

Methods of Determining the Number of Clusters in a Data Set and a New Clustering Criterion

Mingjin Yan

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Statistics

Keying Ye, Chair
Samantha Bates Prins
Eric P. Smith
Dan Spitzner

November, 2005
Blacksburg, Virginia

Keywords: Cluster analysis, DD-weighted gap statistic, Gap statistic, K -means clustering, Multi-layer clustering, Number of clusters, Weighted gap statistic.

Copyright 2005, Mingjin Yan

Methods of Determining the Number of Clusters in a Data Set and a New Clustering Criterion

Mingjin Yan

(ABSTRACT)

In cluster analysis, a fundamental problem is to determine the best estimate of the number of clusters, which has a deterministic effect on the clustering results. However, a limitation in current applications is that no convincingly acceptable solution to the best-number-of-clusters problem is available due to high complexity of real data sets. In this dissertation, we tackle this problem of estimating the number of clusters, which is particularly oriented at processing very complicated data which may contain multiple types of cluster structure. Two new methods of choosing the number of clusters are proposed which have been shown empirically to be highly effective given a clear and distinct cluster structure in a data set. In addition, we propose a sequential type of clustering approach, called multi-layer clustering, by combining these two methods. Multi-layer clustering not only functions as an efficient method of estimating the number of clusters, but also, by superimposing a sequential idea, improves the flexibility and effectiveness of any arbitrary existing one-layer clustering method. Empirical studies have shown that multi-layer clustering has higher efficiency than one layer clustering approaches, especially in detecting clusters in complicated data sets. The multi-layer clustering approach has been successfully implemented in clustering the WTCHP microarray data and the results can be interpreted very well based on known biological knowledge.

Choosing an appropriate clustering method is another critical step in clustering. K -means clustering is one of the most popular clustering techniques used in practice. However, the k -means method tends to generate clusters containing a nearly equal number of objects, which is referred to as the “equal-size” problem. We propose a clustering method which competes with the k -means method. Our newly defined method is aimed at overcoming the so-called “equal-size” problem associated with the k -means method, while maintaining its advantage of computational simplicity. Advantages of the proposed method over k -means clustering have been demonstrated empirically using simulated data with low dimensionality.

Dedication

This dissertation is dedicated to my beloved parents, Wengang Yan and Hong Shi, and my sister, Mingfeng Yan, in China. Their sincerest love has given me courage to overcome the most difficult times in my way of pursuing my dream of studying abroad. It was their unconditional and consistent care, support and understanding that helped me sustain.

ACKNOWLEDGEMENTS

I would like to express my greatest appreciation to my advisor, Dr. Keying Ye, for supervising my graduate study for four and a half years. I feel blessed to have him as my advisor. He helped me not only accomplish my dream of becoming a professional statistician but also develop a more mature personality. I thank him for every piece of his intensive efforts that have been put into this research work. I also thank him for introducing me to the field of computational biology in which I found great interests of applying the theories and methods of statistics.

I would like to thank Dr. Xiao Yang for sharing with me a lot of knowledge in doing statistical research associated with computational biology. He has been instrumental in finding the topic of this dissertation. I would also like to thank Dr. Eric Smith, Dr. Samantha Bates Prins and Dr. Dan Spitzner for their helpful comments and advice in improving this dissertation.

Contents

1	Introduction	1
2	Review of Cluster Analysis	4
2.1	Basic steps in cluster analysis	4
2.1.1	Element selection	5
2.1.2	Variable selection	5
2.1.3	Variable standardization	6
2.1.4	Selecting a measure of association (similarity/dissimilarity)	7
2.1.5	Selection of clustering method	8
2.1.6	Determining the number of clusters	8
2.1.7	Interpretation, validation and replication	9
2.2	Clustering methods	10
2.2.1	Partitioning methods via optimization criteria	10
2.2.2	k -means algorithms	19
2.2.3	Model-based clustering	20
2.2.4	Hierarchical clustering	21
2.2.5	Miscellaneous clustering methods	22
2.3	Determining the number of clusters	23
2.3.1	Global methods	23

2.3.2	Local methods	28
3	Determining the Number of Clusters Using the Weighted Gap Statistic	30
3.1	Introduction	30
3.2	Methods using the weighted gap statistic	33
3.2.1	The weighted gap method	33
3.2.2	DD-weighted gap method	35
3.3	Simulation studies and applications	38
3.3.1	Simulation studies	39
3.3.2	Multi-layer clustering	42
3.3.3	Applications	44
3.4	Discussion	49
4	Clustering of gene expression data using Multi-Layer Clustering: Application to the study of oxidative stress induced by cumene hydroperoxide in yeast	57
4.1	Introduction	58
4.2	Weighted gap statistic and multi-layer clustering approach	59
4.2.1	Weighted gap method and DD-weighted gap method	59
4.2.2	Multi-layer clustering	61
4.3	Clustering application	62
4.3.1	Data and analysis	62
4.3.2	Interpretation and discussion	65
4.4	Conclusion	71
5	A New Partitioning Method	73
5.1	Motivation	73

5.2	Description of the new method	77
5.2.1	Partitioning criterion	77
5.2.2	Algorithm for computation	78
5.3	Comparisons with the k -means method	79
5.3.1	Clustering validation and the adjusted Rand index	80
5.3.2	Simulation studies	81
5.3.3	Clustering of the Iris data	83
5.4	Summary and discussion	84
6	Summaries and Future Research	95

List of Figures

2.1	An example showing the importance of variable selection in clustering. Suppose a 2-variate data contain four distinct clusters, distinguished by different colors of characters in the plots. When both variable 1 and 2 are included in clustering, the four clusters, each indicated by one type of characters in Figure 2.1 (a), can be easily separated by most clustering methods. If the clustering is based on variable 1 only, it is very likely that only two clusters will be revealed which are indicated by the two types of characters in Figure 2.1 (b).	6
3.1	Plots of W_g and \overline{W}_g , the weighted W_g : (a) a two-cluster data; (b) within-clusters dispersion W_g and the weighted within-clusters dispersion \overline{W}_g for the data in (a); (c) a six-cluster data; (d) within-clusters dispersion W_g and the weighted within-clusters dispersion \overline{W}_g for the data in (c). For the convenience of demonstration, W_g/n is actually plotted instead of W_g , where n is the sample size of the data.	36
3.2	Plots of $Gap_n(g)$ and $\overline{Gap}_n(g)$ vs. g : (a) the two-cluster data in Figure 3.1 (b) the six-cluster data in Figure 3.1; plots of $D\overline{Gap}_n(g)$ and $DD\overline{Gap}_n(g)$ vs. g : (c) the two-cluster data studied in Figure 3.1 (d) the six-cluster data studied in Figure 3.1.	37
3.3	Illustration of the simulated data with nested clusters: (a) data; (b) classification result via K -means clustering where $g = 3$. In (b), the three clusters are distinguished by different types of markers.	43
3.4	Results of estimating the number of clusters of real data sets: (a1)~(a3) plots of $Gap_n(g)$, $\overline{Gap}_n(g)$ and $DD\overline{Gap}_n(g)$ vs. g for the Iris data; (b1)~(b3) plots for the Wisconsin breast cancer data; (c1)~(c3) plots for the leukemia data.	48

4.1	Hierarchical structure of WTCHP clusters ($C_1 \sim C_{14}$). The number of genes allocated to each (sub-)cluster at each step of clustering is indicated in the parenthesis. A rectangular represents a homogeneous cluster. An oval means that a cluster is further separated.	64
4.2	Plots of transcript fold change profiles for clusters generated by multi-layer clustering. For each gene, logarithm of its transcript level at 0, 3, 6, 12, 20, 40 and 70 minutes divided by its 0 minute transcript level is plotted against time.	66
4.3	Profiles of clusters enriched in antioxidant defense genes. The gene-wise standardized data were plotted in the graph. (A) k -means clustering with 6 clusters; (B) multi-layer clustering (14 clusters). Different colors represent different clusters. (C) Profiles of the two genes coding for the 2 enzymes of the glutathione biosynthesis.	67
4.4	Profiles of clusters enriched in genes encoding proteasome subunit proteins. The standardized gene expressions were plotted in the graph. (A) k -means clustering with 6 clusters; (B) multi-layer clustering (14 clusters). Different colors in (B) represent different clusters.	71
5.1	Illustration of the “equal-size” problem with k -means clustering. In this plot, data were simulated from 2 well separated normal distributions in 2 dimensions, with 200 and 50 points from each distribution, respectively. Points generated from the 2 different populations were plotted in triangles and squares, respectively. The 2 clusters produced by k -means clustering are indicated separately by solid and hollow markers in the plot.	76
5.2	Comparing our proposed clustering method with k -means method. Univariate Normal clusters with equal cluster sizes: $n_1 = n_2 = 50$, $\mu_1 = 0$, $\mu_2 = 4$, $\sigma_1 = 1$, $\sigma_2 = 0.3$. Summary: the two methods work equally well.	86
5.3	Comparing our proposed clustering method with the k -means method. Univariate Normal clusters with equal cluster variations: $n_1 = 100$, $n_2 = 30$, $\mu_1 = 0$, $\mu_2 = 4$, $\sigma_1 = \sigma_2 = 1$. Summary: the two methods work equally well.	86
5.4	Comparing our proposed clustering method with the k -means method. Univariate Normal clusters with both unequal cluster sizes and unequal cluster variations: $n_1 = 100$, $n_2 = 30$, $\mu_1 = 0$, $\mu_2 = 4$, $\sigma_1 = 1$, $\sigma_2 = 0.3$. Summary: when clusters with larger cluster variations are also associated with larger cluster sizes, our proposed method performs better than the k -means method.	87

5.5	Comparing our proposed clustering method with the k -means method. Univariate Normal clusters with both unequal cluster sizes and unequal cluster variations: $n_1 = 30$, $n_2 = 100$, $\mu_1 = 0$, $\mu_2 = 4$, $\sigma_1 = 1$, $\sigma_2 = 0.3$. Summary: when clusters with larger cluster variations are associated with smaller sample sizes, our proposed method may perform worse than the k -means method.	87
5.6	Comparing our proposed clustering method with the k -means method. Univariate uniform clusters with both unequal cluster sizes and unequal cluster variations: (a1, b1) $n_1 = 100$, $n_2 = 30$, $f_1 \sim U(0, 4)$, $f_1 \sim U(5, 6)$; (a2, b2) $n_1 = 30$, $n_2 = 100$, $f_1 \sim U(0, 4)$, $f_1 \sim U(5, 6)$. The same conclusions as obtained from the above examples of univariate Normal clusters.	88
5.7	Comparing our proposed clustering method with the k -means method. Bivariate Normal clusters with equal cluster sizes: (a1, b1) $n_1 = n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (5, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.3^2 I$; (a2, b2) $n_1 = n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.3^2 I$. Summary: our proposed outperforms the k -means method.	89
5.8	Comparing our proposed clustering method with the k -means method. Bivariate Normal clusters with both unequal cluster sizes and unequal cluster variations: $n_1 = 200$, $n_2 = 50$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$. Summary: when clusters with larger cluster variations are also associated with larger cluster sizes, our proposed method performs better than the k -means method.	90
5.9	Comparing our proposed clustering method with the k -means method. Bivariate Normal clusters with both unequal cluster sizes and unequal cluster variations: (a1, b1) $n_1 = 20$, $n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (4, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$; (a2, b2) $n_1 = 20$, $n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$. Summary: when clusters with larger cluster variations are associated with smaller sample sizes, our proposed method may perform worse than the k -means method.	91

5.10	Comparing our proposed clustering method with the k -means method. Bivariate Normal clusters with both unequal cluster sizes and unequal cluster variations: (a1, b1) $n_1 = 20, n_2 = 100, \mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$; (a2, b2) $n_1 = 40, n_2 = 100, \mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$; (a3, b3) $n_1 = 60, n_2 = 100, \mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$. Summary: when clusters with larger cluster variations are associated with smaller sample sizes, our proposed method may perform worse than the k -means method; as the degree of the discrepancy in cluster sizes decreases, our method will outperform the k -means method.	92
5.11	Comparing our proposed clustering method with the k -means method with the Iris data. To facilitate visualization of the clustering results, scores corresponding to the first two principal components are plotted.	93
5.12	Classification results when the specified number of clusters is larger than the true cluster number in data: (a) our proposed method; (b) the k -means method. . . .	94

List of Tables

2.1	Summary of the important optimization criteria reviewed in Section 2.2.1.	17
3.1	Results of comparing the weighted gap method, the DD-weighted gap method and the gap method in estimating the true number of clusters in simulated data: Model 1 to 9. The last column contains the estimating results when applying multi-layer clustering to each model which is introduced in Section 3.3.2. Numbers given in this table are the percentages of the total 50 data sets generated in each model.	41
3.2	Estimates of the number of clusters of real data sets via the gap method, the weighted gap method, the DD-weighted gap method and multi-layer/pc clustering. G is the known number of clusters in data.	45
3.3	The number of sub-clusters of real data sets estimated in multi-layer clustering. .	45
4.1	Multi-layer clustering results for the WTCHP data. D_1 and D_2 are the two clusters generated at the first layer of analysis. D_{ij} stands for the j_{th} cluster when further separating cluster D_i in to 2 smaller clusters, $i, j = 1$ or 2 . D_{ijk} represents the k_{th} cluster obtained by dividing cluster D_{ij}	63
4.2	Pathway analysis of clusters found by multi-layer clustering. In this table, * stands for clusters containing pathways with p-value < 0.05 or not enriched in just a few specific pathways.	69
4.3	Pathway analysis of clusters found by one-layer clustering analysis with 6 clusters specified.	70

5.1	The contingency table for comparing two partitions of a data into G clusters. n_{ij} , $i = 1, 2, \dots, G$, $j = 1, 2, \dots, G$, is the number of objects simultaneously assigned to cluster i in partition T and cluster j in partition C . $n_{i.} = \sum_{j=1}^G n_{ij}$. $n_{.j} = \sum_{i=1}^G n_{ij}$	81
5.2	Simulation scenarios for comparing our proposed method with the k -means method. The number of variables in a simulated data set is p . Data were simulated from the Normal distribution with parameters indicated in the table, except that in Scenario 5, the uniform clusters were examined. For a particular scenario, 100 data sets were generated and clustered using both the k -means method (kmeans) and our proposed method (wkmeans), and the averaged adjusted Rand index is the mean value of the corresponding 100 adjusted rand indices.	82

Chapter 1

Introduction

Cluster analysis is an important exploratory tool widely used in many areas such as biology, sociology, medicine and business. For example, in computational biology, cluster analysis has been successfully implemented in inferring functions of unknown genes and detecting classes or sub-classes of diseases. The goal of cluster analysis is to assign objects in a data set into meaningful classes such that objects in the same class are more similar to each other than to those in other classes. The reasonable way of summarizing the observed data into classes is determined only based on the information provided by the data since there is no prior knowledge about the classes at the beginning of an investigation. Abundant research on cluster analysis exists in the literature. However, none of them is convincingly acceptable, due to the high complexity of real data sets. This dissertation is dedicated to the methodologies of cluster analysis. Our research goals focus on two critical steps in cluster analysis: determining the number of clusters in a data set and choosing a good clustering technique.

Cluster analysis involves several procedures as summarized by Milligan ([69]): selecting clustering objects and clustering variables, variable standardization, choosing the measure of association, selecting the clustering method, determining the number of clusters and interpretation, validation and replication. Chapter 2 gives a review of cluster analysis: we review the basic steps in a clustering process in Section 2.1; Section 2.2 reviews clustering methods proposed in the literature in detail; finally, several important methods of determining the number of clusters in a data set are reviewed in Section 2.3.

A large number of clustering methods are available for cluster analysis. However, a fundamental problem in applying most of the existing clustering approaches is that the number

of clusters needs to be pre-specified before the clustering is conducted. The clustering results may heavily depend on the number of clusters specified. It is necessary to provide educated guidance for determining the number of clusters in order to achieve appropriate clustering results. At the current stage of research, none of the existing methods of choosing the optimal estimate of the number of clusters is completely satisfactory. The gap method was recently proposed by Tibshirani, *et al.* ([84]). The main idea of the gap method is to compare the within-cluster dispersions in the observed data to the expected within-cluster dispersions assuming that the data came from an appropriate null reference distribution. Simulation results reported by Tibshirani, *et al.* indicated that the gap method is a potentially powerful approach in estimating the number of clusters for a data set. However, recent studies have shown that there are situations where the gap method may perform poorly. For example, when the data contain clusters which consist of objects from well separated exponential populations.

Motivated by the gap method, in Chapter 3, we investigate this best-number-of-clusters problem, which is specifically oriented at processing very complicated data that may contain multiple types of cluster structure. In the current research, we propose a criterion of measuring the goodness-of-fit associated with the classification result given a specific number of clusters. Based on this criterion, the weighted gap and DD-weighted gap methods are developed to search for the optimal estimate of the cluster number over a range of candidate values. The weighted gap approach can determine if the examined data is clustered or homogeneous, that is, if the data contain distinct clusters of observations or just come from one group. The DD-weighted gap method will estimate the cluster number assuming that the data contain multiple clusters, that is, the number of clusters is more than 1. Both simulation studies and real data applications have shown that these two methods are more efficient than the gap method. In addition, we propose a sequential type of clustering approach, called multi-layer clustering, by combining these two methods. Empirical studies have shown that multi-layer clustering has higher effectiveness than the one layer clustering approaches, especially in detecting clusters in complicated data sets.

Potentially, our proposed methods for estimating the number of clusters will be applicable to any arbitrary research context where cluster analysis is a suitable analytical tool. An interesting problem of recovering distinct patterns of changes in a particular feature across a series of experimental conditions presented in a data set has been discussed intensively in applications. An important example is clustering objects (e.g. genes in a computational biology problem) based on their temporal profiles. In Chapter 4, we successfully implement

our methods for determining best-number-of-clusters in clustering WTCHP microarray data, which was obtained from a time course study of genome-wide oxidative stress response in *S. cerevisiae* cultures exposed to cumene hydroperoxide (CHP). We observe that our methods can be used to decide the number of different patterns of changes in transcripts after exposure to CHP in the WTCHP data and to separate various patterns from each other in ways that results can be interpreted very well.

It is obvious that the optimal estimate of the number of clusters depends on the clustering method. Therefore, an important step is the choice of an appropriate clustering method at the onset of any cluster analysis. Among the large number of available clustering techniques, k -means clustering has been widely used mainly because that it is easy to implement in terms of computation. However, shortcomings have been found in applications of this popular clustering method, such as scale-dependence and the tendency to produce “spherical” clusters. A main problem with the k -means method is that it tends to generate clusters containing a nearly equal number of objects, often referred to as the “equal-size” problem. In situations where there exists large discrepancy in cluster sizes, k -means clustering may fail to provide appropriate classification results. In Chapter 5, we propose a clustering method which supplements the k -means method. Our newly defined method is aimed at overcoming the so-called “equal-size” problem associated with the k -means method, while maintaining its advantage of computational simplicity. Advantages of the proposed method over k -means clustering have been demonstrated empirically using simulated data when dimension is low to moderate.

Finally, Chapter 6 summarizes the work that has been done in this dissertation and presents discussions about directions in future research.

Chapter 2

Review of Cluster Analysis

2.1 Basic steps in cluster analysis

“A clustering method refers to the means by which the clusters or groups are formed. A cluster analysis will refer to the overall sequence of steps that represent analysis.” As emphasized by Milligan ([69]), a cluster analysis is distinct from a clustering method. It is an essential step to choose the clustering method in a cluster analysis. The purpose of this section is to review cluster analysis as an integrated set of procedures. The organization of this section largely follows Milligan’s excellent review paper on applied cluster analysis ([69]).

As summarized by Milligan ([69]), there are seven critical steps in cluster analysis: 1) Clustering element selection: objects used in clustering should represent the cluster structure (if any) in the data; 2) Clustering variable selection: variables selected for clustering should provide sufficient and relevant information for the discovery of the correct cluster structure; 3) Variable standardization: the user needs to decide whether the clustering should be based on the raw variable space or the standardized variable space; 4) Choosing a measure of association (dissimilarity/similarity measure): “the measure should reflect those characteristics that are suspected to distinguish the clusters present in the data”; 5) Selection of clustering method: clustering methods should be efficient in recovering the cluster structure underlying the data; 6) Determining the number of clusters: it is necessary to estimate the number of clusters since most clustering methods partition objects into a fixed number of clusters, but they are not designed to determine the “optimal” number of clusters; 7) Interpretation,

validation and replication: the final stage in cluster analysis is to interpret the clustering results under the contexts of practical problems. Validation tests may determine whether the detected cluster structure is significant. The same clustering result should be found in replicated samples. In the following sections, we are going to review these steps in detail.

2.1.1 Element selection

The purpose of cluster analysis is to explore the cluster structure in data without any prior knowledge about the true classification. The result is quite data-driven. In particular, the data structure will be defined by the elements selected for study. An ideal sample should represent the underlying clusters or populations. Inclusion of outliers (data points falling outside the general region of any cluster) should be avoided as much as possible so as to facilitate the recovery of distinct and reliable cluster structures, although they sometimes form a single cluster.

2.1.2 Variable selection

Data investigated in cluster analysis are often recorded on multiple variables. The importance of choosing an appropriate set of variables for clustering is obvious. Figure 2.1 gives an example showing the strong impact that variable selection may have on the cluster structure that could be revealed by a clustering method. In Figure 2.1, the data were generated from four 2-variate normal distributions with the mean vectors $\mu_1 = (0, 0)'$, $\mu_2 = (0, 2)'$, $\mu_3 = (2, 0)'$ and $\mu_4 = (2, 2)'$, respectively, and the same covariance matrix $\Sigma = I$. Intuitively, the data contain four small clusters, each of which corresponds to one of the four normal distributions (Figure 2.1 (a)). However, assume that variable 2 is not selected in the clustering, then it is very likely that only two large clusters can be distinguished (Figure 2.1 (b)), since further separation of the two large clusters into four smaller clusters requires information about variable 2.

First of all, it is necessary to select enough variables to provide sufficient information about the underlying cluster structure. However, inclusion of unnecessary “noise” variables (better termed as “masking variables” by Fowlkes and Mallows [34]) might “dramatically interfere with cluster recovery [69]”. The unfavorable impact of masking variables on searching for real clustering in the data was first studied by Milligan ([65]). When Euclidean distance is used in a hierarchical clustering technique, a possible solution of the problem of

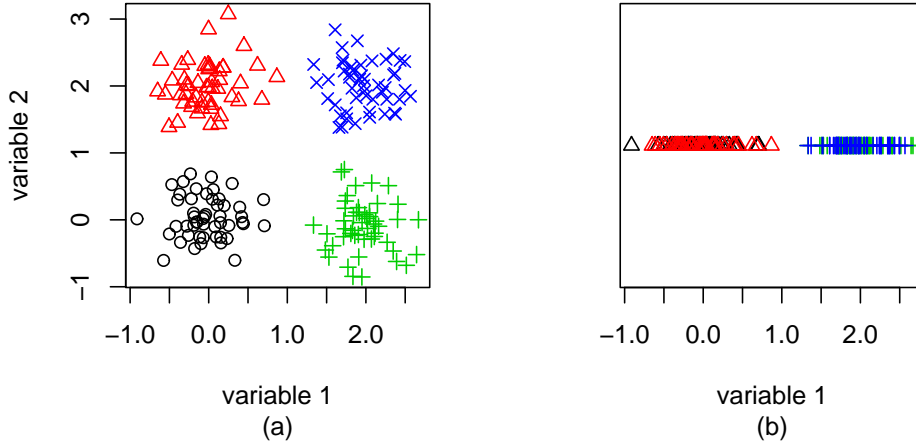


Figure 2.1: An example showing the importance of variable selection in clustering. Suppose a 2-variate data contain four distinct clusters, distinguished by different colors of characters in the plots. When both variable 1 and 2 are included in clustering, the four clusters, each indicated by one type of characters in Figure 2.1 (a), can be easily separated by most clustering methods. If the clustering is based on variable 1 only, it is very likely that only two clusters will be revealed which are indicated by the two types of characters in Figure 2.1 (b).

masking variables is De Soete's optimal weighting method ([23],[24]). This method defines the distance between $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})'$ and $x_j = (x_{j1}, x_{j2}, \dots, x_{jp})'$ as

$$d(i, j) = \left[w_1(x_{i1} - x_{j1})^2 + w_2(x_{i2} - x_{j2})^2 + \dots + w_p(x_{ip} - x_{jp})^2 \right]^{\frac{1}{2}}, \quad (2.1)$$

where w_k , $k = 1, \dots, p$, are the optimal weights computed in a complex way so that the fit of the distance to an ultrametric matrix is optimized. Milligan's study ([68]) showed that this method was effective in dealing with the masking problem. Unfortunately, Gnanadesikan, Kettenring, and Tsao ([41]) didn't find the same favorable evidence for De Soete's method in their trials with differently simulated data sets. Other attempts to solve this problem include different approaches to pursuing optimal variable weighting and variable selection strategies without weighting variables.

2.1.3 Variable standardization

Variable standardization may have striking impact on cluster analysis. The relative distances between pairs of objects may be changed after standardization, hence altering the

cluster structure in the data. Although standardization is usually suggested as an approach to making variables commensurate, it would be a suitable step if the clusters were believed to exist in the transformed variable space ([2],[18],[32],[80]). The other aspect of variable standardization emphasized by Milligan ([69]) is to choose appropriate standardization measures. A comparative study of eight forms of standardization including the unstandardized form of data using simulated data was conducted by Milligan and Cooper (see [72]).

2.1.4 Selecting a measure of association (similarity/dissimilarity)

A basic assumption of cluster analysis is that objects assigned to the same cluster are closer (more similar) to each other than to those in other clusters. A large class of clustering techniques partition objects based on the dissimilarity matrix directly or indirectly. Logically, “the measure should reflect the characteristics that are suspected to distinguish the clusters present in the data (see [69])”. There are numerous definitions of similarity/dissimilarity measures with respect to different types of variables (e.g. interval-scaled variable, nominal variable, ordinal variable or mixed data). In this paper, our studies are restricted to data containing interval-scaled variables, hence, only the similarity/dissimilarity measures defined for continuous data are reviewed here.

For continuous data, the degree of dissimilarity between objects is often measured by the distance between them. Euclidean distance (2.2) and Manhattan distance (2.3) are the most popular choices, being typical examples from the class of Minkowski distances (2.4). Sometimes, users may compute weighted Euclidean distances as in (2.5) so that variables of more importance will receive higher weights.

$$\text{Euc. Dist. } d_1(i, j) = \left((x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots + (x_{ip} - x_{jp})^2 \right)^{\frac{1}{2}}. \quad (2.2)$$

$$\text{Man. Dist. } d_2(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \cdots + |x_{ip} - x_{jp}|. \quad (2.3)$$

$$\text{Min. Dist. } d_3(i, j) = \left(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \cdots + |x_{ip} - x_{jp}|^q \right)^{\frac{1}{q}}, \quad q > 1. \quad (2.4)$$

$$\begin{aligned} \text{W. Euc. Dist. } d_4(i, j) &= \left(w_1(x_{i1} - x_{j1})^2 + w_2(x_{i2} - x_{j2})^2 + \cdots + w_p(x_{ip} - x_{jp})^2 \right)^{\frac{1}{2}}, \\ &\text{where } \sum_{k=1}^p w_k = 1. \end{aligned} \quad (2.5)$$

Other than distance measures, there are correlation-type measures of dissimilarity, which are defined on the basis of correlation coefficients, $R(i, j)$. When correlations between objects

are used to quantify their similarity, it is equivalent to standardizing each row of the data matrix. Then, these measures are 1 when $R(i, j)$ is -1 and 0 when $R(i, j)$ is 1. This will be suitable when interest is in assessing the linear relationship between objects, instead of the difference in size. Two commonly used examples are given by

$$d_5(i, j) = (1 - R_1(i, j))/2, \quad R_1(i, j) = \frac{\sum_{k=1}^p (x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j)}{\left(\sum_{k=1}^p (x_{ik} - \bar{x}_i)^2 \sum_{k=1}^p (x_{jk} - \bar{x}_j)^2\right)^{\frac{1}{2}}},$$

$$d_6(i, j) = (1 - R_2(i, j))/2, \quad R_2(i, j) = \frac{\sum_{k=1}^p x_{ik}x_{jk}}{\left(\sum_{k=1}^p x_{ik}^2 \sum_{k=1}^p x_{jk}^2\right)^{\frac{1}{2}}}.$$

2.1.5 Selection of clustering method

Choosing the clustering method is a critical step in a clustering process. As pointed out by Milligan ([69]), one should consider four aspects when selecting a method. First, the method should be designed to recover the cluster types suspected to be present in the data. Second, the clustering method should be effective at recovering the structures for which it was designed. Third, the method should be able to resist the presence of error in data. Finally, availability of computer software for performing the method is important.

