute pairs have the highest correlation values are (3,2), (2,0) and (3,0). Thus, the three most correlation values are 0, 2 and 3. These attributes will be used to represent all data for computing the initial centriod for the number of clusters of 3. Then, we compared the centroid of 3 clusters computed using three attributes and using all attributes as shown in Table 5.2.

Table 5.2: Iris's Centroid using Few Attributes

Cluster Number	Iris 230			Iris All Attributes			
Cluster 0	1.464	0.244	5.006	1.464	0.244	5.006	3.418
Cluster 1	4.400	1.411	5.916	4.396	1.418	5.888	2.737
Cluster 2	5.732	2.107	6.826	5.702	2.079	6.846	3.082

The results illustrate that the centroid of each cluster between two outcomes is highly closed to each other. We also compute the centroid of the three clusters using other selected attributes having less correlation values. To measure the accuracy, in both cases, we compared all records in each cluster when using few attributes and when using all attributes, and calculated the number of total records that are not overlapped between two respective clusters. Table 5.3 shows the number of different records between two cases.

Table 5.3: Number and Percentage of Different Records for Iris's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
3, 2, 0	1	0.67
1, 2, 3	12	8.00

The results in Table 5.3 indicate that the attribute set of (3, 2, 0) with high correlation values have less percent of errors in clustering than the other set. Thus, using highly correlated attributes for computing centroid of clusters give almost the same accuracy as those using all attributes, but use less computing power.

5.4.2 Ecoli Data Set

Ecoli data set is used to predict protein localization site. It contains seven attributes and 336 records. The correlation values among all attributes are computed and presented in Figure 5.2 which shows three attributes, 0, 5 and 6, having the most correlation values. Thus, these three attributes will be used to represent all attributes. We select to compute the initial centroid of four clusters. The results of four clusters are compared with the centroid of another four clusters using all attributes as shown in Table 5.4.

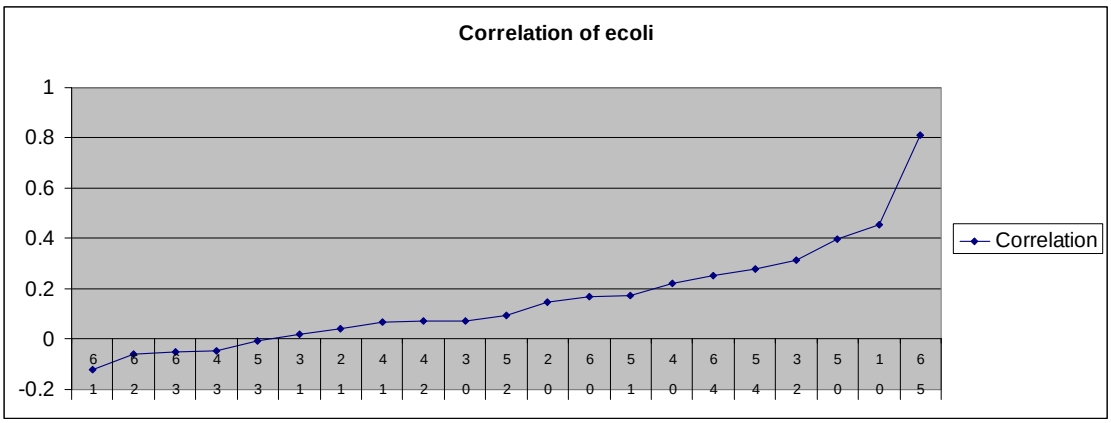

Figure5.2: Ecoli Correlation Values

Table 5.4: Ecoli's Centroid using Few Attributes

Cluster Number	Ecoli 560			Ecoli All Attributes		
Cluster0	0.306	0.386	0.336	0.317	0.390	0.354
Cluster1	0.467	0.343	0.654	0.474	0.336	0.668
Cluster2	0.737	0.744	0.361	0.748	0.751	0.354
Cluster3	0.765	0.773	0.704	0.764	0.772	0.700

The results illustrate that the Ecoli's centroid of each cluster between two outcomes is highly closed to each other. We also compute the Ecoli's centroid of the four clusters using other selected attributes having less correlation values. To measure the accuracy, we compared all records in each cluster when using few attributes and when using all attributes, and calculated the number of total records that are not overlapped between two respective clusters. Table 5.5 shows the number of different records between two cases.

The results in Table 5.5 indicate that the attribute set of (6, 5, 0) has less percent of errors in clustering than other selected sets. As a consequence, using highly correlated attributes for computing centroid of clusters give high accuracy with minimal errors.

Table 5.5: Number and Percentage of Different Records for Ecoli's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
6, 5, 0	13	3.86
1, 5, 6	67	19.94
2, 5, 6	83	24.70
4, 5, 6	98	29.16
3, 5, 6	99	29.46

5.4.3 Yeast Data Set

Yeast data set is used to predict the cellular localization site of proteins. It contains 8 attributes and 1,484 records. The correlation values of all attribute pairs are calculated as shown in Figure 5.3 below.

