12-15-2006

Clustering of Leukemia Patients via Gene Expression Data Analysis

Zhiyu Zhao
University of New Orleans

Clustering of Leukemia Patients via Gene Expression Data Analysis

A Thesis

Submitted to the Graduate Faculty of the
University of New Orleans
in partial fulfillment of the
requirements for the degree of

Master of Science
in
Computer Science

by

Zhiyu Zhao

B.E. Huazhong University of Science and Technology, 1997
M.E. Huazhong University of Science and Technology, 2000

December 2006

Acknowledgement

I would like to thank all the people who gave me the opportunity to complete this master thesis.

I appreciate the kind guidance and instruction of my thesis advisor, Dr. Bin Fu. His intelligent ideas and strict attitude toward academic research impressed me deeply, and I believe I will benefit from what I have learned from him in my whole life.

I wish to thank Dr. Shengru Tu who always gave me valuable advices and kind help, and Dr. Stephen Winters-Hilt who helped me as my co-major professor as well as a member of my thesis committee, and gave precious comments to my thesis.

I would like to thank Dr. Mahdi Abdelguerfi who cared much about my study, my life and my family, and Dr. Diego Liberati who led me into the research field of Bioinformatics while I was studying in Italy.

Finally, I would like to express my deep gratitude to my husband Liqiang Wang, my parents, my brother and his wife for their care, understanding and encouragement throughout these years.

Table of Contents

List of Figures

List of Tables

Abstract

This thesis attempts to cluster some leukemia patients described by gene expression data, and discover the most discriminating a few genes that are responsible for the clustering. A combined approach of Principal Direction Divisive Partitioning and bisect K-means algorithms is applied to the clustering of the selected leukemia dataset, and both unsupervised and supervised methods are considered in order to get the optimal results. As shown by the experimental results and the predefined reference, the combination of PDDP and bisect K-means successfully clusters the leukemia patients, and efficiently discovers some significant genes that can serve as the discriminator of the clustering. The combined approach works well on the automatic clustering of leukemia patients depending merely on the gene expression information, and it has great potential on solving similar problems. The discovered a few genes may provide very important information for the diagnosis of the disease of leukemia.

Chapter 1. Introduction

The rapid development of the DNA micro-array technology is making it more and more convenient to obtain various gene expression datasets with abundant information that can be very helpful for many meaningful biomedical applications such as prediction, prevention, diagnosis and treatment of diseases, development of new drugs, patient-tailored therapy, and so on. However, these datasets are usually very large and unbalanced, with the number of genes (thousands upon thousands) being much greater than the number of patients (generally from tens to hundreds). Consequently, how to analyze effectively this kind of large datasets with few samples and numerous attributes, for example, how to classify according to their gene expression profile the patients suffering from certain disease, or how to determine from thousands of genes the most discriminating ones that are responsible for the corresponding disease, should be viewed as an important issue.

Recently there have been many exciting research results (1-11) in the area of DNA micro-array data mining on the basis of gene expression data analysis. For instances, depending solely on gene expression monitoring to micro-array datasets, Golub et al (1999) classified sample patients of acute leukemia as two sub types, ALL (Acute Lymphoblastic Leukemia) and AML (Acute Myeloid Leukemia), and predicted the sub types of new leukemia cases according to the expression values of the most decisive genes that were discovered during the classification of sample cases; Scott et al (2002) discovered a new sub type of acute leukemia, MLL (Mixed Lineage Leukemia), which was claimed as distinct enough to be separated from ALL or AML; In a hierarchical point of view, Loris et al (2004) classified patients of advanced ovarian cancer and extracted significant genes which characterized each level in the hierarchies; On the basis

of gene expression profile analysis van't Veer et al (2002) predicted the clinical outcome (relapse / non-relapse) of breast cancer and Pomeroy et al (2002) predicted the outcome (survivor / failure) of embryonal tumor of central nervous system; Alon et al (1999) clustered correlated gene families about colon tissues and separated cancerous from non cancerous tissues; Dinesh et al (2002) performed the tumor versus normal classification of prostate cancer and predicted the clinical outcome of prostatectomy; Eng-Juh et al (2002) classified the sub types and predicted the outcome of pediatric acute lymphoblastic leukemia; Gavin et al (2002) separated malignant pleural mesothelioma (MPM), which is not a lung cancer, from adenocarcinoma (ADCA) of the lung; Alizadeh et al (2000) identified two distinct types of diffuse large B-cell lymphoma (DLBCL), the germinal centre B-like DLBCL and the activated B-like DLBCL.

The technologies applied in the analysis of gene expression data are various. In (1) a method of neighborhood analysis is used to select out the most informative genes that are related to the classification of patients, a class predictor is designed by using the sum of the weighted votes from these genes to determine the wining class, and a cross-validation method is adopted to test the accuracy of the predictor. To classify the leukemia patients, a technology of self-organizing maps is applied to obtain two classes. In (3) an unsupervised method is used to cluster both genes and tumors, and a supervised alternative is adopted to identify the outcome of the tumors and extract the most significant genes that are related to the outcome. In (4) Principal Component Analysis (PCA) is applied to determine different types of tumors and the related genes. In (5) a deterministic-annealing algorithm is used to organize both genes and sample tissues into binary trees so that they can be clustered hierarchically. In (9) gene expression ratios are calculated and thresholds are selected to distinguish between

cancer and non-cancer tissues.

In this thesis, an approach based on the collaboration of three algorithms, Principal Component Analysis (PCA), Principal Direction Divisive Partitioning (PDDP), and bisect K-means, is applied to cluster the sample patients from a public leukemia dataset (see 11) which consists of 72 leukemia samples (24 acute lymphoblastic leukemia (ALL), 20 mixed-lineage leukemia (MLL) and 28 acute myeloid leukemia (AML)) with each sample represented by 12,582 gene expression values. In the mean time, a few significant genes that are strongly related to the result of the clustering are discovered. All the algorithms are implemented and the dataset imported using MATLAB, and the experimental results on the clustering of the patients and the discovering of the significant genes are discussed.

The remaining content of the thesis is organized as follows:

Chapter 2 is about the description and the pre-processing of the leukemia dataset that is used in the experiments, chapter 3 describes in detail the clustering algorithms, chapter 4 illustrates the experimental results of the clustering of the leukemia dataset by applying the MATLAB coded algorithms, and chapter 5 is the discussion and conclusion about the results.

Chapter 2. Dataset Description and Pre-processing

2.1 Description of the Dataset

The dataset analyzed in this thesis is the combination of two leukemia datasets processed in (11), where 57 samples (20 ALL, 17 MLL and 20 AML) are used for the training and 15 (4 ALL, 3 MLL and 8 AML) for the test of the clustering of leukemia patients. The original datasets can be found at http://research.dfci.harvard.edu/korsmeyer/MLL.htm or http://sdmc.lit.org.sg/GEDatasets/Data/MLL_Leukemia.zip. Table 2.1 describes the combined dataset with 72 patients (57 training followed by 15 test, with the same order in the original datasets) as rows and 12,582 genes as columns. The class column shows the sub types of the patients which are used as a reference result to compare with the clustering result obtained in this thesis.

Table 2.1 The Leukemia Patient Dataset

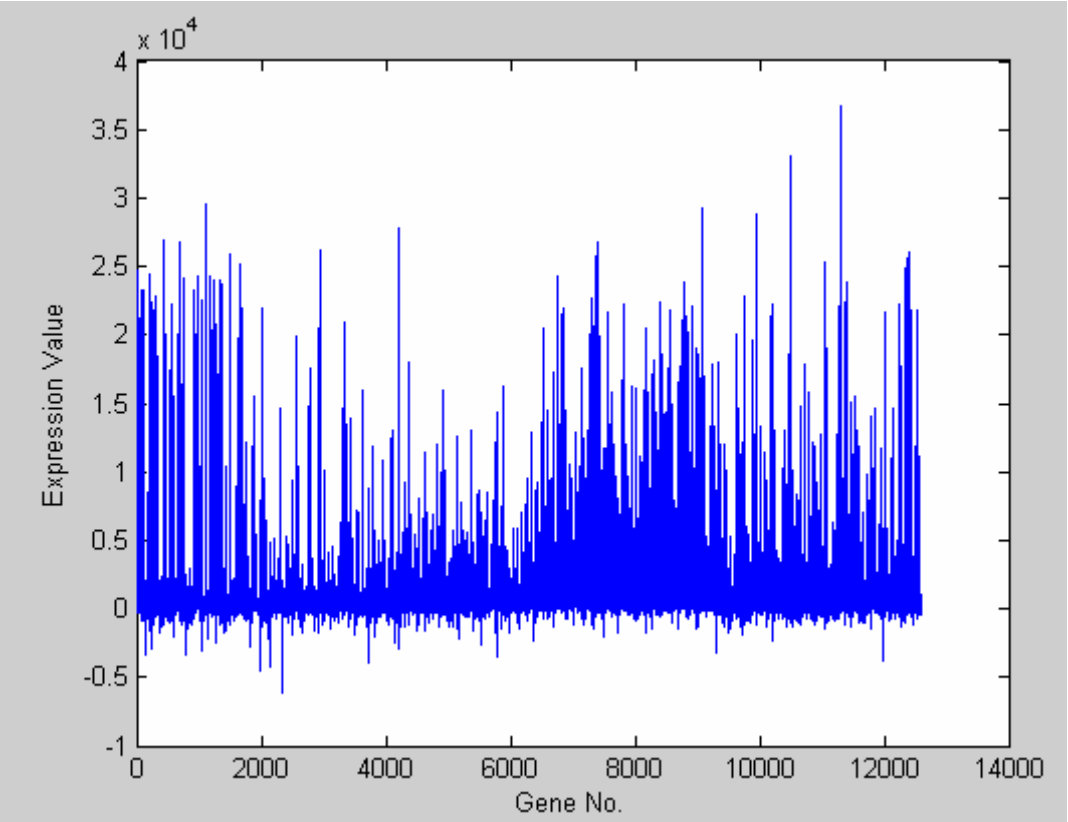
Patient No.	Gene 1	Gene 2	Gene 3	…	Gene 12,582	Class	Original Dataset	Patient No. in Original Dataset
1	-161.8	34.8	-34.4	…	1,115.5	ALL	Training	1
…	…	…	…	…	…	…	…	…
20	-170	-98	-48	…	739	ALL	Training	20
21	-76	-54	4	…	1,432	MLL	Training	21
…	…	…	…	…	…	…	…	…
37	-273	-105	59	…	1,972	MLL	Training	37
38	-336	-49	4	…	1,027	AML	Training	38
…	…	…	…	…	…	…	…	…
57	-71	6	122	…	832	AML	Training	57
58	-163	-199	-7	…	716	ALL	Test	1

…	…	…	…	…	…	…	…	…
61	-130	225	64	…	458	ALL	Test	4
62	-144	36	39	…	760	MLL	Test	5
…	…	…	…	…	…	…	…	…
64	-333	-15	7	…	2,408	MLL	Test	7
65	-53	-7	4	…	1,009	AML	Test	8
…	…	…	…	…	…	…	…	…
72	-109	166	28	…	791	AML	Test	15

(Table 2.1 Continued)

In Table 2.1, each patient is represented as one row. Column 1 is the patient number in the combined dataset, columns 2 to 6 denote the gene expression values corresponding to each patient, column 7 indicates the type of cancer (ALL, MLL or AML) that each patient is classified as in (11), column 8 specifies the original dataset (training or test) that each patient belonged to, and column 9 is the number of each patient in its original dataset. Each patient is determined by a sequence of 12,582 real numbers, each measuring the relative expression of the corresponding gene. See Figure 2.1 for the gene expression plotting of a sample patient.

Figure 2.1 Gene Expression Plotting of a Sample Patient



By exploiting the gene expression values in Table 2.1, the data set can be viewed as 72 points in a 12,582-dimensional Euclidean space. A simple measure of the genomic difference between two patients can be obtained by resorting to the Euclidean distance of two points.

In order to ease the algebraic manipulations of data, the dataset can also be represented as a real 2-D matrix S of size 72×12,582; the entry s_{ij} of S measures the expression of the j^{th} gene of the i^{th} patient.

2.2 Pre-processing of the Dataset

The leukaemia dataset is a very large matrix with more than ten thousand genes as its columns, while a great portion of them, with small changes of values between different patients, provides much less information related to the patient clustering than the rest small portion, in which large differences of values can be found between different patients or patient types. Figure 2.2 plots a

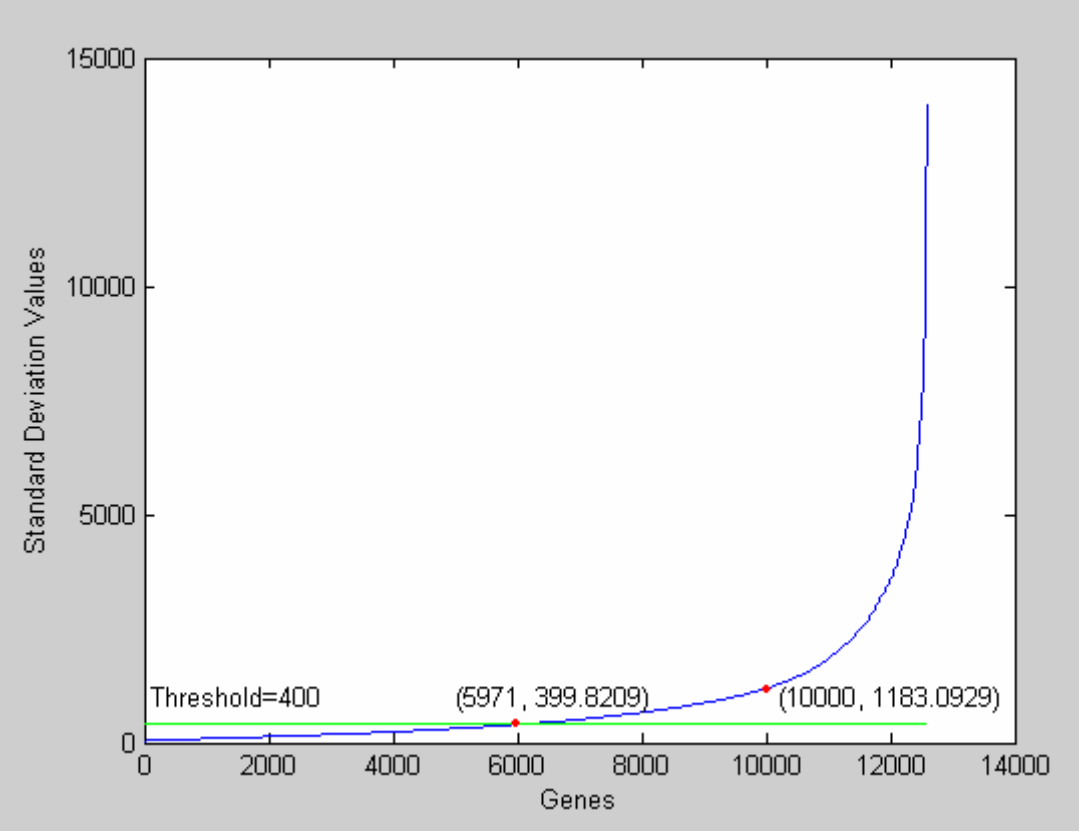
non-decreasing curve of the importance of each gene in the dataset in terms of the standard deviation value which is used here as a measurement of the degree of difference within a gene column. There are two common definitions for the standard deviation s of a data vector $X=(x_1, x_2, \ldots, x_n)$:

$$(1)\quad s=\left(\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2\right)^{\frac{1}{2}},$$

$$(2)\quad s=\left(\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2\right)^{\frac{1}{2}},$$

where $\bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i$ is the mean value of X and n is the number of elements in X. The two equations differ only in n-1 versus n in the divisor. In this thesis the standard deviation values are calculated by using equation (1).