There are a large number of clustering methods proposed in the literature. A lot of work has been done in comparing clustering methods in terms of the ability to recover cluster structures. For example, Milligan and Cooper ([71]) performed a comprehensive comparison of agglomerative hierarchical algorithms by examining the effects of a number of factors on the performances of these clustering methods. Milligan ([65]) studied the influence of initial seeds on the k -means algorithms. Marriott ([62]) discussed the properties of several clustering criteria by examining the effect of adding a single point into a data set. A summary of pre-1996 validation results for partitioning clustering methods was provided by Milligan ([69]). Although conclusions drawn from these studies can not be generalized due to the limited simulations that have been done, it did provide helpful implications for applied work. More details about clustering methods will be reviewed in the Section 2.2.

2.1.6 Determining the number of clusters

A fundamental problem in cluster analysis is to determine the number of clusters, which is usually taken as a prior in most clustering algorithms. Clustering solutions may vary

as different numbers of clusters are specified. A clustering technique would most possibly recover the underlying cluster structure given a good estimate of the true number of clusters.

A number of strategies for estimating the optimal number of clusters have been proposed. A very extensive comparative evaluation was conducted by Milligan and Cooper ([70]), where they compared 30 proposed methods in estimating the true number of clusters when applying hierarchical clustering algorithms to simulated data with well-separated clusters. According to their work, Calinski and Harabasz's index ([12]) is the most effective one, followed by Duda and Hart's method ([26]) and the C -index. Discussions about Milligan and Cooper's study can also be found in [44]. Developments in this area of cluster analysis will be reviewed in more detail later in Section 2.3.

2.1.7 Interpretation, validation and replication

The ultimate usefulness of a cluster analysis depends on the final interpretation of the resulting classification with respect to the specific problems under study. This often requires special knowledge and expertise in particular areas. Clusters may be characterized by simple descriptive statistics on variables used in clustering as well as exogenous variables not included in clustering. Graphical representations could be very useful in interpreting the resultant cluster structure. For a helpful discussion on the use of graphical methods, see [51].

Although cluster analysis is conceived as an *unsupervised* process, cluster validity can be assessed via the so-called *external* or *internal* approaches. Commonly used *external criteria* include the Rand index, the adjusted Rand index, the Fowlkes and Mallows index and the Jaccard index. An *external* approach evaluates the results of a clustering process based on external classification information independent of the clustering procedure. In a less formal manner, internal indices can be computed to reflect the good-ness-of-fit between the data and the partitioning result. A good source for critical reviews on internal criteria is [66], where 30 internal criteria were compared through Monte Carlo studies. More formal internal validation tests are available. Gordon ([44]) categorized such tests by the type of class structure that is under study as follows:

1. the complete absence of class structure,
2. the validity of an individual cluster,

3. the validity of a partition,
4. the validity of a hierarchical classification.

Limitations in using these more formal tests were also discussed by Gordon. As he commented, tests about the complete absence of class structure depend on how the null model is specified, which could be data-influenced. At the same time, there are several possible ways to define the alternative models. The power of such tests are not well studied.

2.2 Clustering methods

In this section, we review various clustering methods that have been proposed in the literature. The organization of this section is as follows: Section 2.2.1 reviews a class of methods which partition a data with n objects into g clusters such that certain criteria are optimized; Section 2.2.2 discusses a widely used partitioning technique: the k -means clustering; model-based clustering, which is a recently developed powerful clustering technique assuming finite mixture models for data, is introduced in Section 2.2.3; Section 2.2.4 summarizes another large class of clustering methods: hierarchical clustering; finally, miscellaneous clustering techniques are briefly discussed in Section 2.2.5.

2.2.1 Partitioning methods via optimization criteria

In this section, discussions are focused on partitioning techniques which are based on various optimization criteria. For this class of clustering methods, clusters of observations are generated such that certain numerical criterion is optimized. The number of groups should be specified before conducting the partition. Methods that can help users in determining the number of clusters will be discussed in Section 2.3.

Partitioning techniques vary in both the optimization criteria and the optimization algorithms used in computation. Some criteria are aimed at obtaining groups with certain good qualities, such as high within-group homogeneity and large between-group separation. Some definitions of optimization criteria are related to multivariate normal mixture models. Computational algorithms are critical in application since it is certainly impossible to search through all the possible partitions of n objects into g groups except for very small n . Efficient algorithms are required for finding the optimal or near-optimal partition quickly. Behaviors

of different partitioning methods can be compared through simulation studies or real data analysis where information about the true classification is available.

Optimization criteria defined on dissimilarity matrices

Cluster analysis has never been straightforward, and probably one main reason of this is for the lack of clear definition of clusters and the variety of different types of clusters. With multivariate data, grouping structures can be far more complicated than those can be imagined. Cormack ([18]) and Gordon ([44]) gave a suggestive definition of cluster via *homogeneity* and *separation*, which reflect internal cohesion and external isolation of clusters, respectively. As an ideal partition, objects assigned to the same cluster should be homogeneous while clusters should be well separated from each other. Numerically, homogeneity and separation are usually measured on the basis of the one-mode dissimilarity matrix $(d_{ij})_{(n \times n)}$, of which element d_{ij} measures the dissimilarity between object i (x_i) and object j (x_j). For multivariate data in p dimensions, a popular choice of the dissimilarity measure between $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})'$ and $x_j = (x_{j1}, x_{j2}, \dots, x_{jp})'$ is Euclidean distance (2.2).

The *variance minimization techniques*, described by Singleton and Kautz ([79]), Forgy ([33]), Edwards and Cavalli-Sforza ([28]), Jancey ([52]), MacQueen ([60]) and Ball and Hall ([4]), also fall into this class of clustering methods. In these cases, a cluster is characterized by the cluster mean, \bar{x}_m , namely by its *centroid*. These techniques aim at achieving the optimal partition which minimizes the error sum of squares *ESS*, given by

$$ESS = \sum_{m=1}^g \sum_{l=1}^{n_m} (x_{ml} - \bar{x}_m)' (x_{ml} - \bar{x}_m). \quad (2.6)$$

It is easy to show that the criterion (2.6) can be rewritten as (2.7) shown below. Thus, minimizing *ESS* is equivalent to minimizing the cluster criterion (2.7) based on the squared Euclidean distance.

$$ESS = \sum_{m=1}^g \frac{1}{2n_m} \sum_{l=1}^{n_m} \sum_{v=l, v \neq l}^{n_m} d_{ml, mv}^2 \quad (2.7)$$

Optimization criteria derived from probabilistic models

Suppose we have multivariate data containing n objects in p dimensions. Each object can be expressed as $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})'$, $i = 1, \dots, n$. We define the dispersion matrix of each

group by

$$W_m = \sum_{l=1}^{n_m} (x_{ml} - \bar{x}_m)(x_{ml} - \bar{x}_m)', m = 1, \dots, g. \quad (2.8)$$

Then the pooled within-group dispersion matrix W is defined by

$$W = \sum_{m=1}^g \sum_{l=1}^{n_m} (x_{ml} - \bar{x}_m)(x_{ml} - \bar{x}_m)'. \quad (2.9)$$

The between-group dispersion matrix is defined by

$$B = \sum_{m=1}^g n_m (\bar{x}_m - \bar{x})(\bar{x}_m - \bar{x})', \quad \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.10)$$

and the total dispersion matrix of the data is T , where

$$T = \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})'. \quad (2.11)$$

It is a well known matrix identity that

$$T = W + B.$$

Optimization criteria reviewed in this subsection are expressed as numerical functions of W , W_m and/or B . Although some of them were defined heuristically, most of them were derived based on assumptions about the underlying distribution of data, especially finite multivariate normal mixture models.

I) *Minimization of the trace of W*

In the literature on cluster analysis, the criterion of minimizing (2.6) is more frequently known as the minimization of $tr(W)$ (the trace of W) criterion, which is equivalent to maximizing the trace of B , $tr(B)$. This criterion first appeared in the context of hierarchical clustering in [88]. It was explicitly suggested as a partitioning criterion by Singleton and Kautz ([79]). Moronna and Jacovkis ([73]) exhibited the equivalence between minimizing $tr(W)$ and using the metric $E(x, G)$ in (2.12), the Euclidean distance between x and the mean of cluster G (c_G), in their discussions about clustering procedures with different metrics.

$$E(x, G) = d^2(x, c_G; I) \quad (2.12)$$

II) *Minimization of the determinant of W*

Minimization of $tr(W)$ has the advantage of ease in computation. However, it has two main drawbacks: it is not invariant under nonsingular linear transformation of the data (e.g. changes in the measurement units [37]) and it tends to produce clusters of “spherical” shape. Motivated by the problem of scale-dependency in minimizing $tr(W)$, Friedman and Rubin ([37]) proposed two scale independent criteria by maximizing the trace of $W^{-1}B$, $tr(W^{-1}B)$, and the ratio of the determinants of T and W , $\frac{det(T)}{det(W)}$, respectively. Obviously, maximizing $\frac{det(T)}{det(W)}$ is equivalent to minimizing $det(W)$. Expressed by the eigenvalues of $W^{-1}B$, we have equations (2.13) and (2.14). It is known that all the eigenvalues of $W^{-1}B$ are invariant to any nonsingular linear transformation, hence, so are the two criteria. Friedman and Rubin made the conclusion that the $\frac{det(T)}{det(W)}$ criterion was preferable to $tr(W^{-1}B)$ since it demonstrated a stronger sensitivity to the local structure of data.

$$tr(BW^{-1}) = \sum_{k=1}^p \lambda_k, \quad (2.13)$$

$$\frac{det(T)}{det(W)} = \prod_{k=1}^p (1 + \lambda_k), \quad (2.14)$$

where $\lambda_1, \dots, \lambda_p$ are p eigenvalues of $W^{-1}B$.

III) *Scott and Symons' criterion*

Assume that each object independently comes from one of g p -dimensional normal distributions, which are uniquely determined by the mean vector μ_m and the variance covariance matrix Σ_m , $m = 1, \dots, g$. Let $\gamma = (\gamma_1, \dots, \gamma_n)'$ with γ_i denoting the classification parameter associated with x_i , where $\gamma_i = m$ if x_i comes from the m th sub-population. Then, the log likelihood density function of the sample $X = (x_1, x_2, \dots, x_n)'$ is

$$l(\theta; X) = -\frac{1}{2} \sum_{m=1}^g \left[\sum_{x_l \in C_m} (x_l - \mu_m)' \Sigma_m^{-1} (x_l - \mu_m) + n_m \log |\Sigma_m| \right], \quad (2.15)$$

where $\theta = (\gamma, \mu_1, \dots, \mu_g, \Sigma_1, \dots, \Sigma_g)$, C_m is the set of x_i 's allocated to the m th group, n_m is the number of objects in C_m and $|\Sigma_m|$ is the determinant of Σ_m .

Scott and Symons ([76]) showed that the $det(W)$ criterion could be derived by maximizing $l(\theta)$. Their argument relies on the fact that the maximum likelihood estimate (MLE) of μ_m , whatever the MLE of Σ_m and γ , is

$$\hat{\mu}_m(\gamma) = \bar{x}_m = \frac{1}{n_m} \sum_{x_l \in C_m} x_m. \quad (2.16)$$

Substituting $\hat{\mu}_m(\gamma)$ for μ_m in (2.15), we have that maximizing (2.15) is equivalent to minimizing

$$\sum_{m=1}^g \left[\text{tr}(W_m \Sigma_m^{-1}) + n_g \log |\Sigma_m| \right]. \quad (2.17)$$

If $\Sigma_m = \Sigma$, $m = 1, \dots, g$, and Σ is *unknown*, then expression (2.17) reduces to $\text{tr}(W \Sigma^{-1}) + n \log |\Sigma|$. Also, for fixed γ , we have the MLE of Σ as W/n . It follows that minimization of (2.17) is further reduced to minimization of $\det(W)$.

Under the situation where Σ_m , $m = 1, \dots, g$, are not restricted to be the same, the MLE of Σ_m , given γ , is W_m/n_m . Then, the $\hat{\gamma}$ which minimizes expression (2.17) is the grouping which minimizes

$$\prod_{m=1}^g |W_m|^{n_m}. \quad (2.18)$$

Both the $\text{tr}(W)$ criterion and the $\det(W)$ criterion have the tendency to generate clusters of equal shape. Besides, these two criteria have been found to produce clusters with roughly equal numbers of objects ([30]), although the $\det(W)$ criterion is able to identify elliptical clusters. Scott and Symons ([76]) suggested that minimizing the criterion (2.18) might solve the “equal-size” problem (illustrated later in Figure 5.1) that occurs within the $\det(W)$ criterion. Alternative criteria have been proposed allowing searching for groups with different shapes and/or different sizes. For example, Moronna and Jacoviks ([73]) suggested minimizing $\sum |W_m|^{\frac{1}{p}}$.

IV) Symons’s criteria based on Normal mixture models

Symons ([82]) derived new clustering criteria under the assumption that each object x_i comes from a mixture population with g components, each of which is a p -variate normal distribution with mean vector μ_m and covariance matrix Σ_m , $m = 1, \dots, g$. The density of x_i is

$$\begin{aligned} f(x_i) &= \sum_{m=1}^g \pi_m f_m(x_i | \mu_m, \Sigma_m) \\ &= \sum_{m=1}^g \pi_m \frac{\exp\{-\frac{1}{2}(x_i - \mu_m)' \Sigma_m^{-1} (x_i - \mu_m)\}}{(2\pi)^{p/2} |\Sigma_m|^{1/2}}, \end{aligned} \quad (2.19)$$

where the term π_m is the probability of x_i coming from the m th component.

Denote the unknown mixture component origin for x_i by z_i , where $z_i = m$ if x_i comes from the m th component. The clustering problem can be viewed as the problem of estimating the value of the vector $\underline{z} = (z_1, z_2, \dots, z_n)'$. Let C_m indicate the set of x_i ’s allocated to the

m th cluster, n_m the number of objects in C_m , n the total number of objects in data and $|\Sigma_m|$ the determinant of Σ_m . Let $\theta = (\pi_1, \pi_2, \dots, \pi_g, \mu_1, \dots, \mu_g, \Sigma_1, \dots, \Sigma_g)$, then the MLE of \underline{z} is obtained by maximizing the likelihood density function $L(X|\theta, \underline{z})$, given by

$$L(X|\theta, \underline{z}) = \prod_{m=1}^g (\pi_m^{n_m} |\Sigma_m|^{-\frac{1}{2}n_m}) \exp \left\{ -\frac{1}{2} \sum_{m=1}^g \sum_{i \in C_g} (x_l - \mu_m)' \Sigma_m^{-1} (x_l - \mu_m) \right\}, \quad (2.20)$$

or equally, the log likelihood density function, given by

$$l(X|\theta, \underline{z}) = -\frac{1}{2} \sum_{m=1}^g \left[-2n_m \log \pi_m + n_m \log |\Sigma_m| + \sum_{x_l \in C_m} (x_l - \mu_m)' \Sigma_m^{-1} (x_l - \mu_m) \right]. \quad (2.21)$$

Under the assumption that $\Sigma_m = \Sigma$ ($m = 1, \dots, g$) and Σ is *unknown*, the MLE of θ , for any fixed value of \underline{z} (\hat{z}), is $\hat{\theta} = (\hat{\pi}_1, \hat{\pi}_2, \dots, \hat{\pi}_g, \hat{\mu}_1, \dots, \hat{\mu}_g, \hat{\Sigma}_1, \dots, \hat{\Sigma}_g)$, where

$$\hat{\pi}_m(\hat{z}) = n_m/n, \quad (2.22)$$

$$\hat{\mu}_m(\hat{z}) = \bar{x}_m = \frac{1}{n_m} \sum_{x_l \in C_m} x_m, \quad (2.23)$$

$$\hat{\Sigma}_m(\hat{z}) = \frac{1}{n} W. \quad (2.24)$$

Substituting $\hat{\theta}$ into 2.21, we can obtain the MLE of z by minimizing

$$n \log |W| - 2 \sum_{m=1}^g n_m \log(n_m). \quad (2.25)$$

Thus the ML optimal partition of the data is achieved by assigning the n objects into g groups such that (2.25) is minimized.

Without the equal covariance matrices assumption, the MLE of π_m and μ_m are the same as (2.22) and (2.23), while Σ_m is estimated by

$$\hat{\Sigma}_m(\hat{z}) = \frac{1}{n_m} W_m = \frac{1}{n_m} \sum_{x_l \in C_m} (x_l - \bar{x}_m)(x_l - \bar{x}_m)'. \quad (2.26)$$

Then the ML optimal allocation, \hat{z} , is achieved by assigning the n objects into g groups such that

$$\sum_{m=1}^g n_m \log |W_m| - 2 \sum_{m=1}^g n_m \log(n_m) \quad (2.27)$$

is minimized.

Assuming that $\Sigma_m = \Sigma$, $m = 1, \dots, g$, the Bayesian approach estimates the optimal allocation, \hat{z} , as the mode of the marginal posterior density of z , $L(X|z)$, given by

$$L(X|z) = \int L(X|\theta, z)p(\theta)d\theta,$$

where $p(\theta)$ is the prior distribution of θ . Under the assumption that $\Sigma_m = \Sigma$ ($m = 1, \dots, g$) and Σ is *unknown*, with a vague prior on θ , namely

$$p(\theta) = p(\pi_1, \pi_2, \dots, \pi_g)p(\mu_1, \dots, \mu_g|\Sigma)p(\Sigma) \propto \left(\prod_{m=1}^g \pi_m\right)^{-1} |\Sigma|^{-\frac{1}{2}(p+1)}, \quad (2.28)$$

the Bayesian optimal allocation (\tilde{z}) is obtained by minimizing

$$(n - G) \log |W| + \sum_{m=1}^g p \log(n_m) - 2 \log \Gamma(n_m). \quad (2.29)$$

When Σ_m , $m = 1, \dots, g$, are not assumed to be equal, a similar vague prior as in (2.28) is used, except for independent prior information used for each of the g matrices Σ_m . In such cases, the Bayesian optimal allocation (\tilde{z}) is such that

$$\begin{aligned} & \sum_{m=1}^g ((n_m - G) \log |W_m| + p \log(n_m) - p(n_m + p) \log 2 \\ & - 2[\log \Gamma(n_m) + \sum_{i=1}^p \log \Gamma\{\frac{1}{2}(n_m + p + 1 - i)\}]) \end{aligned} \quad (2.30)$$

is minimized.

Symons ([82]) studied behaviors of these criteria together with the $tr(W)$ criterion and the $det(W)$ criterion using two real data sets which have heterogeneous covariance matrices. He showed that the two criteria derived under the unequal covariance matrices assumption are preferable to those associated with equal covariance matrices assumption.

V) *Banfield and Raftery's criterion*

Using the same mixture model as (2.19) and assuming that $\Sigma_m = \lambda_m^2 I$, Banfield and Raftery ([5]) suggested that a generalization of the $tr(W)$ criterion is to minimize

$$\sum_{m=1}^g n_m \log tr\left(\frac{W_m}{n_m}\right). \quad (2.31)$$

To summarize discussions in this section, Table 2.1 lists all the partitioning criteria reviewed in this section.

Table 2.1: Summary of the important optimization criteria reviewed in Section 2.2.1.

<i>Criteria</i>	<i>Origin</i>	<i>Main assumptions for derivation</i>
$\text{tr}(\mathbf{W})$	Ward (1963)	no distribution assumptions
$\text{trace}(\mathbf{B}\mathbf{W}^{-1})$	Friedman and Rubin (1967)	no distribution assumptions
$\det(\mathbf{W})$	Friedman and Rubin (1967), Scott and Symons (1971)	g sub-populations with equal covariance matrices
$\prod_{m=1}^g \mathbf{W}_m ^{\frac{\mathbf{n}_m}{p}}$	Scott and Symons (1971)	g sub-populations with unequal covariance matrices
$\sum \mathbf{W}_m ^{\frac{1}{p}}$	Moronna and Jacoviks (1974)	
$\mathbf{n} \log \mathbf{W} - 2 \sum_{m=1}^g \mathbf{n}_m \log(\mathbf{n}_m)$	Symons (1981)	finite normal mixture model with equal covariance matrices
$(\mathbf{n} - G) \log \mathbf{W} $ $+ \sum_{m=1}^g \mathbf{p} \log(\mathbf{n}_m) - 2 \log \Gamma(\mathbf{n}_m)$	Symons (1981)	finite normal mixture model with equal covariance matrices, Bayesian approach with vague prior
$\sum_{m=1}^g \mathbf{n}_m \log \mathbf{W}_m $ $- 2 \sum_{m=1}^g \mathbf{n}_m \log(\mathbf{n}_m)$	Symons (1981)	finite normal mixture model with unequal covariance matrices
$\sum_{m=1}^g ((\mathbf{n}_m - G) \log \mathbf{W}_m $ $+ \mathbf{p} \log(\mathbf{n}_m) - \mathbf{p}(\mathbf{n}_m + \mathbf{p}) \log 2$ $- 2[\log \Gamma(\mathbf{n}_m) +$ $\sum_{i=1}^{\mathbf{p}} \log \Gamma\{\frac{1}{2}(\mathbf{n}_m + \mathbf{p} + 1 - i)\}])$	Symons (1981)	finite normal mixture model with unequal covariance matrices, Bayesian approach with vague prior
$\sum_{m=1}^g \mathbf{n}_m \log \text{tr}(\frac{\mathbf{W}_m}{\mathbf{n}_m})$	Banfield and Raftery (1993)	$\sum_m = \lambda_m^2 I$

Optimization algorithms

Once a suitable clustering criterion has been defined, an efficient optimization algorithm is required to search for the optimal partition. The total number of possible partitions of n objects into g groups is, given by Liu ([58]),

$$N(n, g) = \frac{1}{g!} \sum_{m=1}^g (-1)^{g-m} \binom{g}{m} m^n.$$

It will not be feasible to search all possible partitions except for very small n and g . For example, $N(100, 5) \approx 6.6 \times 10^{67}$. In practice, *hill-climbing* algorithms ([37]) have been developed to approach this problem. The basic idea is to update the current partition by a better arrangement of objects if it provides an improvement in terms of a particular clustering criterion. The essential steps of these algorithms (see [30]) are as follows:

1. Find an initial partition of the n objects into g groups;
2. Calculate the change in the clustering criterion caused by moving each object from its own cluster to other clusters;
3. Make the change which leads to the greatest improvement in terms of a clustering criterion;
4. Repeat the previous two steps until no movement of a single object can bring further improvement to the cluster criterion.

These algorithms differ in two main aspects. First, the initial partition can be obtained in various ways. It might be chosen based on prior knowledge. Or, it could be chosen randomly. Alternatively, clustering results from other cluster methods, such as hierarchical techniques (Section 2.2.4), could be input as the initial partition. Additionally, the scheme for rearranging objects can be different. For example, in minimizing $tr(W)$, Forgy's method ([33]) will reassign each object in the data to its closest cluster center simultaneously in one pass without changing the current cluster centers, but with MacQueen's method ([60]), the cluster means needs to be updated after the movement of every single object.

Obviously, these algorithms are limited in finding the *global* optimum partition in the sense that only partitions improving the cluster criteria with a single object movement are considered at each stage. It has been found that choice of the initial partition may have a dramatic influence on the final optimum solution ([37, 11]). Different initial partitions could

lead to different *local* optima of a clustering criterion. One possible way to avoid finding the *local* optimum is to run an algorithm many times with varying sets of starting partitions, and choose among the resulting partitions the one which gives the best result.

2.2.2 *k*-means algorithms

The well known *k*-means clustering is a typical example of partitioning techniques, of which the purpose is to minimize the trace of W (2.9). It has become one of the most popular clustering methods because it is computationally easy to implement and it is generally accessible in most statistical softwares and clustering packages.

Various algorithms have been developed to search for the optimal partition of the *k*-means clustering, which are frequently referred to as *k*-means algorithms since it involves the calculation of the mean (*centroid*) of each cluster. We only introduce the two widely used *k*-means algorithms in this section. First, Forgy ([33]) suggested a *k*-means algorithm consisting of the following steps:

1. Start with k randomly-selected initial centers (seed points). Obtain the initial partition by assigning each object to its closest center.
2. Recompute the centroids with the current arrangement of objects.
3. Assign each object to the cluster with the nearest centroid. The centroids remain unchanged for an entire pass through the set of objects.
4. If no movement of an object has occurred during a pass, stop. Otherwise, repeat step 2 and step 3.

The second one is proposed by MacQueen ([60]), and is the most widely used *k*-means algorithm. The main steps in this method are very similar to Forgy's algorithm, but it updates the cluster means after every single relocation of an object. A fault with Forgy's algorithm is that, during step 3, it could happen that none of the objects is assigned to certain centroids, thus the resulting partition actually has less than k clusters. This is not a problem with MacQueen's method, because an object can't change its membership if it is the only member in a cluster. However, a drawback with both of the algorithms is that "the results (sometimes strongly) depend on the order of the objects in the input file ([54])". Moreover, like all the other *hill-climbing* algorithms, different sets of initial seed points might lead to different local optima of the *k*-means algorithms.

2.2.3 Model-based clustering

Model-based clustering is a recently developed powerful clustering technique assuming finite mixture models for data. Since explicit probabilistic models are associated with data, the optimal partition of objects is determined such that the likelihood of the observed data is maximized. The Expectation-Maximization (EM) algorithm is employed to find the maximum likelihood estimate (MLE) of parameters in the model.

For quite a long time, finite mixture models of multivariate normals have been considered by many authors in cluster analysis. Examples are Day ([22]), Scott and Symons ([76]), Binder ([10]) and Symons ([82]). In a more recent work, Banfield and Raftery ([5]) proposed criteria more general than the $\det(W)$ criterion but more parsimonious than those based on the completely unconstrained covariance matrices assumption. The core of their argument is to express the covariance matrix for the m th component or cluster in the form

$$\Sigma_m = D_m \Lambda_m D_m', \quad (2.32)$$

where D_m is the matrix of eigenvectors determining the orientation and Λ_m is a diagonal matrix with the eigenvalues of Σ_m on the diagonal. Λ_m can be further written as $\Lambda_m = \lambda_m A_m$, where λ_m is the largest eigenvalues of Σ_m and $A_m = \text{diag}\{1, \alpha_2, \alpha_3, \dots, \alpha_p\}$. The size of the j th cluster is controlled by λ_m and its shape A_m . Thus, such reparameterization of Σ_m allows geometric features of the m th cluster to vary in terms of its orientation, size and shape. Restrictions on parameters D_m , λ_m and A_m incorporate particular assumptions about the cluster structure. Common parameterizations and the corresponding geometric interpretation is found in [8].

The finite mixture model of multivariate normals (2.19) together with the parameterization of the covariance matrix Σ_m makes up the main framework of model-based clustering. The MLE of parameters in the model can be estimated through the Expectation-Maximization (EM) algorithm ([25], [63]). In the EM algorithm, each object x_i is associated with a multinomial categorical variable $z_i = (z_{i1}, z_{i2}, \dots, z_{ig})$, z_{im} taking the value 1 if x_i is assigned to the m th component. In the context of cluster analysis, the complete data is $y_i = (x_i, z_i)$, where z_i is considered to be missing. Starting with initial values of z_i , the EM algorithm updates the estimate of parameters in the model iteratively, and stops when the convergence criteria are satisfied (see [35] for details).