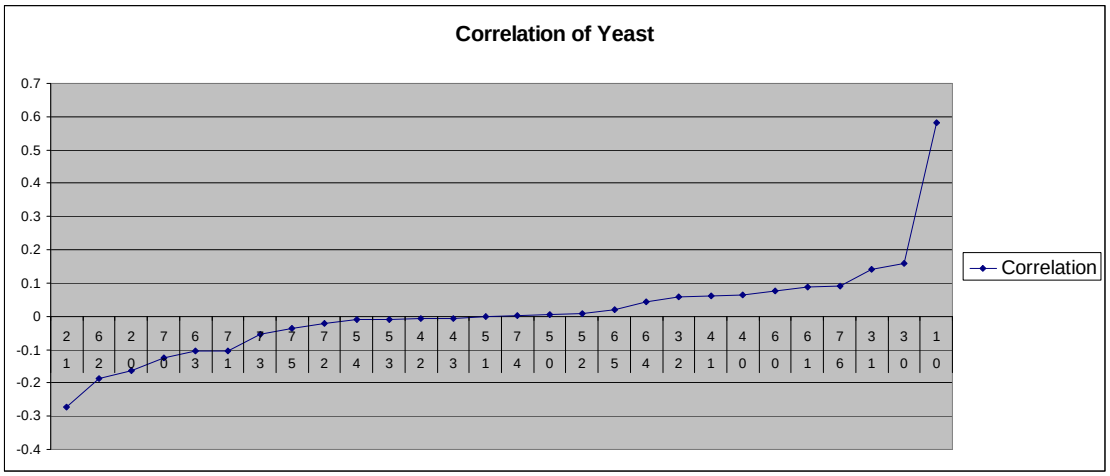

Figure 5.3: Yeast Correlation Values

The results indicate that the attribute set of (0, 1, 3) has the most correlation values, and they can be used to represent all attributes. Hence, these three attributes are used to find the initial centroid of four clusters. The resulting centroid of the four clusters using the selected three attributes is compared with the centroid computed from all attributes as shown in Table 5.6.

Table 5.6: Yeast's Centroid using Few Attributes

Cluster Number	Yeast_013			Yeast All Attributes		
Cluster0	0.507	0.505	0.5100	0.503	0.500	0.505
Cluster1	0.380	0.393	0.206	0.380	0.395	0.204
Cluster2	0.519	0.526	0.199	0.521	0.527	0.199
Cluster3	0.748	0.693	0.291	0.752	0.694	0.297

The results illustrate that the Yeast's centroid of each cluster between two outcomes is highly closed to each other. We also compute the Yeast's centroid of the four clusters using other selected attributes having less correlation values. To measure the accuracy, we compared all records in each cluster when using few attributes and when using all attributes, and calculated the number of total records that are not overlapped between two respective clusters. Table 5.7 shows the number of different records between two cases.

Table 5.7: Number and Percentage of Different Records for Yeast's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
0, 1, 3	212	14.28
4, 0, 1	696	47.04
7, 0, 1	741	49.93
6, 0, 1	833	56.13
5, 0, 1	851	57.34
2, 0, 1	870	58.62

The results in Table 5.7 indicate that the attribute set of (0, 1, 3) has higher common records with the all-attribute clusters than other attribute sets. Therefore, using highly correlated attributes for computing centroid of clusters would give comparable accuracy while minimizing computation efforts.

5.4.4 Abalone Data Set

Abalone data set is used to predict the age of abalone from physical measurements. The age of abalone is determined by cutting the shell through the cone,

staining it, and counting the number of rings through a microscope. The data set consists of eight attributes and 4,177 records. The correlation values among all pairs of attributes are computed and presented as shown in Figure 5.4.

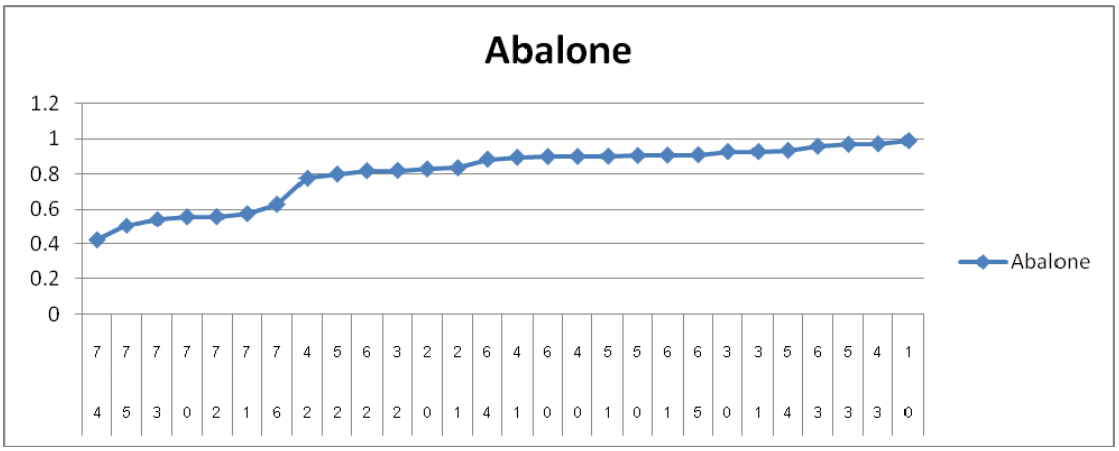

Figure 5.4: Abalone Correlation Values

The correlation of attributes in this data set is different from the previous data sets since the correlation values among attribute pairs are not obvious. Thus, only the two highest correlated attribute pairs are selected to be the representatives, and they are (0, 1, 3, 4) attribute set. In addition, the initial centroid of four clusters is computed based on only these four attributes. The outputs are shown in Table 5.8 below.