Figure 2.2 Standard Deviation Plotting of the Dataset



In Figure 2.1 the genes are sorted according to an ascending order of their corresponding standard deviation values. It can be observed that a very large

portion of genes has relatively small standard deviation values, although the values vary from 0 to 15,000. For example, at least 10,000 values are less than 1,200, as #10,000 is with value 1183.1. Therefore, prior to the patient clustering, it is possible to apply a filter to remove those genes of little importance. On the other hand, in order to analyze such a huge dataset without any filters, a very high complexity of time and storage is inevitable, and a large amount of computational resources is required as well. The removing of less important genes can help decrease the complexity of analysis and the requirement of computational resources. Furthermore, the removing of those genes may also reduce the interference caused by noise.

By taking all these factors into account, a pre-processing of the dataset is applied first to remove those genes with small standard deviation values. A threshold 400 is used to filter out the genes with standard deviation values less than it. The dataset after this pre-processing becomes a 72×6,611 matrix with the removing of 5,971 gene columns. The reason for using 400 as the threshold is that it keeps a large portion (more than a half) of the data, so that the important information will not be ignored, and at the same time removes another large portion of data to speed up the clustering procedures. In the following chapters, unless otherwise specified, all the analysis is based on the 72×6,611 dataset after the pre-processing with threshold th = 400.

Chapter 3. Description of Algorithms

The clustering analysis of the leukemia dataset is based on three steps. First, with the principal component analysis, all the genes in the dataset are sorted according to their significance to the patient clustering. Then, the dataset is clustered using a modified bisect K-means algorithm which is essentially the combination of the principal direction divisive partitioning and the K-means. Finally, by referring to a predefined clustering result, the minimum set of genes that can produce a result with the least clustering errors is discovered. This gene set consists of a few necessary and sufficient genes in the sense of the clustering approach applied in this thesis, and the discovered genes may provide very useful information for the diagnosis of the corresponding sub types of leukemia.

3.1 Principal Component Analysis (PCA)

It is well known that the PCA method (12-14) works very well on measuring the contribution of attributes to the clustering of samples, when the dataset can be partitioned linearly. The extraction of principal components is briefly described as follows:

Given a p×N dataset S where p and N are respectively the numbers of samples and attributes. If dataset S is an unbiased matrix where each column (i.e. attribute) of S has zero mean value, then the first principal component of S should be the eigenvector corresponding to the largest eigenvalue of the covariance matrix of S, namely S^TS, the second principal component of S should be the eigenvector corresponding to the second largest eigenvalue of S^TS, and so on. A simple proof is given out in (12).

The principal components can be obtained from the singular value decomposition (SVD) (14) of S, with which matrix S is decomposed as the product of three special matrices: the orthonormal unitary square matrix $U_{P\times P}$ (i.e.

$U^{-1}=U^T$), the diagonal matrix $\Sigma_{P\times N}$, and the orthonormal unitary square matrix $V_{N\times N}$ (i.e. $V^{-1}=V^T$). Any non-zero diagonal element of matrix Σ is called a singular value of matrix S (i.e. the square root of an eigenvalue of matrix S^TS), and the columns of matrix V (i.e. the eigenvectors of S^TS) corresponding to the largest singular values are in turn the principal components of S.

When a principal component, generally the one corresponding to the largest singular value, is selected out, the degree of contribution of the attributes to the clustering of samples can be quantified by comparing the absolute values of the elements in the principal component vector. The positions of the largest absolute values point out the most discriminating attributes for the sample clustering.

When the dataset matrix S is biased, with the mean values of some attributes being non-zeros, the SVD should be performed on the unbiased form of S so as to equally weight the contribution from each attribute.

3.2 Principal Direction Divisive Partitioning (PDDP)

3.2.1 The PDDP algorithm

The PDDP algorithm is proposed by Boley (15) in 1998. It has the following steps:

(1) For the matrix S (in general S is not unbiased) in section 3.1, first calculate the mean value vector $w=[w_1, w_2, \ldots, w_N]$ for all the samples. The mean value vector is the centroid of the samples, where $w_j=\frac{1}{p}\sum_{i=1}^{p} s_{ij}$ $(1\le j\le N)$ and s_{ij} is the element in the i^{th} row and j^{th} column of S.

(2) Calculate matrix S_0, the unbiased form of S, as $S_0 = S - ew$ and $e = [1,1,\Lambda,1]^T$. Then, by the PCA analysis described in section 3.1, decompose S_0 as $S_0 = U\Sigma V$.

(3) Select an appropriate principal component $v = [v_1, v_2, \ldots, v_N]^T$ for S_0, where vector v is determined manually or automatically by the method described in section 3.3.

(4) Write matrix S as $[S_1, S_2, \ldots, S_p]^T$. If $(S_i - w)v \leq 0$, then $S_i \in S_L$, otherwise $S_i \in S_R$, where $1 \leq i \leq p$.

3.2.2 A PDDP example

A simple example is given here to make all the steps clear.

Let dataset $S = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 5 & 4 \\ 1 & 2 & 3 & 4 \end{bmatrix}$. Intuitively, since the first and third samples (rows) of S are identical, they should be clustered into the same class, and the second sample (row) of S should be clustered into another class. By applying PDDP, matrix S is first converted to its unbiased form

$$S_0 = S - ew = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 5 & 4 \\ 1 & 2 & 3 & 4 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \times [1+1+1 \;\; 2+2+2 \;\; 3+5+3 \;\; 4+4+4]/3$$

$$= \begin{bmatrix} 0 & 0 & -0.6667 & 0 \\ 0 & 0 & 1.3333 & 0 \\ 0 & 0 & -0.6667 & 0 \end{bmatrix}.$$

Then, by the singular value decomposition,

$$S_0 = U\Sigma V = \begin{bmatrix} -0.4082 & 0.8165 & -0.4082 \\ 0.8165 & 0.5266 & 0.2367 \\ -0.4082 & 0.2367 & 0.8816 \end{bmatrix} \times \begin{bmatrix} 1.6330 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Since matrix S_0 has only one singular value 1.6330 which is the first

diagonal element of Σ, its corresponding vector $v_1=\begin{bmatrix}0\\0\\1\\0\end{bmatrix}$, the first column of V, is selected as the principal component.

Finally, by calculating $S_0v_1=\begin{bmatrix}0 & 0 & -0.6667 & 0\\0 & 0 & 1.3333 & 0\\0 & 0 & -0.6667 & 0\end{bmatrix}\times\begin{bmatrix}0\\0\\1\\0\end{bmatrix}=\begin{bmatrix}-0.6667\\1.3333\\-0.6667\end{bmatrix}$, where the first and third elements of vector $\begin{bmatrix}-0.6667\\1.3333\\-0.6667\end{bmatrix}$ are less than zero while the second one is larger than zero, the first and third rows of S are clustered into $S_L=\begin{bmatrix}1 & 2 & 3 & 4\\1 & 2 & 3 & 4\end{bmatrix}$, and the second row into $S_R=\begin{bmatrix}1 & 2 & 5 & 4\end{bmatrix}$. The result is exactly the same with what is discussed before the clustering. Furthermore, by comparing the absolute values of the elements in vector v_1, we know that the third attribute (column) of dataset S, corresponding to the third elements in v_1 with the largest absolute value, is the most discriminating attribute in the sense of the sample clustering. This conclusion is consistent with what we observe directly from S. In dataset S attributes (columns) 1, 2, and 4 have identical values, thus have no ability to discriminate different samples, but attribute 3 works well.

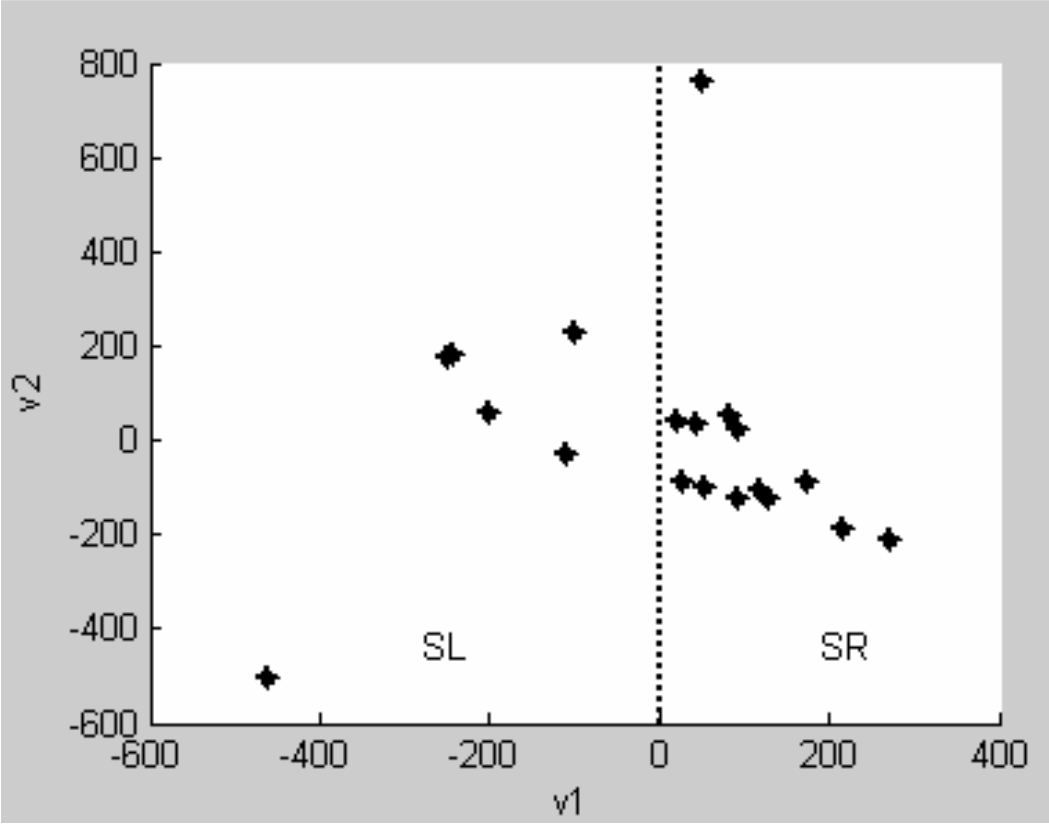
3.2.3 The geometric interpretation of PDDP

The theory of PDDP can be interpreted geometrically. The p×N dataset is first transformed to an N-dimensional coordinates system which takes the centroid of the dataset as its origin and all the N component vectors (principal or not) as N coordinates. Suppose a principal component is selected to do PDDP, then the data

points are separated as two clusters by an (N-1)-dimensional hyperplane which passes through the origin and is perpendicular to this principal component vector. Generally speaking, some distance based methods such as the minimum distance and the average distance between two different clusters can be used to measure the difference between them.

Figure 3.1 shows the projection of a dataset to the 2-D plane formed by its first two principal components, v_1 and v_2. In Figure 3.1, the PDDP clustering is performed on the basis of v_1, and v_2 is a reference principal component that is used only for the illustration purpose. All the data points on the left side of the dashed line, which is actually the projection of the hyperplane passing through the origin (i.e. the centeroid of the dataset) and perpendicular to the direction of v_1, are clustered into S_L, and all those on the right side are clustered into S_R.

Figure 3.1 The Geometrical Illustration of PDDP



It should be pointed out that PDDP can be applied repeatedly to any cluster

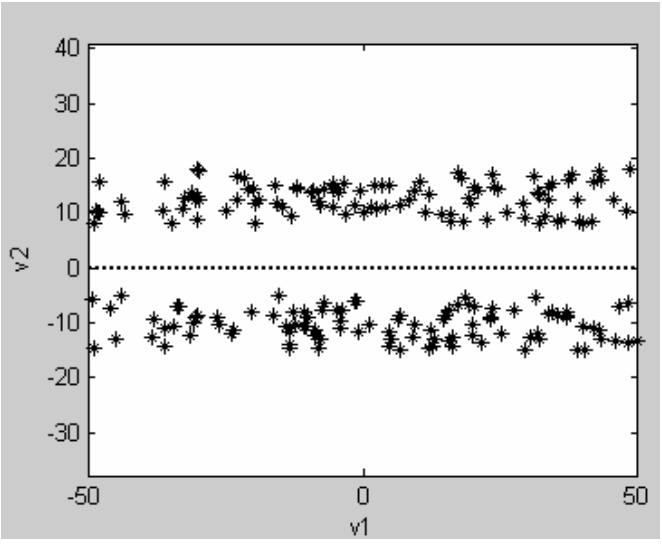
to get two sub clusters; therefore any number of clusters can be obtained by using this algorithm. Savaresi et al. (16) have proposed a method to tell which one of two given clusters is more suitable to split further, and Kruengkrai et al. (17) have described in their paper how to determine if a cluster can be split again or not.

3.3 The Selection of Principal Components

3.3.1 A problem of principal component selection

The selection of an appropriate principal component is the prerequisite of the success of PDDP clustering. In general, the first principal component is appropriate because it represents the primary direction of the dataset and the direction itself is the very foundation of the PDDP algorithm. However, the first principal component may not always be a good choice, for example when a dataset is similar to the one in Figure 3.2. In this case the primary direction of the data points is still indicated by the first principal component (shown as v_1), but obviously another principal component (shown as v_2) splits the dataset much better, therefore this principal component, even though not being the first one, should be selected as the input of the PDDP algorithm.

Figure 3.2 A Special Case of Principal Component Selection



3.3.2 The automatic selection of principal components

The selection of a principal component is easy for the supervised PDDP clustering, because we can simply find out from a set of given candidates, for example, the first three principal components, the best one that produces the result closest to the reference. However, when an unsupervised PDDP clustering algorithm is applied, the selection of an appropriate principal component should be on the automatic other than the manual basis. In (16) a method that is originally used for selecting clusters to split is also helpful for selecting principal components, just after slight modification. Following is the description of the modified algorithm.

Suppose the matrices S_0 and V have been worked out from section 3.2, and a candidate principal component set $P = \{v_1, v_2, \ldots, v_q\}$ (usually $P = \{ v_1, v_2, v_3\}$) has been given out.