For a review on model-based clustering, see [36]. Applications of model-based clustering can be found in [13], [20], [74], [87], [90], [99]. For clustering large data sets, improved

strategies are proposed by Wehrens et al. ([89]) and Fraley et al. ([36]).

2.2.4 Hierarchical clustering

Clustering techniques discussed above share the property that objects in a data set are partitioned into a specific number of clusters at a single step. Therefore, they all fall into the large class of clustering methods known as partitional clusterings. In contrast, hierarchical clusterings produce nested clusters through a series of partitions. Hierarchical clusterings can be either agglomerative, with fewer clusters at the higher level (by fusing clusters generated at the lower level), or divisive, which separate the n objects into more and finer groups in sequential steps. Hierarchical clustering methods differ in the numerical criteria used to determine distances between the clusters that will be merged (*agglomerative*) or subdivided (*divisive*) at the subsequent stage in a clustering process. A common shortcoming for hierarchical algorithms is that “divisions or fusions, once made, are irrevocable so that when an agglomerative algorithm has joined two individuals they cannot subsequently be separated, and when a divisive algorithm has made a split it cannot be undone ([30])”.

Agglomerative methods

Agglomerative hierarchical clustering starts with n clusters, each of which contains a single object in the data. In the second step, the two clusters that have the closest between-cluster distance are fused and they will be treated as a single cluster in the next step. As the procedure continues, it results in a single cluster containing all the n objects. Agglomerative methods vary in the ways of defining the distance between two clusters when more than one object are present in either of them. For example, the single linkage method considers the shortest pairwise distance between objects in two different clusters as the distance between the two clusters. In contrast with the complete linkage method, the distance between two clusters is defined as the distance between the most distant pair of objects. While, in the average linkage clustering, the average of the pairwise distances between all pairs of objects coming from each of the two clusters is taken as the distance between two clusters.

Divisive methods

Divisive hierarchical clustering also generates partitions of objects with a hierarchical structure, but it performs the partition in a direction opposite to agglomerative methods. It proceeds by splitting a cluster into two smaller groups. At the final stage, there are totally n clusters each containing only one object. Kaufman and Rousseeuw ([54]) commented that divisive methods have been largely ignored in the literature mainly because of limitations in the computational aspects. Notice that the first step of a divisive algorithm needs to compare $2^{n-1} - 1$ possible divisions of an n -object data set into two clusters (if this can be realized). It is certainly too computationally demanding, given the fact that $2^{n-1} - 1$ grows exponentially as n increases. In practice, divisive algorithms may only consider subsets of all possible partitions. Divisive algorithms following such ideas are either monothetic or polythetic.

Monothetic divisive methods (e.g. [91], [19]) are applicable to data only consisting of binary variables. At each stage, these algorithms choose a single variable as the basis to separate a cluster into two sub-groups. The variable is selected such that the distance between the two sub-groups is maximized ([43]). A cluster is then split according to the presence or absence of the selected variable. Advantages of monothetic methods include computational simplicity, the straightforward assignment of new objects and admissibility of objects with missing values. However, problems may arise if there exists a particular variable which is either rare or rarely found in combination with others ([30]).

Polythetic divisive methods use all variables at each stage. As Gordon ([43]) concisely describes, these algorithms “divide a cluster into two by successively removing objects from the cluster” ([59], [85]) or “select the pair of objects in the cluster with the largest pairwise dissimilarity to act as the seeds for the two sub-classes” ([49]).

2.2.5 Miscellaneous clustering methods

In the proceeding sections, we have reviewed the partitioning techniques, the model-based clustering and the hierarchical clustering methods. Beside these three major categories of clustering approaches, there are many other clustering methods that can not be categorized into any of them. We will briefly mention some important methods here. If the cluster is conceived as a region in the space with high density of points, then clusters can be found by density search clustering techniques (see [14], [15], [39], [16], [53], [93], [92], [95] and [96]). In

the cases where an object is not restricted to be assigned to a single cluster, techniques such as the *ADCLUS* method ([78]) and the *MAPCLUS* method ([3]) can be used to find overlapping clusters. In applications where the membership of objects is restricted by external information, clustering with constraints is necessary (e.g. the contiguity matrix approach [44]). Instead of assigning each object to a particular cluster, the fuzzy cluster analysis associates each object with a membership function indicating its *strength of membership* in all or some of the clusters. For references on fuzzy cluster analysis, see [54], [97], [9] and [47]. Recent developments in cluster analysis involves the application of neural networks. A well-known example is the self-organizing map (SOM) due to Kohonen ([55], [56]).

2.3 Determining the number of clusters

From the above discussions about various clustering techniques, it is clear that the number of clusters is generally an unknown parameter which needs to be either specified by users based on their prior knowledge or estimated in a certain way. A variety of methods have been proposed to estimate the number of clusters. Gordon ([44]) divided these methods into two categories: global methods and local methods. With the global methods, the quality of clustering given a specific number of clusters, g , is measured by a criterion, and the optimal estimate of g , \hat{G} , is obtained by comparing the values of the criterion calculated in a range of values of g . A disadvantage of most global methods is that there is no guidance for whether the data should be partitioned ($\hat{G} > 1$) or not ($\hat{G} = 1$). However, it will not be a problem if users have good reasons to believe that there are clusters present in data. The local methods are intended to test the hypothesis that a pair of clusters should be amalgamated. They are suitable for assessing only hierarchically-nested partitions. As Gordon commentes, the significance levels should not be interpreted strictly since multiple tests are involved in the procedure.

2.3.1 Global methods

In this section, we review several important global methods for estimating the best number of clusters in data. The behavior of these methods has been compared in recent research (see [84] and [81]).

Calinski and Harabasz's method

Milligan and Cooper ([70]) conducted a very comprehensive comparative study of 30 methods of determining the number of clusters in data. Among the methods examined in their work, the global method suggested by Calinski and Harabasz ([12]) generally outperformed the others. This approach determines \hat{G} by maximizing the index $CH(g)$ over g , where $CH(g)$ is given by

$$CH(g) = \frac{B(g)/(g-1)}{W(g)/(n-g)}, \quad (2.33)$$

and $B(g)$ and $W(g)$ are the between- and within-cluster sum of squared errors, calculated as the trace of matrix B (2.10) and W (2.9), respectively. $CH(g)$ is only defined for g greater than 1 since $B(g)$ is not defined when $g = 1$.

Hartigan's method

Hartigan ([45]) proposed the following index

$$Har(g) = \left[\frac{W(g)}{W(g+1)} - 1 \right] / (n - g - 1). \quad (2.34)$$

Intuitively, the smaller the value of $W(g)$, the higher similarity between objects which have the same cluster memberships. For fixed values of g and $W(g)$, $Har(g)$ will be sufficiently large if and only if $W(g+1)$ is sufficiently small. Thus, the idea is to start with $g=1$ and to add a cluster if $Har(g+1)$ is significantly large. The distribution of $Har(g)$ can be approximated by a F -distribution, which provides an approximated cut-off point. A simpler decision rule suggested by Hartigan is to add a cluster if $Har(g) > 10$. Hence, the cluster number is best estimated as the smallest g , $g = 1, 2, \dots$, such that $H(g) \leq 10$.

Krzanowski and Lai's method

As mentioned in section 2.2.1, Friedman and Rubin ([37]) proposed minimization of $|W|$ as a clustering criterion. Concerned with the problem of finding \hat{G} when using this method, Marriott ([61]) studied properties of $|W|$ in detail and proposed an approach based on $g^2|W|$. Krzanowski and Lai ([57]) examined the behavior of Marriott's $g^2|W|$ criterion by the Monte Carlo methods. They calculated the sample value of $g^2|W|/|T|$ when sampling from a homogeneous uniform population. The results showed that there was large discrepancy between the estimated value and the predicted value, especially when p and g were large. Instead, a

similar criterion using $g^2W(g)$ demonstrated much better consistency between the estimated value and the predicted value. Define

$$DIFF(g) = (g - 1)^{2/p}W(g - 1) - g^{2/p}W(g), \quad (2.35)$$

and

$$KL(g) = \left| \frac{DIFF(g)}{DIFF(g + 1)} \right|. \quad (2.36)$$

According to Krzanowski and Lai's argument ([57]), $KL(g)$, $g = 1, 2, \dots$, are expected to be randomly distributed around zero if the data came from a homogeneous uniformly distributed distribution. If the data came from a distribution with k modes, it can be expected that there is a dramatic decrease in $W(g)$ at $g = k$, and $W(g)$ only decreases by a little after $g = k$. Thus, they suggested a better criterion for which the optimum value of g is the one that maximizes $KL(g)$.

Silhouette statistic

Kaufman and Rousseeuw ([54]) proposed the *silhouette* index as to estimate the optimum number of clusters in the data. The definition of the *silhouette* index is based on the silhouettes introduced by Rousseeuw ([75]), which are constructed to show graphically how well each object is classified in a given clustering output. To plot the silhouette of the m th cluster, for each object in C_m , calculate $s(i)$ as

$$\begin{aligned} a(i) &= \text{average dissimilarity of object } i \text{ to all other objects in the } m\text{th cluster} \\ d(i, C) &= \text{average dissimilarity of object } i \text{ to all other objects in cluster } C, C \neq C_m \\ b(i) &= \min_{C \neq C_m} d(i, C) \\ s(i) &= \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}. \end{aligned}$$

The *silhouette* index, denoted by $\bar{s}(g)$, is defined as the average of the $s(i)$ for all objects in the data. $\bar{s}(g)$ is called the *average silhouette width for the entire data set*, reflecting the within-cluster compactness and between-cluster separation of a clustering. Compute $\bar{s}(g)$ for $g = 1, 2, \dots$. The optimum value of g is chosen such that $\bar{s}(g)$ is maximized over all g :

$$\hat{G} = \arg \max_g \bar{s}(g). \quad (2.37)$$

Gap method

Tibshirani *et al.* ([84]) proposed an approach to estimating the number of clusters in a data set via the gap statistic. This method is designed to be applicable to any cluster technique and distance measure d_{ij} . The idea is to compare the change in $W(g)$ as g increases for the original data with that expected for the data generated from a suitable reference null distribution. The best value of g is estimated as the value \hat{G} such that $\log(W(\hat{G}))$ falls the farthest below its expected curve.

Define

$$Gap_n(g) = E_n^*\{\log(W(g))\} - \log(W(g)), \quad (2.38)$$

where $E_n^*\{\log(W(g))\}$ indicates the expected value of $\log(W(g))$ under the null distribution. Then, the estimated number of clusters in data will be the value of g maximizing $Gap_n(g)$.

To apply this method, it is important to choose an appropriate reference null distribution. Considering k -means clustering, Tibshirani, *et al.* ([84]) proved that if $p=1$, the uniform distribution is the most likely to produce spurious clusters based on the gap test among all unimodal distributions. Unfortunately, they also proved that in the multivariate case ($p > 1$), there is no such generally applicable reference distribution: it may depend on the geometry of the particular null distribution. However, stimulated by the univariate case, they suggested two ways of generating reference data sets:

1. generate each reference variable uniformly over the range of the observed values for that variable;
2. generate the reference variables from a uniform distribution over a box aligned with the principal components of the data.

The first method is advantageous for its simplicity. The second one may be more effective in recovering the underlying cluster structure since it takes into considerations the shape of the multivariate distribution.

In detail, the computational procedures of the gap method are:

1. Cluster the data under investigation for fixed cluster number, g , where $g = 1, 2, \dots$. Compute $W(g)$ for all values of g ;
2. Generate B reference data sets in the way described above. Cluster each of the B reference data sets and calculate $W_b^*(g)$, $b = 1, 2, \dots, B$ and $g = 1, 2, \dots$. Compute

the gap statistic

$$Gap(g) = (1/B) \sum_b \log(W_b^*(g)) - \log(W(g));$$

3. Compute the standard deviation

$$sd_g = [(1/B) \sum_b \{\log(W_b^*(g)) - \bar{l}\}^2]^{1/2},$$

where $\bar{l} = (1/B) \sum_b \log(W_b^*(g))$;

4. Define $s_g = sd_g \sqrt{1 + 1/B}$. The optimum number of clusters is given by the smallest g such that $Gap(g) \geq Gap(g+1) - s_{g+1}$.

Jump method

Sugar and James ([81]) proposed an information-theoretic approach to estimating the number of clusters in a data set. Assume a random variable X comes from a mixture distribution of G components, each with covariance matrix Γ . Let c_1, \dots, c_g be a set of candidate cluster centers, and let c_x be the one closest to X . Define the minimum achievable distortion as

$$d(g) = \frac{1}{p} \min_{c_1, \dots, c_g} E[(X - c_x)' \Gamma^{-1} (X - c_x)].$$

Notice that if Γ is the identity matrix, the distortion is simply the mean squared error. $d(g)$ defined above can be estimated by \hat{d}_g , the minimum distortion obtained by applying the k -means clustering of the data. \hat{d}_g measures the within-cluster dispersion when partitioning the data into g clusters. Define the jump statistic as

$$J(g) = \hat{d}_g^{-Y} - \hat{d}_{g-1}^{-Y}, \quad g = 2, 3, \dots, \quad (2.39)$$

where Y is a transformation parameter, typically chosen as $p/2$. Define $\hat{d}_0^{-Y} \equiv 0$, so $J(1)$ is always equal to \hat{d}_1^{-Y} . With this method, the optimum number of clusters is estimated such that $J(g)$ is maximized over all values of g .

Based on asymptotic rate distortion theory results, Sugar and James provided rigorous theoretical proofs about properties of the jump statistic $J(g)$, assuming mixture distribution of X . The major result is that the jump statistic $J(g)$ will be maximized when $g = G$ provided that 1) each component of the distribution of X has finite fourth moments; 2)

there is sufficient separation between centers; 3) an appropriate transformation is used, that is a reasonable value of Y is chosen. However, the theoretical result only indicates the existence of Y , instead of providing an explicit guidance for the choice of Y . In practice, the authors suggested the use of $Y = p^*/2$, where p^* is the *effective dimension* in a real data. Empirically, the jump method, a quite non-parametric approach, performs very well provided the presence of well-separated clusters in data and a correct choice of Y . Nevertheless, our experiences with this method are that it may be difficult to estimate the *effective dimension* in a data set given little knowledge about the data structure. It is possible to try several values of Y , but it has been found that small variation in Y could lead to very different estimate of G .

2.3.2 Local methods

Among the five top performers in Milligan and Cooper's comparative study, two of them are local methods. The first one is proposed by Duda and Hart ([26]). In their method, the null hypothesis that the m th cluster is homogeneous is tested against the alternative that it should be subdivided into two clusters. The test is based on comparing the within-cluster sum of squared errors of the m th cluster, $J_1^2(m)$, with the within-cluster sum of squared distances when the m th cluster is optimally divided into two, $J_2^2(m)$. If the m th cluster contains n_m objects in p dimensions, then the null hypothesis should be rejected if

$$J_2^2(m)/J_1^2(m) < 1 - 2/(\pi p) - z[2(1 - 8/(\pi^2 p))/(n_m p)]^{1/2}, \quad (2.40)$$

where z is the cutoff value from a standard normal distribution specifying the significance level.

The second method proposed by Beale ([6]) tests the same hypothesis with a pseudo- F statistic, given by

$$F \equiv \left(\frac{J_1^2(m) - J_2^2(m)}{J_2^2(m)} \right) / \left(\left(\frac{n_m - 1}{n_m - 2} \right) 2^{2/p} - 1 \right). \quad (2.41)$$

The homogeneous one cluster hypothesis is rejected if the value of the F statistic is greater than the critical value from an $F_{p, (n_m - 1)p}$ distribution. In both tests, given the rejection of the null hypothesis, it follows that the subdivision of the m th cluster into two sub clusters is significantly better than treating it as a single homogeneous cluster.

To conclude this section, it is critical to mention that the decision about the optimum estimate of the number of clusters in the data should be based on results from several methods

instead of a single one. Although most of the methods are designed to be applicable to any clustering technique, the performance of a method may depend on assumptions of the cluster structure, as it is for the clustering technique itself. It is desirable to find consistent (at least partially) answers derived from the chosen methods, which may suggest the presence of clear cluster structure in data. If no agreement between methods, then the different answers should be interpretable under the particular context of research. Milligan ([69]) commented that non-interpretable results may indicate that there is no strong evidence for significant cluster structure in data.

Chapter 3

Determining the Number of Clusters Using the Weighted Gap Statistic

Estimating the number of clusters in a data set is a crucial step in cluster analysis. In this chapter, motivated by the gap method ([84]), we propose two approaches for estimating the number of clusters in data using the weighted within-clusters sum of errors: a robust measure of the within-clusters homogeneity. In addition, a “multi-layer” analytic approach is proposed which is particularly useful in detecting the nested cluster structure of data. The methods are applicable when the input data contain only continuous measurements and are partitioned based on any clustering method. Simulation studies and real data applications are used to show that the proposed methods are more accurate than the original gap method in determining the number of clusters.

3.1 Introduction

Cluster analysis, also referred to as the unsupervised classification, is a powerful statistical technique in exploring data structure. Although various clustering methods are available, most of them require the number of clusters, usually unknown in practice, to be pre-specified before conducting the clustering procedure. Since the resulting partition of the data objects depends on the specification of the cluster number, it is crucial to develop efficient methods of determining the number of clusters.

A large number of methods have been proposed to deal with this best-number-of-clusters

problem. In earlier works, Milligan and Cooper ([70]) did a comprehensive comparison of 30 criteria for estimating the number of clusters. Although constrained to the types of data considered in their simulations, Milligan and Cooper provided good indications of the performances of these criteria in clustering well-separated data. As discussed in [44], several criteria, such as Calinski and Harabasz’s index ([12]), demonstrated better properties under most situations considered in Milligan and Cooper’s study. Other important methods discussed in the literature include Hartigan’s rule ([45]), Krzanowski and Lai’s index ([57]) and the *silhouette* statistic suggested by Kaufman and Rousseeuw. Examples of more recent developments in determining the number of clusters include an estimating approach using the approximated Bayes factor in model-based clustering developed by Fraley and Raftery ([35] and [36]), the gap method proposed by Tibshirani *et al.* ([84]), the prediction-based resampling method (Clest) by Dudoit and Fridlyand ([27]) and the jump method by Sugar and James ([81])

Consider a multivariate observation $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})'$, $i = 1, \dots, n$, containing n independent objects measured on p variables. For any partition of the n objects into g clusters (P_g), denote by C_m the set of objects allocated to the m th cluster and by n_m the number of objects in C_m , $m = 1, \dots, g$. Denote by $d_{i,i'}$ the distance between objects i and i' . The sum of pairwise distances between objects in the m th cluster is given by

$$D_m = \sum_{i,i' \in C_m} d_{i,i'}. \quad (3.1)$$

For a fixed value of g , define

$$W_g = \sum_{m=1}^g \frac{1}{2n_m} D_m. \quad (3.2)$$

Note that W_g in (3.2) is a typical measure of the within-clusters homogeneity associated with P_g , a small value of which reflects a good fit of a classification to the “true” cluster structure of data.

In the above definition of W_g , $d_{i,i'}$ can be any arbitrary measure of distance. If the squared Euclidean distance is used, simple mathematical derivation shows that W_g is monotonically decreasing in g . Hence, W_g is not informative in choosing the optimal number of clusters by itself. However, for data strongly grouped around G centers, it is expected that the value of function W_g will drop quickly as g increases until it reaches the “true” number of clusters in the data. Intuitively, W_g will decrease at a much slower rate when $g > G$ since with more than G centers, objects belonging to the same cluster will be partitioned.

Therefore, an “elbow” point in the curve of W_g may indicate the optimal estimate of the number of cluster in data.

In estimating the number of clusters in a data set, methods based on the W_g criterion are aimed at appropriately determining the “elbow” point in W_g , where W_g is sufficiently small (e.g. [84] and [81]). The idea of the gap method is to compare the curve of W_g from the original data to the curve of the expected W_g ($E_n^*\{\log(W_g)\}$) under an appropriate null reference distribution. The best estimate of the cluster number is \hat{g} if W_g falls farthest below the expected curve at $g = \hat{g}$. Defining the gap statistic as

$$Gap_n(g) = E_n^*\{\log(W_g)\} - \log(W_g), \quad (3.3)$$

the estimate \hat{g} is the value of g which maximizes $Gap_n(g)$.

An essential step of the gap method is to generate suitable reference data sets which are used to obtain the benchmark of the within-clusters dispersion for comparison. The reference data can be generated by incorporating information about the shape of the data distribution. By definition, application of the gap method does not depend on the clustering method used. For example, Tibshirani *et al.* ([84]) implemented the gap method under the contexts of both K -means and hierarchical clustering methods in their research. Simulation studies showed that the gap method is a potentially powerful procedure in estimating the number of clusters for a data set. Moreover, the gap method has the advantage over most of the other estimating methods (see [84]) that it can be used to test the null hypothesis about homogeneous non-clustered data against the alternative of clustered data.

However, a deficiency of the gap method in finding the correct number of clusters has been demonstrated in more recent studies. For example, the gap method failed to detect the 4-cluster structure in the simulated data which contain well-separated clusters generated from distinct exponential distributions ([81]). In microarray data analysis, Dudoit and Fridlyand ([27]) developed the Clest method and compared it with several other existing methods including the gap method. They noted that the gap method tends to overestimate the number of clusters. One possible reason for such a deficiency in using the gap method may be because W_g , a statistic summarizing the within-clusters homogeneity, is not suitable in measuring the clustering adequately.

In this paper, we propose two alternative methods for determining the number of clusters which will be shown to perform more efficiently than the original gap method under many situations. These two methods are described in Sections 3.2.1 (the weighted gap method) and 3.2.2 (the DD-weighted gap method), respectively. Furthermore in Section 3.3.2, we

propose a “multi-layer” clustering approach that has a benefit of determining number of clusters not too “aggressively” nor too “conservatively.” Simulation studies and real data analysis are conducted in Section 3.3. Finally, Section 3.4 contains discussion.

3.2 Methods using the weighted gap statistic

3.2.1 The weighted gap method

Consider the m th cluster (C_m) and its associated sum of pairwise distances D_m defined in (3.1). There are $n_m \times (n_m - 1)$ pairs of different objects in C_m , thus $D_m/[n_m(n_m - 1)]$ is the averaged sum of the pairwise distances between all points in cluster m . Define $\bar{D}_m = D_m/[2n_m(n_m - 1)]$ and the weighted W_g as

$$\bar{W}_g = \sum_{m=1}^g \bar{D}_m = \sum_{m=1}^g \frac{1}{2n_m(n_m - 1)} D_m. \quad (3.4)$$

In (3.4), let $\bar{D}_m = 0$ when $n_m = 1$. In this paper, we are concerned about data containing only continuous variables, thus we consider the squared Euclidean distance $d_{i,i'}$, the most popular distance measure for continuous data. However, the approaches discussed here may be easily extended to other distance measures. If $d_{i,i'}$ is the squared Euclidean distance between objects i and i' , then \bar{D}_m is the average squared distance between objects in cluster m and its cluster mean. In the univariate case ($p=1$), \bar{D}_m will be the sample variance of the objects assigned to the m th cluster. It is easy to show that \bar{D}_m is an unbiased estimate of the population variance associated with cluster m , a constant. Provided that enough points are observed to represent the population of cluster m , this estimate is quite stable. This is in contrast to the D_m criterion, which is highly dependent on the number of points allocated to the m th cluster at a certain stage of clustering. In other words, the \bar{W}_g criterion is supposedly more robust than W_g in quantifying the overall within-clusters homogeneity with respect to variations in the observed or reference samples as long as each of the g clusters contains sufficient elements of estimating its within-clusters variation. This is a desirable property since it will guarantee that the same estimating result can be duplicated in data replicates. Hence, we expect that an estimation criterion using \bar{W}_g would provide a better solution to the best-number-of-clusters problem.

Suppose the data are well separated into G clusters. It is expected that \bar{W}_g will decrease dramatically as g increases until it reaches the true number of clusters (G) in a data set; for

$g > G$, the value of \overline{W}_g may go up or down, but it should only differ from \overline{W}_G by a little. Based on such an expectation about \overline{W}_g , it falls farthest below the expected curve at $g = G$. Thus, we propose a new criterion of estimating the number of clusters in data based on \overline{W}_g . Define the weighted gap statistic as

$$\overline{Gap}_n(g) = E_n^*\{\log(\overline{W}_g)\} - \log(\overline{W}_g). \quad (3.5)$$

The best estimate of the number of clusters is \hat{g} such that $\overline{Gap}_n(g)$ is maximized at $g = \hat{g}$.

Computationally, the weighted gap statistic is calculated following the same routine as the computation of the original gap statistic except that W_g is replaced by \overline{W}_g . There are two options for generating reference data sets. The weighted gap/uni method simply generates each reference variable uniformly over the range of the observed values of that variable. The weighted gap/pc method, where “pc” stands for principal components, is used to account for the shape of data distribution. Denote by X the original data and by X^* the centered data, obtained by subtracting the sample mean of each column of X from elements in the corresponding column of X . Apply the singular value decomposition to X^* and assume that $X^* = UDV'$. Let $X^{**} = X^*V$ and draw uniform data Y over the ranges of variables in X^{**} . The final reference data is $Z = YV'$.

Specifically, computational procedures of the weighted gap method are as follows.

Step 1: Using an appropriate clustering method to cluster the data with the number of clusters fixed at $g = 1, 2, \dots, K$; compute the corresponding value of \overline{W}_g , $g = 1, 2, \dots, K$.

Step 2: Generate B reference data sets using one of the two methods (gap/uni or gap/pc) described above, each containing n objects; cluster each reference data set to obtain \overline{W}_{bg} , $b = 1, 2, \dots, B$, $g = 1, 2, \dots, K$. Compute the weighted gap statistic

$$\overline{Gap}_n(g) = \frac{1}{B} \sum_b \log(\overline{W}_{bg}^*) - \log(\overline{W}_g).$$

Step 3: Let $\bar{l} = \frac{1}{B} \sum_b \log(\overline{W}_{bg}^*)$, compute the standard deviation

$$sd_g = \left[\frac{1}{B} \sum_b \{\log(\overline{W}_{bg}^*) - \bar{l}\}^2 \right]^{1/2}.$$

Step 4: Let $s_g = sd_g \sqrt{(1 + 1/B)}$. The best estimate of the number of clusters is determined via the “1-standard-error” style of rule such that

$$\hat{G} = \text{smallest } g \text{ such that } \overline{Gap}_n(g) \geq \overline{Gap}_n(g+1) - s_{g+1}.$$

In Figure 3.1 we compare the behavior of functions \overline{W}_g and W_g . Two simulated data sets, one with two clusters and the other with six clusters, are investigated, of which the K -means clustering method is used to classify the data with the number of clusters specified as 2, 3, ..., 10. Observe that the function W_g is monotonically decreasing with increasing g . However, \overline{W}_g is not necessarily decreasing in g . We applied both the gap and the weighted gap methods to estimate the true number of clusters in the data.

In Figure 3.2(a) and (b), we plot the corresponding gap and weighted gap functions. As expected, in both cases, the weighted gap is monotonically increasing in g as $g \leq G$ and the maximum value of $\overline{Gap}_n(g)$ is achieved when g is equal to G . Hence, the weighted gap method correctly finds the true numbers of clusters in these two data sets. However, for the six-cluster example, the gap curve has a local peak at $g = 2$, which then provides the estimate of G as 2 according to the “1-standard-error” style of rule. In Section 3.3, more intensive comparative studies will be presented to examine the behavior of the weighted gap method under different scenarios.

3.2.2 DD-weighted gap method

In the preceding section, we proposed the weighted gap method of estimating the number of clusters. Following the idea of the original gap method, the weighted gap method employs the “1-standard-error” rule in obtaining the optimal estimate in order to avoid unnecessary clusters. An advantage of such a rule is that it can be used to test the null hypothesis ($G = 1$) against the alternative ($G > 1$). However, when $G > 1$, it has been found that the original gap method has the tendency to overestimate the number of clusters in practice (see examples in [27] and discussions in Section 3.3). Our experience with the weighted gap method is that it may also overestimate G , although performing better than the original gap method (see Section 3.3.3 for an example).