Table 5.8: Abalone's Centroid using Few Attributes

Cluster Number	Abalone_0134				Abalone All Attributes			
Cluster0	0.682	0.538	1.713	0.751	0.681	0.538	1.706	0.748
Cluster1	0.607	0.476	1.131	0.492	0.604	0.474	1.111	0.483
Cluster2	0.516	0.401	0.677	0.289	0.510	0.395	0.648	0.277
Cluster3	0.362	0.274	0.251	0.108	0.355	0.268	0.234	0.101

The results illustrate that the Abalone's centroid of each cluster between two outcomes is still highly closed. We also compute the Abalone's centroid of the four clusters using other attribute sets that have less correlation values. To measure the accuracy, we compared all records in each cluster when using few attributes and when using all attributes, and calculated the number of total records that are not overlapped

between two respective clusters. Table 5.9 shows the number of different records between two cases.

Table 5.9: Number and Percentage of Different Records for Abalone's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
0,1,3,4	192	4.5
0,1,3,2	213	5.1
0,1,3,5	258	6.18
0,1,3,6	527	12.61

The results in Table 5.9 illustrate that the attribute set of (0, 1, 3, 4) has less number of different records in clustering than other selected sets. Thus, using highly correlated attributes for computing centroid of clusters would give a certain degree of accuracy with minimal errors.

5.4.5 Page Blocks Data Set

Page Blocks data set is used to classify all the blocks of the page layouts of documents detected by a segmentation process. The data set consists of 10 attributes and 5,473 records. The correlation values between each pair of all attributes are computed and the outputs are presented in Figure 5.5.

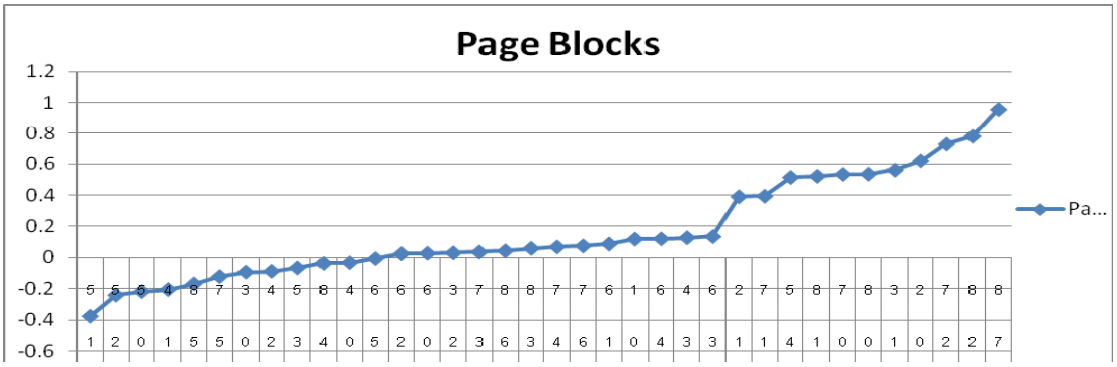

Figure 5.5: Page Blocks Correlation Values

Figure 5.5 shows the highest correlation values are among the attributes 7, 8 and 2, and these three attributes are used to represent all attributes. The initial centroid of four clusters using these three attributes is computed, and compared with

the corresponding clusters using all attributes. The results are shown in Table 5.10 below.

Table 5.10: Page Block's Centroid using Few Attributes

Cluster Number	Page_Blocks_278			Page_Blocks All Attributes		
Cluster0	1432.61	2900.89	4424.44	1429.31	2895.23	4418.76
Cluster1	17529.80	26486.00	101419.00	17529.80	26486.00	101419.00
Cluster2	147.73	347.88	476.08	147.59	347.50	475.13
Cluster3	8827.62	13325.70	22906.40	8827.62	13325.70	22906.40

The results illustrate that the Page Block's centroid of each cluster between two outcomes is still highly closed. We also compute the Page Block's centroid of the four clusters using other attribute sets that have less correlation values. To measure the accuracy, we compared all records in each cluster when using few attributes and when using all attributes, and calculated the number of total records that are not overlapped between two respective clusters. Table 5.11 shows the number of different records between two cases.

Table 5.11: Number and Percentage of Different Records for Page Block's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
7, 8, 2	2	0.037
7, 8, 0	135	2.47
7, 8, 1	132	2.41
7, 8, 3	135	2.47
7, 8, 4	135	2.47
7, 8, 5	135	2.47
7, 8, 6	302	5.52

The results in Table 5.11 depict that the attribute set of (7, 8, 2) gives only two different records in all clusters when compared with clusters using all attributes.

5.4.6 White Wine Data Set

White Wine data set is the samples of white vinho verde wine from the north of Portugal. The goal is to model the wine quality based on physicochemical tests. The data set contains 11 attributes and 4,898 records. The correlation values of each attribute pair are calculated and presented as shown in Figure 5.6 below.