(1) Write matrix S_0 as $[S_{0,1},S_{0,2},\ldots,S_{0,p}]^T$. For each principal component v_j in the given set P, calculate scalar $k_{i,j} = S_{0,i} \cdot v_j$ $(1 \le i \le p,\ 1 \le j \le q)$. If $k_{i,j} \le 0$, then $k_{i,j} \in K_{j,L}$, otherwise $k_{i,j} \in K_{j,R}$. Write $K_{j,L}$ and $K_{j,R}$ as two row vectors $K_{j,L} = [k_{j,L,1}, k_{j,L,2}, \ldots, k_{j,L,l}]$ and $K_{j,R} = [k_{j,R,1}, k_{j,R,2}, \ldots, k_{j,R,r}]$.

(2) Let $K_{j,L} = K_{j,L} / \min (K_{j,L})$ and $K_{j,R} = K_{j,R} / \max (K_{j,R})$. This normalizes $K_{j,L}$ and $K_{j,R}$ so that all their absolute values range from 0 to 1.

(3) Let scalars $w_{j,L}$and $w_{j,R}$ be the mean values of $K_{j,L}$ and $K_{j,R}$, respectively, and $w'_{j,L}$ and $w'_{j,R}$ be the mean values of $[(k_{j,L,1}-w_{j,L})^2, (k_{j,L,2}-w_{j,L})^2, \ldots, (k_{j,L,l}-w_{j,L})^2]$ and $[(k_{j,R,1}-w_{j,R})^2, (k_{j,R,2}-w_{j,R})^2, \ldots, (k_{j,R,r}-w_{j,R})^2]$, respectively.

Calculate ratio $R_j = \dfrac{w'_{j,L}+w'_{j,R}}{w_{j,L}^2 + w_{j,R}^2}$.

(4) Select the principal component with the minimum ratio R.

Other tentative methods for automatically finding out the best principal

component are presented in the appendix. See section A.2 for the MATLAB implementation of five methods including the one described above.

3.4 K-means and Bisect K-means

3.4.1 The basic K-means algorithm

K-means (18-19) is a famous iterative clustering method. The clustering is based on some randomly selected "center points". The number of random points is predefined and determines the number of clusters that the algorithm will output. The basic principle of K-means is as follows:

(1) Randomly select k points (c_1, c_2, …, c_k) from a dataset $S=[S_1,S_2,\ldots,S_p]^T$ in which S_i ($1\leq i\leq p$) denotes the i^{th} sample. These k random points are viewed as the initial "center points" of k clusters and refined later.

(2) For each sample S_i ($1\leq i\leq p$), find out a number m, so that for any $j\neq m$ ($1\leq m, j\leq k$), $||S_i-c_m|| \leq ||S_i-c_j||$, then $S_i \in C_m$, where $||S_i-c_m||$ and $||S_i-c_j||$ are respectively the distances, for example the Euclidean distances, from S_i to c_m and c_j, and C_m denotes the m^{th} cluster.

(3) Calculate the new center points i.e. the mean values w_1, w_2, …, w_k for the clusters C_1, C_2, …, C_k.

(4) If for each cluster j ($1\leq j\leq k$), $c_j=w_j$, then stop; otherwise let $c_j=w_j$ for each j, and go to step (2).

K-means algorithm is iteratively convergent, and, if the initial "center points" are selected well, that is to say, they are close to the true center points, then K-means will converge more rapidly, and the clustering result will be more accurate. However, it may not be easy to select good initial center points if one does not know in advance what the distribution of the data points is. This is the reason why to take random points as the initial centers. On the other hand, to

apply K-means, the total number of clusters must be determined prior to the clustering.

3.4.2 The bisect K-means algorithm

One kind of K-means, which can be repeatedly applied to form multiple clusters by separating one cluster at a time to get two sub clusters, is called bisect K-means. Similarly, bisect K-means algorithm has the following steps:

(1) Randomly select two "center points", c_1 and c_2, from the dataset $S=[S_1,S_2,\ldots,S_p]^T$.

(2) If $||S_i-c_1||\leq||S_i-c_2||$, then $S_i \in C_1$; otherwise $S_i \in C_2$, $(1\leq i\leq p)$, where $||S_i-c_1||$ and $||S_i-c_2||$ are the distances, for example the Euclidean distances, from S_i to c_1 and c_2, respectively, and C_1 and C_2 denote the two sub clusters.

(3) Calculates the new center points w_1 and w_2 for the two sub clusters C_1 and C_2.

(4) If $c_1=w_1$ and $c_2=w_2$, then stop; otherwise let $c_1=w_1$ and $c_2=w_2$, and go to step (2).

To get more sub clusters, one can select a cluster, replace dataset S with it, and simply repeat the above steps. Such a procedure can be repeated until a desired number of clusters is obtained.

3.5 Combining PDDP with Bisect K-means

3.5.1 The weakness of K-means

K-means algorithm performs well when the distance information between data points is important to the clustering. However, K-means has an intrinsic disadvantage. The clustering result depends greatly on the selection of initial "center points". Cited from (18), Figures 3.4 and 3.5 show the different results of

applying K-means on the same dataset (see Figure 3.3) but with different choices of initial “center points”.

Figure 3.3 The Dataset of the K-means Example

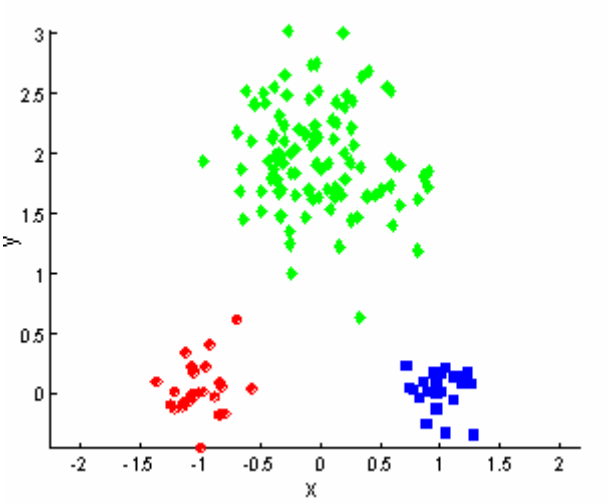


Figure 3.4 K-means Result from Initial “Center Point” Set 1

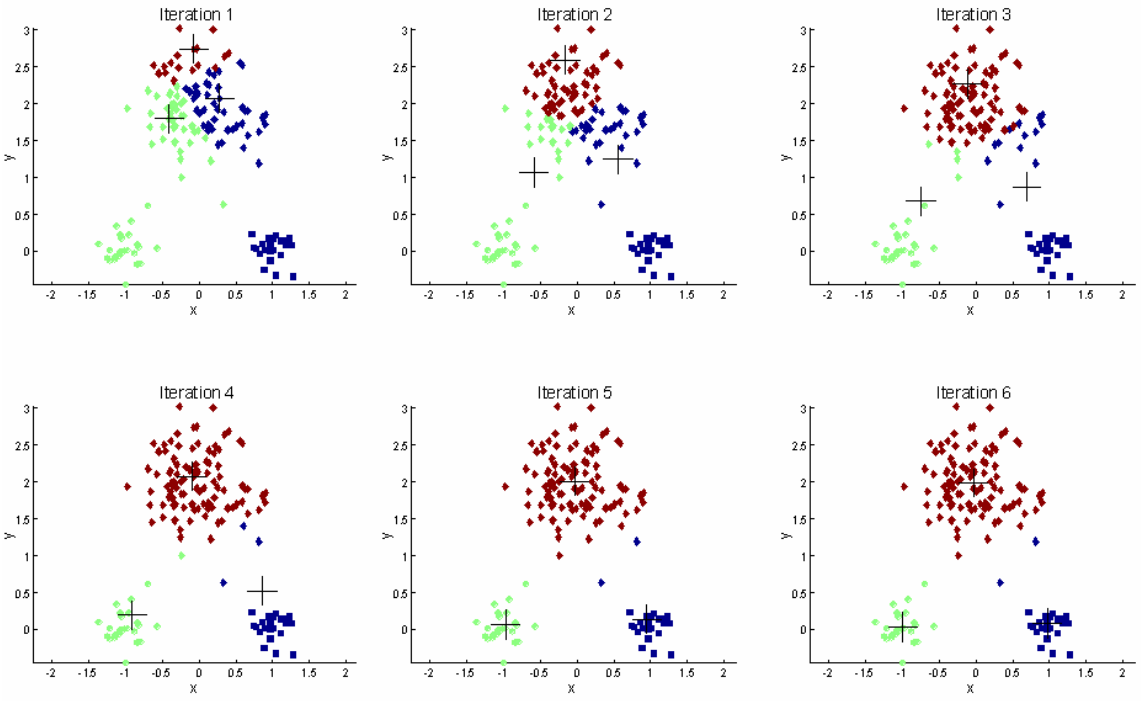
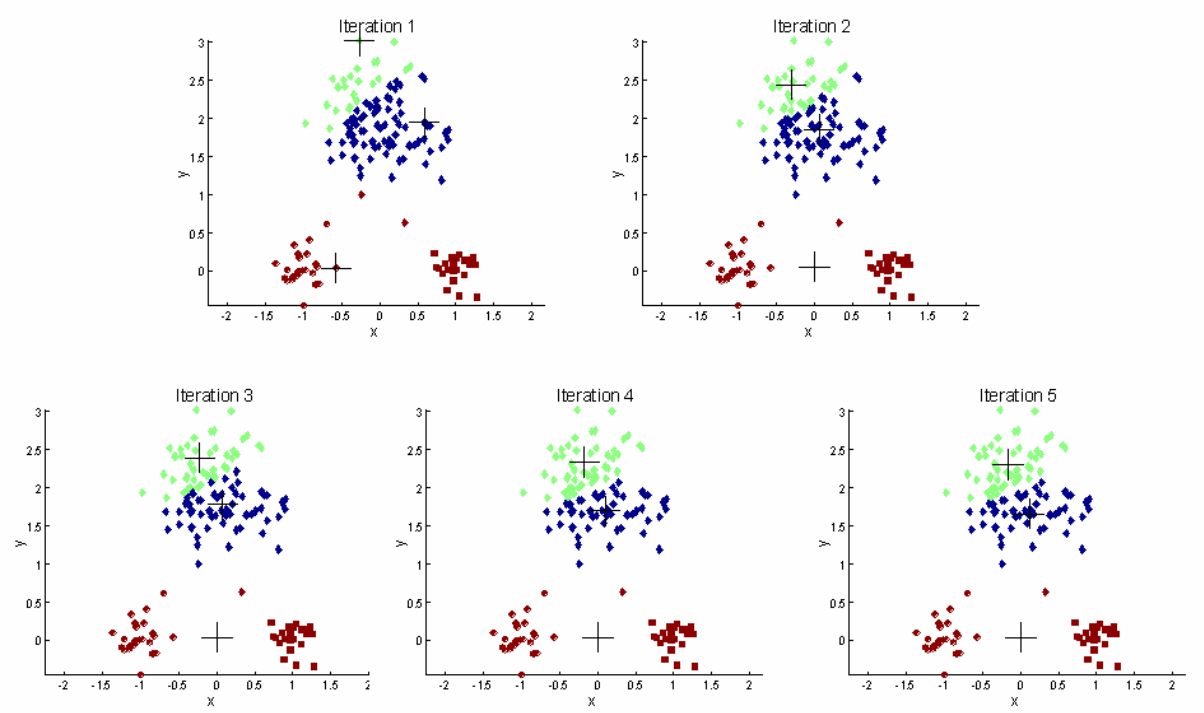


Figure 3.5 K-means Result from Initial "Center Point" Set 2

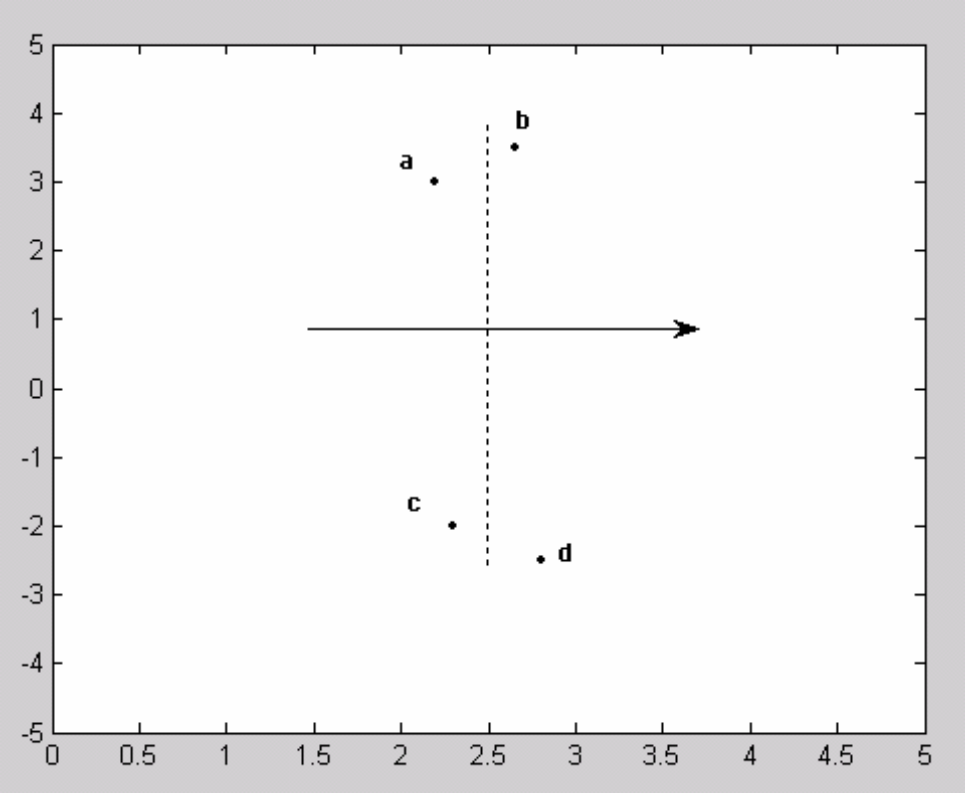


In Figure 3.3, the original dataset consists of three distinct clusters colored as red, green, and blue, respectively. In iteration 1 of Figure 3.4, where three initial center points are assigned and represented as three crosses, each data point is clustered according to the closet center point, and the data points that so far belong to the same cluster are rendered the same color. Iterations 2 to 6 illustrate the changing of center points and clusters, and, finally in iteration 6, neither the center points nor the clusters change any more. Similar iterations are illustrated in Figure 3.5, except that the selection of initial center points is different. Comparing iteration 6 of Figure 3.4 with iteration 5 of Figure 3.5, we see that the former converges to an excellent clustering result which is consistent with the one expected in the original dataset, while the latter does not produce a good result by cutting the green cluster in Figure 3.3 into two parts and merging the red and the blue into one. The great difference of final results in Figures 3.4 and 3.5 is merely caused by the selection of different initial center points.

3.5.2 The weakness of PDDP

PDDP has its own weakness, too. Since the partition of PDDP is only on the basis of the projection from the data points to a selected principal direction, the distance information between these data points is ignored. Figure 3.6 shows an example of such neglect.