In this section, a new stopping rule called DD-weighted gap method is proposed which can be used to find the best estimate of the cluster number more effectively. The basic idea is to search for the optimal estimate of G such that the observed \overline{W}_g is sufficiently small compared with its expected value under an appropriate reference distribution. However, instead of choosing the optimal cluster number as to maximize $\overline{Gap}_n(g)$ over g , the best estimate of G is chosen in a way that clustering with a specific number of clusters provides a much better fit to the data than clustering with one cluster less and, at the same time, adding one more cluster brings only little or even no improvement in terms of the $\overline{Gap}_n(g)$

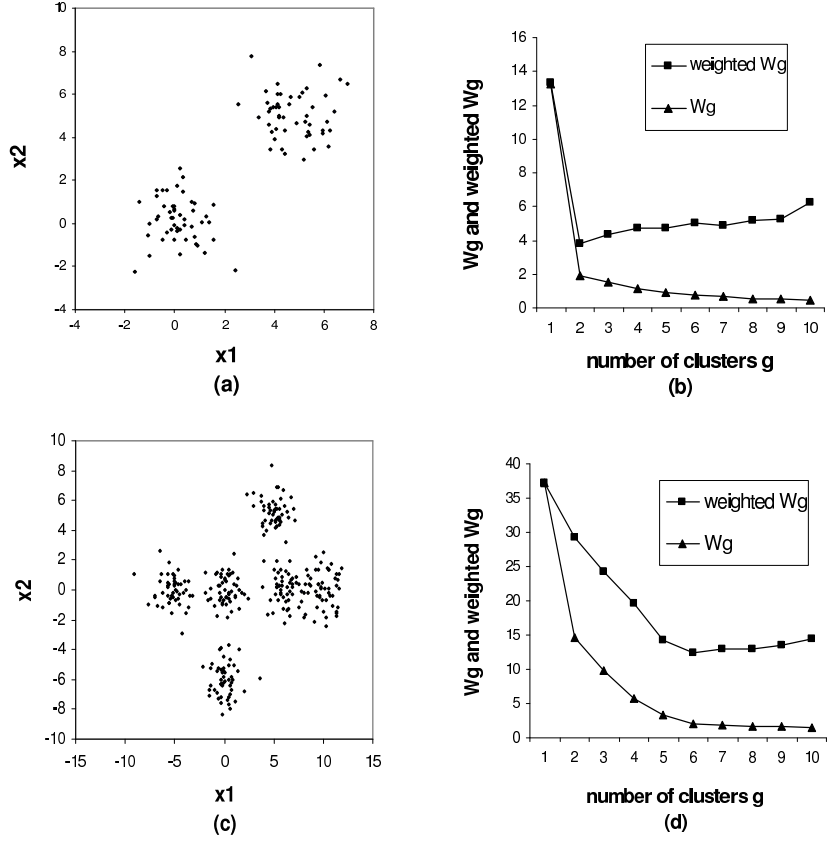


Figure 3.1: Plots of W_g and \overline{W}_g , the weighted W_g : (a) a two-cluster data; (b) within-clusters dispersion W_g and the weighted within-clusters dispersion \overline{W}_g for the data in (a); (c) a six-cluster data; (d) within-clusters dispersion W_g and the weighted within-clusters dispersion \overline{W}_g for the data in (c). For the convenience of demonstration, W_g/n is actually plotted instead of W_g , where n is the sample size of the data.

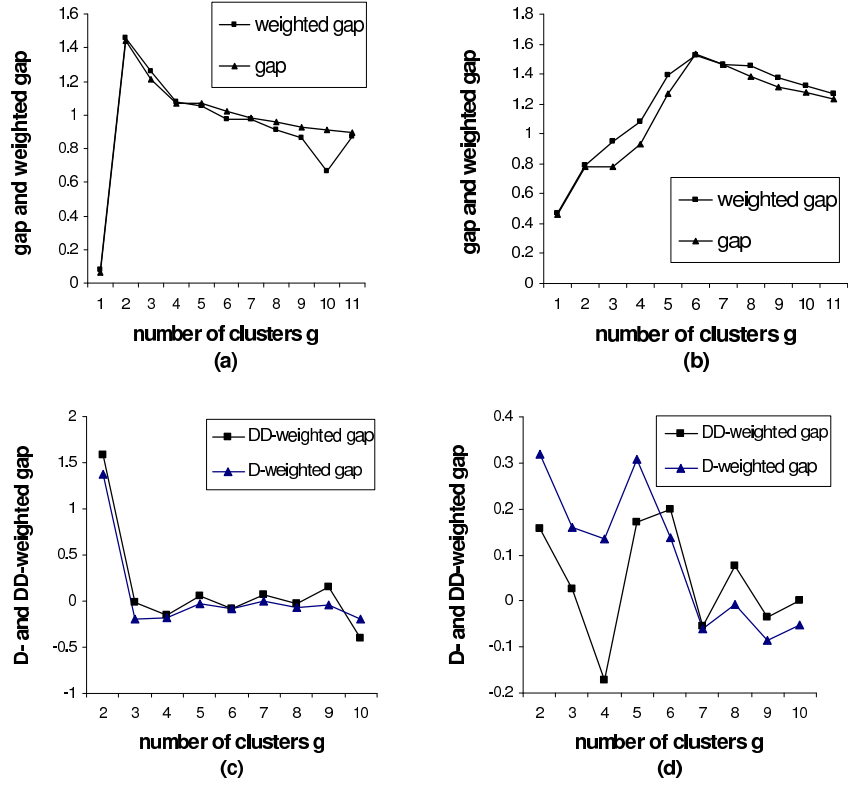


Figure 3.2: Plots of $Gap_n(g)$ and $\overline{Gap}_n(g)$ vs. g : (a) the two-cluster data in Figure 3.1 (b) the six-cluster data in Figure 3.1; plots of $D\overline{Gap}_n(g)$ and $DD\overline{Gap}_n(g)$ vs. g : (c) the two-cluster data studied in Figure 3.1 (d) the six-cluster data studied in Figure 3.1.

criterion. In particular, such a “prominent” point in function $\overline{Gap}_n(g)$ will be determined by comparing $\overline{Gap}_n(g)$ with its adjacent neighbors: $\overline{Gap}_n(g-1)$ and $\overline{Gap}_n(g+1)$. Stopping rules based on the successive differences as a criterion have been discussed elsewhere (e.g., the KL index defined by Krzanowski and Lai [57]).

Denote by $D\overline{Gap}_n(g)$ the difference in the value of function $\overline{Gap}_n(g)$ when the cluster number increases from $g-1$ to g ($g \geq 2$) as

$$D\overline{Gap}_n(g) = \overline{Gap}_n(g) - \overline{Gap}_n(g-1). \quad (3.6)$$

Suppose the data are strongly grouped around G modes, based on our expectation about \overline{W}_g , it is expected that $D\overline{Gap}_n(g) > 0$ for $g \leq G$ and $D\overline{Gap}_n(g)$ will be close to zero when $g > G$, while $D\overline{Gap}_n(g+1) > 0$ for $g < G$ and $D\overline{Gap}_n(g+1) \approx 0$ when $g \geq 0$. Define

$$DD\overline{Gap}_n(g) = D\overline{Gap}_n(g) - D\overline{Gap}_n(g+1). \quad (3.7)$$

It is expected that $DD\overline{Gap}_n(g)$ will be maximized when g is equal to the true number of clusters. Hence, we propose a new rule of estimating the number of clusters based on $\overline{Gap}_n(g)$: $\hat{G} = g^*$ if $DD\overline{Gap}_n(g)$ is maximized at g^* . The procedure of determining the number of clusters based on this rule will be referred to as the DD-weighted gap method in the following discussions.

Figure 3.2 illustrates the expected properties of function $DD\overline{Gap}_n(g)$ via the two-cluster and six-cluster examples studied in Section 3.2. The plots of $D\overline{Gap}_n(g)$ coincide with the expectation stated above: $D\overline{Gap}_n(g)$ has positive values given $g \leq G$ and it gets close to zero (being either positive or negative) for $g > G$. Correspondingly, $DD\overline{Gap}_n(g)$ is maximized when g is equal to the true number of clusters in the data sets ($G = 2$ in Figure 3.2 (c) and $G = 6$ in Figure 3.2 (d)). Performance of this proposed method will be further examined in Section 3.3.

3.3 Simulation studies and applications

According to Gordon’s categorization ([44]), the weighted gap, the DD-weighted gap and the original gap methods are all global approaches which determine the best estimate of G by optimizing a specific criterion over a range of values of the number of clusters g . The purpose of this section is to compare the performance of our proposed methods with that of the original gap method using both simulated and real application data. For the simulation

studies, we generated data sets from 9 different models that contain clear cluster structures with known number of clusters. Results using the three different methods are summarized in Table 3.1. For a specific method, its efficiency is measured by the percentage of the data sets for which the number of clusters is correctly estimated, out of the total 50 simulated samples. To examine the performances of these methods in application, we applied them to three real data sets containing grouped data in Section 3.3.3 and found that our proposed methods can appropriately determine the best number of clusters while the gap method generally overestimated the cluster number.

3.3.1 Simulation studies

Simulation models considered in our study are described as follows:

- *Model 1*: homogeneous data in ten dimensions - 200 data points generated from a uniform distribution containing 10 independent $U(0,1)$ s.
- *Model 2*: six clusters in two dimensions - The clusters are standard normal variables, each containing 50 observations, centered at $(10,0)$, $(6,0)$, $(0,0)$, $(-5,0)$, $(5,5)$, $(0,-6)$.
- *Model 3*: four clusters in two dimensions - The clusters are generated from an exponential distribution with mean 1 in a square box with length 2. The means of the two features for the four clusters are $(0,0)$, $(0,-2.5)$, $(-2.5,-2.5)$, $(-2.5,0)$, respectively. There are 50 observations in each cluster.
- *Model 4*: two clusters in two dimensions with different within cluster variations and different sample sizes - The clusters contain two independent normal variables with mean vector $(0,0)$, identity covariance matrix I and sample size 100 for one cluster and mean vector $(5,0)$, covariance matrix $0.1I$ and sample size 15 for the other cluster.
- *Model 5*: four clusters in two dimensions with different nonidentity covariance matrices - The four cluster centers are $(-1.6,5)$, $(-5,-5)$, $(5,5)$ and $(8.5,8.5)$ and the within-cluster correlation coefficients of the 4 clusters are -0.7, -0.3, 0.3, 0.7, respectively. Each cluster has 50 sample points.
- *Model 6*: two elongated clusters in two dimensions - To get the two elongated clusters, set $x = y = t + z$ with t increasing by 0.01 from -0.5 to 0.5 and z being Gaussian noise with standard variance 0.1. Finally, x for all the points in the second cluster is decreased by 1.

- *Model 7*: three clusters in ten dimensions - Each of the three clusters contains 50 objects generated from the standard normal distribution with fixed cluster mean vector $(1.6, 1.6, \dots, 1.6)$, $(0, 0, \dots, 0)$ or $(-1.6, -1.6, \dots, -1.6)$.
- *Model 8*: four clusters in ten dimensions - The mean of each cluster is randomly generated from $N(0_{10}, 3.6I_{10})$. Each cluster contains 25 data points simulated from the normal distribution with identity covariance matrix and the generated mean vector. A simulated data set will be discarded if the Euclidean distance between any pair of points in two different clusters is less than 1.
- *Model 9*: six clusters in two dimensions - Each data set consists of six clusters with two independent standard normal variables. Clusters are centered at $(0,0)$, $(-1,5)$, $(10,-10)$, $(15,-10)$, $(10,-15)$ and $(25,25)$, with 50 data points in each.

We generated 50 data sets under each of the simulation contexts described above. The simulated data sets were clustered via K -means clustering, where $g = 1, 2, \dots, 10$. The gap, the weighted gap and the DD-weighted gap methods were implemented to estimate the true number of clusters (G) in the simulated data. To avoid the sensitivity of our methods due to non-optimal K -means clustering result, for each data set and each value of g , the classification result was obtained by running the K -means algorithm 200 times with different randomly-chosen starting partitions. Table 3.1 summarizes the results corresponding to each method under different simulation scenarios. As described in Section 3.2.1, Gap/uni , \overline{Gap}/uni and $DD\overline{Gap}/uni$ are the gap, weighted gap and DD-weighted gap methods respectively with reference data sets generated from appropriate uniform distributions; Gap/pc , \overline{Gap}/pc and $DD\overline{Gap}/pc$ are the gap, the weighted gap and the DD-weighted gap methods with reference data sets generated using the second option, respectively.

We see that in *Model 1*, the gap method and the weighted gap method were both effective in detecting the homogeneous data by estimating G as 1. Note that by definition, the DD-weighted gap statistic is not defined at $g = 1$ hence it can not be applied to determine whether $G = 1$ or $G > 1$. *Models 2 to 6* focus on data in 2 dimensions, in which the spatial structure of clusters can be conveniently visualized. Data generated from *Models 2, 3, 4* and *5* contain well separated clusters and it is known that $G > 1$. Clearly, we see that the two approaches we proposed in this paper (the weighted gap method and the DD-weighted gap method) performed much better than or equivalent to the gap method in estimating the true number of clusters G . *Model 6* is a situation where both the gap method and the weighted gap method had difficulty in revealing the 2-elongated-cluster structure, but the

Table 3.1: Results of comparing the weighted gap method, the DD-weighted gap method and the gap method in estimating the true number of clusters in simulated data: Model 1 to 9. The last column contains the estimating results when applying multi-layer clustering to each model which is introduced in Section 3.3.2. Numbers given in this table are the percentages of the total 50 data sets generated in each model.

<i>Model</i>	<i>Method</i>						<i>Multi – layer</i>
	<i>Gap</i> <i>/uni</i>	<i>Gap</i> <i>/pc</i>	\overline{Gap} <i>/uni</i>	\overline{Gap} <i>/pc</i>	$DD\overline{Gap}$ <i>/uni</i>	$DD\overline{Gap}$ <i>/pc</i>	
1	100%	100%	100%	100%	N/A	N/A	100%
2	38%	64%	98%	100%	90%	88%	98%
3	40%	30%	90%	64%	100%	100%	60%
4	20%	20%	94%	94%	96%	96%	94%
5	100%	100%	94%	96%	92%	96%	86%
6	6%	0	0	0	24%	88%	52%
7	100%	100%	100%	100%	0	96%	100%
8	100%	100%	98%	88%	54%	62%	96%
9	100%	100%	94%	100%	0	0	100%

DD-weighted gap/pc method appeared to be efficient enough in finding the two clusters in the data.

Since cluster analysis often deals with data measured on many variables, in *Model 7* and *8*, we examined performances of the proposed methods using simulated data of high dimensionality. For *Model 7*, the proposed two methods successfully detected the 3 clusters. *Model 8* is an interesting scenario concerned with random clusters with randomly generated cluster means. Table 3.1 shows that the weighted gap method as well as the original gap method gave satisfactory estimating results, while the DD-weighted gap method didn't detect the four-cluster data structure for nearly half of the total 50 simulated data sets. However, we should not be surprised by such an “unfavorable” outcome due to the unusual data structure in *Model 8*. It should be noted that the four clusters may not be evenly spaced in this model, which means that the distances between pairs of cluster centers (randomly generated) might be very different, so that the data actually contain nested clusters (e.g. the data in Figure 3.3 (a)). In the presence of such nested cluster structures, results given by the DD-weighted gap method will be determined by the dominant structure in a data set (i.e. the higher level 3-cluster structure shown in Figure 3.3 (b)). In *Model 8*, data could be dominated by a structure with less than four clusters, which correspondingly will be associated with an estimate of G less than four.

Model 9 is purposely designed to examine the behavior of the DD-weighted gap method given data that contain nested clusters. Figure 3.3 demonstrates the cluster structure in *Model 9* and the K -means clustering result given the number of clusters specified as 3. As displayed by the plot, this is a situation where the data contain multiple types of cluster structure: a dominant 3-cluster structure with smaller sub-clusters present within each cluster. Not surprisingly, the DD-weighted gap method estimated the number of clusters as 3 for each of the 50 trials since it tends to pick up the dominant cluster structure in a data set. As a strategy for revealing the nested structure in *Model 9*, the DD-weighted gap method can be applied to examine each of the three clusters found at the previous stage and determine if further separation is necessary. An analysis in such a sequential manner will be referred to as “multi-layer” clustering in next section.

3.3.2 Multi-layer clustering

It is worth emphasizing that the weighted gap method can be applied to test if the number of clusters in a data set (or the partial data) is equal to 1 (non-clustered structure) or

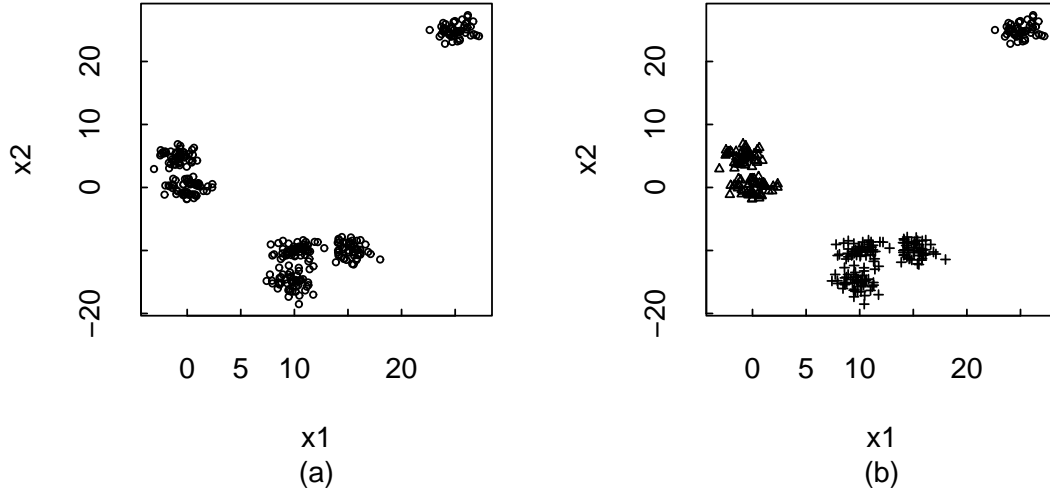


Figure 3.3: Illustration of the simulated data with nested clusters: (a) data; (b) classification result via K -means clustering where $g = 3$. In (b), the three clusters are distinguished by different types of markers.

greater than 1 (clustered structure), but the DD-weighted gap method is only defined for $g \geq 2$. Similar to the original gap method, the weighted gap method may also overestimate the number of clusters. Hence, we propose a “multi-layer” clustering which combines the weighted gap method and the DD-weighted gap method proceeds as follows:

1. Implement the weighted gap method on the data of interest to test the non-clustered situation against the clustered alternative;
2. If the estimate of the number of clusters in the data is 1, then stop and conclude that the data is non-clustered;
3. If the number of clusters was estimated as some number larger than 1, apply the DD-weighted gap method to determine the best estimate of the cluster number in the data;
4. Apply the above three steps to each of the clusters found at step 3);
5. Repeat step 4) until no subdivision within any clusters is needed.

We performed “multi-layer” clustering using the data sets generated from the models described in Section 3.3.1. In “multi-layer” clustering, there are two choices in carrying out

this approach: either use the uniformly distributed reference data or use the “pc” type of reference data. Results given in Table 3.1 show that the DD-weighted gap/pc method is an overall better performer in detecting evenly-spaced clusters. Thus, we chose to generate the “pc” type of reference data in order to take consideration of the shape of data distribution. Indicated by the multi-layer/pc method, the last column of Table 3.1 gives the corresponding estimating results. It should be emphasized that “multi-layer” clustering is particularly informative in that it may provide useful information about the optimal separation of a data at each stage of clustering. For example, in *Model 6*, it is known that the data were generated to contain two elongated clusters. In terms of the two elongated clusters, the multi-layer/pc method seems to perform worse than the DD-weighted gap/pc method (see Table 3.1), however, it should be noted that the multi-layer/pc method did find the two-elongated-cluster structure for 44 (88%) data sets simulated from this model at the first stage of this analysis. In further steps of analysis, the two main elongated clusters might be divided into smaller sub-clusters.

From the practical point of view, the DD-weighted gap method would be particularly suitable in detecting the presence of a “hierarchical” cluster structure in a data set, as demonstrated in *Models 8* and *9*. In “multi-layer” clustering, the hierarchy is formed depending on the nesting relationship between natural clusters in a data, which is different from the hierarchy considered in the widely used hierarchical clustering which is simply a property of this type of clustering method. Hence, in practice, the final clustering solution can be interpreted in two aspects: the property of each cluster in a specific application and the hierarchical structure represented by clusters.

3.3.3 Applications

In this section, we consider three data sets. Table 3.2 summarizes the estimates of the “true” number of clusters G for these three data sets when the gap method, the weighted gap method and the DD-weighted gap method are employed independently. We also applied “multi-layer” clustering to these real data and the estimated number of (sub-)clusters in the “second-layer” analysis are given in Table 3.3.

1) Iris data

The Iris data, studied by Fisher ([31]) in linear discriminant analysis, contains 150 objects in total, each of which was measured on four variables: length of sepal, width of

Table 3.2: Estimates of the number of clusters of real data sets via the gap method, the weighted gap method, the DD-weighted gap method and multi-layer/pc clustering. G is the known number of clusters in data.

<i>Method</i>	Iris ($G = 2/3$)	Breast-Cancer ($G = 2$)	Leukemia ($G = 3$)
\overline{Gap}/uni	6/8	9	13
\overline{Gap}/pc	4	9	3
$\overline{G\overline{ap}}/uni$	6	2	3
$\overline{G\overline{ap}}/pc$	4	2	3
$\overline{DDG\overline{ap}}/uni$	2	2	3
$\overline{DDG\overline{ap}}/pc$	2	2	3
<i>Multi – layer/pc</i>	3	3	3

Table 3.3: The number of sub-clusters of real data sets estimated in multi-layer clustering.

<i>Data</i>	\overline{Gap}/pc	$\overline{DDG\overline{ap}}/pc$
Iris C_1	1	4
Iris C_2	2	2
Breast Cancer C_1	2	2
Breast Cancer C_2	1	7
Leukemia C_1	1	8
Leukemia C_2	1	6
Leukemia C_3	N/A	N/A

sepal, length of petal and width of petal. The objects are categorized by three species (Iris sotosa, Iris versicolor and Iris verginica). It has been found that Iris sotosa could be well separated from the other two species, while Iris versicolor and Iris verginica are somewhat overlapping.

The Iris data set does not have clear cluster structure because of the presence of the two overlapping species. And it will be reasonable to conclude that the data contain either two or three clusters. When each method was utilized independently, both the gap and the weighted gap methods overestimated G , while the DD-weighted gap method provided an appropriate estimate of G ($\hat{G} = 2$) (see Table 3.2).

In terms of the whole Iris data set with 150 objects, when $g = 2$, the data were separated into two clusters containing 97 (Iris C_1) and 53 (Iris C_2) data points, respectively. Following the idea of “multi-layer” clustering, we explored further into the two clusters found at this stage. For Iris C_1 , the weighted gap/pc method estimated the cluster number as 1 (see Table 3.3), so no separation of this cluster is needed. It is a reasonable solution because, based on our prior knowledge, Iris C_1 corresponds to Iris versicolor and Iris verginica, two overlapping species, which can not be well separated using the K -means method. The result for Iris C_2 is clear, which determined two sub-clusters within this cluster. According to the classification of Iris C_2 when $g = 2$, one sub-cluster of Iris C_2 contains the 50 Iris sotosa samples, and the other one is formed by the remaining 3 Iris versicolor objects misclassified by the K -means method in the previous stage. Hence, it explains that why Iris C_2 needs to be further separated into two sub-clusters.

II) Wisconsin breast cancer data

In order to show that our proposed methods are applicable to data of high dimensionality, the Wisconsin breast cancer data ([94]) which contain measurements on nine variables recorded on biopsy specimens of 683 patients were chosen. This data consists of two distinct clusters which correspond to one group of the 444 benign specimens and the other group containing the remaining 239 malignant specimens. For the whole data set, both the weighted gap method and the DD-weighted gap method found the correct estimate of G ($\hat{G} = 2$) in data, but the gap method overestimated G in this case as well.

In addition, “multi-layer” clustering was conducted to determine if there are any sub-clusters nested within the two clusters found above. By the estimating results presented in Table 3.3, one of the two clusters of the breast cancer data (Breast Cancer C_1) needs further separation into two sub-clusters. Interpretations of the properties of the sub-clusters

detected may rely on specific knowledge about this data set. No sub-clusters was found in Breast Cancer C_2 because the estimate of the number of sub-clusters given by the weighted gap/pc method is 1.

III) Leukemia data

The leukemia data is an example with measurements on very large number of variables. This data set came from a microarray experiment exploring two types of acute leukemia: acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). More detailed descriptions about this data set can be found in [42]. It contains measurements of the gene expression levels of 6817 genes for three classes of mRNA samples: 38 ALL B-cell, 9 ALL T-cell and 25 AML. Prior to cluster analysis, the data were processed (missing data imputation, log transformation, standardization and gene selection) according to the procedures described in [27] so that only the top 100 most variable genes were selected for estimating the number of clusters in the data. Finally, the 72 mRNA samples were clustered based on a 72×100 data matrix.

Based on one-step analyses, all the methods, except for the gap/uni method, consistently provided the correct estimate of G as 3 (see Table 3.2). The “multi-layer” analysis was also performed to detect the possible existence of nested sub-clusters. Since Leukemia C_3 only contains a few (9) samples, there is no need to divide it any more. Each of the other two clusters (Leukemia C_1 and Leukemia C_2) was analyzed via the weighted gap/pc and the DD-weighted gap/pc methods. Since the weighted gap/pc method gives 1 as the estimated sub-cluster number in these two clusters, we conclude that the leukemia data is represented by a distinct 3-cluster structure.