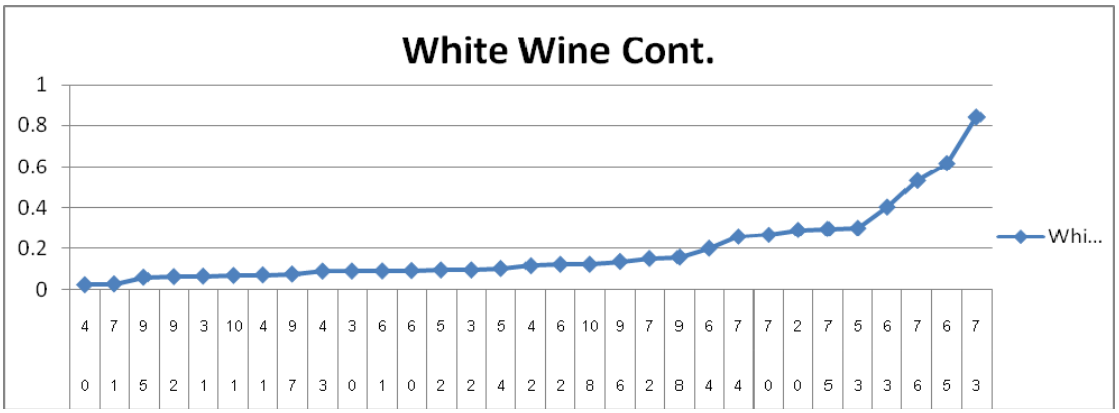

Figure 5.6: White Wine Correlation Values

Figure 5.6 shows that the highest correlation values are among the attributes: 7, 3 and 6. Thus, these three attributes are the representatives for finding initial centroid of four clusters. The results are compared with the centroid computed using all attributes as shown in Table 5.12, and show that the White Wine's centroid of each cluster between two outcomes is highly closed.

Table 5.12: White Wine's Centroid using Few Attributes

Cluster Number	White_Wine_736			White_Wine All Attributes		
Cluster0	121.32	0.32	30.17	121.10	0.32	30.04
Cluster1	160.33	0.34	42.30	160.11	0.34	42.33
Cluster2	206.68	0.35	52.72	206.67	0.35	52.73
Cluster3	83.22	0.31	20.59	83.09	0.31	20.54

When we compute the White Wine's centroid of the four clusters using other attribute sets having less correlation values, and compared all records in every cluster when using all attributes, we found that the attribute set (7, 3, 6) gives the least

total number of records that are not overlapped between two respective clusters. Table 5.13 shows the number of different records of some selected attribute sets.

Table 5.13: Number and Percentage of Different Records for White Wine's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
7, 3, 6	23	0.47
7, 3, 5	286	5.84
7, 3, 1	518	10.58
7, 3, 2	349	7.13
7, 3, 4	463	9.45
7, 3, 8	307	6.27
7, 3, 9	284	5.79
7, 3, 10	303	6.19
7, 3, 11	349	7.13

5.4.7 Magic Gamma Telescope Data Set

Magic Gamma Telescope data set is the data generated to simulate the registration of high energy gamma particles in an atmospheric Cherenkov telescope using the imaging technique. The data set consists of 11 attributes and 19,020 records. The correlation values of each attribute pair are computed and presented as shown in Figure 5.7.

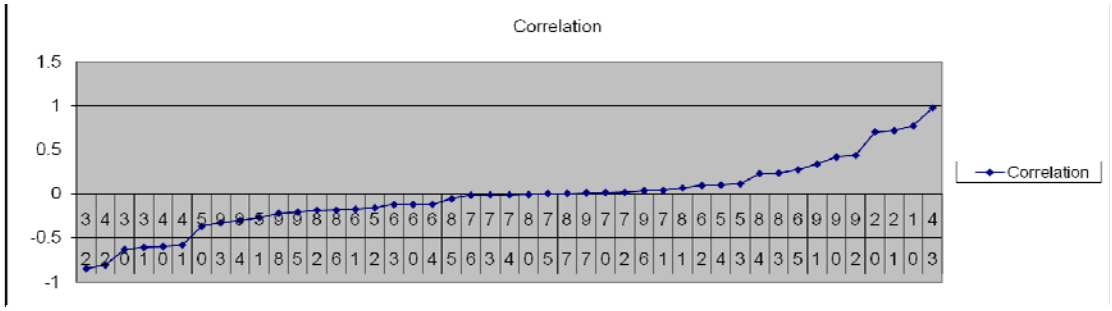

Figure 5.7: Magic Gamma Telescope Correlation Values

From Figure 5.7, the six attributes with the highest correlation values are 4, 3, 0, 1, 2 and 9, and they are used to represent all attributes. The centroid of four

clusters computed from the six attributes and the centroid of four clusters computed from all attributes are compared as shown in Table 5.14.