Figure 3.6 Neglect of PDDP to the Distance Information



In Figure 3.6, suppose the line with an arrow indicates the selected principal direction, the dashed line is the projection of the hyperplane passing through the origin and perpendicular to the principal direction, and a, b, c, and d are four data points. By applying PDDP, points a, c are clustered into the left class, and points b, d into the right class. However, one may notice that, when the distances between points are considered, a result which clusters a, b into one class and c, d into another class also makes sense, since b is much closer to a than c is, and similarly, c is much closer to d than b is.

3.5.3 The merit of PDDP + bisect K-means

In spite of the fact that in many cases neither PDDP nor K-means alone is good enough for deriving desirable clustering results, according to the theory of Savaresi and Boley etc. (20-22), the combination of PDDP and bisect K-means keeps the merits of both algorithms, and usually performs better than either single one does. PDDP, although is weak at taking advantage of distance information, can provide bisect K-means good initial center points that are close to true ones, therefore the accuracy of bisect K-means clustering can be improved. The difference between the combined method and the traditional bisect K-means lies in the selection of the initial center points, c_1 and c_2. With the combined method, the two center points of bisect k-means are not selected randomly but according to the clustering result of PDDP, that is to say, c_1 and c_2 should be the sample mean values of the PDDP clusters S_L and S_R, respectively. The combination of PDDP and bisect K-means makes the selection of c_1 and c_2 more reasonable by reducing the risk caused by a random selection. See Figure 3.7 (a-d) for an example. The data of this example come from another leukemia dataset (1, 23) with 72 patients and two sub types of leukemia, ALL and AML. The detailed analysis of this dataset can be found in (23).

Figure 3.7 The Merit of PDDP + Bisect K-means

Figure 3.7 (a)

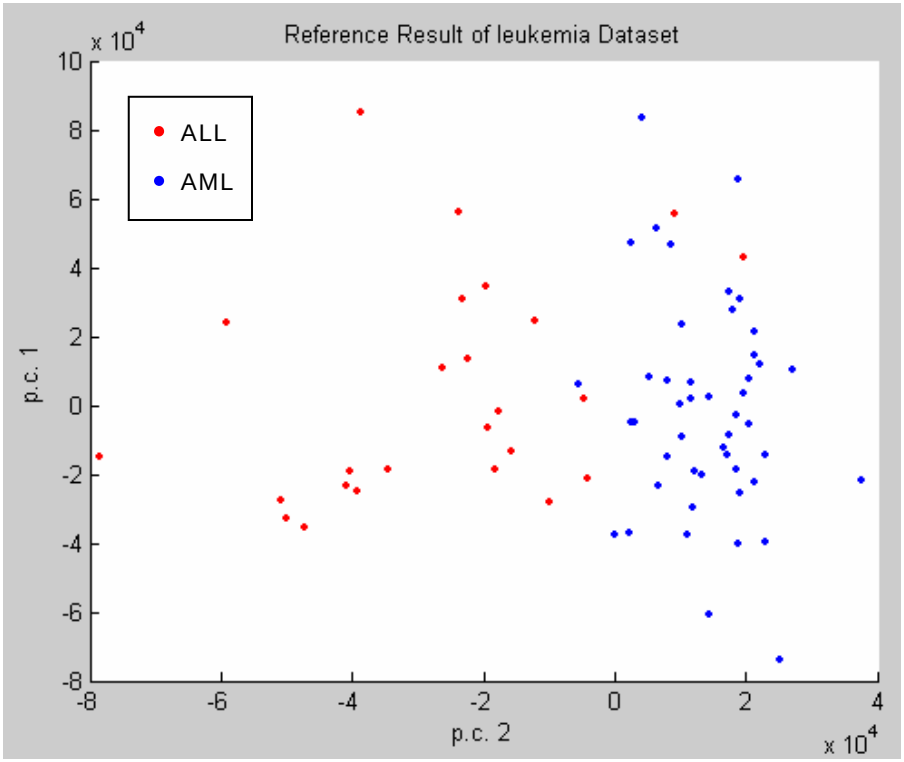


Figure 3.7 (b)

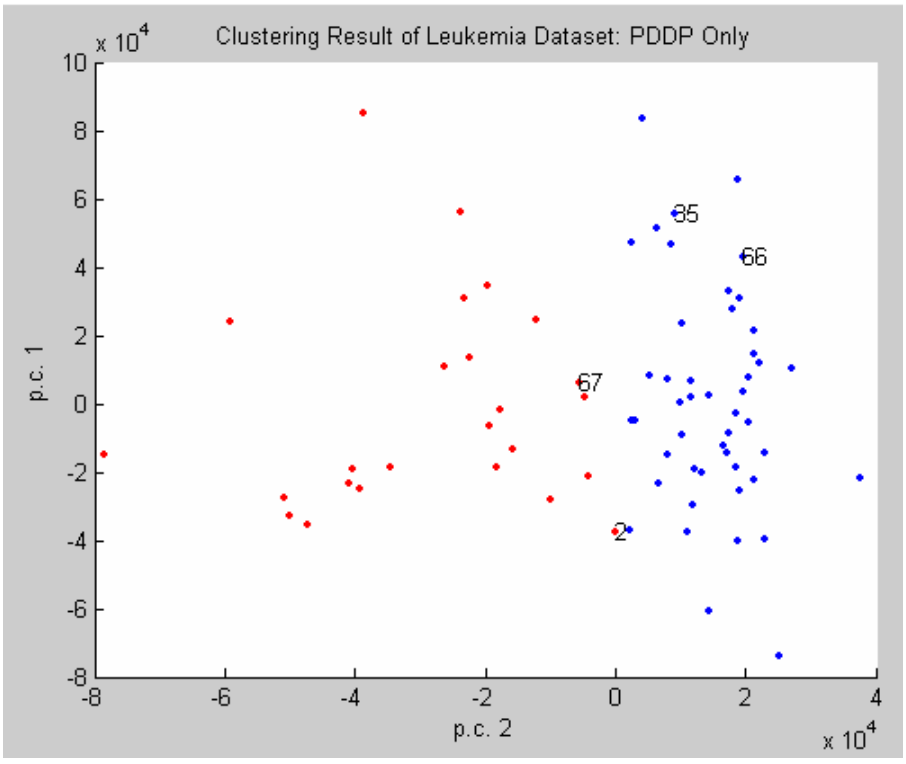


Figure 3.7 (c)

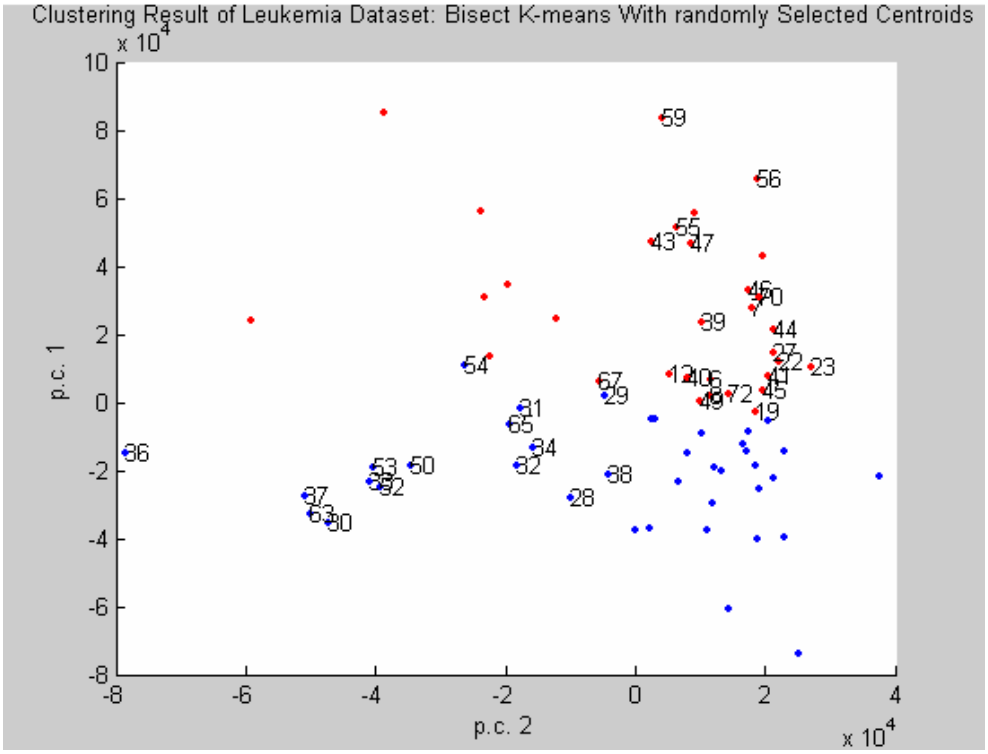
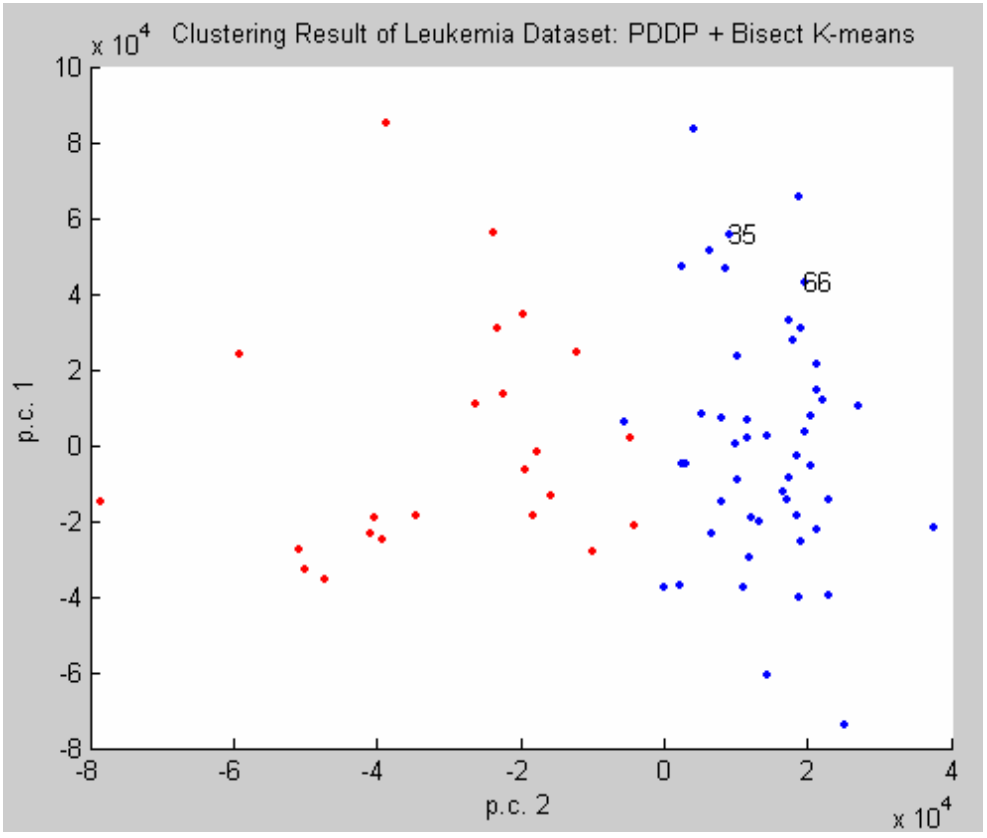


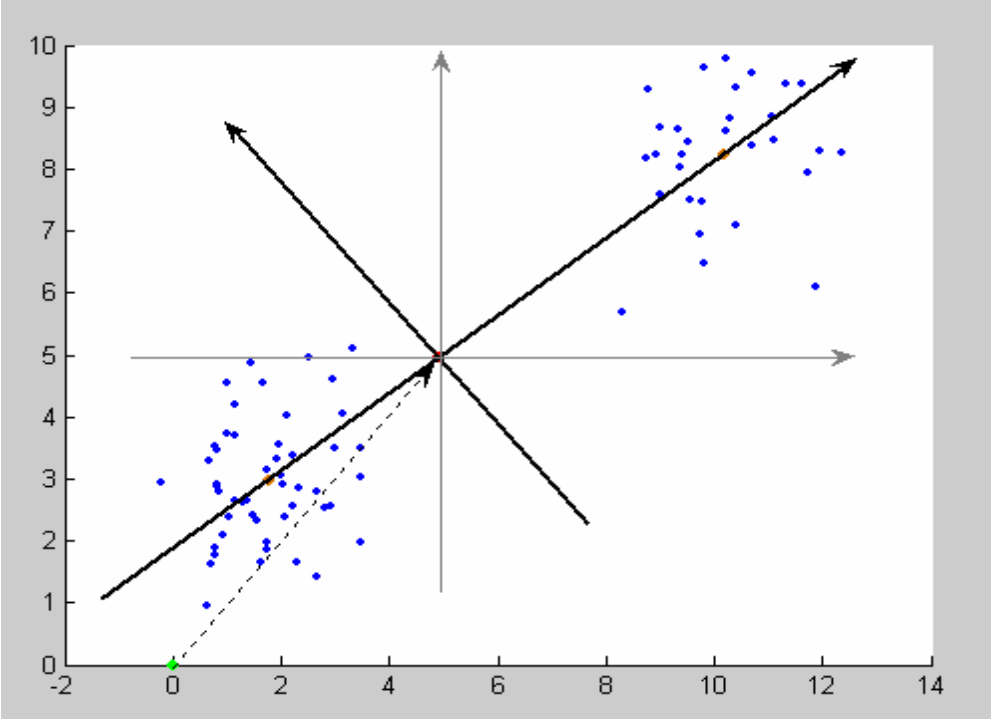
Figure 3.7 (d)



3.5.4 An illustration of PDDP + bisect K-means

Figure 3.8 is a 2-D illustration of the PDDP plus bisect K-means algorithm. In the figure, suppose a 2-D dataset is clustered using the combined method, the data points are represented as blue dots, and their origin is the green dot with coordinates (0, 0). First, by PCA analysis, the origin is moved to the centroid of the dataset (shown as a red dot) along the direction indicated by the dashed arrow, and a principal component is selected with its direction indicated by the black arrow which passes through the new origin and two orange dots. Then, by PDDP, the dataset is separated by another black arrow which passes through the new origin and is perpendicular to the principal direction. The two black arrows actually compose the two coordinates of the new coordinates system. Finally, after PDDP, the centroids of both clusters (shown as two orange dots) are selected as the initial center points of bisect K-means, and the dataset is clustered based on this selection.

Figure 3.8 A 2-D Illustration of PDDP + Bisect K-means



3.6 The Extraction of Significant Attributes

As having been mentioned at the beginning of this thesis, the extraction of the significant attributes that are strongly related to clustering is also a very important issue, besides the clustering itself. To achieve this, one should first know the degree of significance of each attribute. Fortunately, principal component analysis itself can also provide quantitative information to measure the significance. Following is a method of extracting the most significant attributes based on PCA analysis:

(1) Suppose vector $v_j=[v_{1j},v_{2j},\ldots,v_{Nj}]^T$ is the j-th principal component of S_0 (i.e. column j of V where $S_0=U\sum V$) and v_j is selected to do PDDP. Sort vector v_j in a descending order of $|v_{ij}|$ $(1\le i\le N)$ and write it as $v'_j=[v'_{1j},v'_{2j},\ldots,v'_{Nj}]^T$. Since the significance of each attribute is reflected by the absolute value of the corresponding element in the principal component, now v'_{1j} is the significance coefficient of the most important attribute,v'_{2j} is that of the second most attribute, and so on.