In practice, other than using a given method to automatically estimate the number of clusters, further useful information may be obtained by utilizing graphical tools. For the three data sets considered in this section, Figure 3.4 presents the corresponding curves of the gap statistic, the weighted gap statistic and the DD-weighted gap statistic. Examining carefully the plots of function $Gap_n(g)$ associated with the three data sets, we see that there are no obvious peaks in these curves, which is not what would be expected for well-grouped data. It then indicates that the results should be cautiously treated. Correspondingly, the gap method did overestimate the number of clusters in these applications. Similarly, a warning signal arose when applying the weighted gap method to the Iris data where this method also incorrectly overestimated the cluster number. In addition, if the data contain multiple types of cluster structures, more than one outstanding peaks may be observed in

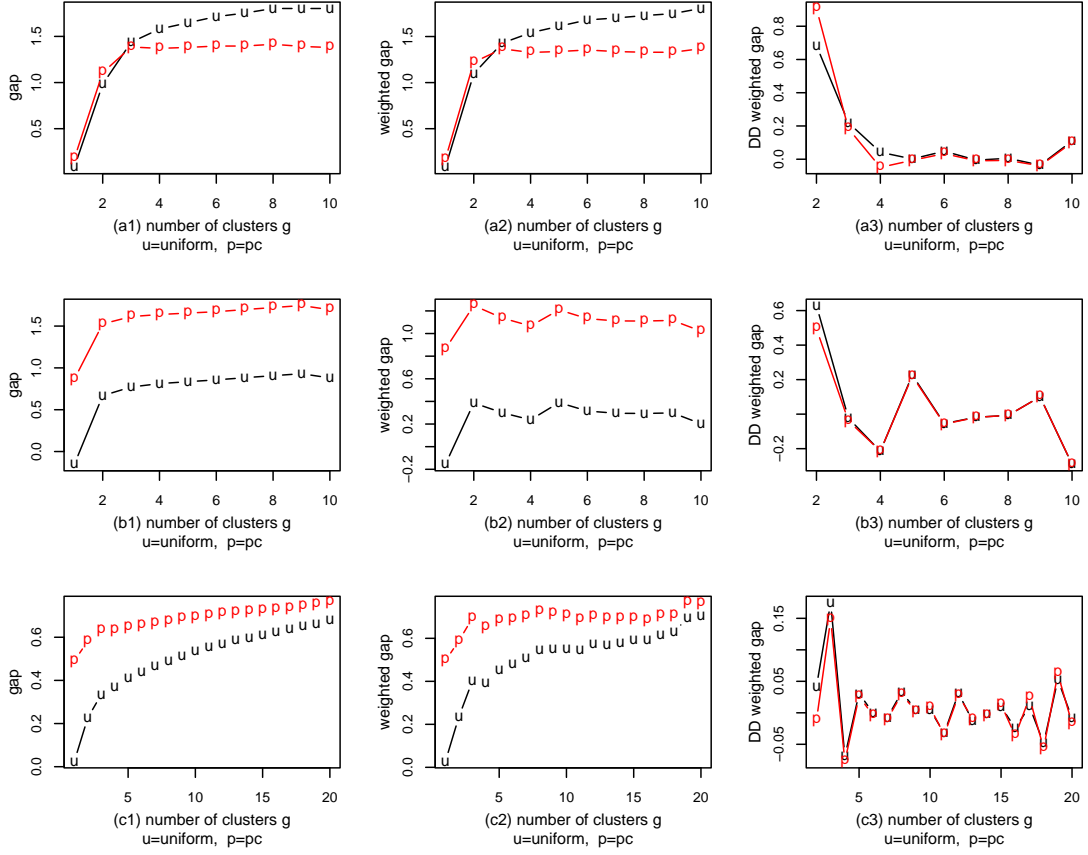


Figure 3.4: Results of estimating the number of clusters of real data sets: (a1)~(a3) plots of $Gap_n(g)$, $\overline{Gap}_n(g)$ and $DD\overline{Gap}_n(g)$ vs. g for the Iris data; (b1)~(b3) plots for the Wisconsin breast cancer data; (c1)~(c3) plots for the leukemia data.

the curves of the criterion statistics. In terms of the Iris data, the second largest value of $DD\overline{Gap}_n(g)$ occurs at $DD\overline{Gap}_n(3)$ (0.207), which is still quite large compared with other points in the function. Thus we would suspect that further separation of the 2 most distinct clusters might be necessary, which was supported by the “multi-layer” clustering result described above. Similarly, a plot of the DD-weighted gap function corresponding to the Wisconsin breast cancer data also provides strong evidence for the need for second layer clustering. Based on the above discussion, it is highly recommended that the user examine plots of the weighted gap statistic and the DD-weighted gap statistic vs. the number of clusters g in an application.

3.4 Discussion

Based on the comparative studies discussed in the previous section, we conclude that the weighted gap and the DD-weighted gap methods are highly effective in determining the number of clusters in a data set. In practice, estimates of G for the same data set provided by the two proposed methods do not have to be the same, particularly when there is no distinct separation between clusters in the data. An advantage of the weighted gap method is that it is defined for $g = 1$, so that it can be applied to test the null case ($G = 1$) against the alternative case ($G \geq 2$) while the DD-weighted gap method is applicable only when $G \geq 2$. In our experience, answers given by the DD-weighted gap method could be a locally optimal estimate of G instead of the global optimum, especially when the data cluster structure is complicated. An important situation is when the data contain nested clusters where larger clusters consisting of smaller clusters represent the dominant cluster structure in the data. Then it is highly possible that the DD-weighted gap method will be dominated by the cluster pattern represented by the larger clusters. It will be helpful to check the plot of the $DD\overline{Gap}_n(g)$ function, which may indicate the presence of such hierarchical cluster structure by showing more than one prominent local peaks. In such cases, the “multi-layer” analysis proposed in Section 3.3.2 should be used to achieve better understanding of the cluster structure in the data.

The pooled within-clusters variation is probably the most popular criterion in cluster analysis. A lot of approaches for estimating the number of clusters in data have been developed aiming at optimizing this criterion. Such approaches have the merit of simplicity in computation. In this paper, we suggest to use the weighted within-clusters dispersion as a criterion for measuring goodness-of-fit in cluster analysis. Our proposed criterion retains the advantage of computational convenience. More importantly, the weighted within-clusters dispersion enables better robustness in estimating the number of clusters. We observed the effect of such a property in the analysis of the Iris data in Section 3.3.3. Although the weighted within-clusters dispersion can be related to a measure of cluster homogeneity based on the dissimilarity matrix (see [30]), it has not been applied to cluster analysis according to our knowledge.

In terms of the best-number-of-clusters problem, two estimation rules are proposed for determining the best estimate of the number of clusters. Both methods may effectively find the correct number of clusters in a data set when used independently. Moreover, for data which contain multiple types of cluster structures, the two methods can be combined

in application. This provides an extremely efficient solution in finding complicated cluster structure in data. In the literature, the idea of sequentially detecting cluster structure is not new, but it has never been formally developed into an applicable procedure. It is expected that the strategy of utilizing the weighted gap method and the DD-weighted gap method in combination would be particularly efficient in analyzing large-scale data such as microarray gene expression data.

Although we only considered the k -means clustering method in our discussions, the definition of our proposed methods actually allows us to use any arbitrary clustering technique. It is worthwhile to mention that the resulting estimation may be dependent on the clustering method. It is hence a prerequisite to choose an appropriate clustering method which may provide reasonable classification given the correct number of clusters is specified. In practice, the choice of clustering method can be guided by prior knowledge in certain research fields. If no such prior information is available, the final estimation result is justified if it is interpretable. In the case that the result has no meaningful interpretation, it should raise cautions about either the choice of clustering technique or the method of estimating the number of clusters.

Appendix A: Proof of the monotonically decreasing property of W_g in g

Suppose the observed data contain n objects with p variables. Each object can be expressed as $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})'$, $i = 1, \dots, n$. For a specific partition of the n objects into g clusters, let C_m indicate the set of objects allocated to the m th group, n_m the number of objects in C_m , $m = 1, \dots, g$. Define the dispersion matrix for each cluster, $W(m)$, as

$$W(m) = \sum_{i \in C_m} (x_i - \bar{x}_m)(x_i - \bar{x}_m)',$$

where

$$\bar{x}_m = \frac{1}{n_m} \sum_{i \in C_m} x_i.$$

The pooled within-cluster dispersion matrix W is given by

$$W = \sum_{m=1}^g \sum_{i \in C_m} (x_i - \bar{x}_m)(x_i - \bar{x}_m)'.$$

Then, the within-clusters sum of errors corresponding to this partition is simply the trace of matrix W , denoted by W_g .

When $g = 1$,

$$W_1 = \text{tr} \left[\sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})' \right] = \sum_{i=1}^n (x_i - \bar{x})' (x_i - \bar{x}),$$

where $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$.

For any arbitrary partition of the n objects into 2 clusters ($g = 2$), the within-clusters sum of errors is

$$\begin{aligned} W_2 &= \text{tr} \left[\sum_{m=1}^2 \sum_{i \in C_m} (x_i - \bar{x}_{2m})(x_i - \bar{x}_{2m})' \right] \\ &= \sum_{m=1}^2 \sum_{i \in C_m} (x_i - \bar{x}_{2m})' (x_i - \bar{x}_{2m}), \end{aligned}$$

where $\bar{x}_{2m} = \frac{1}{n_m} \sum_{i \in C_m} x_i$, $m = 1$ or 2 .

Rewrite W_1 as the follows

$$\begin{aligned} W_1 &= \sum_{i=1}^n (x_i - \bar{x})' (x_i - \bar{x}) \\ &= \sum_{m=1}^2 \sum_{i \in C_m} (x_i - \bar{x})' (x_i - \bar{x}) \\ &= \sum_{m=1}^2 \sum_{i \in C_m} (x_i - \bar{x}_{2m} + \bar{x}_{2m} - \bar{x})' (x_i - \bar{x}_{2m} + \bar{x}_{2m} - \bar{x}) \\ &= \sum_{m=1}^2 \sum_{i \in C_m} [(x_i - \bar{x}_{2m})' (x_i - \bar{x}_{2m}) + (\bar{x}_{2m} - \bar{x})' (\bar{x}_{2m} - \bar{x}) + 2(x_i - \bar{x}_{2m})' (\bar{x}_{2m} - \bar{x})] \end{aligned}$$

Note that $\sum_{i \in C_m} (x_i - \bar{x}_{2m})' (\bar{x}_{2m} - \bar{x}) = 0$. Then

$$\begin{aligned} W_1 &= \sum_{m=1}^2 \left[\sum_{i \in C_m} (x_i - \bar{x}_{2m})' (x_i - \bar{x}_{2m}) + n_m (\bar{x}_{2m} - \bar{x})' (\bar{x}_{2m} - \bar{x}) \right] \\ &> \sum_{m=1}^2 \sum_{i \in C_m} (x_i - \bar{x}_{2m})' (x_i - \bar{x}_{2m}) = W_2. \end{aligned}$$

The inequality holds strictly since for $g = 2$, it must be true that at least one of \bar{x}_{2m} , $m = 1$ or 2 , is not equal to \bar{x} . Then, $\sum_{m=1}^2 \sum_{i \in C_m} 2(x_i - \bar{x}_{2m})' (\bar{x}_{2m} - \bar{x}) > 0$.

Suppose that when $g = 2$, the resulting cluster C_1 contains more than 1 observations, we can split it into two smaller nonempty clusters, say C_1^* and C_2^* . Then, cluster C_1^* , C_2^* and C_2

forms an arbitrary clustering of the data for $g = 3$. Denote the corresponding within-clusters sum of errors as W_3^* . Based on the above derivation, it is obvious that $W_2 > W_3^*$. But, since K -means clustering searches for the optimal solution which minimizes W_3 , it must be true that $W_3^* \geq W_3$. It follows that $W_1 > W_2 > W_3$. When applying the same argument to the situations where $g > 3$, we have $W_1 > W_2 > W_3 > W_4 > \dots$. Thus we prove that W_g , as a function of the number of clusters g , is monotonically decreasing in g .

Appendix B: Proof of the invariance of W_g and \bar{W}_g before and after centering a data set at any value of g , given a fixed partition of the original data.

The purpose of this proof is to show that the value of W_g and \bar{W}_g computed using the centered data is equal to that based on the original uncentered data. Hence, when reference data sets are generated from a box aligned with the principal components of the centered data, it is reasonable to compare W_g (\bar{W}_g) of the observed data with the expected value of W_g (\bar{W}_g) which is virtually computed based on the centered data.

I. Invariance of W_g

Suppose the observed data are expressed in the matrix format X , the i th row of which corresponds to the i th observation in the data. Let

- n = the total number of observations in a data set
- p = the number of variables for each observation
- g = the number of clusters specified in clustering the data
- x = the n by p matrix of data
- \bar{X} = the g by p matrix of cluster means
- Z = the n by g allocation matrix, where
- z_{ij} = 1 if the i th observation was assigned to the j th cluster.

Then, $\bar{X} = (Z'Z)^{-1}Z'X$ and the pooled within-clusters dispersion matrix is

$$\begin{aligned}
W &= (X - Z\bar{X})'(X - Z\bar{X}) \\
&= (X - Z(Z'Z)^{-1}Z'X)'(X - Z(Z'Z)^{-1}Z'X) \\
&= X'(I - Z(Z'Z)^{-1}Z')(I - Z(Z'Z)^{-1}Z')X \\
&= X'(I - Z(Z'Z)^{-1}Z')X \\
&= X'(I - H)X
\end{aligned}$$

where $H = I - Z(Z'Z)^{-1}Z'$. Note that H and $I-H$ are both idempotent matrices, since $HH=H$ and $(I-H)(I-H)=I-H$.

Denote the centered data by \bar{X}^* , then $X^* = (I - \frac{1}{n}J)X$ and $\bar{X}^* = (Z'Z)^{-1}Z'X^* = (Z'Z)^{-1}Z'(I - \frac{1}{n}J)X$. Hence, for the centered data, the pooled within-clusters covariance matrix is

$$\begin{aligned}
W^* &= (X^* - Z\bar{X}^*)'(X^* - Z\bar{X}^*) \\
&= ((I - \frac{1}{n}J)X - Z(Z'Z)^{-1}Z'(I - \frac{1}{n}J)X)'((I - \frac{1}{n}J)X - Z(Z'Z)^{-1}Z'(I - \frac{1}{n}J)X) \\
&= ((I - \frac{1}{n}J)X - H(I - \frac{1}{n}J)X)'((I - \frac{1}{n}J)X - H(I - \frac{1}{n}J)X) \\
&= X'((I - H)(I - \frac{1}{n}J))'(I - H)(I - \frac{1}{n}J)X \\
&= X'(I - H)X
\end{aligned}$$

The last equality used the fact that

$$HJ = Z(Z'Z)^{-1}Z'J = ZJ = J,$$

$$(I - H)(I - \frac{1}{n}J) = \frac{1}{n}HJ - \frac{1}{n}J + I - H = I - H,$$

and

$$(I - H)'(I - H) = (I - H)(I - H) = (I - H).$$

Thus, $W^* = W$. It follows immediately that $W_g^* = \text{tr}(W^*) = \text{tr}(W) = W_g$. That is, when the data is centered, W_g remains invariant.

II. Invariance of \bar{W}_g

Let I^* be the block diagonal matrix

$$I^* = \begin{pmatrix} \frac{1}{n_1}I_1 & 0 & \dots & 0 \\ 0 & \frac{1}{n_2}I_2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{n_g}I_g \end{pmatrix}$$

where I_i is the n_i by n_i identity matrix. For the original data, the weighted within-clusters

dispersion matrix is given by

$$\begin{aligned}
\bar{W} &= (X - Z\bar{X})' I^* (X - Z\bar{X}) \\
&= (X - Z(Z'Z)^{-1}Z'X)' I^* (X - Z(Z'Z)^{-1}Z'X) \\
&= X'(I - Z(Z'Z)^{-1}Z') I^* (I - Z(Z'Z)^{-1}Z')X \\
&= X'(I - H) I^* (I - H)X.
\end{aligned}$$

The weighted within-clusters covariance matrix for the centered data is given by

$$\begin{aligned}
\bar{W}^* &= (X^* - Z\bar{X}^*)' I^* (X^* - Z\bar{X}^*) \\
&= ((I - \frac{1}{n}J)X - Z(Z'Z)^{-1}Z'(I - \frac{1}{n}J)X)' I^* ((I - \frac{1}{n}J)X - Z(Z'Z)^{-1}Z'(I - \frac{1}{n}J)X) \\
&= ((I - \frac{1}{n}J)X - H(I - \frac{1}{n}J)X)' I^* ((I - \frac{1}{n}J)X - H(I - \frac{1}{n}J)X) \\
&= X'((I - H)(I - \frac{1}{n}J))' I^* (I - H)(I - \frac{1}{n}J)X.
\end{aligned}$$

We have shown that $(I - H)(I - \frac{1}{n}J) = \frac{1}{n}HJ - \frac{1}{n}J + I - H = I - H = (I - H)'$, thus $\bar{W}^* = X'(I - H) I^* (I - H)X = \bar{W}$. Hence, $\bar{W}_g^* = tr(\bar{W}^*) = tr(\bar{W}) = \bar{W}_g$. That is, when the data is centered, \bar{W}_g remains invariant.

Thus, we prove that both W_g and \bar{W}_g will be invariant after the data are centered.

Appendix C: Estimates of the number of clusters in simulation studies.

Method	Estimates of the number of clusters: \hat{G}									
	1	2	3	4	5	6	7	8	9	≥ 10
<i>Model 1: uniform data in 10 dimensions</i>										
Gap/uniform	50*	0	0	0	0	0	0	0	0	0
Gap/pc	50*	0	0	0	0	0	0	0	0	0
$\overline{\text{Gap}}$ /uniform	50*	0	0	0	0	0	0	0	0	0
$\overline{\text{Gap}}$ /pc	50*	0	0	0	0	0	0	0	0	0
DDGap/uniform	0*	7	6	5	5	4	7	2	10	4
DDGap/pc	0*	1	2	6	7	4	6	6	10	8
Hartigan	0*	0	0	1	34	14	1	0	0	0
KL	0*	0	48	2	0	0	0	0	0	0
Silhouette	0*	0	0	0	0	0	0	1	2	47
CH	0*	0	50	0	0	0	0	0	0	0
<i>Model 2: 6 clusters in 2 dimensions</i>										
Gap/uniform	0	31	0	0	0	19*	0	0	0	0
Gap/pc	0	18	0	0	0	32*	0	0	0	0
$\overline{\text{Gap}}$ /uniform	0	0	1	0	0	49*	0	0	0	0
$\overline{\text{Gap}}$ /pc	0	0	0	0	0	50*	0	0	0	0
DDGap/uniform	0	1	0	0	4	45*	0	0	0	0
DDGap/pc	0	0	0	0	6	44*	0	0	0	0
Hartigan	0	0	0	0	0	0*	0	0	0	50
KL	0	48	0	0	0	2*	0	0	0	0
Silhouette	0	0	0	0	9	41*	0	0	0	0
CH	0	0	0	0	0	50*	0	0	0	0
<i>Model 3: 4 exponential clusters in 2 dimensions</i>										
Gap/uniform	3	27	0	20*	0	0	0	0	0	0
Gap/pc	18	17	0	15*	0	0	0	0	0	0
$\overline{\text{Gap}}$ /uniform	5	1	0	44*	0	0	0	0	0	0
$\overline{\text{Gap}}$ /pc	17	1	0	32*	0	0	0	0	0	0
DDGap/uniform	0	0	0	50*	0	0	0	0	0	0
DDGap/pc	0	0	0	50*	0	0	0	0	0	0
Hartigan	0	0	0	0*	0	0	0	0	0	50
KL	0	0	0	39*	1	7	2	1	0	0
Silhouette	0	0	0	50*	0	0	0	0	0	0
CH	0	0	0	50*	0	0	0	0	0	0
<i>Model 4: 2 clusters in 2 dimensions with different within-clusters covariance</i>										
Gap/uniform	2	12*	9	17	0	0	0	0	0	0
Gap/pc	4	10*	19	17	0	0	0	0	0	0
$\overline{\text{Gap}}$ /uniform	3	47*	0	0	0	0	0	0	0	0
$\overline{\text{Gap}}$ /pc	3	47*	0	0	0	0	0	0	0	0
DDGap/uniform	2	48*	0	0	0	0	0	0	0	0
DDGap/pc	2	48*	0	0	0	0	0	0	0	0
Hartigan	0	0*	0	0	0	0	0	0	0	50
KL	0	19*	2	12	9	1	3	1	2	1
Silhouette	0	50*	0	0	0	0	0	0	0	0
CH	0	5*	0	0	4	0	2	3	7	29

Method	Estimates of the number of clusters: \hat{G}									
	1	2	3	4	5	6	7	8	9	≥ 10
<i>Model 5: 4 clusters in 2 dimensions with different nonidentity covariance matrices</i>										
Gap/uniform	0	0	0	50*	0	0	0	0	0	0
Gap/pc	0	0	0	50*	0	0	0	0	0	0
$\overline{\text{Gap}}$ /uniform	0	2	1	47*	0	0	0	0	0	0
$\overline{\text{Gap}}$ /pc	0	1	1	48*	0	0	0	0	0	0
$\overline{DD\text{Gap}}$ /uniform	0	0	4	46*	0	0	0	0	0	0
$\overline{DD\text{Gap}}$ /pc	0	0	2	48*	0	0	0	0	0	0
Hartigan	0	0	0	0*	0	0	0	1	3	46
KL	0	0	0	40*	3	1	1	2	2	1
Silhouette	0	0	14	36*	0	0	0	0	0	0
CH	0	0	0	43*	0	1	0	3	1	2
<i>Model 6: 2 elongated clusters in 2 dimensions</i>										
Gap/uniform	0	3*	0	33	0	14	0	0	0	0
Gap/pc	0	0*	0	48	0	2	0	0	0	0
$\overline{\text{Gap}}$ /uniform	0	0*	0	3	24	23	0	0	0	0
$\overline{\text{Gap}}$ /pc	0	0*	0	0	15	18	13	4	0	0
$\overline{DD\text{Gap}}$ /uniform	0	12*	0	38	0	0	0	0	0	0
$\overline{DD\text{Gap}}$ /pc	0	44*	0	6	0	0	0	0	0	0
Hartigan	0	0*	0	0	0	0	0	0	0	50
KL	0	41*	0	2	0	5	0	0	0	2
Silhouette	0	50*	0	0	0	0	0	0	0	0
CH	0	0*	0	0	0	14	4	19	3	10
<i>Model 7: 4 random clusters in 3 dimensions</i>										
Gap/uniform	1	0	0	49*	0	0	0	0	0	0
Gap/pc	7	0	0	43*	0	0	0	0	0	0
$\overline{\text{Gap}}$ /uniform	0	2	0	48*	0	0	0	0	0	0
$\overline{\text{Gap}}$ /pc	3	3	1	43*	0	0	0	0	0	0
$\overline{DD\text{Gap}}$ /uniform	0	7	11	32*	0	0	0	0	0	0
$\overline{DD\text{Gap}}$ /pc	0	8	12	30*	0	0	0	0	0	0
Hartigan	0	0	0	0*	1	4	1	1	8	35
KL	0	0	0	39*	0	3	2	3	2	1
Silhouette	0	3	11	36*	0	0	0	0	0	0
CH	0	0	0	50*	0	0	0	0	0	0
<i>Model 8: 4 random clusters in 10 dimensions</i>										
Gap/uniform	0	0	0	50*	0	0	0	0	0	0
Gap/pc	0	0	0	50*	0	0	0	0	0	0
$\overline{\text{Gap}}$ /uniform	0	0	1	49*	0	0	0	0	0	0
$\overline{\text{Gap}}$ /pc	1	1	4	44*	0	0	0	0	0	0
$\overline{DD\text{Gap}}$ /uniform	0	15	8	27*	0	0	0	0	0	0
$\overline{DD\text{Gap}}$ /pc	0	11	8	31*	0	0	0	0	0	0
Hartigan	0	0	0	50*	0	0	0	0	0	0
KL	0	0	0	50*	0	0	0	0	0	0
Silhouette	0	5	8	37*	0	0	0	0	0	0
CH	0	5	6	39*	0	0	0	0	0	0
<i>Model 9: 5 clusters in 10 dimensions</i>										
Gap/uniform	0	0	0	0	50*	0	0	0	0	0
Gap/pc	0	0	38	1	11*	0	0	0	0	0
$\overline{\text{Gap}}$ /uniform	0	0	0	0	50*	0	0	0	0	0
$\overline{\text{Gap}}$ /pc	0	0	21	29	0*	0	0	0	0	0
$\overline{DD\text{Gap}}$ /uniform	0	2	48	0	0*	0	0	0	0	0
$\overline{DD\text{Gap}}$ /pc	0	0	50	0	0*	0	0	0	0	0
Hartigan	0	0	0	0	50*	0	0	0	0	0
KL	0	0	0	0	50*	0	0	0	0	0
Silhouette	0	0	50	0	0*	0	0	0	0	0
CH	0	0	50	0	0*	0	0	0	0	0

Chapter 4

Clustering of gene expression data using Multi-Layer Clustering: Application to the study of oxidative stress induced by cumene hydroperoxide in yeast

In computational biology, cluster analysis has been successfully implemented in inferring functions of unknown genes and detecting classes or sub-classes of diseases. An important application is to cluster genes based on their temporal profiles. In this chapter, we will show that our methods for determining best-number-of-clusters proposed in Chapter 3 are successfully implemented in clustering the microarray gene expression data obtained from a time course study of the genome-wide oxidative stress response in *S. cerevisiae* cultures exposed to cumene hydroperoxide (CHP). We observe that our methods can be used to decide the number of different patterns of changes in transcripts after exposure to CHP in this specific microarray data set and can also separate various patterns from each other. The clustering results are validated utilizing known biological knowledge about the yeast genome.

4.1 Introduction

The development of microarray experiments provides a convenient experimental tool in monitoring the behaviors of gene in an organism in one experiment simultaneously. Microarray hybridization experiment plays an important role in the study of functional genomics. Since biological networks are highly complicated and the microarray data consist of large number of genes, cluster analysis has been particularly useful in exploring gene expression data. The goal of clustering microarray gene data focuses on two main aspects: classifying the tissue samples based on gene expression profiles and finding groups of genes on the basis of general gene expression patterns, that is, the relative changes in gene expression levels across various experimental conditions. An important application of the classification of tissues is to discover classes (or subclasses) of cancer (see [29], [7], [1], [42], [46], [64] for examples). In terms of gene classifications, it is expected that a good clustering method is useful in grouping functionally associated genes, such as genes with similar functions in a certain biochemical pathway or genes controlled under similar regulatory mechanism.

Many clustering techniques are available for the gene expression data analysis. These include classical approaches, such as k -means clustering, hierarchical clustering and model-based clustering, and several recently proposed clustering algorithms: the CAST algorithm ([7]), self-organizing maps ([83]), biclustering ([17]), *etc.* For most of the existing non-hierarchical clustering methods, the number of clusters is required to be pre-specified before conducting clustering. Because of the deterministic effect of the choice of the cluster number on the clustering results, a practitioner needs not only to choose an appropriate clustering method but also to decide the number of clusters. However, in the literature on clustering gene expression data, not much attention has been paid to determining the number of clusters compared to the abundant discussions about the selection of clustering techniques. In this paper, we offer a solution to the problem of estimating the number of clusters in gene expression data analysis described in Chapter 3.

As a fundamental problem in cluster analysis, choosing the best estimate of the number of clusters of a data set is not straightforward since there is no clear definition of *cluster*. In addition, the relationship between the thousands of genes in a microarray data can be highly complicated. Hence, an effective approach for estimating the number of clusters is necessary to achieve the correct answer about the best-number-of-clusters problem. Among existing methods, a multi-layer clustering proposed by Yan and Ye ([98]) has been demonstrated to be useful and robust in detecting complex cluster structure of data. In contrast to the

other methods, which simply give an estimate of the numerical value of the cluster number, multi-layer clustering can also reveal the hierarchical structure of clusters. This is an appealing property in gene expression analysis because it provides more information in terms of interpretation of the clustering results. In this paper, we examine the performance of multi-layer clustering in clustering gene expression data, more specifically, in finding groups of genes with similar expression patterns across various experimental conditions. Obviously, the estimate of the number of clusters is dependent on the type of the clustering technique employed. In our study, the estimation is based on the k -means clustering method, the simplest partitioning algorithm. Comparative studies by Yeung *et al.* ([99]) and Datta and Datta ([21]) have shown that the k -means clustering method is an appropriate choice for gene expression data.

This paper is organized as follows. Section 4.2 introduces the essential idea of multi-layer clustering. In Section 4.3, we apply the multi-layer clustering approach to a specific example, the time series data of gene expression obtained when *S. cerevisiae* cells are exposed to cumene hydroperoxide. The background information about this study is presented and we discuss the results focusing on the biological significance of clustered genes. Finally, we summarize the contributions of our work in Section 4.4.

4.2 Weighted gap statistic and multi-layer clustering approach

4.2.1 Weighted gap method and DD-weighted gap method

Suppose the observed data are x_1, x_2, \dots, x_n , where $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})'$, $i = 1, 2, \dots, n$. For an arbitrary number of clusters, g , apply any clustering method to partition the n objects in the data into g clusters. Denote the corresponding partition by P_g . For the m th cluster of P_g , $m = 1, 2, \dots, g$, denote by C_m the set of objects allocated to the m th cluster and n_m the number of objects in C_m . Denote by $d_{i,i'}$ the distance between objects i and i' . Then, the sum of pairwise distances between objects allocated to the m th cluster is given by

$$D_m = \sum_{i,i' \in C_m} d_{i,i'}. \quad (4.1)$$

Define a measure of the within-clusters homogeneity of P_g as

$$\overline{W}_g = \sum_{m=1}^g \bar{D}_m = \sum_{m=1}^g \frac{1}{2n_m(n_m - 1)} D_m. \quad (4.2)$$

The essential idea of the weighted gap method is to compare the value of \overline{W}_g for the observed data to the expected value of \overline{W}_g assuming that the data came from a suitable reference distribution. Denote the expected value of $\log(\overline{W}_g)$ by $E^*\{\log(\overline{W}_g)\}$, and define the weighted gap statistic as

$$\overline{Gap}(g) = E^*\{\log(\overline{W}_g)\} - \log(\overline{W}_g). \quad (4.3)$$

Given a data well separated into G clusters, it is expected that the greatest difference between the observed and expected values of \overline{W}_g occurs when $g = G$. Thus, the number of clusters is estimated such that $\overline{Gap}(g)$ is maximized.