Table 5.14: Magic Grammar Telescope's Centroid using Few Attributes

Cluster Number	Magic Grammar_430129					
Cluster0	0.13	0.24	91.76	35.23	3.29	303.85
Cluster1	0.23	0.41	38.40	16.34	2.70	204.28
Cluster2	0.09	0.17	175.12	66.79	3.46	198.04
Cluster3	0.253	0.44	33.44	16.16	2.62	111.09
Cluster Number	Magic_Grammar All Attributes					
Cluster0	0.14	0.24	90.11	33.75	3.25	295.49
Cluster1	0.21	0.38	45.98	18.90	2.79	214.84
Cluster2	0.11	0.20	162.43	63.83	3.45	266.34
Cluster3	0.25	0.45	33.46	16.18	2.61	120.27

When we compute the Magic Grammar Telescope's centroid of the four clusters using other attribute sets having less correlation values, and compared all records in every cluster when using all attributes, we found that the attribute set (4,3,0,1,2,9) gives the least total number of records that are not overlapped between two respective clusters. Table 5.15 shows the number of different records of some selected attribute sets.

Table 5.15: Number and Percentage of Different Records for Magic Grammar Telescope's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
4,3,0,1,2,9	3,292	17.31
4,3,0,1,2,5	8,371	44.01
4,3,0,1,2,6	9,970	52.42
4,3,0,1,2,7	9,855	51.81
4,3,0,1,2,8	8,980	47.21

5.4.8 Letter Recognition Data Set

Letter Recognition data set has the objective to identify a large number of black-and-white rectangular pixel displays as one of the 26 capital letters in English alphabets. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli. Each stimulus was converted into 16 primitive numerical attributes which were scaled to fit into a range of integer values from 0 through 15. The data consists of 16 attributes and 20,000 records in total. The correlation values among all attributes are computed as shown in Figure 5.8.

Figure 5.8 shows the graph of correlation values among all attributes of Letter Recognition data set. The graph is rather flat because most attributes are closely related. However, we selected four attributes having the highest correlation values, and they are 0, 1, 2, and 3 to represent all attributes. The centroid of four clusters computed from four attributes and those from all attributes are compared. But, the results shown in Table 5.16 depict that all centroids of four clusters are somewhat far apart. For higher accuracy, we selected eight attributes (0, 1, 2, 3, 4, 6, 10, 12) which are closely related. Then, the centroid of four clusters computed from eight attributes and those from all attributes are compared as shown in Table 5.17 below.

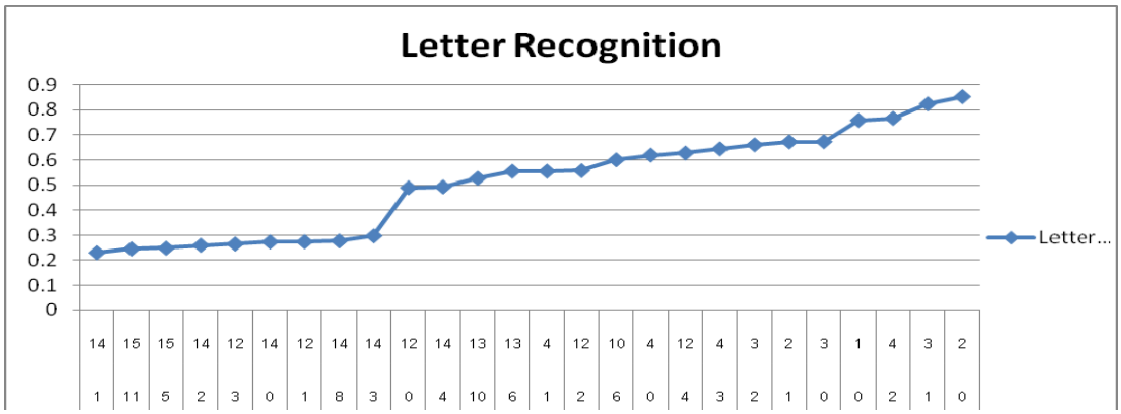

Figure 5.8: Letter Recognition Correlation Values

Table 5.16: Letter Recognition's Centroid using Four Attributes

Cluster Number	Letter Recognition_0213				Letter Recognition_All Attributes			
Cluster0	4.32	5.46	8.62	6.56	4.37	5.47	7.66	5.61
Cluster1	3.13	4.37	5.55	4.50	3.76	4.67	7.40	5.95
Cluster2	6.72	7.72	10.87	7.58	5.67	6.94	9.67	7.04
Cluster3	1.79	2.66	1.89	1.88	2.02	3.09	2.99	2.57

When we compute the Letter Recognition's centroid of the four clusters using a few attribute sets of different correlation values, and compared all records in every cluster when using all attributes, we found that the attribute set (0, 1, 2, 3, 4, 6, 10, 12) gives the least total number of records that are not overlapped between two respective clusters as shown in Table 5.18.