(2) Redo the PDDP + bisect K-means clustering using the reduced principal component $u_m=[v'_{1j},v'_{2j},\ldots,v'_{mj},0,0,\ldots,0]^T$ $((1\le m\le N)$, and find out the minimum value of m that outputs the best clustering result that is the closest to a reference result, then the m corresponding attributes are the solution.

For example, if we have a principal component $v=[2.5\ -3.0\ 1.2\ 4.1]^T$, then after the sorting, $v'=[4.1\ -3.0\ 2.5\ 1.2]^T$. Now we try to use $u_1=[4.1\ 0\ 0\ 0]^T$, $u_2=[4.1\ -3.0\ 0\ 0]^T$, $u_3=[4.1\ -3.0\ 2.5\ 0]^T$, and $u_4=v'=[4.1\ -3.0\ 2.5\ 1.2]^T$ to do PDDP, respectively, and compare the results with a reference. Suppose u_3 and u_4 get exactly the same result with the reference, while u_2 gets one error and u_1 gets two, then u_3 with m = 3 is selected, and attributes 4, 2, and 1, which correspond to the three largest absolute values of coefficients, 4.1, 3.0, and 2.5, should be the

minimum attribute set in the sense of clustering.

3.7 Supervised and Unsupervised Clustering

With a supervised clustering approach, some a priori knowledge such as a pre-defined reference result and the number of clusters can be used to guide the process of clustering. However, such a priori knowledge is not always available before clustering; they may be known only when the clustering is successfully completed. In this case, an unsupervised alternative can be considered when applicable. The PDDP + bisect K-means algorithm is capable of dividing data points into two clusters in either supervised or unsupervised way, as described in the following procedures:

3.7.1 Procedure PCA

Procedure PCA

Input: p×N data matrix S.

Output: sorted principal component vector v and index vector x.

Begin

Calculate the unbiased matrix S_0 of S;

Do singular value decomposition with S_0 and get the principal components;

Select a principal component manually or automatically;

Sort its elements in the descending order of their absolute values, and get the index of each attribute corresponding to the order;

Return v (the sorted principal component vector) and x (the index vector);

End

3.7.2 Procedure PDDP_Bisect_K-means_Unsupervised

Procedure PDDP_Bisect_K-means_Unsupervised

Input: matrix S, vector v (output of procedure PCA), and vector x (output of procedure PCA).

Output: two clusters S_L and S_R and the significant attribute set A

Begin

Use matrix S and vector v to do PDDP + Bisect K-means clustering, and get two clusters S_L and S_R;

For (i <- 1 to N-1)

v_i <- v;

Set the last N-i elements in v_i to 0;

Use S and v_i to do PDDP + Bisect K-means, and get two clusters S_{Li} and S_{Ri};

If ((S_{Li}=S_L) *and* (S_{Ri}=S_R))

Break;

End If

End For

A <- the first i indices in x;

Return S_L, S_R, and A;

End

3.7.3 Procedure PDDP_Bisect_K-means_Supervised

Procedure PDDP_Bisect_K-means_Supervised

Input: matrix S, vector v (output of procedure PCA), vector x (output of procedure PCA), and vector c as the reference result of clustering.

Output: two clusters S_L and S_R and the significant attribute set A.

Begin

Get two clusters S_{Lc} and S_{Rc} from matrix S and reference result c;

err <- p;

m <- 0;

For (i <- 1 to N)

v_i <- v;

Set the last N-i elements in v_i to 0;

Use S and v_i to do PDDP + Bisect K-means, get two clusters S_{Li} and S_{Ri} and the clustering result c_i;

Calculate err_i, the number of differences between c and c_i;

If (err_i < err)

err <- err_i;

m <- i;

End If

End For

S_L <- S_{Lm};

S_R <- S_{Rm};

A <- the first m indices in x;

Return S_L, S_R, and A;

End

Chapter 4. Experimental Results

This chapter is focused on some experimental results about the clustering of the leukemia gene expression dataset described previously. The original dataset S consists of 72 samples (24 ALL, 20 MLL and 28 AML patients) and each sample is represented by 12,582 gene expression values. Dataset S is stored as a 72×12,583 matrix, because there is an extra column, column 12,583, which represents the clustering result presented in (11). In this column, classes ALL, MLL, and AML are represented as 0, 1, and 2, respectively. This column serves as the reference result of all the following experiments. In other words, the experiment results are compared with the reference, and any different clustering cases are reported as "errors" and analyzed later. Before any experiments, a threshold th = 400 is applied to remove those genes with standard deviation values less than 400, since they are with little possibility to be significant attributes. To verify the effectiveness of the threshold, every experiment is then repeated with th = 0 i.e. all the genes included. The exactly same results and much less execution time show that the threshold applied is reasonable and effective. All the experiments are based on the MATLAB implementation of the algorithms described in chapter 3. See the appendix for the MATLAB source code.

4.1 The Unsupervised Clustering of Dataset S

With threshold th = 400, the input dataset S becomes a 72×6,611 matrix. See Figure 2.2 for the standard deviation plotting of all the 12,582 genes. With the first principal component and all the 6,611 genes, a clustering result is shown in Figure 4.1. According to the reference result (see Figure 4.2), 21 "errors" are shown in Figure 4.1 as points with patient numbers, and in Table 4.1 as cells with gray shadings. Note that almost all the 21 "errors" (except #3) are classified as MLL in Figure 4.2, implying that the PDDP + Bisect K-means approach correctly

identified 23 of 24 ALL and 28 of 28 AML patients.

Figure 4.1 The Unsupervised Clustering Result of Dataset S

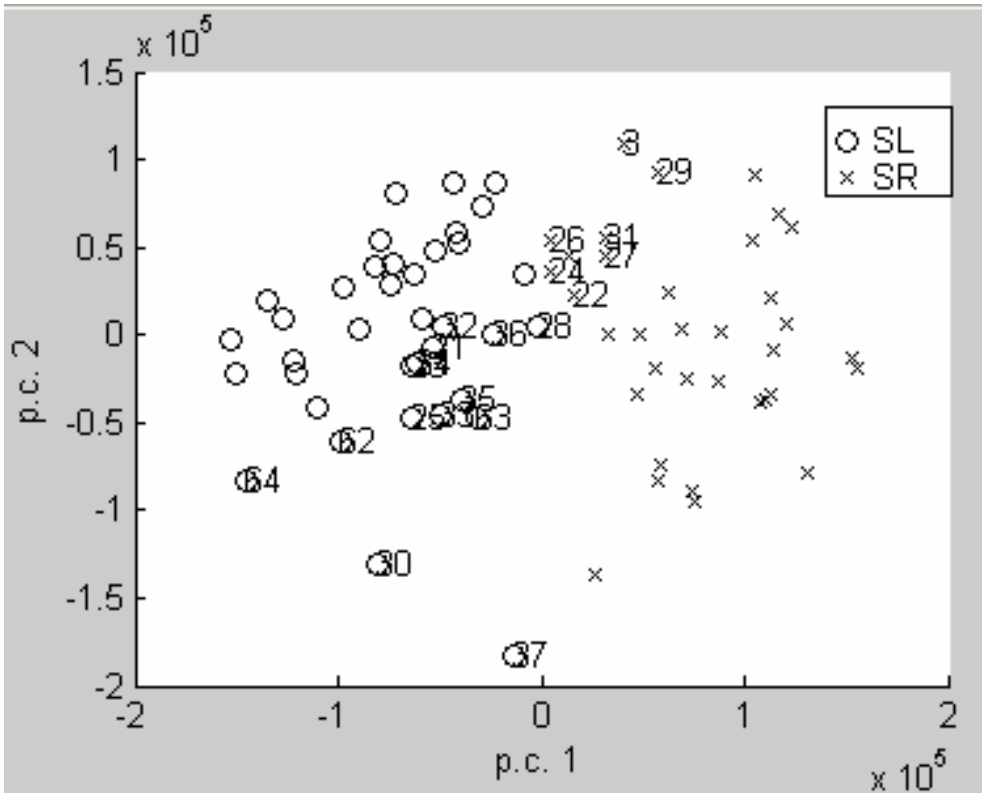


Figure 4.2 The Reference Result of Dataset S

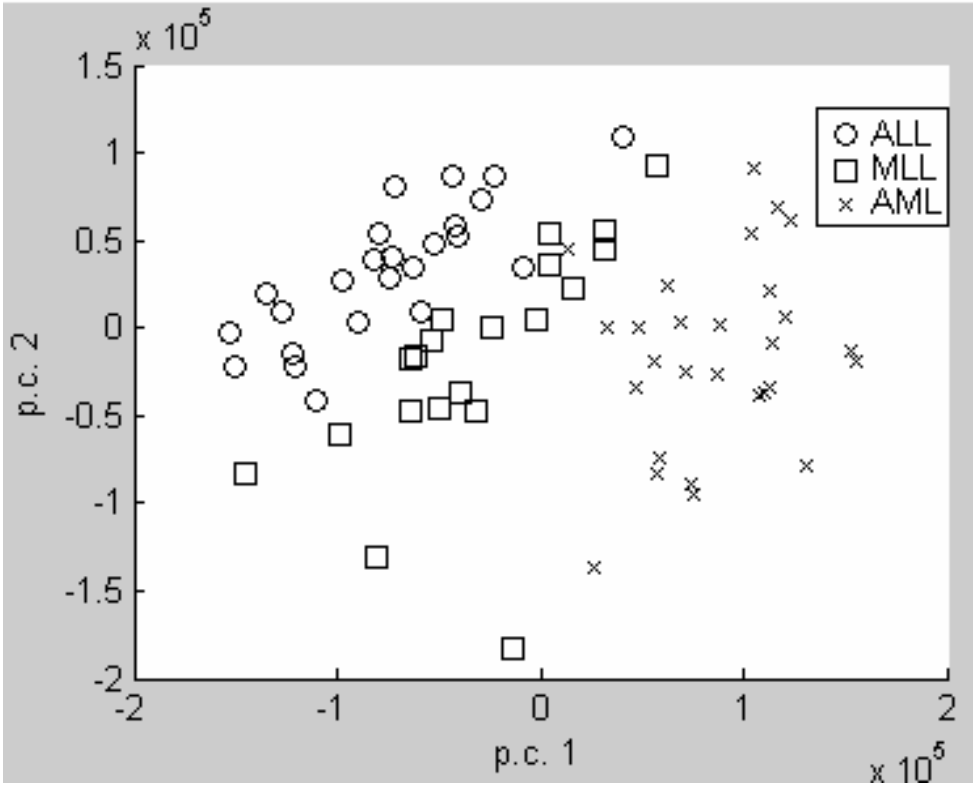


Table 4.1 The Unsupervised Clustering Result of Dataset S

	Patient Numbers																			
S_L	1	2	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
S_R	3	22	24	26	27	29	31	38	39	40	41	42	43	44	45	46	47	48	49	50
	Patient Numbers																			
S_L	23	25	28	30	32	33	34	35	36	37	58	59	60	61	62	63	64			
S_R	51	52	53	54	55	56	57	65	66	67	68	69	70	71	72					

The minimum gene set that produces the above result consists of only two genes: #28 (the index in the original 12,582-attribute dataset) with the name AFFX-HUMGAPDH/M33197_5_at and #12,430 with the name 256_s_at. Table 4.2 gives out the significance coefficient information about these two genes. The significance coefficients are obtained by taking the absolute values of the corresponding elements in the first principal component, the average coefficient is the mean of the absolute values of all the 6,611 coefficients, and the normalized coefficients, which are used as the contribution indicator of the genes to the clustering, are the quotients of the significance coefficients and the average coefficient.

Table 4.2 The Significant Genes for the Clustering of Dataset S

Gene #	Gene Name	Significance Coefficient	Average Coefficient	Normalized Coefficient
28	AFFX-HUMGAPDH/M33197_5_at	0.1113	0.0073	15.2466
12,430	256_s_at	0.0984		13.4795

From Figures 4.3 and 4.4, the plotting of the 72 expression values of these two genes, we can visually separate S_L (with relatively low expression values) and S_R (with relatively high expression values) to a certain extent, although a few

exceptional cases exist. The rationale of the extraction of these two genes is thus illustrated in such a manner.

Figure 4.3 The Expression Values of Gene #28

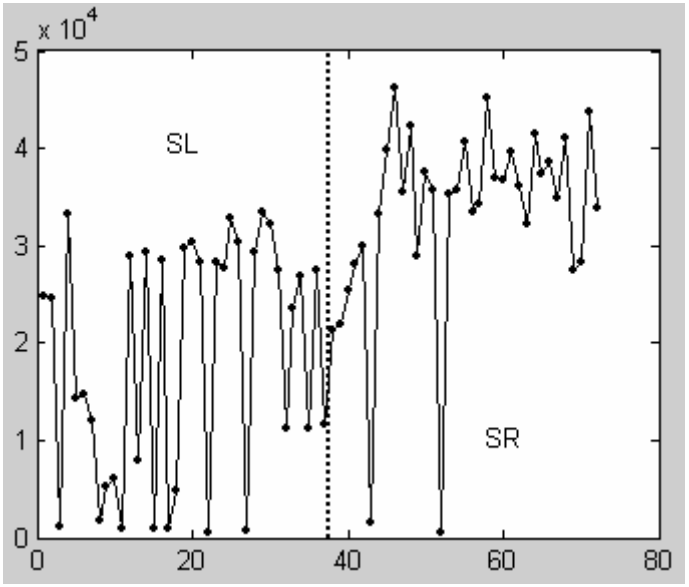
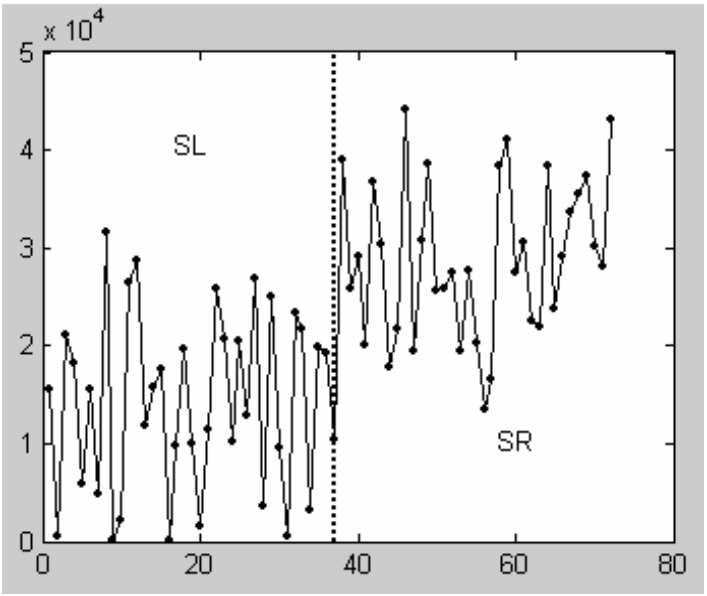


Figure 4.4 The Expression Values of Gene #12,430



It is natural that the initial clustering does not give out any useful information about the MLL samples, because the PDDP based approach only produces two clusters after a single application. For this reason, further clustering is needed to hopefully reveal the aspect of the MLL part.