In practice, $E^*\{\log(\overline{W}_g)\}$ is computed based on a set of reference data sets generated from the reference distribution which describes the underlying distribution of the observed data assuming that the data are not clustered, that is, the true number of clusters is 1. The uniform distribution is recommended for the calculation of the weighted gap statistic. In addition, there are two different ways for obtaining reference data sets. In the first way, each reference variable is generated uniformly over the range of the observed values of that variable. The other method utilizes information about the shape of the data distribution. Suppose X^* is the centered data obtained by subtracting the sample mean of each column of the original data from the elements in that column. Apply the singular value decomposition to X^* and assume that $X^* = UDV'$. Let $X^{**} = X^*V$ and draw uniform data Y over the ranges of the variables in X^{**} . The final reference data are $Z = YV'$. In order to avoid unnecessary clusters, the optimal estimate of the cluster number is determined via the '1-std-error' rule. More details about the weighted gap method are provided in [98].

It has been found that the weighted gap method may overestimate the number of clusters. Hence, Yan and Ye ([98]) proposed the DD-weighted gap method which has been shown to be more effective in choosing the appropriate estimate of the number of clusters given the data contain more than one cluster. This method is aimed at finding the "sufficiently large" point, instead of the maximum, in the function of \overline{W}_g . The value of $\overline{Gap}_n(g)$ is considered to be large enough if the clustering stops gaining much from adding one more cluster. Numerically, such a point is determined by comparing $\overline{Gap}_n(g)$ with its adjacent neighbors:

$\overline{Gap}_n(g-1)$ and $\overline{Gap}_n(g+1)$. Define

$$DD\overline{Gap}_n(g) = D\overline{Gap}_n(g) - D\overline{Gap}_n(g+1), \quad (4.4)$$

where $D\overline{Gap}_n(g) = \overline{Gap}_n(g) - \overline{Gap}_n(g-1)$, $g \geq 2$. Then, the best estimate of the number of clusters is equal to \hat{G} if $DD\overline{Gap}_n(g)$ is maximized at \hat{G} .

4.2.2 Multi-layer clustering

Both the weighted gap method and the DD-weighted gap method will simply provide an estimate of the number of clusters for a data set. In simple cases, knowing the cluster number would be sufficient to obtain the appropriate partition of the data. However, in the presence of more complicated data, we may need more information to fully reveal the relationships between objects in a data set. Multi-layer clustering has been proposed as a solution to this type of data. A special case is when the data contain a natural “hierarchical” cluster structure. Such a hierarchical structure exists when smaller sub-clusters are nested within the larger clusters that construct the dominant cluster structure of the data.

By definition, the weighted gap method can be used to test the null hypothesis (the cluster number is equal to 1) against the alternative of clustered data, and the DD-weighted gap method is only defined when g is greater than 1. The multi-layer clustering approach combines these two methods and sequentially detect clusters (sub-clusters) in a data set in multiple steps. In more detail, multi-layer clustering proceeds as follows. First, apply the weighted gap method to the whole data set to determine if the data need to be partitioned. If the estimated cluster number is equal to 1, it is concluded that the data is non-clustered and the clustering process stops. Otherwise, estimate the number of clusters in the data (K_1) using the DD-weighted gap method and partition the data into K_1 clusters. Perform the above clustering procedures separately using the partial data in each of the K_1 clusters obtained at this stage. Repeat the previous step until no further separation of any cluster is necessary.

In addition to reporting the total number of clusters and the corresponding classification of data, the multi-layer clustering analysis also produces the hierarchy formed by the clusters detected. To fully interpret the multi-layer clustering results, it is important to examine the hierarchical structure in practical problems which may provide insights into the association between clusters. It is also worth pointing out that the ultimate classification of a data is achieved step by step through the analysis, which may be quite different from the one

time classification produced by the clustering method with the same fixed total number of clusters.

4.3 Clustering application

4.3.1 Data and analysis

Oxidative stress has been related to processes such as ageing, apoptosis and cancer (see [38], [48], [86] and [100]). Information obtained in work done with a model eukaryotic cell like *Saccharomyces cerevisiae* has been helpful in clarifying questions involving higher eukaryotes. In this work, we clustered data from the genome-wide study of the kinetics of the yeast response to oxidative stress induced by cumene hydroperoxide (CHP) ([77]). The changes in gene expression that occur during the yeast response to oxidative stress induced by CHP were analyzed at the transcriptional level, from a dynamical point of view, spanning a time range of 3 to 70 min after the addition of the oxidant. The time course (7 time points) of events was monitored by collecting samples at defined time points (0, 3, 6, 12, 20, 40 and 70 min) and studying changes at the genome-wide level, using **Affymetrix**TM oligonucleotide arrays (Yeast Genome S98). Detailed information about this microarray experiment is found in [77]. In the following discussions, we refer to this microarray data set as the WTCHP data.

The main purpose of this study is to identify clusters of genes which have similar pattern of change in expression level after the exposure to cumene hydroperoxide (CHP). Thus, in our clustering, we measured the similarity between genes based on the correlation type of distance measure, instead of the distance measuring the difference between the absolute magnitudes of gene expressions. The data was gene-wise standardized such that for each gene, the mean of its expressions at the 7 time points is 0 and the variation is 1. It is easy to show that the Euclidean distance between a pair of genes calculated using the standardized data is equivalent to the corresponding distance based on the Pearson correlation coefficient. Hence, we may conclude that genes assigned to the same cluster in this analysis behave similarly as time elapses after being treated with CHP.

With the gene-wise standardized data, we estimated the number of clusters in the WTCHP data using the multi-layer clustering method. Similar to the weighted gap method, as discussed in Section 4.2.1, there are two ways of obtaining the reference data sets in multi-

Table 4.1: Multi-layer clustering results for the WTCHP data. D_1 and D_2 are the two clusters generated at the first layer of analysis. D_{ij} stands for the j_{th} cluster when further separating cluster D_i into 2 smaller clusters, $i, j = 1$ or 2 . D_{ijk} represents the k_{th} cluster obtained by dividing cluster D_{ij} .

$Data$	\overline{Gap}/pc	\overline{DDGap}/pc
D	6	2
D_1	2	2
D_2	5	2
D_{11}	3	8
D_{12}	1	14
D_{21}	1	7
D_{22}	4	4
$D_{11m}, m = 1, 2, \dots, 8$	1	≥ 2
$D_{22n}, n = 1, 2, 3, 4$	1	≥ 2

layer clustering. As Yan and Ye suggested ([98]), we generated the reference data based on the principal components of the observed data, that is, we only considered the estimated results given by the weighted gap/pc and the DD-weighted gap/pc method.

At the beginning, the k -means clustering method was used to cluster the whole data set (D) containing all 5262 genes with the number of clusters specified as $g, g = 1, 2, \dots, 40$. For each fixed value of cluster number, the optimal k -means separation was obtained based on 500 sets of randomly selected initial partitions. Since the estimated number of clusters given by the weighted gap/pc method is greater than 1 (see Table 4.1), we determined the cluster number using the DD-weighted gap/pc method and found two well separated clusters which contain 2879 and 2773 genes, respectively. In the next step, the 2879-gene cluster (D_1) and the 2773-gene cluster (D_2) were analyzed separately following the same procedures as in the first step. The results showed that D_1 and D_2 both contain two sub-clusters, denoted by D_{11} , D_{12} , D_{21} and D_{22} . So, in the third step, we conducted the estimation process for each of the four sub-clusters found in the previous step. Based on the weighted gap/pc estimation, two clusters D_{11} and D_{22} need to be further divided, while no separation of clusters D_{12} and D_{21} is necessary. When we continued with analyzing the smaller clusters found in D_{11} and D_{22} , the weighted gap/pc estimates were all equal to 1, thus the multi-layer clustering analysis stopped at this stage. In summary, the WTCHP data were decomposed into 14 clusters of

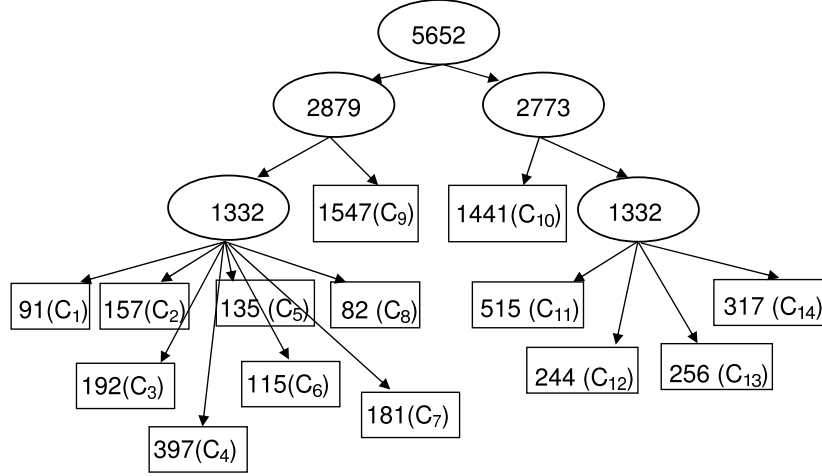


Figure 4.1: Hierarchical structure of WTCHP clusters ($C_1 \sim C_{14}$). The number of genes allocated to each (sub-)cluster at each step of clustering is indicated in the parenthesis. A rectangular represents a homogeneous cluster. An oval means that a cluster is further separated.

genes with similar expression profiles across the 7 time points considered in our analysis. The estimating results at each step of the multi-layer clustering process are presented in Table 4.1.

As mentioned in Section 4.2.2, it is possible to examine the hierarchical relationships of clusters in multi-layer clustering. In this application, the hierarchical structure of the 14 clusters is demonstrated in Figure 4.1. In the graph, we use C_1, C_2, \dots, C_{14} to represent the 14 clusters in the WTCHP data set. To explain the presence of such a hierarchical structure in the WTCHP data, we examined the 14 clusters by graphing the temporal profiles of genes in each cluster. Instead of showing the absolute values of expressions, for a specific gene, we actually plotted the fold change of its expression at any time point relative to its initial expression at the onset of the experiment (0 minute time point) (see Figure 4.2). In this way, it facilitates our understanding of the global expression pattern represented by a particular cluster. From Figure 4.2, we see that clusters C_1 to C_9 contain genes which were generally up-regulated after been exposed to CHP; clusters C_{10} to C_{14} are genes with down-regulated expressions either from a very early time or from a later time. The presence of these two most dominant expression patterns in this data set explains the separation of the data into 2 large clusters in the hierarchy. On the other hand, expressions of genes in clusters C_1 to C_8 levelled off or decreased after a certain time of exposure to CHP, but genes in cluster C_9 have constantly increasing expressions until the end of the experiment. This

explains why clusters C_1 to C_8 were closer to each other than to cluster C_9 . In a similar vein, cluster C_{10} could be separated from clusters C_{11} to C_{14} . Hence, by interpreting the hierarchy automatically generated during the multi-layer clustering process, it provides us a convenient way of understanding the relationships between clusters, especially when a data contain a large number of clusters.

4.3.2 Interpretation and discussion

In this section, we discuss the results obtained by multi-layer clustering in terms of biological significance. In order to demonstrate the advantage of applying the multi-layer method over the commonly used one-layer style of analyses, we also present a comparison between the one-layer and multi-layer results. For one-layer clustering, we take the results from k -means clustering given 6 clusters which is the number of clusters estimated using the weighted gap/pc method (see Table 4.1). To evaluate the performance of the clustering methods, pathway analysis was used to evaluate each cluster generated by each method. Pathway analysis was performed in Database for Annotation, Visualization and Integrated Discovery (DAVID) ([40]). Biological meanings, in terms of pathway analysis, of the clusters detected using these two different types of analysis are summarized in Tables 4.2 and 4.3. More specific discussions about distinctions in the clustering results are as follows.

The first response of cells to oxidative stress at the transcriptional level is the activation of genes that encode antioxidant defense proteins (W. Sha, A. Martins, P. Mendes and V. Shulaev, unpublished results). This group includes proteins that keep the redox state of the cell (thioredoxin system), the glutathione system, and enzymes involved in the detoxification of reactive oxygen species (ROS) ([48]). These genes are clustered together when k -means clustering is used - Figure 4.3 (A), blue cluster (cluster 1). However, when multi-layer clustering is used, this group is resolved in 3 different clusters (4.3 (B), clusters in dark blue - cluster 12, red - cluster 13, and green - cluster 5). It is then possible to distinguish 3 different patterns that correspond to 3 different time of response to stress. Hence, the blue cluster (cluster 12) contains genes that respond to stress very early (6-12 min), while the green cluster (cluster 5) contains genes that respond later (20-40 min). This distinction is very important when one is studying the kinetics of a certain phenomenon. An interesting result is the induction of *GSH1* and *GSH2*, the two genes encoding enzymes involved in the biosynthesis of glutathione (γ -glutamylcysteine synthetase and glutathione synthetase, respectively). The 2 enzymes catalyze sequential reactions in the *GSH* biosynthesis. Genes

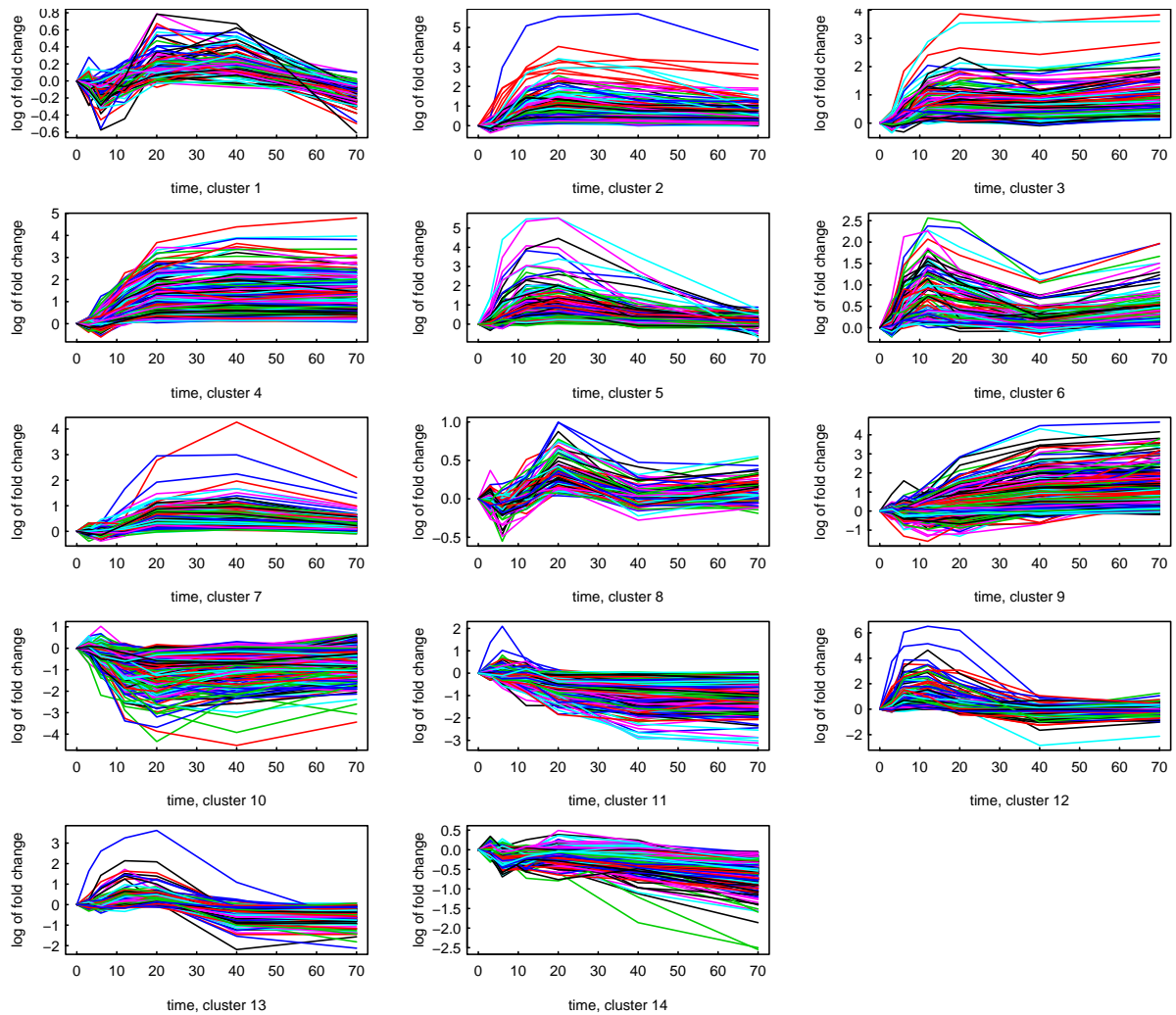


Figure 4.2: Plots of transcript fold change profiles for clusters generated by multi-layer clustering. For each gene, logarithm of its transcript level at 0, 3, 6, 12, 20, 40 and 70 minutes divided by its 0 minute transcript level is plotted against time.

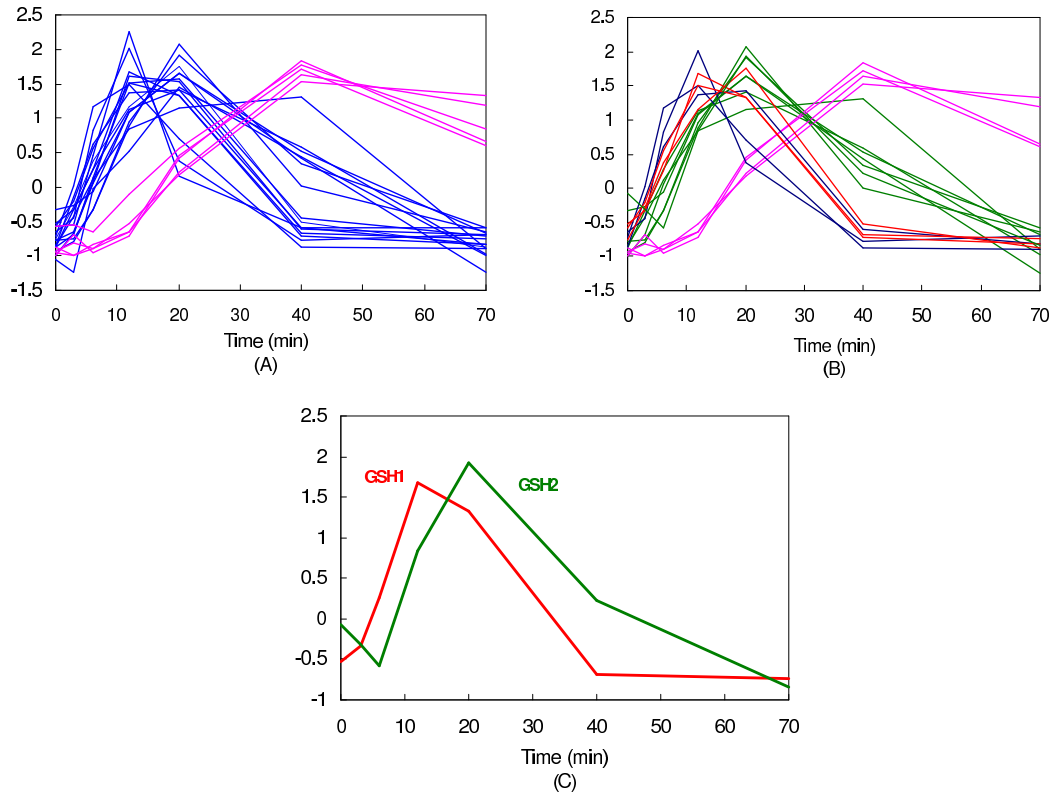


Figure 4.3: Profiles of clusters enriched in antioxidant defense genes. The gene-wise standardized data were plotted in the graph. (A) k -means clustering with 6 clusters; (B) multi-layer clustering (14 clusters). Different colors represent different clusters. (C) Profiles of the two genes coding for the 2 enzymes of the glutathione biosynthesis.

GSH1 and *GSH2* are clustered together using k -means clustering (blue cluster, Figure 4.3 (A)) but they are resolved into 2 different clusters using multi-layer clustering (Figure 4.3 (C),) - *GSH1* is induced at 12 min, belonging to the red cluster while *GSH2* is induced later, at 20 min, like most of the genes in the green cluster (Figure 4.3 (B),). Both methods distinguish a cluster of genes induced later (40 min), represented in pink (Figure 4.3 (A) and (B)), and corresponding to cluster 9 in multi-layer clustering, and to cluster 6, in k -means clustering.

During oxidative stress, ROS are formed and these react with and damage macromolecules, mainly DNA and proteins ([48]). Misfolded and malfunctioning proteins must be scavenged and degraded ([100]). In eukaryotic cells, most cytosolic and nuclear proteins are

degraded via the ubiquitin-proteasome pathway ([86]). The proteasome is a macromolecular machine built from ~ 31 different subunits, which degrades proteins ([86], [100]). In the response of yeast to oxidative stress induced by CHP, it was very interesting to observe that almost all the genes encoding proteasome subunit proteins were up-regulated at the same time, 20 to 40 min after the addition of the oxidant (W. Sha, A. Martins, P. Mendes and V. Shulaev, unpublished results). Profiles of this group of genes are shown in Figure 4.4. We observe that the genes are clustered together, either using one layer clustering (the enriched cluster is cluster 6) or using multi-layer clustering (cluster 7; in this case only 2 genes (- plotted in pink and green) are clustered separately but in clusters with a pattern very similar to that of cluster 7: clusters 4 and 9). Since the proteasome is constituted of all these subunits, it makes sense that all these genes are up-regulated at the same time. The regulation of this transcriptional process in the cell should be very tight and, since the expression of these genes is so similar, it was expected that they would fall in the same cluster, regardless of the method used.

A similar result is obtained with genes related to the cell cycle. In order to respond to oxidative stress, cells arrest growth and this phenotype is largely supported by the transcriptomics results: cell cycle-related genes are down-regulated since very early after the addition of the oxidant (W. Sha, A. Martins, P. Mendes and V. Shulaev, unpublished results). In this case, just like in the case of proteasome, genes are clustered together, either when using multi-layer clustering (cluster 10) or using k-means clustering (cluster 2) (Tables 4.2 and 4.3). Other pathways also cluster together with the cell cycle one, all related in cell growth: RNA polymerase, pyrimidine metabolism, purine metabolism and DNA polymerase (Tables 4.2 and 4.3).

In summary, to study the response to stress in terms of pathways, how genes change globally, *k*-means clustering with 6 clusters (one-layer) probably would be enough. However, to study the kinetics of response within a pathway or system (like in Figure 4.3, genes encoding antioxidant defense proteins) the multi-layer clustering provides better resolution. This may apply not only to the specific results analyzed in this paper - oxidative response in yeast - but also to other studies, as long as the chosen time range is appropriate to the problem in question.

Table 4.2: Pathway analysis of clusters found by multi-layer clustering. In this table, * stands for clusters containing pathways with p-value < 0.05 or not enriched in just a few specific pathways.

Cluster ID	Number of genes	Characteristics (based on average of cluster)	Main pathways (p-value < 0.05)
1	91	Decreases 3-6 min then increases until 20-40 min and goes back to basal levels at 70 min	*
2	157	Increases from 3 to 20 min, then gradually returns to basal levels from 20 to 70 min	*
3	192	Increases from 3 to 12 min, keeps high until 70 min	Starch and sucrose metabolism (0.00816) MAPK Signaling Pathway (0.01586)
4	379	Increases from 6 to 20 min and keeps high until 70 min	MAPK Signaling Pathway (0.048)
5	135	Increases at 6 min and is back to basal state at 70 min	Glutathione metabolism (0.01305)
6	115	Increases at 3 min, peaks at 12-20 min, goes back to basal levels after 40 min	Ubiquitin mediated proteolysis (0.0326)
7	181	Increases from 12-20 min and begins going back to initial at 70 min	Proteasome (7.59E-30)
8	82	Decreases at 6 min, increases until 20 min, goes back to basal levels at 40 min and slightly increases at 70 min	MAPK Signaling Pathway (0.00376)
9	1547	Increases at 20 min and keeps high until 70 min	Biotin metabolism Basal transcription factors
10	1441	Decreases at 6-12 min, begins going back to basal levels after that	Cell cycle (9.37E-11) RNA polymerase (2.10E-10) Pyrimidine metabolism (7.34E-10) Purine metabolism (0.0003) DNA polymerase (0.00957)
11	515	Decreases at 12-20 min and keeps down until 70 min	Ribosome (7.76E-12)
12	244	Increases 3- 6 min until 12 min, goes back to basal levels at 40 min (very early expressed genes)	Starch and sucrose metabolism (0.00485)
13	256	Increases at 12-20 min, goes back to initial state at 40 min	Aminoacyl-tRNA biosynthesis (0.00522)
14	317	Decreases at 6 min and keeps on decreasing	Ribosome (1.52E-46) ATP synthesis (0.0139)

Table 4.3: Pathway analysis of clusters found by one-layer clustering analysis with 6 clusters specified.

Cluster ID	Number of genes	Characteristics (based on average of cluster)	Main pathways (p-value < 0.05)
1	618	Increases after 3 min, reaches maximum at 12-20 min and decreases to basal levels after 20 min	Glutathione metabolism (0.00126)
2	1131	Decreases after 3 min until 20 min, after that the expression increases again	Cell cycle (3.29E-10) RNA polimerase (9.04E-10) Pyrimidine metabolism (1.51E-8) Purine metabolism (0.000124) DNA polymerase (0.003803)
3	341	Decreases after 0 min until 6 min, then begins increasing until 20- 40 min, decreases again after 40 min	ATP synthesis (0.000334)
4	1091	Decreases all the way to 120 min	Ribosome (1.10E-46) Aminoacyl tRNA biosynthesis (0.01632)
5	920	Decrease from 6 to 12 min, increases from 20 to 70 min	Biotin metabolism (0.006238) Basal transcription factors (0.018542)
6	1551	Increases after 6 min until 40 min, stable from 40 to 70 min	Proteasome (2.54E-10) Starch and sucrose metabolism (0.005355) Ubiquitin mediated proteolysis (0.006248)

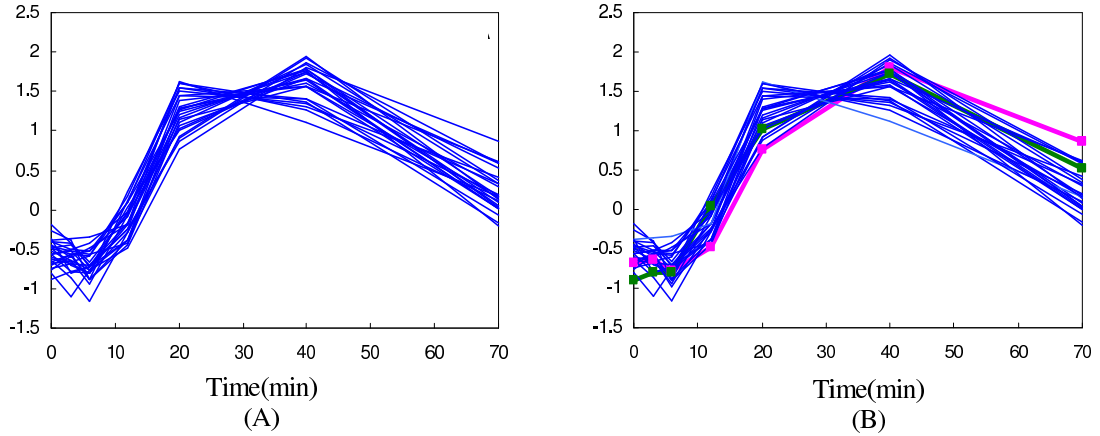


Figure 4.4: Profiles of clusters enriched in genes encoding proteasome subunit proteins. The standardized gene expressions were plotted in the graph. (A) k-means clustering with 6 clusters; (B) multi-layer clustering (14 clusters). Different colors in (B) represent different clusters.