Table 5.17: Letter Recognition's Centroid using Eight Attributes

Cluster Number	Letter Recognition_0123461012							
Cluster0	4.12	5.18	7.35	5.64	2.83	2.40	10.05	9.95
Cluster1	3.70	4.82	7.56	5.74	3.25	2.34	6.39	4.89
Cluster2	6.09	7.31	10.04	7.34	6.01	5.25	7.32	6.24
Cluster3	1.98	2.95	2.52	2.33	1.44	1.97	7.41	6.36
	Letter Recognition_All Attributes							
Cluster0	4.37	5.47	7.66	5.61	2.97	2.42	5.28	10.18
Cluster1	3.76	4.67	7.40	5.95	3.23	2.65	6.58	6.82
Cluster2	5.67	6.94	9.67	7.04	5.69	4.70	7.89	6.80
Cluster3	2.02	3.09	2.99	2.57	1.54	1.90	7.24	6.96

Table 5.18: Number and Percentage of Different Records for Letter Recognition's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
0,2,1,3	9,566	47.83
0,2,1,3,4	9,119	45.59
0,2,1,3,4,12	8,591	42.95
0,2,1,3,4,12,6,10	4,916	24.58

Selected Attributes	Number of Different Records	In Percent (%)
0,2,1,3,4,12,6,5	5,012	25.06
0,2,1,3,4,12,6,7	5,243	26.26
0,2,1,3,4,12,6,8	5,835	29.18
0,2,1,3,4,12,6,9	8,936	44.68

5.4.9 Breast Cancer Data Set

Each record in the Cancer data set represents the follow-up data for individual breast cancer case. The data are collected from consecutive patients seen by Dr. Wolberg since 1984, and include only cases that exhibit invasive breast cancer and show no evidence of distant metastases at the time of diagnosis. The data consists of 34 attributes and 198 records in total. The correlation values among all attribute pairs are computed as shown in Figure 5.9. However, the attributes in this data set are highly related as the graph shown is rather flat. Thus, we choose the first three attributes with the highest correlation values. They are attributes (1, 3, 4) to compute the initial centroid using the K-means method, and compare the results with those using all attributes.

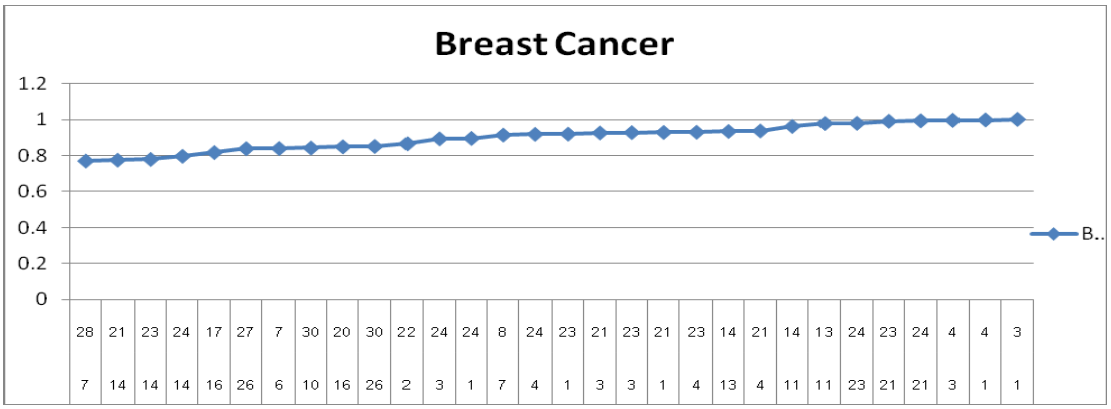

Figure 5.9: Breast Cancer Correlation Values

Table 5.19: Breast Cancer's Centroid using Three Attributes

Cluster Number	Breast Cancer_134			Breast Cancer_All Attributes		
Cluster0	13.94	91.79	604.75	14.14	93.04	622.43
Cluster1	19.84	130.97	1223.32	20.02	132.68	1249.31
Cluster2	16.99	111.68	899.18	17.47	114.71	948.93
Cluster3	23.40	156.22	1699.06	23.07	153.29	1671.60

The results of choosing only three attributes do not give high accuracy when compared with the centroid obtained from using all attributes. Therefore, we extend the number of selected attributes to be 5 which are (1, 3, 4, 21, 24) for use in computing the centroid. The results are shown in Table 5.20 below.

Table 5.20: Breast Cancer's Centroid using Five Attributes

Cluster Number	Breast Cancer_134_2124				
Cluster0	14.14	93.04	622.43	16.74	862.99
Cluster1	20.00	132.51	1247.32	24.32	1815.89
Cluster2	17.44	114.55	945.55	20.69	1309.00
Cluster3	23.07	153.29	1671.60	30.15	2794.33
Cluster Number	Breast Cancer_All Attributes				
Cluster0	14.14	93.04	622.43	56.22	21.68
Cluster1	20.02	132.68	1249.31	37.63	22.11
Cluster2	17.47	114.71	948.93	48.36	22.48
Cluster3	23.07	153.29	1671.60	28.93	24.84

With five closely related attributes, their computed centroid is very closed to the centroid computed from all attributes. In addition, when we compute the centroid of other attribute sets having less correlation values and compared them with the centroid computed from all attributes, the number and percentage of different records in all clusters as shown in Table 5.21 illustrate that our selected five attributes give the highest accuracy.

Table 5.21: Number and Percentage of Different Records for Breast Cancer's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
1, 3, 4, 21, 24	1	0.5
1, 3, 4, 21, 23	33	16.67
1, 3, 4, 21, 13	33	16.67
1, 3, 4, 21, 0	34	17.17
1, 3, 4, 21, 2	33	16.67

5.4.10 Waveform Data Set

Waveform data set is obtained from the CART book's waveform domain. Each instance is generated by adding noise in each attribute. The data consists of 40 attributes and 5,000 records in total. The correlation values between each pair of attributes are computed and presented in Figure 5.9.