4.2 The Unsupervised Clustering of Sub Dataset S_L

According to the result of the initial clustering, 37 samples are classified as S_L; among them 23 are actually ALL samples and 14 are MLL. In order to see if the PDDP based approach can successfully identify these ALL samples from the non ALL ones (i.e. the MLL ones, according to the reference result), the clustering of the subclass S_L is continued. With the first principal component, 5,962 genes (threshold th = 400), and two significant genes, a result that is exactly the same with the reference is obtained, as shown in Figures 4.5 and 4.6. Table 4.3 lists the patient numbers and the subclasses that they belong to, where S_{LL} and S_{LR} are actually ALL and a part of MLL, respectively. Table 4.4 gives out the two significant genes and quantifies their contribution to the clustering. Figures 4.7 and 4.8 plot the 37 expression values of these two genes.

Figure 4.5 The Unsupervised Clustering Result of Sub Dataset S_L

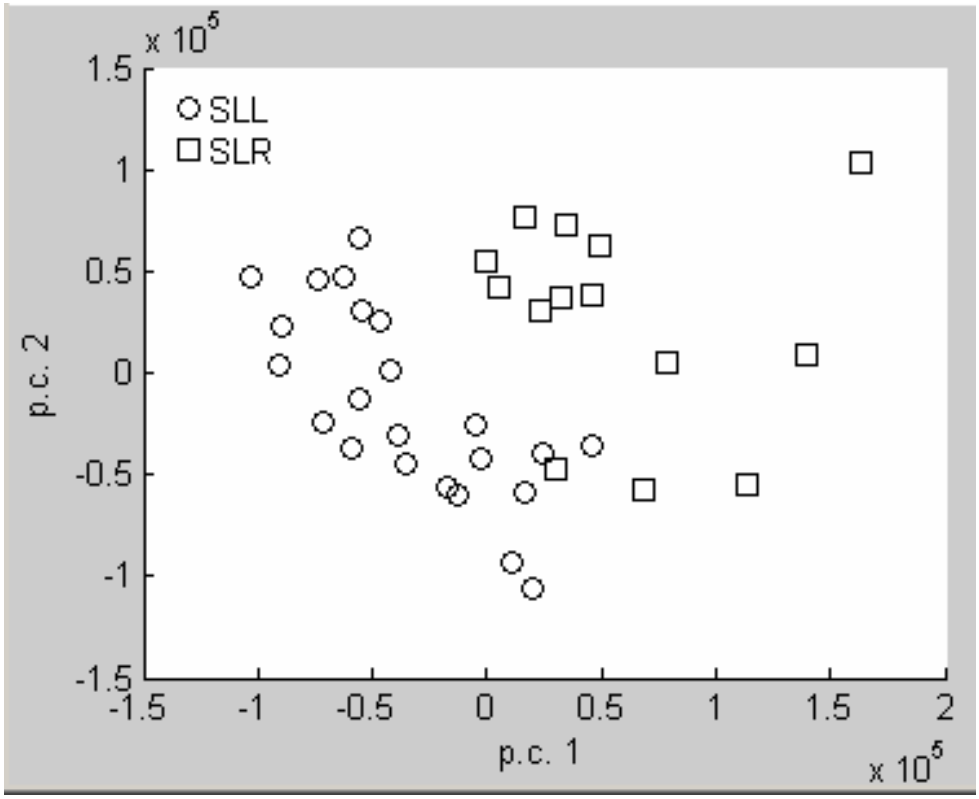


Figure 4.6 The Reference Result of Sub Dataset S_L

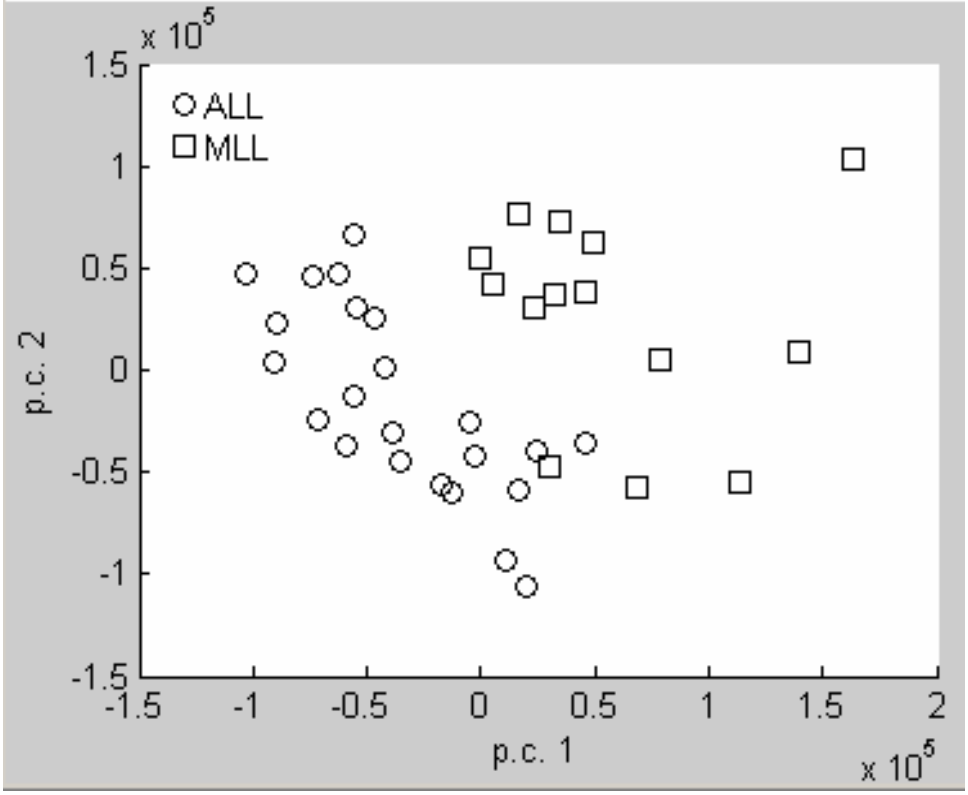


Table 4.3 The Unsupervised Clustering Result of Sub Dataset S_L

	Patient Numbers																			
S_{LL}	1	2	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	58
S_{LR}	21	23	25	28	30	32	33	34	35	36	37	62	63	64						
	Patient Numbers																			
S_{LL}	59	60	61																	
S_{LR}																				

Table 4.4 The Significant Genes for the Clustering of Sub Dataset S_L

Gene #	Gene Name	Significance Coefficient	Average Coefficient	Normalized Coefficient
7,754	33412_at	0.1533	0.0072	21.2917
11,924	769_s_at	0.1083		15.0472

Figure 4.7 The Expression Values of Gene #7,754

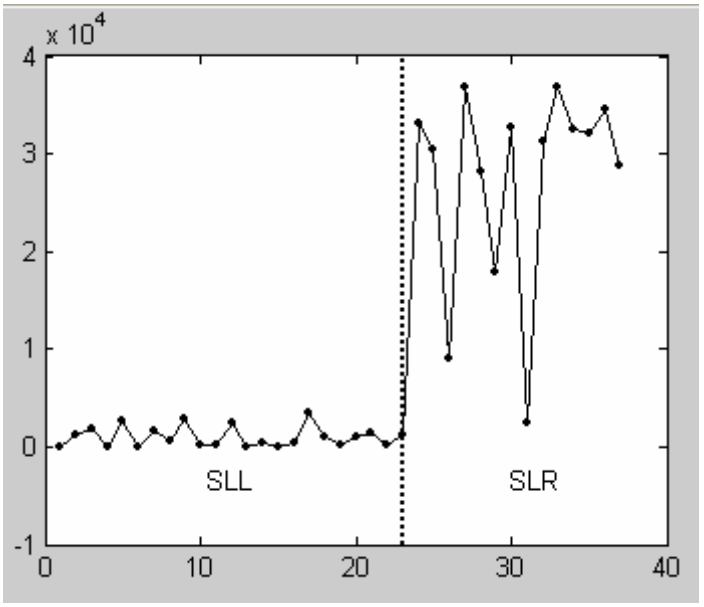
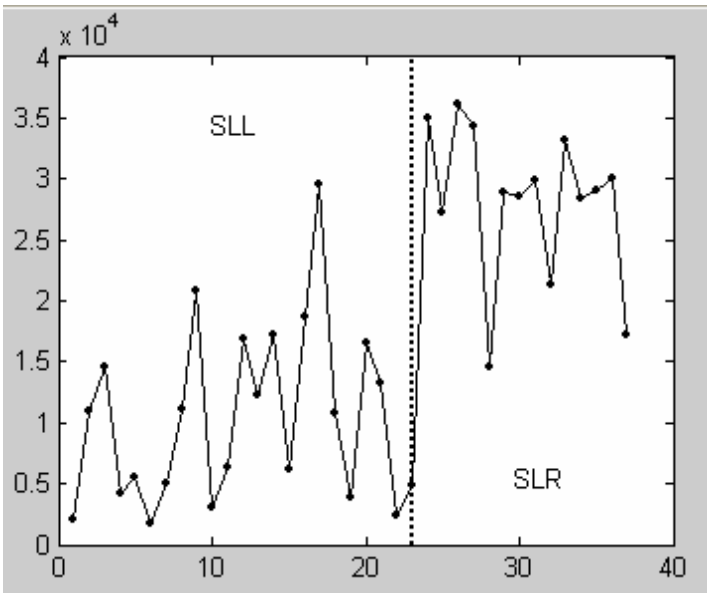


Figure 4.8 The Expression Values of Gene #11,924



4.3 The Unsupervised Clustering of Sub Dataset S_R

Since the initial clustering of dataset S is not adequate for identifying the MLL samples, a similar clustering of the subclass S_R is then performed to see if those MLL samples can be separated successfully.

According to the result of the initial clustering, 35 samples are classified as S_R. Among them are 28 AML, 6 MLL, and one misclassified ALL. With the first

principal component and 6,191 genes (threshold th = 400), the result is shown in Figure 4.9. See Figure 4.10 for the reference result. The minimum gene set with the clustering result in Figure 4.9 consists of 219 genes; they are not reported here. The clustering seems not to be successful, with many AML samples and all the MLL samples clustered together into S_{RL}. However, an interesting observation is that no MLL sample is clustered into S_{RR}. Table 4.4 lists the patient numbers with their sub clusters, where the MLL patients are shown with grey shadings.

Figure 4.9 The Unsupervised Clustering Result of Sub Dataset S_R

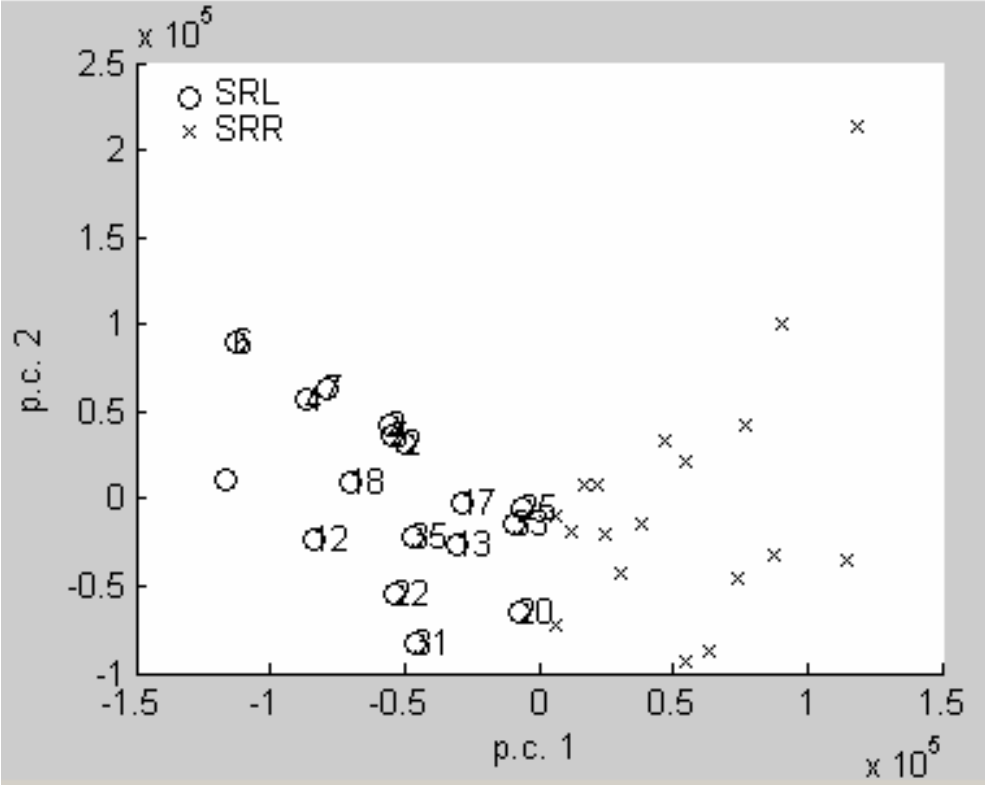


Figure 4.10 The Reference Result of Sub Dataset S_R

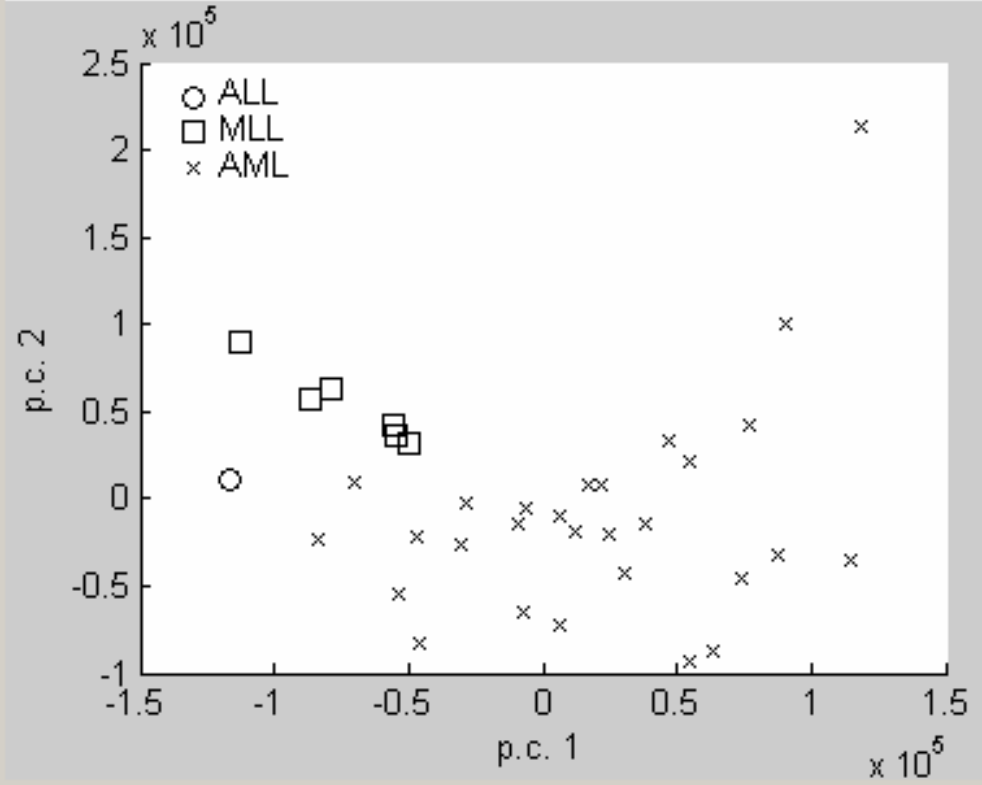


Table 4.5 The Unsupervised Clustering Result of Sub Dataset S_R

	Patient Numbers																	
S_{RL}	3	22	24	26	27	29	31	42	43	47	48	50	52	55	68	70	72	
S_{RR}	38	39	40	41	44	45	46	49	51	53	54	56	57	65	66	67	69	71

4.4 The Supervised Clustering of Sub Dataset S_{RL}

Because all the 6 MLL samples are classified as S_{RL} in section 4.3, it may be interesting to continue clustering the sub cluster S_{RL}. With the first principal component and 5,877 genes (threshold th = 400), an unsupervised result with two errors is obtained and shown in Figure 4.11. Figure 4.12 is the reference result. The minimum gene set of the result in Figure 4.11 consists of 103 genes which are not reported here. However, when the clustering is performed under the supervision of the reference result, a better clustering result is obtained with only one error at patient #3, as shown in Figure 4.13. Table 4.6 lists the patient numbers

and their sub clusters according to this supervised clustering, where the MLL patients are shown with gray shadings. The minimum gene set for this result consists of 9 genes. They are listed in Table 4.7.