4.4 Conclusion

With the advance of -omics technologies (genomics, proteomics, metabolomics etc.), huge amounts of data are being produced. Bioinformatics tools are then crucial to analyze all the data and extract as much valuable information as possible. Clustering is one widely used technique of exploratory analysis of expression data. In the studies of functional genomics, the interesting problem of recovering distinct patterns of changes in a particular feature across a series of experimental conditions has been discussed intensively in applications. An important example is to cluster genes based on their temporal profiles observed from a microarray experiment. However, a major difficulty in current applications is that there is no convincingly acceptable method of determining the number of clusters, although an appropriate decision about the cluster number is critical in utilizing most clustering techniques.

The multi-layer clustering method is motivated largely by the problem of estimating the number of cluster in analyzing data containing complicated cluster structure. In this chapter, we applied multi-layer clustering to a real microarray data set which in nature has high complexity. Interpretation and validation of the resulting clusters shows that multi-layer clustering performs satisfactorily in choosing the number of different expression patterns and separating various patterns from each other. In particular, merits of the multi-layer clustering solution, compared with the one-layer analysis results, are listed in our discussions.

We expect multi-layer clustering to be successful in broader applications besides microarray experiments.

Chapter 5

A New Partitioning Method

In this chapter, we propose a clustering method that defines a partitioning criterion based on the n objects in a data set that are split into g mutually exclusive clusters. Proposal of this method is motivated by the “equal-size” problem with the widely used k -means clustering which will be addressed in Section 5.1 (see Figure 5.1). Computationally, a hill-climbing algorithm is designed to search for the partition of a data which optimizes our proposed clustering criterion. Similar to the k -means method, this new clustering method proceeds with a fixed number of clusters specified in advance. We leave the problem of estimating the optimal number of clusters open in this chapter. We will mainly focus on discussing the performance of our proposed method given that the “true” number of clusters is used in clustering. For the purpose of practical application, similar studies as described in Chapter 3 should be conducted in order to determine the appropriate method of finding the best estimate of the number of clusters when applying the clustering method proposed in this chapter.

5.1 Motivation

A large family of clustering methods focus on dividing or partitioning objects in a data set into a pre-specified number of clusters such that a particular clustering criterion defined to measure the adequacy of a clustering is optimized. K -means clustering (see Ball and Hall [4], Edwards and Cavalli-Sforza [28], Forgy [33], Jancey [52], MacQueen [60] and Singleton and Kautz [79]) is a widely used partitioning technique, where the optimization involves mini-

minimizing the sum of squared distances between objects and the means of clusters (centroids). The idea of k -means clustering is as follows.

Suppose that the observed data contain n objects in p dimensions. Each object can be expressed as $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})'$, $i = 1, \dots, n$. For a specific partition of the n objects into g clusters, let C_m indicate the set of objects allocated to the m th group, n_m the number of objects in C_m , $m = 1, \dots, g$, and x_{ml} the l th object in C_m , $l = 1, \dots, n_m$. Define the dispersion matrix for each cluster, W_m , as

$$W_m = \sum_{l=1}^{n_m} (x_{ml} - \bar{x}_m)(x_{ml} - \bar{x}_m)', \quad (5.1)$$

where

$$\bar{x}_m = \frac{1}{n_m} \sum_{l=1}^{n_m} x_{ml}. \quad (5.2)$$

The pooled within-cluster dispersion matrix W is given by

$$W = \sum_{m=1}^g \sum_{l=1}^{n_m} (x_{ml} - \bar{x}_m)(x_{ml} - \bar{x}_m)'. \quad (5.3)$$

The optimization criterion of the k -means method is to minimize the trace of matrix W , $tr(W)$. Minimization of $tr(W)$ is equivalent to minimization of the sum of squared Euclidean distance between each object and its cluster mean, $d^2(x_{ml}, \bar{x}_m)$, $m = 1, \dots, g$, $l = 1, \dots, n_m$, since

$$tr(W) = \sum_{m=1}^g \sum_{l=1}^{n_m} d^2(x_{ml}, \bar{x}_m) = \sum_{m=1}^g \sum_{l=1}^{n_m} |x_{ml} - \bar{x}_m|^2.$$

In terms of k -means clustering, a cluster is defined by the mean of the cluster which is estimated by the sample mean of objects assigned to the cluster. More generally, consider the population version of the k -means approach. Suppose that x_1, x_2, \dots, x_n is a random sample of n points, each of which came independently from a population with probability density $p(x)$ in R_p . Given a g -tuple $a = (a_1, a_2, \dots, a_g)$, $a_i \in R_p$, $i = 1, 2, \dots, g$, a partition of R_p , denoted by $S = (S_1, S_2, \dots, S_g)$, is given by

$$S_i = \left\{ x \mid |x - a_i| = \min_{j=1,2,\dots,g} |x - a_j| \right\}.$$

Set

$$V(a_1, a_2, \dots, a_g) = \sum_{i=1}^g \int_{S_i} |x - a_i|^2 p(x) dx.$$

Then, the optimal partition of R_p defined by the k -means method is $S = (S_1^*, S_2^*, \dots, S_g^*)$, where

$$S_i^* = \left\{ x \mid |x - a_i^*| = \min_{j=1,2,\dots,g} |x - a_j^*| \right\},$$

and $a^* = (a_1^*, a_2^*, \dots, a_g^*) = \operatorname{argmin}_a V(a_1, a_2, \dots, a_g)$.

Suppose $\bar{x}_m, m = 1, 2, \dots, g$, are the corresponding cluster means when separating x_1, x_2, \dots, x_n into g clusters, where \bar{x}_m is defined in (5.2). Then the k -means algorithm is intended to minimize $V(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_g)$ over all possible partitions of this sample. Denote by $V(\bar{x}_1^*, \bar{x}_2^*, \dots, \bar{x}_g^*)$ the optimal solution found by the k -means method. The principal theoretical result derived by MacQueen ([60]) is that $V(\bar{x}_1^*, \bar{x}_2^*, \dots, \bar{x}_g^*)$ converges to $V(u_1, u_2, \dots, u_g)$, where $u_m = (\int_{S_m^*} xp(x)dx) / \int_{S_m^*} p(x)dx, m = 1, 2, \dots, g$, the population cluster means. It follows that the variance within the sample clusters determined by the k -means algorithm converges to the variance of the optimal clustering of the population.

Empirical studies have shown some drawbacks of k -means clustering. For example, it is not invariant under nonsingular linear transformation of the data, so that changes in the measurement units might lead to different partitions of the same data. It has been found that this method tends to find equal sized clusters and it is not adequate in distinguishing clusters with different shapes. One explanation for such a tendency of producing equal clusters is that a partition based on the minimization of $\operatorname{tr}(W)$ criterion is equivalent to the maximum likelihood partition when the data is assumed to come from multivariate normal mixture distributions with equal covariance matrices ([82]). The “equal-size” problem with k -means clustering is demonstrated in Figure 5.1. As indicated by markers of two different shapes (triangles and squares) in the plot, the data contain 2 well-separated clusters, each of which corresponds to a distinct normal population in 2 dimensions. It is known the true cluster sizes are 200 and 50 for the two clusters, respectively. However, when the k -means method was applied to the data, a lot of objects with true memberships of the larger cluster were misclassified to the cluster with smaller size.

Marriott ([62]) investigated the properties of several optimization criteria for clustering by examining the changes in the values of these criteria when a single object is added to the current data set. Relevant to his discussions, there is an explanation for the presence of such an “equal-size” problem in k -means clustering. Consider W_m defined in (3.1), adding a point x (a vector) to C_m changes W_m to $W_m^* = W_m + d_m d_m'$, where

$$d_m = (x - \bar{x}_m) \left(\frac{n_m}{n_m + 1} \right)^{\frac{1}{2}}. \quad (5.4)$$

Correspondingly, $\operatorname{tr}(W)$ changes to $\operatorname{tr}(W^*) = \operatorname{tr}(W) + d_m' d_m = \operatorname{tr}(W) + (1 - \frac{1}{n_m + 1}) |x - \bar{x}_m|^2$,

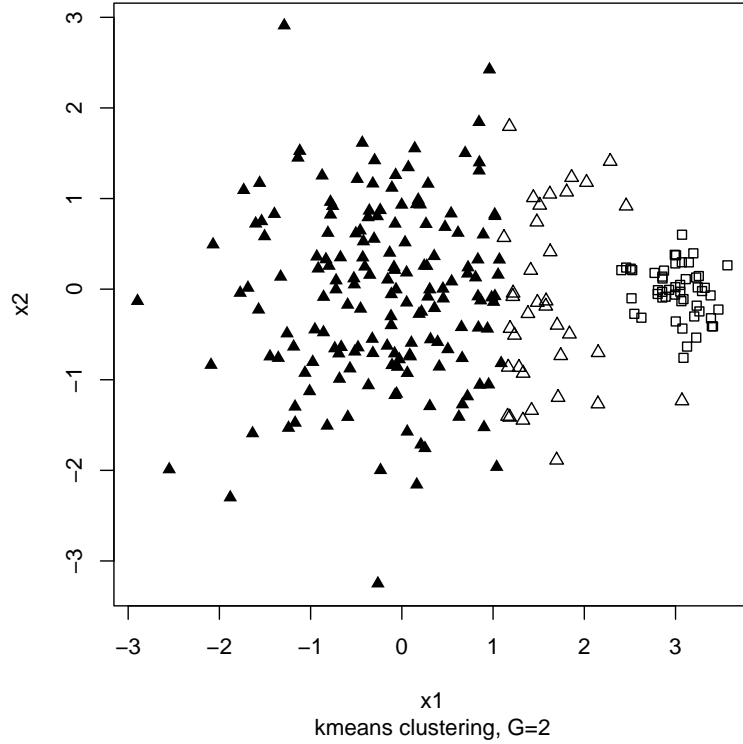


Figure 5.1: Illustration of the “equal-size” problem with k -means clustering. In this plot, data were simulated from 2 well separated normal distributions in 2 dimensions, with 200 and 50 points from each distribution, respectively. Points generated from the 2 different populations were plotted in triangles and squares, respectively. The 2 clusters produced by k -means clustering are indicated separately by solid and hollow markers in the plot.

where $|x - \bar{x}_m|$ denotes the Euclidean distance between x and the cluster mean of C_m . Assume that, at the current stage, the distances between x and \bar{x}_i , \bar{x}_j , $i \neq j$ (two distinct cluster centers) are equal, that is $|x - \bar{x}_i| = |x - \bar{x}_j|$. If $n_i \gg n_j$, then $(1 - \frac{1}{n_i+1})|x - \bar{x}_i|^2 \gg (1 - \frac{1}{n_j+1})|x - \bar{x}_j|^2$. Thus, x will be assigned to C_j instead of C_i under the minimization of $tr(W)$ criterion. Hence, clusterings based on minimization of $tr(W)$ have a tendency to generate clusters of equal sizes.

In spite of the disadvantages mentioned here, k -means clustering has become one of the most popular clustering technique mainly because that the k -means algorithms for finding the optimal partition are computationally economical. In the following section, we propose a partitioning method in an attempt to deal with the potential problem with the k -means method in the presence of large discrepancy in cluster sizes in a data set, while taking

advantage of the computational simplicity of computing the trace of a matrix.

5.2 Description of the new method

5.2.1 Partitioning criterion

Suppose that we have a data set containing n objects in p dimensions. In the matrix format, the data can be written as

$$X = \begin{pmatrix} x_1' \\ x_2' \\ \vdots \\ x_n' \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \ddots & & \vdots \\ x_{n1} & \dots & & x_{np} \end{pmatrix},$$

where x_{ij} represents the measurement on the j th variable of the i th object in the data, $i = 1, 2, \dots, n$, $j = 1, 2, \dots, p$. For any partition of X into g clusters, let C_m denote the set of observations allocated to the m th cluster and n_m the total number of observations in C_m . For $m = 1, \dots, g$, define

$$\hat{Q}_m = \frac{1}{n_m - 1} \sum_{\substack{1 \leq l \leq n \\ x_l \in C_m}} (x_l - \bar{x}_m)(x_l - \bar{x}_m)', \quad \bar{x}_m = \frac{1}{n_m} \sum_{\substack{1 \leq l \leq n \\ x_l \in C_m}} x_l, \quad (5.5)$$

where \hat{Q}_m is the estimated covariance matrix of all the observations in C_m .

Since $W_m = (n_m - 1)\hat{Q}_m$, where \hat{Q}_m is the estimate of the covariance matrix of cluster m , Marriott ([62]) argued that, for criteria defined on the basis of \hat{Q}_m , the tendency to generate equal clusters may be increased or reduced. Hence, we seek to define an optimization criterion based on \hat{Q}_m , with the expectation that the equal cluster tendency can be reduced. It is based on the fact that each cluster is characterized by its own covariance matrix and a good estimation of it can be obtained provided that enough objects are collected for that cluster. Although criteria based on the determinant of W_m have been shown to perform better than those based on the trace of W_m in capturing clusters of unequal shape, they require the non-singularity of matrix W_m . This might be a problem for data with many dimensions. Thus we propose a new partitioning criterion utilizing the trace of \hat{Q}_m mainly for the convenience of computation.

Given a fixed number of clusters (g), we propose a new partitioning criterion which is to minimize

$$\sum_{m=1}^g tr(\hat{Q}_m) \equiv Q(g), \quad (5.6)$$

where $tr(\hat{Q}_m)$ is the trace of \hat{Q}_m . With the same definition of W_m as (5.1), minimizing (5.6) is equivalent to minimizing

$$\sum_{m=1}^g \frac{1}{n_m - 1} tr(W_m). \quad (5.7)$$

5.2.2 Algorithm for computation

Computationally, an exhaustive search for the best partition of n objects into g clusters which minimizes the proposed partitioning criterion will be extremely time consuming except for very small n . Several approaches have been developed to overcome such difficulty by comparing only a part of all the possible partitions. The hill-climbing algorithm suggested by Friedman and Rubin ([37]) determines the optimal partition through the manipulation of a single point at each step. This approach has been shown to be simple and efficient given the presence of good data structures. Hartigan ([45]) described the procedures of a hill-climbing algorithm for k -means clustering. We design an algorithm to obtain the optimal solution of our partitioning method following the idea of Hartigan's method.

Suppose the data is separated into g clusters, C_1, \dots, C_g . For $m = 1, \dots, g$, definitions of n_m , \bar{x}_m , $tr(W_m)$ and $tr(\hat{Q}_m)$ are given by (5.1) and (5.5). Assume that the i th object, x_i , is currently assigned to C_k . When x_i is moved from C_k to C_l ($k \neq l$), the value of the criterion in (5.6) changes from $Q(g)$ to $Q^*(g)$. Denote the resulting estimated covariance matrices of C_k and C_l by $\hat{Q}_{k(i-)}$ and $\hat{Q}_{l(i+)}$, respectively. It can be easily shown that the difference between $Q(g)$ and $Q^*(g)$ is given by

$$\begin{aligned} \Delta(k \rightarrow l, i) &= Q(g) - Q^*(g) \\ &= [tr(\hat{Q}_k) + tr(\hat{Q}_l)] - [tr(\hat{Q}_{k(i-)}) + tr(\hat{Q}_{l(i+)})] \\ &= \frac{n_k d(i, C_k) - tr(W_k)}{(n_k - 1)(n_k - 2)} + \frac{tr(W_l)}{(n_l - 1)n_l} - \frac{d(i, C_l)}{n_l + 1}, \\ d(i, C_k) &= (x_i - \bar{x}_k)'(x_i - \bar{x}_k), \\ d(i, C_l) &= (x_i - \bar{x}_l)'(x_i - \bar{x}_l). \end{aligned}$$

Let

$$\Delta(i) = \max_{\substack{1 \leq l \leq g \\ l \neq k}} \Delta(k \rightarrow l, i),$$

and

$$c(i) = \arg \max_{\substack{1 \leq l \leq g \\ l \neq k}} \Delta(k \rightarrow l, i).$$

Note that, given the other objects' memberships are fixed, moving x_i from C_k to another cluster will reduce $Q(g)$ if $\Delta(i)$ is positive. In terms of the optimization criterion in (5.6), the best improvement of the current partition will be achieved by moving x_i from C_k to cluster $C_{c(i)}$ (defined above). If iteratively updating the partition by changing a single object's membership which brings the largest reduction in $Q(g)$ at each step, we will end up with a local optimum of the criterion.

Specifically, a hill-climbing algorithm for minimizing $\sum_{m=1}^g \text{tr}(\hat{Q}_m)$ contains the following steps:

1. Randomly select g points in the R^p space as the initial centers of g clusters. Generate the starting partition by assigning each object to the cluster whose center it is nearest to.
2. Recompute the cluster centers as the sample mean of the objects assigned to each cluster in step 1.
3. For $i = 1, \dots, n$, calculate the value of $\Delta(i)$. If $\Delta(i)$ is positive, move the i th object from its current cluster to cluster $c(i)$. Update the centers of the i th object's previous and current cluster after its membership was changed.
4. Repeat step 3. If there is no movement of a single object after a complete pass through all objects, then stop.

A general concern with hill-climbing algorithms is that it is not ensured that the local optimum will coincide with the global optimum. The final solution depends on the initial partition used to start the iteration and the order of objects in data. It is possible to overcome this problem by repeating the above procedures with different starting partitions and choosing the best partition among all resultant solutions.

5.3 Comparisons with the k -means method

In this section, we compare the behavior of the method proposed in the previous section to the k -means method in clustering the same data under different situations. First, performances of

these two methods are examined using simulated data where the true classification of a data is known. Secondly, we applied the proposed method to cluster the Iris data published by Fisher ([31]), a well studied benchmark data set in cluster analysis. For both the simulation studies and the Iris data analysis, the clustering results are demonstrated using scatter plots, where objects with the same true classification indices (known as a *prior*) in a data set are plotted with the same color and points with the same type of character in a plot are assigned to the same cluster when a particular clustering method is used. As indicated by the subtitle of a plot, “wkmeans” and “kmeans” refer to the classification results obtained by our proposed method and k -means clustering, respectively. A capital G in the subtitle of a plot gives the true number of clusters in simulated data, which is also the number of clusters fixed in clustering. When clustering the Iris data, the number of clusters specified in clustering is indicated by g .

5.3.1 Clustering validation and the adjusted Rand index

The purpose of clustering validation is to determine the validity associated with a resulting clustering result. A higher cluster validity reflects a higher agreement between the clustering result and the true cluster structure of data. There are two major types of cluster validity testing approaches in the literature: testing procedures based on either external criteria or internal criteria. External approaches evaluate the agreement between the results of a clustering process and an external classification independent of the clustering procedure. Internal analysis only relies on information inherited from the data, usually some goodness-of-fit measures reflecting degree of agreement between the input data and the resulting classification of data.

Validity of the clustering solution obtained from an arbitrary clustering method can be conveniently measured using external criteria since the true cluster structure of a data set is known in simulation studies. Hence, external validating approaches can be used in comparing the performances of different clustering techniques. Among existing external criteria, the adjusted Rand index ([50]) was recommended by Milligan ([67]).

Given a set of n objects $O = \{x_1, x_2, \dots, x_n\}$ and the true number of clusters G , denote by $T = \{t_1, t_2, \dots, t_G\}$ the partition of O corresponding to the true memberships of objects in the simulated data and $C = \{c_1, c_2, \dots, c_G\}$ a clustering result. For our proposed method and k -means clustering, $\bigcup_{m=1}^G t_m = S = \bigcup_{m=1}^G c_m$ and $t_m \cap t_{m'} = \emptyset = c_m \cap c_{m'}$ for $1 \leq m \neq m' \leq G$. Consider both T and C , the assignment of the n objects can be

summarized into Table 5.1. Then, the adjusted Rand index ([50]) can be expressed as:

$$\frac{\sum_{i,j} \binom{n_{ij}}{2} - [\sum_i \binom{n_{i\cdot}}{2} \sum_j \binom{n_{\cdot j}}{2}] / \binom{n}{2}}{\frac{1}{2} [\sum_i \binom{n_{i\cdot}}{2} + \sum_j \binom{n_{\cdot j}}{2}] - [\sum_i \binom{n_{i\cdot}}{2} \sum_j \binom{n_{\cdot j}}{2}] / \binom{n}{2}}. \quad (5.8)$$

The expected value of the adjusted Rand index is 0 assuming the generalized hypergeometric model. A large value of the adjusted Rand index indicates a good agreement between the true cluster structure and the resulting partition generated by a particular clustering method. More details about the adjusted Rand index can be found in [50]. In Section 5.3.2, we conduct simulation studies and utilize the adjusted Rand index as the criterion to compare the performances of our proposed method and the k -means method.

Table 5.1: The contingency table for comparing two partitions of a data into G clusters. n_{ij} , $i = 1, 2, \dots, G$, $j = 1, 2, \dots, G$, is the number of objects simultaneously assigned to cluster i in partition T and cluster j in partition C . $n_{i\cdot} = \sum_{j=1}^G n_{ij}$. $n_{\cdot j} = \sum_{i=1}^G n_{ij}$.

<i>Cluster</i>	c_1	c_2	\dots	c_G	Sums
t_1	n_{11}	n_{12}	\dots	n_{1G}	$n_{1\cdot}$
t_2	n_{21}	n_{22}	\dots	n_{2G}	$n_{2\cdot}$
\vdots	\vdots	\vdots		\vdots	\vdots
t_G	n_{G1}	n_{G2}	\dots	n_{GG}	$n_{G\cdot}$
Sums	$n_{\cdot 1}$	$n_{\cdot 2}$	\dots	$n_{\cdot G}$	n

5.3.2 Simulation studies

Table 5.2 lists the contexts under which data were simulated for the purpose of comparing our proposed method with k -means clustering. 100 data sets were simulated under each scenario considered in our studies. The averaged adjusted Rand index is calculated where a larger value reflects a better fit of the clustering results to the true known cluster structure in the simulated data.

So far, our comparative studies were focused on data with low dimensionality. Thus, the behavior of a clustering method can be easily examined using scatter plots. Provided that clusters in the data had either roughly equal cluster sizes or roughly equal cluster variations, our method performed equally well as or outperformed the k -means method (see Figure 5.2,

Table 5.2: Simulation scenarios for comparing our proposed method with the k -means method. The number of variables in a simulated data set is p . Data were simulated from the Normal distribution with parameters indicated in the table, except that in Scenario 5, the uniform clusters were examined. For a particular scenario, 100 data sets were generated and clustered using both the k -means method (kmeans) and our proposed method (wkmeans), and the averaged adjusted Rand index is the mean value of the corresponding 100 adjusted rand indices.

Scenario Plot Index	p	Cluster Size	Distribution Parameters	Averaged Adjusted Rand Index	
				wkmeans	kmeans
1) Figure 5.2	1	$n_1 = n_2 = 50$	$\mu_1 = 0, \mu_2 = 4, \sigma_1 = 1, \sigma_2 = 0.3$	0.9597	0.9549
2) Figure 5.3	1	$n_1 = 100, n_2 = 30$	$\mu_1 = 0, \mu_2 = 4, \sigma_1 = \sigma_2 = 1$	0.9026	0.9011
3) Figure 5.4	1	$n_1 = 100, n_2 = 30$	$\mu_1 = 0, \mu_2 = 4, \sigma_1 = 1, \sigma_2 = 0.3$	0.9778	0.8975
4) Figure 5.5	1	$n_1 = 30, n_2 = 100$	$\mu_1 = 0, \mu_2 = 4, \sigma_1 = 1, \sigma_2 = 0.3$	0.9161	0.9748
5) Figure 5.6	1	(a1, b1) $n_1 = 100, n_2 = 30$ (a2, b2) $n_1 = 30, n_2 = 100$	$f_1 \sim U(0, 4), f_2 \sim U(5, 6)$ $f_1 \sim U(0, 4), f_2 \sim U(5, 6)$	1 0.7096	0.4197 0.9158
6) Figure 5.7	2	(a1, b1) $n_1 = n_2 = 100$ (a2, b2) $n_1 = n_2 = 100$	$\mu_1 = (0, 0)', \mu_2 = (5, 0)', \Sigma_1 = I, \Sigma_2 = 0.3^2 I$ $\mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.3^2 I$	0.9908 0.8689	0.9851 0.8387
7) Figure 5.8	2	$n_1 = 200, n_2 = 50$	$\mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$	0.9472	0.6095
8) Figure 5.9	2	(a1, b1) $n_1 = 20, n_2 = 100$ (a2, b2) $n_1 = 20, n_2 = 100$	$\mu_1 = (0, 0)', \mu_2 = (4, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$ $\mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$	0.8024 0.4783	0.9778 0.9449
9) Figure 5.10	2	(a1, b1) $n_1 = 20, n_2 = 100$ (a2, b2) $n_1 = 40, n_2 = 100$ (a3, b3) $n_1 = 60, n_2 = 100$	$\mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$ $\mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$ $\mu_1 = (0, 0)', \mu_2 = (3, 0)', \Sigma_1 = I, \Sigma_2 = 0.1^2 I$	0.4783 0.8264 0.9926	0.9449 0.9401 0.9525

Figure 5.3 and Figure 5.7). In the cases where huge unbalance existed in both the sizes and the variances among clusters, the difference in behaviors of the two methods is most interesting. If clusters with larger variances were also associated with larger sizes (see Figure 5.6 (a1), (b1), Figure 5.4 and Figure 5.8), the simulated results showed that our proposed method worked much better than the k -means method, where objects were divided by a hyperplane, the “normal bisectors of the joins of the cluster means” ([62]). On the other hand, our current research found that an adverse situation for our method is when much fewer objects were observed for clusters with larger variations compared with those with much smaller variations (see Figure 5.5, Figure 5.6 (a2), (b2) and Figure 5.9). However, as demonstrated in Figure 5.10, as the discrepancy in cluster sizes decreased, the degree of misclassification might be greatly reduced.

5.3.3 Clustering of the Iris data

The Iris data is a well studied benchmark data set in cluster analysis. This data set contains 150 objects in total, each of which was measured on four variables: length of sepal, width of sepal, length of petal and width of petal. The objects are categorized by three species (Iris setosa, Iris versicolor and Iris virginica). Classification of the Iris data based on the species information is frequently compared with classifications obtained by other clustering (classification) techniques. The Iris data was first used by Fisher ([31]) in studying his procedure of Linear Discriminant analysis. Applications of this data set can also be found in [37], [76], [57] and [81]. It has been found that Iris setosa could be well separated from the other two species by most methods, while Iris versicolor and Iris virginica are overlapped somewhat.

We applied our partitioning criterion of minimizing (5.6) to the Iris data. When the number of clusters was fixed at $g = 2$, we see that Iris sotosa was perfectly separated from Iris versicolor and Iris verginica according to our method (Figure 5.11 (a1)). The k -means method also distinguished Iris sotosa from the other two species. However, three objects which are obviously distinct from the Iris sotosa objects were misclassified as Iris sotosa (Figure 5.11 (b1)). As the cluster number was fixed at $g = 3$, both methods had problem in isolating Iris versicolor and Iris verginica. The reason for this is probably because of the inadequacy of both methods in revealing non-spherical clusters. In general, our method has shown better properties than the k -means method.