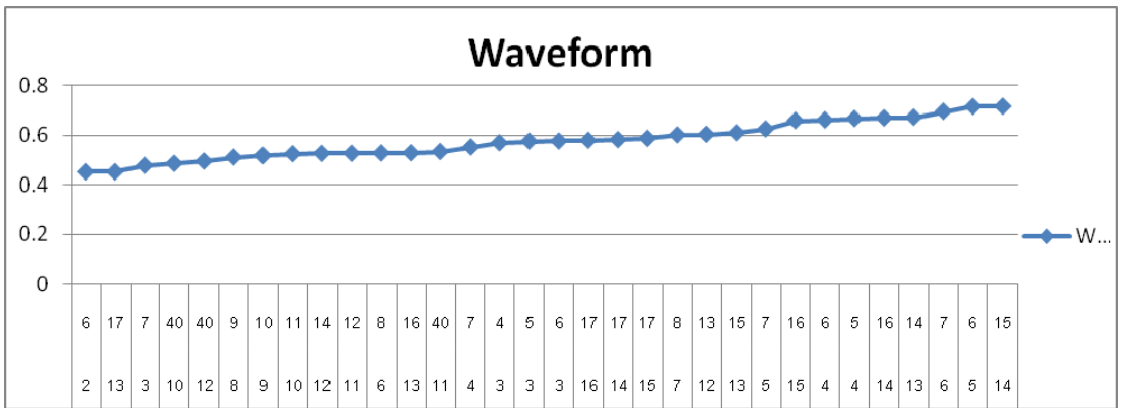

Figure 5.9: Waveform Correlation Values

Similar to Breast Cancer data set, all attributes in Waveform are closely related since the graph in Figure 5.9 is rather flat and the correlation values among them are closed to each other. Thus, if we selected a few number of attributes, the computed centroid of clusters may not be closed to the computed centroid of clusters using all attributes. In addition, since the Waveform data set is large and has high number of attributes, a small number of attributes cannot be used to represent all attributes. Table 5.22 shows the computed centroid of four clusters using four attribute set of (5, 6, 14, 15) and using all attributes.

Table 5.22: Waveform's Centroid using Four Attributes

Cluster Number	Waveform_5-6-14-15				Waveform_All Attributes			
Cluster0	0.52	0.38	4.38	5.30	0.74	0.49	3.98	4.92
Cluster1	5.21	4.24	0.32	0.52	4.93	3.97	0.38	0.62
Cluster2	1.62	0.95	2.48	3.39	2.14	1.29	1.31	2.20
Cluster3	3.08	2.27	1.09	1.69	3.12	2.55	2.50	2.92

Table 5.22 shows that the computed centroid using the most closely related four does not give high accuracy when compared with the centroid computed using all attributes since the attributes of Waveform are highly related and the data set has high

number of attributes. Hence, only four attributes cannot represent all attributes. Later, we selected additional highly related attributes having the most correlation value. The attribute set of (5, 6, 14, 15, 7, 13, 16) is chosen. Its centroid is computed and compared with those using all attributed. The results are shown in Table 5.23 below.

Table 5.23: Waveform's Centroid using Seven Attributes

Cluster Number	Waveform_7_Attributes						
Cluster0	0.65	0.47	4.19	5.09	4.45	0.69	0.24
Cluster1	5.25	4.27	0.31	0.51	0.55	4.56	3.40
Cluster2	1.76	1.01	2.01	2.98	3.28	2.36	0.32
Cluster3	3.42	2.62	1.15	1.60	1.99	3.45	1.79
Cluster Number	Waveform_All Attributes						
Cluster0	0.74	0.49	3.98	4.92	4.37	0.84	0.22
Cluster1	4.93	3.97	0.38	0.62	0.79	4.40	3.09
Cluster2	2.14	1.29	1.31	2.20	2.86	2.91	0.39
Cluster3	3.12	2.55	2.50	2.92	2.48	2.55	2.12

With seven closely related attributes, their computed centroid is highly closed to the centroid computed from all attributes. In addition, when we compute the centroid of other attribute sets having less correlation values and compared them with the centroid computed from all attributes, the number and percentage of different records in all clusters as shown in Table 5.24 illustrate that our selected seven attributes give the highest accuracy. Thus, they can represent other attributes in the data set. Moreover, even though the attribute set of 8 attributes is selected, its results do not give better accuracy than the attribute set of 7 attributes that we choose.