Figure 4.11 The Unsupervised Clustering Result of Sub Dataset S_{RL}

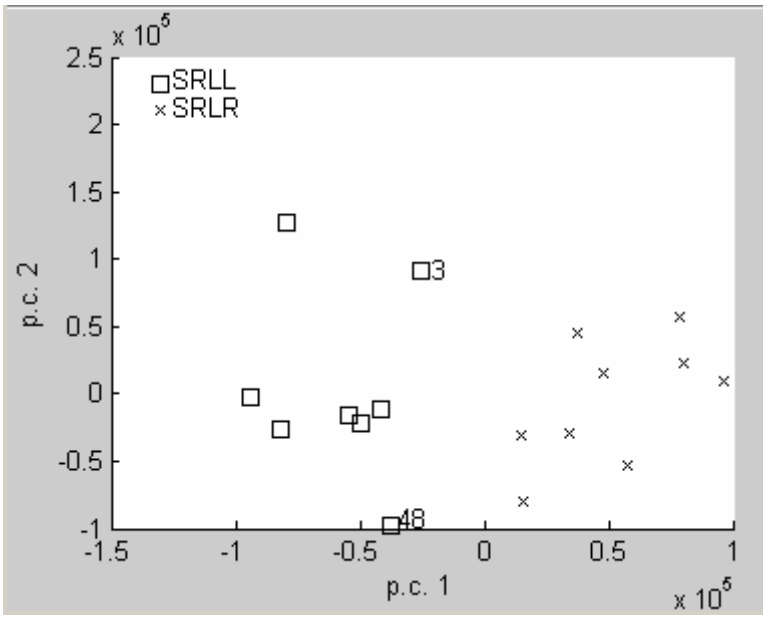


Figure 4.12 The Reference Result of Sub Dataset S_{RL}

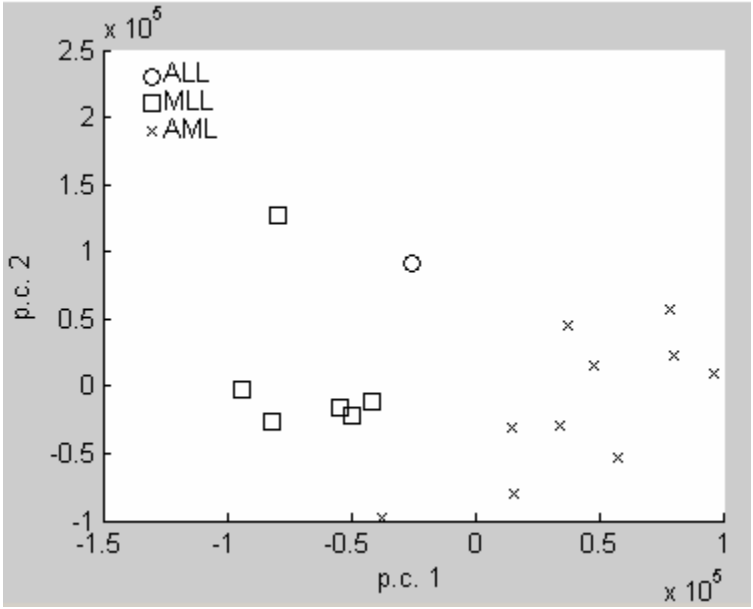


Figure 4.13 The Supervised Clustering Result of Sub Dataset S_{RL}

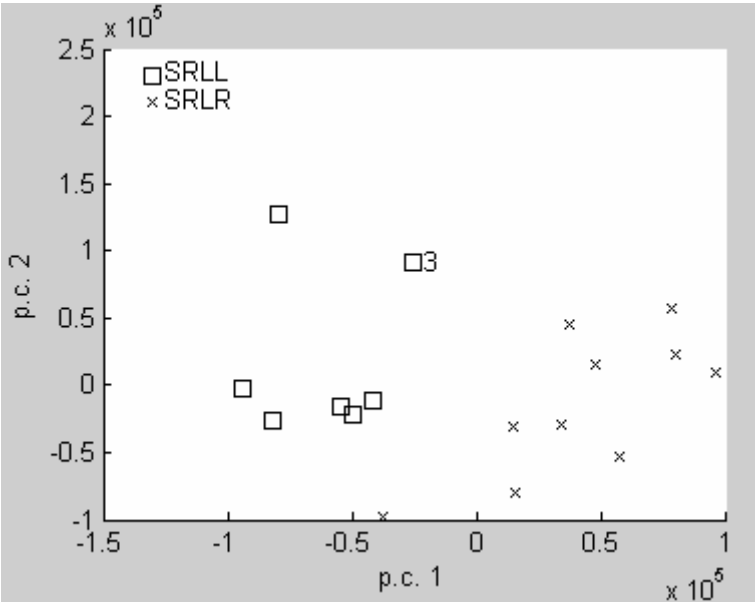


Table 4.6 The Supervised Clustering Result of Sub Dataset S_{RL}

	Patient Numbers									
S_{RLL}	3	22	24	26	27	29	31			
S_{RLR}	42	43	47	48	50	52	55	68	70	72

Table 4.7 The Significant Genes for the Clustering of Sub Dataset S_{RL}

Gene #	Gene Name	Significance Coefficient	Average Coefficient	Normalized Coefficient
12,357	319_g_at	0.1106	0.0080	13.8250
31	AFFX-HSAC07/X00351_5_at	0.1106		13.8250
32	AFFX-HSAC07/X00351_M_at	0.0995		12.4375
7,754	33412_at	0.0993		12.4125
1,904	33516_at	0.0989		12.3625
28	AFFX-HUMGAPDH/M33197_5_at	0.0985		12.3125
1,316	35083_at	0.0950		11.8750
8,428	36122_at	0.0940		11.7500
3,634	39318_at	0.0933		11.6625

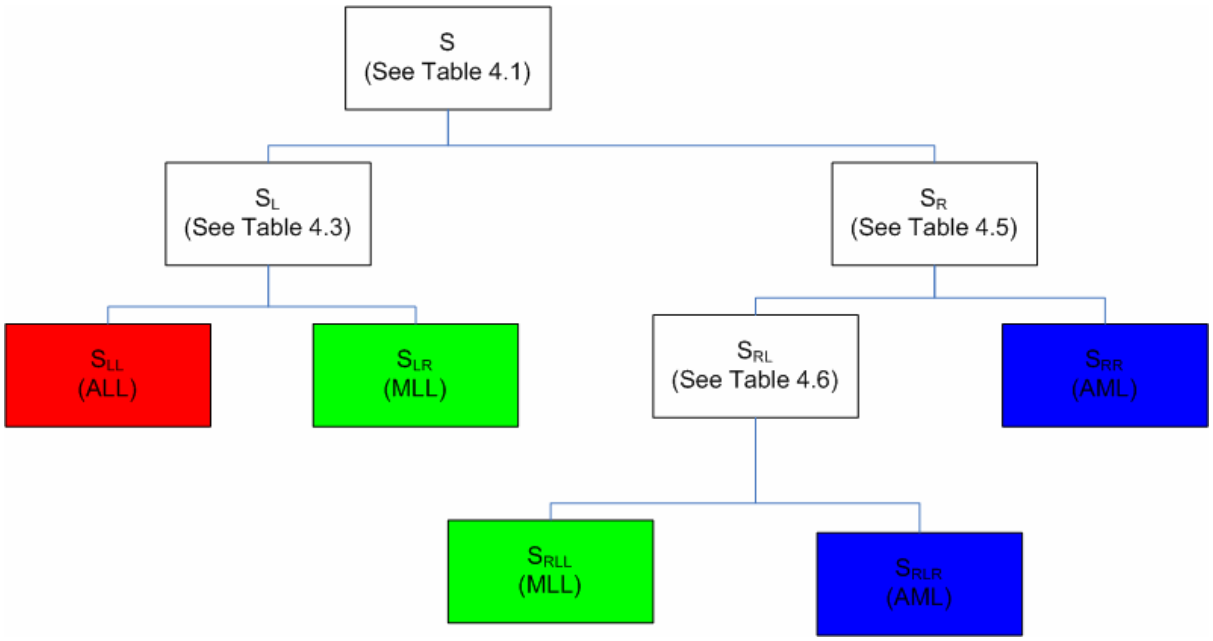
Chapter 5. Discussion and Conclusion

5.1 Discussion about the Experimental Results

5.1.1 Discussion about the clustering results

According to the clustering results in chapter 4, the leukemia dataset S can be clustered to the following hierarchy:

Figure 5.1 The Hierarchy of the Leukemia Dataset



In Figure 5.1, if we name cluster S_{LL} is as ALL, clusters S_{LR} and S_{RLL} together as MLL, and clusters S_{RLR} and S_{RR} together as MLL, then there is only one error occurs with such a name conversion. By Table 4.3, almost all the 24 ALL patients are identified in cluster S_{LL}, except that patient #3 is eventually misclassified into cluster S_{RLL}; this is the only error that occurs. By tables 4.3 and 4.6, 14 MLL patients are identified in cluster S_{LR} and other 6 are identified in S_{RLL}; these two clusters include all the MLL patients with on error. By tables 4.5 and 4.6, 18 AML patients are identified in cluster S_{RR} and other 10 are identified in cluster S_{RLR}; these two clusters include all the AML patients with no error.

It should be noted from the hierarchy that, except ALL, both MLL and AML

patients are divided into two sub clusters. This implies that there might exist other sub types for MLL and AML, although in (1) only two sub types of leukemia (ALL and AML) and in (11) three sub types (ALL, MLL, and AML) are proposed.

5.1.2 Discussion about the significant genes

First, by reviewing the gene extraction results in chapter 4, we see that the different levels of expression values of gene #28 (AFFX-HUMGAPDH/M33197_5_at) and #12,430 (256_s_at) separate well ALL and AML patients. Second, in the initial clustering of the dataset, most MLL patients are classified into the ALL part; this means that MLL and ALL share similarity to a great extent. The difference between ALL and MLL is discovered very well by gene #7,754 (33412_at) and #11,924 (769_s_at). On the other hand, a small portion of MLL patients are classified into the AML part, showing that some MLL and AML cases have common characteristics. The size of the minimum set of genes which separates MLL from AML is very large, implying that the clinical diagnosis of AML-like MLL patients may be more difficult than that of the ALL-like MLL patients. Finally, the contribution of genes to the corresponding clustering results is quantified so that the significance of them can be compared quantitatively. For examples, gene #28 (normalized significance coefficient (NSC) = 15.2466) and #12,430 (NSC = 13.4795) have basically equal significance to the discrimination between ALL and AML, gene #7,754 (NSC = 21.2917) has greater significance than #11,924 (NSC = 15.0472) to the discrimination between MLL and ALL, and so on.

5.2 Conclusion

With the combined approach of PDDP and bisect K-means, the 72 leukemia patients are successfully clustered as ALL, MLL and AML, respectively. Among all the 12,582 genes, the most discriminating a few ones that are responsible for

the clustering are efficiently discovered at the same time. Furthermore, both the clustering of the patients and the discovering of the significant genes are performed automatically to a great extent, and depend merely on the gene expression data which can be obtained conveniently by using the popular DNA micro array technology.

In conclusion, the combination of PDDP and bisect K-means is an efficient approach for the clustering of the leukemia patient dataset described in this thesis, and hopefully also efficient for other similar problems. Moreover, the significant genes discovered among tens of thousands of genes may provide very important information for the diagnosis of the disease of leukemia.

References

[1] Golub, T.R., D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield and E.S. Lander: "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring". Science, 286:531-537, October 1999.

[2] Bittanti S., Garatti S. and Liberati D.: “From DNA micro-arrays to disease classification: an unsupervised clustering approach”. 15th IFAC World Congress, Prague, Czech Republic, 2005.

[3] van't Veer LJ, Dai HY, van de Vijver MJ, He YDD, Hart AAM, Mao M, Peterse HL, van der Kooy K, Marton MJ, Witteveen AT, Schreiber GJ, Kerkhoven RM, Roberts C, Linsley PS, Bernards R and Friend SH: "Gene Expression Profiling Predicts Clinical Outcome of Breast Cancer". Letters to Nature, Nature, 415:530-536, 2002.

[4] Pomeroy SL, Tamayo P, Gaasenbeek M, Sturla LM, Angelo M, McLaughlin ME, Kim JY, Goumnerova LC, Black PM, Lau C, Allen JC, Zagzag D, Olson JM, Curran T, Wetmore C, Biegel JA, Poggio T, Mukherjee S, Rifkin R, Califano A, Stolovitzky G, Louis DN, Mesirov JP, Lander ES and Golub TR: "Prediction of Central Nervous System Embryonal Tumour Outcome Based on Gene Expression". Letters to Nature, Nature, 415:436-442, January 2002.

[5] Alon U, Barkai N, Notterman DA, Gish K, Ybarra S, Mack D and Levine AJ: "Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays". Proceedings of National Academy of Sciences of the United States of American, 96:6745-6750, 1999.