Table 5.24: Number and Percentage of Different Records for Waveform's Few Attributes

Selected Attributes	Number of Different Records	In Percent (%)
14,15,5,6	1,829	36.58
14,15,5,6,7	1,465	29.30
14,15,5,6,7,13	1,414	28.28

Selected Attributes	Number of Different Records	In Percent (%)
14,15,5,6,7,13,16	1,398	27.96
14,15,5,6,7,13,16,4	1,495	29.90
14,15,5,6,7,13,0	1,983	39.66
14,15,5,6,7,13,1	1,826	36.52
14,15,5,6,7,13,2	1,534	30.68
14,15,5,6,7,13,3	1,462	29.24
14,15,5,6,7,13,4	1,435	28.70
14,15,5,6,7,13,16,4	1,495	29.90

5.5 Experimental Conclusion

Our experiments are performed on a variety of data sets. They range from few to high number of attributes and have a few hundred of records to many thousands of records. However, the significant factors contribute to the accuracy of our method is not the number of attributes, but how close they are related. The graph of correlation values show the relationship among all attributes. If the graph shown is rather flat, it would be hard for us to separate the most related attributes from others. As a result, the accuracy of the computed centroid obtained from the selected attribute set would not be high as well.

Thus, our propose method performs well with high accuracy for the data set that some attributes are closely related. If all attributes of the data set are highly related, the correlation values between any pair of attributes will be almost equal and the computed centroid will be deviated from those using the full attribute set.

CHAPTER VI
DISCUSSION AND CONCLUSIONS

This chapter concludes our work by summarizing and discussing the advantages and the disadvantages of the proposed system as well as suggesting some future work for further development.

6.1 Summary of the Proposed Work

The problem of K-means algorithm is to set the initial centroids for each cluster. If the initial centroid is poorly selected, it would take a long time to get results. In addition, the initial centroid usually includes all attributes of data. This work is based on the assumption that some attributes are closely related while other attributes may not be related. Thus, we propose a new algorithm to get the centroids of each cluster by using a few attributes that are closely correlated. Our work found the set of correlated attributes and used them for computing the centroids. Thus, the number of attributes used in the calculation of K-means is reduced, and the computation time would be decreased as well.

Our experiments used the data set from the UCI machine learning repository, and the results showed that the centroids using highly correlated attributes are closer to the centroids using all attributes than those centroids using lowly correlated attributes. Thus, few correlated attributes can represent all attributes in the calculation of K-means clustering. However, for some data set that all attributes are not closely correlated, selecting only few attributes to represent all attributes does not give the good accuracy in the calculation of centroids.

6.2 Suggestions for Future Work

Our suggestions for the future work are as follows.

1. Since computing the K-means clustering using few correlated attributes is not suitable for data set whose attributes are closely related, and that would make it difficult to select attributes that represent all data, defining the ratio of attributes that are correlated to be used in the calculation would help in the selection process. However, it depends on how much accuracy we would like to obtain for the K-means clustering of each data set.

2. When the data are clustered using our algorithm and the correlated attributes are sorted, the data are divided into groups. Thus, the parallel K-means clustering can be used for these data groups.

REFERENCES

1 F. Yuan, Z. Meng, H. Zhang and C.R. Dong. "A New Algorithm to Get the Initial Centroids". Proceedings of the 3rd International Conference on Machine Learning and Cybernetics, Shanghai, August 2004.

2 S.S. Khan and A. Ahmad. "Cluster Center Initialization Algorithm for K-means Clustering", Pattern Recognition Letters, 25 (2004), pp. 1293-1302.

3 P.S. Bradley and U. M. Fayyad. "Refineing Initial Points for K-means Clustering". Proceeding of The Fifteenth International Conference on Machine Learning, Morgan Kaufmann, San Francisco, CA, 1998, pp. 91-99.

4 S. Deelers and S. Auwatanamongkol. "Enhancing K-Means Algorithm with Initial Cluster Centers Derived from Data Partitioning along The Data Axis with the Highest Variance". International Journal of Computer Science, Volume 2, Number 4, 2007.

5 C. Ding and X. He, "K-Means Clustering via Principal Component Analysis", Proceedings of the 21st International Conference on Machine Learning, Banff, Canada, 2004.

6 I. Jolliffe, Principal Component Analysis, 2nd Edition, Springer, 2002.

7 J. Han and M. Kamber. "Data Mining: Concepts and Technique", 2nd Edition, Morgan Kaufmann, 2006.

8 C.L. Blake, C.J. Merz. UCI Repository of Machine Learning Databases. University of California, Irvine, Department of Information and Computer Science, 1998.

9 K. R. Zalik, "An Efficient K-means Clustering Algorithm", Pattern Recognition Letters, 29 (2008), pp. 1385-1391.

10 E. Yom-Tov, E. Pednault, R. Natarajan, D. Pelleg, H. Toledano and E. Aharoni. "Parallel Machine Learning (PML)". Verification Solution and Machine Learning Group, IBM Haifa Research Lab and Data Analytics Department, IBM T.J. Watson Research Lab, November 2007.

11 Rocks Cluster. Available at http://www.rocks.org

BIOGRAPHY

NAME	Miss Thipprapa Popiyatrakul
DATE OF BIRTH	7 April 1983
PLACE OF BIRTH	Buriram, Thailand
INSTITUTIONS ATTENDED	Khon-Kaen University, 2001-2004 Bachelor of Science (Computer Science) Mahidol University, 2006-2010 Master of Science (Computer Science)
HOME ADDRESS	234/8 Jira Rd., Muang, Buriram 31000 Thailand E-mail: noon62@hotmail.com
EMPLOYMENT ADDRESS	-
PUBLICATION / PRESENTATION	National Computer Science and Engineering Conference 2009 5-6 November 2009 Bangkok, Thailand