[6] De Cecco L, Marchionni L, Gariboldi M, Reid JF, Lagonigro MS, Caramuta S, Ferrario C, Bussani E, Mezzanzanica D, Turatti F, Delia D, Daidone MG, Oggionni M, Bertuletti N, Ditto A, Raspagliesi F, Pilotti S, Pierotti MA, Canevari S, and Schneider C: "Gene expression profiling of advanced ovarian cancer: characterization of a molecular signature involving fibroblast growth factor 2 ". Oncogene, 23(49):8171-8183, October, 2004.

[7] Dinesh Singh, Phillip G. Febbo, Kenneth Ross, Donald G. Jackson, Judith Manola, Christine Ladd, Pablo Tamayo, Andrew A. Renshaw, Anthony V. D'Amico, Jerome P. Richie, Eric S. Lander, Massimo Loda, Philip W. Kantoff, Todd R. Golub and William R. Sellers: "Gene Expression Correlates of Clinical Prostate Cancer Behavior". Cancer Cell, 1:203-209, March, 2002.

[8] Eng-Juh Yeoh, Mary E. Ross, Sheila A. Shurtleff, W. Kent Williams, Divyen Patel, Rami Mahfouz, Fred G. Behm, Susana C. Raimondi, Mary V. Relling, Anami Patel, Cheng Cheng, Dario Campana, Dawn Wilkins, Xiaodong Zhou, Jinyan Li, Huiqing Liu, Ching-Hon Pui, William E. Evans, Clayton Naeve, Limsoon Wong and James R. Downing: "Classification, Subtype Discovery, and Prediction of Outcome in Pediatric Acute Lymphoblastic Leukemia by Gene Expression Profiling". Cancer Cell, 1:133-143, March, 2002.

[9] Gavin J. Gordon, Roderick V. Jensen, Li-Li Hsiao, Steven R. Gullans, Joshua E. Blumenstock, Sridhar Ramaswamy, William G. Richards, David J. Sugarbaker and Raphael Bueno: "Translation of Microarray Data into Clinically Relevant Cancer Diagnostic Tests Using Gege Expression Ratios in Lung Cancer And Mesothelioma". Cancer Research, 62:4963-4967, 2002.

[10] Alizadeh AA, Eisen MB, Davis RE, Ma C, Lossos IS, Rosenwald A, Boldrick JC, Sabet H, Tran T, Yu X, Powell JI, Yang L, Marti GE, Moore T, Hudson J Jr, Lu L, Lewis DB, Tibshirani R, Sherlock G, Chan WC, Greiner TC, Weisenburger DD, Armitage JO, Warnke R, Levy R, Wilson W, Grever MR, Byrd JC, Botstein D, Brown PO and Staudt LM: "Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling". Nature, 403:503-511, February 2000.

[11] Scott A. Armstrong, Jane E. Staunton, Lewis B. Silverman, Rob Pieters, Monique L. den Boer, Mark D. Minden, Stephen E. Sallan, Eric S. Lander, Todd R. Golub and Stanley J. Korsmeyer: "MLL Translocations Specify A Distinct Gene Expression Profile that Distinguishes A Unique Leukemia". Nature Genetics, 30:41-47, January 2002.

[12] Hand, D., H. Mannila, P. Smyth (2001): “Principles of Data-Mining”. The MIT press, Cambridge, Massachusetts, USA.

[13] O'Connel M.J.: “Search Program for Significant Variables”. Computer Physics Communications, 1974. 8: p. 49-55.

[14] Wall ME, Rechtsteiner A and Rocha LM: “Singular value decomposition and principal component analysis". A Practical Approach to Microarray Data Analysis. (Berrar DP, Dubitzky W, Granzow M, eds.), pp. 91-109, Kluwer:Norwell, MA (2003).

[15] Boley, D.L. (1998): "Principal Direction Divisive Partitioning". Data Mining and Knowledge Discovery, 2(4), 325-344.

[16] Savaresi, S., Boley, D., Bittanti, S. and Gazzaniga, G. (2002): “Choosing the cluster to split in bisecting divisive clustering algorithms”. Second SIAM International Conference on Data Mining (SDM'2002)

[17] Canasai Kruengkrai, Virach Sornlertlamvanich and Hitoshi Isahara: “Refining A Divisive Partitioning Algorithm for Unsupervised Clustering”. The 3rd International Conference on Hybrid Intelligent Systems (HIS'03), December 14-17, 2003

[18] Pang-ning Tan, Michael Steinbach and Vipin Kumar: “Introduction to Data Mining”, Addison Wesley Publishing Company, 2005.

[19] J. MacQueen: “Some methods for classification and analysis of multivariate observations”. L. M. LeCam and J. Neyman, editors, Proceedings Fifth Berkeley Symposium on Math. Stat. and Prob., pages 281--297. University of California Press, 1967.

[20] Savaresi, S.M. and D.L. Boley (2001): "On the performance of bisecting K-means and PDDP". 1st SIAM Conference on Data Mining, Chicago, IL, USA, paper n.5, pp.1-14.

[21] Savaresi, S.M., D.L. Boley, S. Bittanti and G. Gazzaniga (2002): "Cluster selection in divisive clustering algorithms". 2nd SIAM International Conference on Data Mining, Arlington, VI, USA, pp.299-314.

[22] Savaresi, S.M. and D.L. Boley (2004): “A Comparative Analysis on the Bisecting K-Means and the PDDP Clustering Algorithms”. International Journal on Intelligent Data Analysis, 8(4), pp. 345-362.

[23] Diego Liberati, Simone Garatti, Zhiyu Zhao, Marco Pappalettera and Sergio Bittanti (2005): “Classification of Leukaemia via Micro-Arrays Data Analysis”, submitted to IEEE Trans. on BME.

Appendix: MATLAB Implementation of the Algorithms

A.1 MATLAB Code for PCA

File name: PCA.m

```
% The Principal Component Analysis based on the Singular Value Decomposition.
%
% Usage: [X, INDEX1, U, Z, V]=PCA (s, th);
% Input
% s: the original dataset (each row is a sample and each column is an attribute.)
% th: the threshold. Some “unimportant” attributes will be removed by applying the threshold. If th is 0, no threshold will be applied, and all the attributes kept.
% Output
% X: the dataset after the threshold (columns with standard deviation values less than th have been removed).
% INDEX1: the positions of the columns in s with standard deviation values >= th.
%U, Z, and V: the result matrices of the singular value decomposition of X.
function [X, INDEX1, U, Z, V] = PCA(s, th)
global S OBJ_NUM VAR_NUM X INDEX1 V TH % Declaration of global variables.
OBJ_NUM=size(s,1); % OBJ_NUM <- number of samples
VAR_NUM=size(s,2); % VAR_NUM <- number of attributes
S=s; clear s; % S <- s
TH=th; % TH <- th
```

INDEX1=find(std(S)>=TH); % INDEX1 <- positions of the columns with standard deviation values >= TH

X=[S(:,INDEX1)]; % X <- columns of S with their indices in INDEX1

X=X-ones(size(X,1),1)*mean(X); % X <- the unbiased form of X

[U,Z,V]=svd(X); % Uses the Singular Value Decomposition to decompose X as the product of U, Z, and V.

A.2 MATLAB Code for Find_PC

File name: Find_PC.m

```
% On the basis of the specified method, automatically find out a principal component from the first Num ones.
%
% Usage: pc = Find_PC(Num, Method);
% Input
% Num: the Number of principal components that will be checked. For example, if Num is 3, then p.c.1 to p.c. 3 will be checked.
% Method: the method that will be used. Five methods are available.
% Output
% pc: the found principal component. For example, if pc = 1, then the first principal component is found out.
function pc = Find_PC (Num, Method)
global X V % Declaration of global variables.
switch (Method)
    case 1 % Method 1: the recommended method. See section 3.3.
        pc=0; temp=inf;
        for i=1:Num
            K=X*V(:,i); KL=K(K<=0); KR=K(K>0);
```

```
            KL=KL/min(KL); KR=KR/max(KR);
            wL=mean(KL); wR=mean(KR);
            r=(mean((KL-wL).^2)+mean((KR-wR).^2))/(wL^2+wR^2);
            if (temp > r)
                pc=i; temp=r;
            end
        end
    case 2 % Method 2.
        pc=0; temp=inf;
        for i=1:Num
            K=X*V(:,i); KL=X(K<=0,:); KR=X(K>0,:);
            cL=mean(KL); cR=mean(KR);
            rL=mean(sum((KL-ones(size(KL,1),1)*cL).^2,2));
           rR=mean(sum((KR-ones(size(KR,1),1)*cR).^2,2));
            r=sqrt((rL+rR))/norm(cL-cR);
            if (temp > r)
                pc=i; temp=r;
            end
        end
    case 3 % Method 3.
        pc=0; temp=inf;
        for i=1:Num
            K=X*V(:,i); KL=K(K<=0); KR=K(K>0);
            KL=KL/min(KL); KR=KR/max(KR);
            wL=mean(KL); wR=mean(KR);
           r=(mean(abs(KL-wL))/abs(wL)+mean(abs(KR-wR)))/abs(wR);
```

```
            if (temp > r)
                pc=i; temp=r;
            end
        end
    case 4 % Method 4.
        pc=0; temp=0;
        for i=1:Num
            K=X*V(:,i);
            cL=mean(K(K<=0)); cR=mean(K(K>0));
            r=cR-cL;
            if (temp < r)
                pc=i; temp=r;
            end
        end
    case 5 % Method 5.
        pc=0; temp=0;
        for i=1:Num
            K=X*V(:,i);
            r=min(K(K>0))-max(K(K<=0));
            if (temp < r)
                pc=i; temp=r;
            end
        end
end
```

A.3 MATLAB Code for PDDP

File name: PDDP.m

```
% The Principal Direction Divisive Partitioning based on the Principal Component Analysis.
%
% Usage: [PC, COEFF, INDEX2, K, CLS_PDDP, XL, XR, wL, wR] = PDDP (pc, i);
% Input
% pc: the position of a specific principal component. If pc is 1, the first principal component will be used for PDDP; if pc is 2, the second will be used, and so on. If pc is 0, a principal component will be automatically selected.
% i: the amount of attributes that will be used by PDDP. For example, if i is 5, then the first 5 significant attributes (according to the selected p.c.) will be used. If i is 0, all the attributes will be used.
% Output
% PC: the position of the principal component used for PDDP.
% COEFF: the significance coefficients of the selected attributes.
% INDEX2: the positions of the significant attributes.
% K: the projection vector of the samples against the principal direction.
% CLS_PDDP: the clustering result of PDDP.
% XL and XR: the sub datasets after the PDDP clustering.
% wL and wR: the center points of XL and XR, respectively.
function [PC, COEFF, INDEX2, K, CLS_PDDP, XL, XR, wL, wR] = PDDP (pc, i)
global S X V INDEX2 PC I wL wR % Declaration of global variables.
if (pc>0)
    PC=pc;    % PC <- the specified principal component.
else
```

PC=Find_PC(3, 1); % PC <- the automatically selected principal component from the first three ones. Method 1 is used for Find_PC ().

end

if i==0

i=size(X,2); % If i is 0, then i <- the number of all the attributes in X.

end

I=i; % I <- i.

[COEFF,INDEX2]=sort(abs(V(:,PC))); % Sorts the absolute values of the PC-th column of V in an ascending order. COEFF <- the result of sorting, and INDEX2 <- the corresponding positions of the elements in COEFF.

COEFF=V(:,PC); % COEFF <- the PC-th column of V.

COEFF(INDEX2(1:size(X,2)-I))=0; % The significance coefficients of the unimportant attributes <- 0.

K=X*COEFF; % K <- the projection vector of the samples against the selected principal component.

CLS_PDDP=zeros(1,size(X,1)); %Initializes the clustering result vector.

CLS_PDDP(find(K>0))=1; % Clusters the samples with the PDDP method. The elements in CLS_PDDP with K > 0 are set to 1.

XL=[X(find(~CLS_PDDP),:)]; XR=[X(find(CLS_PDDP),:)]; % Separates the samples into two sub datasets, XL and XR.

wL=mean(XL); wR=mean(XR); % Calculates the center points of XL and XR.

A.4 MATLAB Code for Bisect K-means

File name: K_Means.m

% The Bisect K-means clustering

%

```
% Usage: [CLS_KM,XL,XR,wL,wR]=K_Means()
% Input
% None.
% Output
% CLS_KM: the clustering result of K-means.
% XL and XR: the sub datasets after the K-means clustering.
% wL and wR: the center points of XL and XR, respectively.
function [CLS_KM, XL, XR, wL, wR] = K_Means ()
global X CLS_KM wL wR % Declaration of global variables.
cL=wL*2; cR=wR*2; % Initializes cL and cR.
while (~isequal(cL,wL)) & (~isequal(cR,wR)) % Looping
    cL=wL; cR=wR;
 a=sum((X-ones(size(X,1),1)*wL).^2-(X-ones(size(X,1),1)*wR).^2,2);
    XL=[X(find(a<=0),:)]; XR=[X(find(a>0),:)];
    wL=mean(XL); wR=mean(XR);
end
CLS_KM=zeros(size(X,1),1);
CLS_KM(find(a>0))=1;
```

A.5 MATLAB Code for PDDP + Bisect K-means

A.5.1 MATLAB code for unsupervised PDDP + bisect K-means

```
% The main procedure of the unsupervised PDDP + bisect K-means clustering.
% Data and parameters
% S0: a dataset with samples as rows and attributes as columns.
% th: the threshold.
```

```
%
PCA(S0,th);
pc=PDDP(0,0);
cluster =K_Means;
for i=1:size(S0,2)-1
      PDDP(pc,i);
      cls_temp=K_Means;
      if isequal(cls_temp,cluster)
            break;
      end
end
```

A.5.2 MATLAB Code for supervised PDDP + bisect K-means

```
% The main procedure of the supervised PDDP + bisect K-means clustering.
% Data and parameters
% S0: a dataset with samples as rows and attributes as columns.
% th: the threshold.
% pc: the specified principal component.
% cluster: the clustering result of reference.
%
PCA(S0,th);
n=inf; g=0;
for i=1:size(S0,2)
       PDDP(pc,i);
       cls_temp=K_Means;
       err =sum(xor(cluster,cls_temp));
```

```
            if (err<n)
                g=i;
                n=err;
            end
        end
        PDDP(pc,g);
K_Means;
```

Vita

Zhiyu Zhao was born in Jingzhou, a small historic city in Hubei, China. In 1997 she graduated with a Bachelor's Degree of Engineering in Computer Science and Engineering, and a Bachelor's Degree of Law in Economic Laws from the Huazhong University of Science and Technology (HUST), China. In 2000 she received her Master's Degree of Engineering in Computer Science and Engineering from HUST. She worked for the College of Computer Science and Technology of HUST as an instructor and researcher for three years, and then she studied at the Polytechnic Institute of Milan, Italy for two semesters as a visiting researcher, before she came to New Orleans. In the fall 2005 semester, she studied at the Louisiana State University as a visiting student. She will study for a doctorate degree in the Department of Computer Science at the University of New Orleans.