5.4 Summary and discussion

In this chapter, we developed a new partitioning method for cluster analysis. Originally motivated by the “equal size” problem associated with application of the popular k -means clustering, we proposed an optimization criterion which has demonstrated good properties in overcoming this problem. The main advantage of our method is that its implementation is very convenient: it is defined without assuming any underlying distribution of the data; it only requires specification of the number of clusters to proceed with the computation; the designed algorithm for finding the optimal partition is time-efficient. Empirical studies have shown advantages of our method over the k -means method under most of the simulation contexts considered in our study.

Base on our current studies, we expect better performances of our method than the k -means method in general, except that it will be an adverse situation for our method if the data consists of clusters with both different within-cluster variations and unequal sizes while certain clusters with large variances are represented by much fewer observations compared with other clusters with small variances. In our future research, we intend to adjust the optimization criterion in our method such that it may provide appropriate classification results even in the unfavorable cases that we just mentioned. A possible direction for doing this is to change the criterion in (5.6) to

$$\sum_{m=1}^g \text{tr}(\hat{Q}_m) w_m \equiv \hat{Q}(g), \quad (5.9)$$

where $\sum_{m=1}^g w_m = 1$ and w_m is a function of the cluster sizes n_1, n_2, \dots, n_g . The “weight” put on $\text{tr}(\hat{Q}_m)$ might eliminate the deficiency with the current criterion.

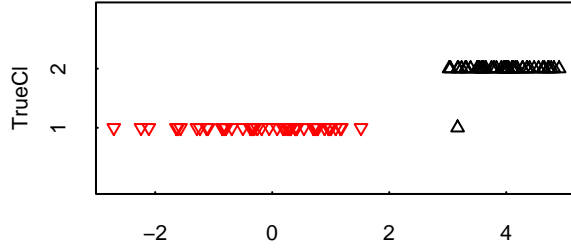
One aspect of future research is to examine asymptotic behaviors of our proposed clustering criterion. It will be insightful if we can derive large sample properties of the criterion. Instead of empirical results, it will formalize the comparisons of our method and k -means clustering.

Furthermore, it would be interesting to compare the outcome of classification provided by our method and the k -means method when unnecessary centers are added into the data. Figure 5.12 gives an example for comparing the behavior of our method with that of the k -means method when the specified number of clusters is larger than the “true” number of cluster in a data set. As indicated by the two different types of markers in each plot, the data contains two distinct clusters in two dimensions. Suppose the cluster number is fixed at

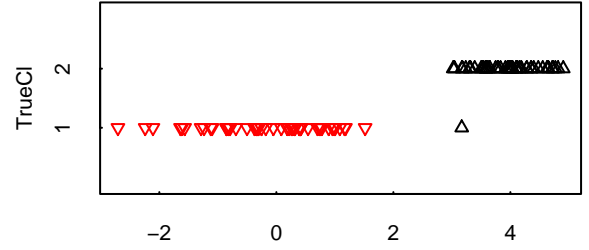
$g = 4$. Then the resultant four clusters corresponding to our method and the k -means method are distinguished by the different colors of markers in Figure 5.12 (a) and Figure 5.12 (b), respectively. We see that, with our method, it will form clusters containing only a few objects within a very small region around the centers, but the k -means method tends to separate the data into “equal” clusters of smaller sizes. Hence, we have two expectations in the cases where the cluster number is overestimated: first, compared with the true classification, our method should have higher level of classification accuracy than the k -means method in term of the total number of pairs of objects correctly assigned into the same clusters; secondly, when using our proposed method, the presence of clusters with both extremely small variances and small cluster sizes when compared with other clusters may provide hints on the need for reducing the number of clusters in analysis. Based on the second expectation, it might be possible to develop a typical method of estimating the cluster number for our proposed partitioning method.

Another direction for ongoing research concerns finding the best estimate of cluster number when applying our proposed clustering method. Although many global methods (see Section 2.3.1) are generally applicable to determine the number of clusters independent of clustering method, it will be suggestive for the practical purpose if we can examine the behaviors of various estimating methods when used in combination with our proposed clustering method.

Finally, in terms of the current simulation studies, we only considered data with 1 or 2 variables so far. Since the real data for cluster analysis are frequently recorded on many variables, we need to examine the performances of our method in clustering data of high dimensionality.

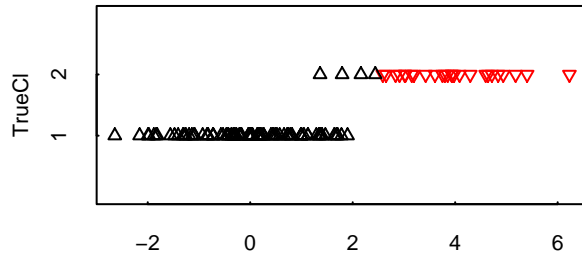


(a) wkmeans, $p=1$, $G=2$

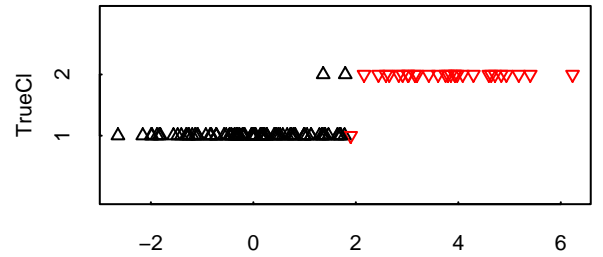


(b) kmeans, $p=1$, $G=2$

Figure 5.2: Comparing our proposed clustering method with k -means method. Univariate Normal clusters with equal cluster sizes: $n_1 = n_2 = 50$, $\mu_1 = 0$, $\mu_2 = 4$, $\sigma_1 = 1$, $\sigma_2 = 0.3$. Summary: the two methods work equally well.



(a2) wkmeans, $p=1$, $G=2$



(b2) kmeans, $p=1$, $G=2$

Figure 5.3: Comparing our proposed clustering method with the k -means method. Univariate Normal clusters with equal cluster variations: $n_1 = 100$, $n_2 = 30$, $\mu_1 = 0$, $\mu_2 = 4$, $\sigma_1 = \sigma_2 = 1$. Summary: the two methods work equally well.

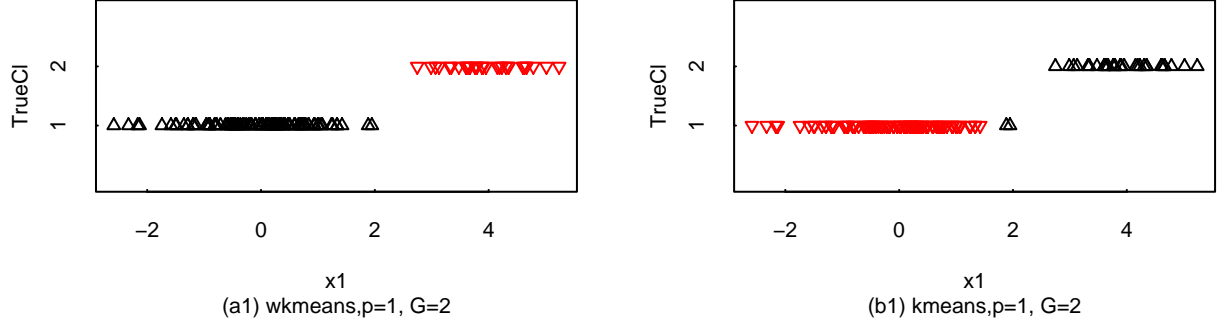


Figure 5.4: Comparing our proposed clustering method with the k -means method. Univariate Normal clusters with both unequal cluster sizes and unequal cluster variations: $n_1 = 100$, $n_2 = 30$, $\mu_1 = 0$, $\mu_2 = 4$, $\sigma_1 = 1$, $\sigma_2 = 0.3$. Summary: when clusters with larger cluster variations are also associated with larger cluster sizes, our proposed method performs better than the k -means method.

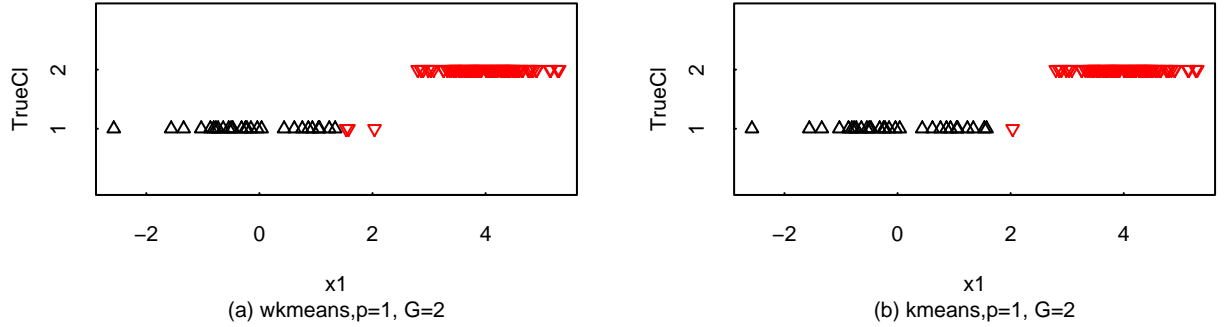


Figure 5.5: Comparing our proposed clustering method with the k -means method. Univariate Normal clusters with both unequal cluster sizes and unequal cluster variations: $n_1 = 30$, $n_2 = 100$, $\mu_1 = 0$, $\mu_2 = 4$, $\sigma_1 = 1$, $\sigma_2 = 0.3$. Summary: when clusters with larger cluster variations are associated with smaller sample sizes, our proposed method may perform worse than the k -means method.

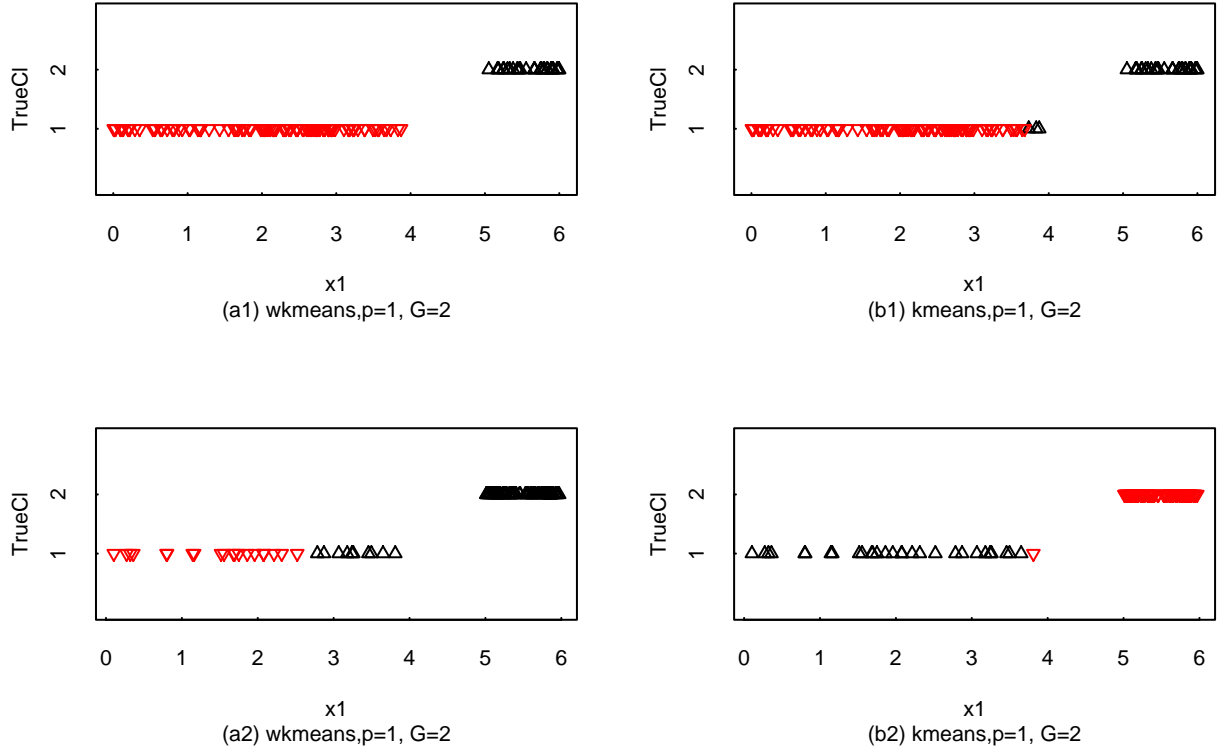
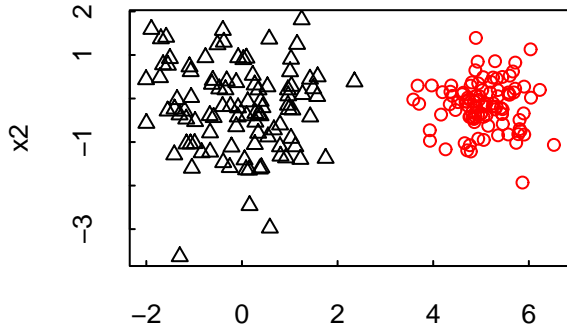
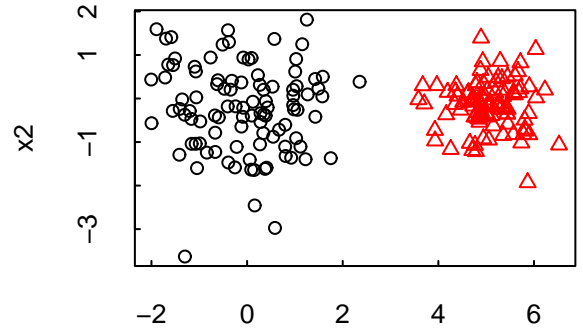


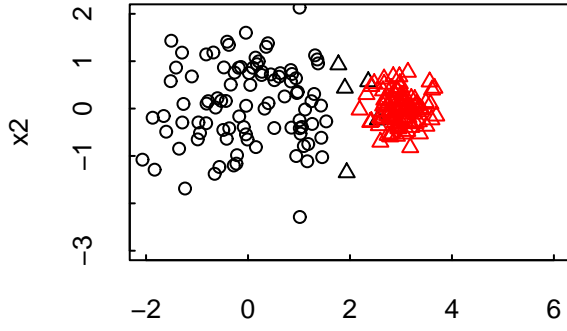
Figure 5.6: Comparing our proposed clustering method with the k -means method. Univariate uniform clusters with both unequal cluster sizes and unequal cluster variations: (a1, b1) $n_1 = 100$, $n_2 = 30$, $f_1 \sim U(0, 4)$, $f_2 \sim U(5, 6)$; (a2, b2) $n_1 = 30$, $n_2 = 100$, $f_1 \sim U(0, 4)$, $f_2 \sim U(5, 6)$. The same conclusions as obtained from the above examples of univariate Normal clusters.



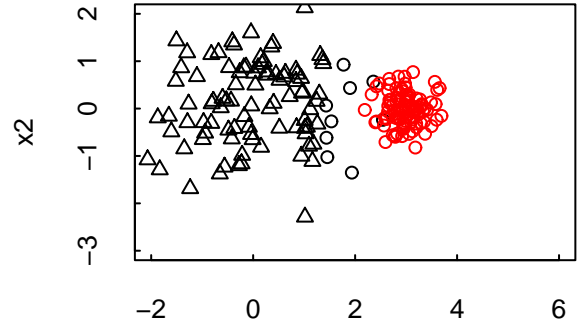
(a1) wkmeans, $p=2$, $G=2$



(b1) kmeans, $p=2$, $G=2$



(a2) wkmeans, $p=2$, $G=2$



(b2) kmeans, $p=2$, $G=2$

Figure 5.7: Comparing our proposed clustering method with the k -means method. Bivariate Normal clusters with equal cluster sizes: (a1, b1) $n_1 = n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (5, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.3^2 I$; (a2, b2) $n_1 = n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.3^2 I$. Summary: our proposed outperforms the k -means method.

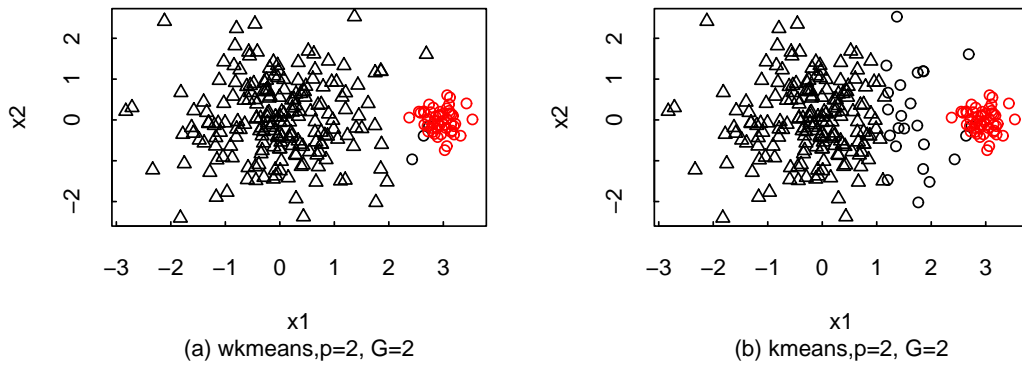


Figure 5.8: Comparing our proposed clustering method with the k -means method. Bivariate Normal clusters with both unequal cluster sizes and unequal cluster variations: $n_1 = 200$, $n_2 = 50$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$. Summary: when clusters with larger cluster variations are also associated with larger cluster sizes, our proposed method performs better than the k -means method.

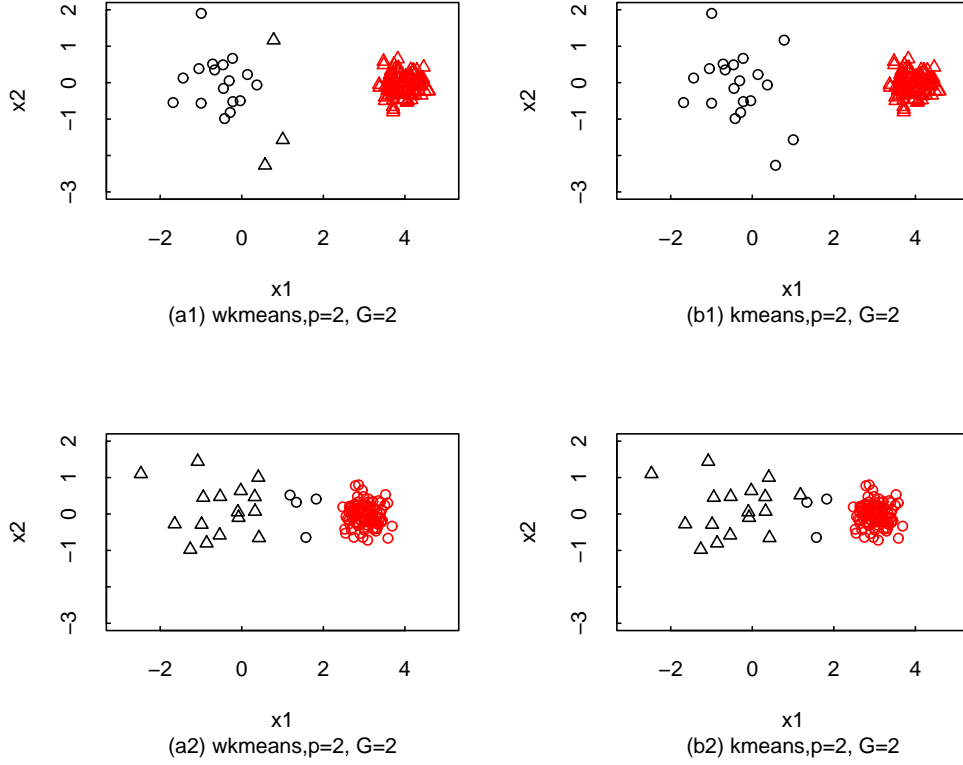


Figure 5.9: Comparing our proposed clustering method with the k -means method. Bivariate Normal clusters with both unequal cluster sizes and unequal cluster variations: (a1, b1) $n_1 = 20$, $n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (4, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$; (a2, b2) $n_1 = 20$, $n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$. Summary: when clusters with larger cluster variations are associated with smaller sample sizes, our proposed method may perform worse than the k -means method.

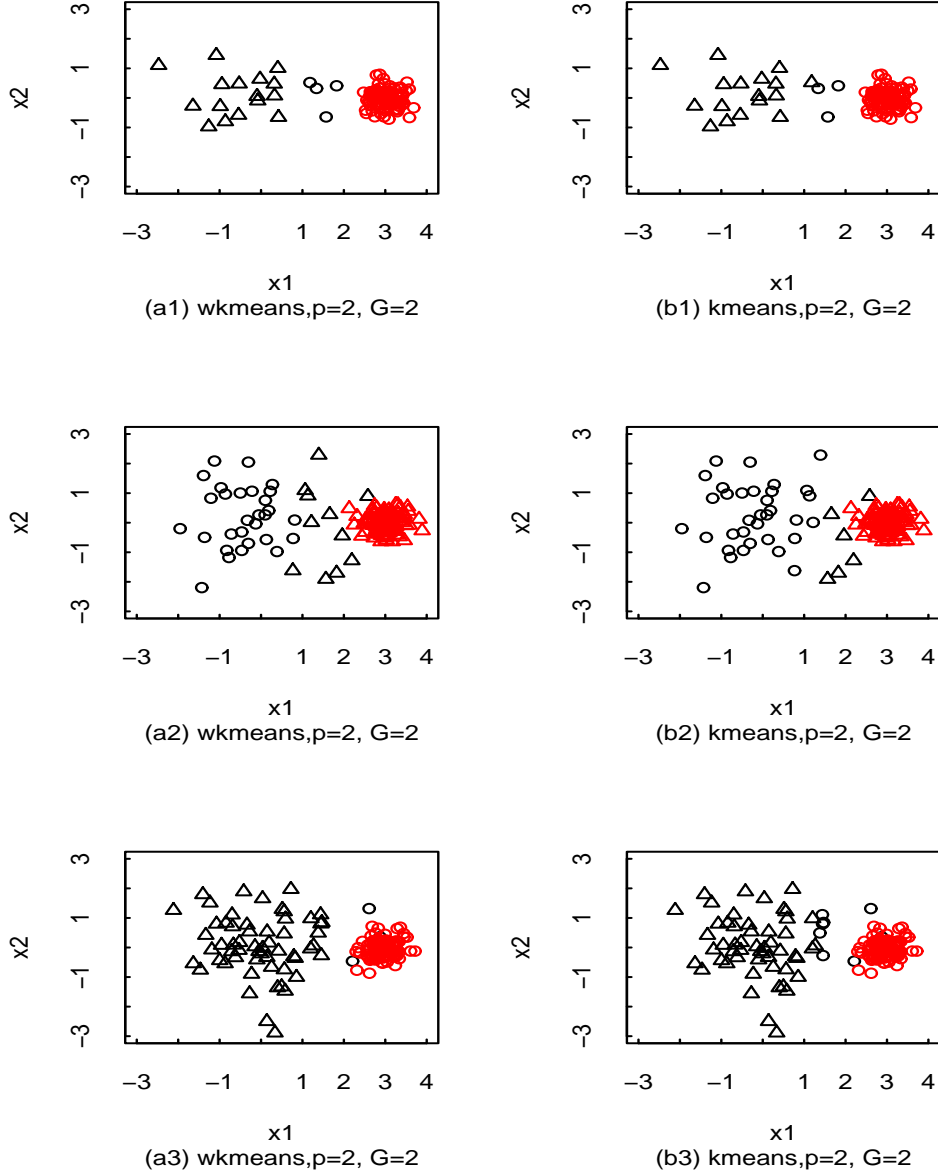


Figure 5.10: Comparing our proposed clustering method with the k -means method. Bivariate Normal clusters with both unequal cluster sizes and unequal cluster variations: (a1, b1) $n_1 = 20$, $n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$; (a2, b2) $n_1 = 40$, $n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$; (a3, b3) $n_1 = 60$, $n_2 = 100$, $\mu_1 = (0, 0)'$, $\mu_2 = (3, 0)'$, $\Sigma_1 = I$, $\Sigma_2 = 0.1^2 I$. Summary: when clusters with larger cluster variations are associated with smaller sample sizes, our proposed method may perform worse than the k -means method; as the degree of the discrepancy in cluster sizes decreases, our method will outperform the k -means method.

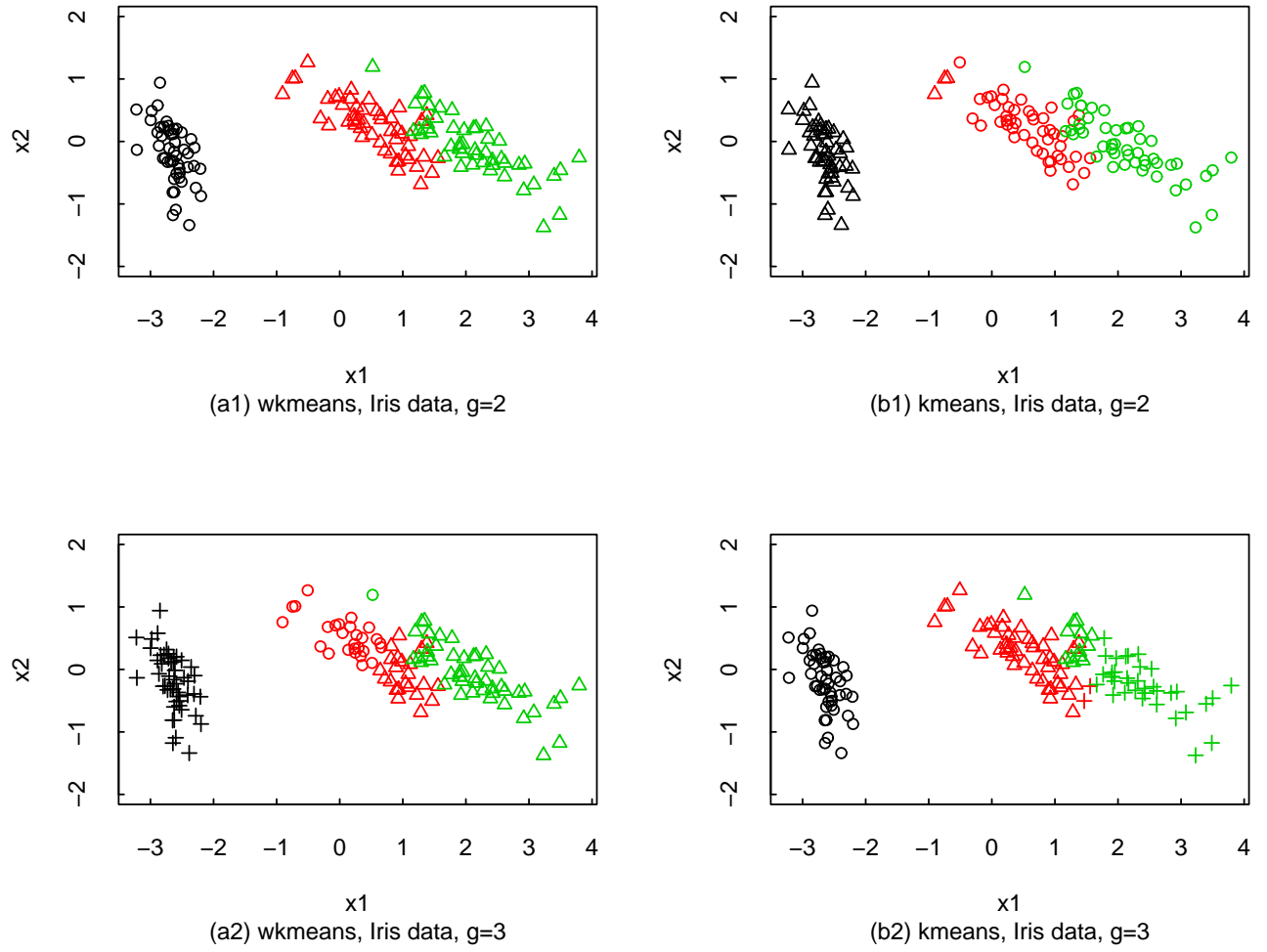


Figure 5.11: Comparing our proposed clustering method with the k -means method with the Iris data. To facilitate visualization of the clustering results, scores corresponding to the first two principal components are plotted.

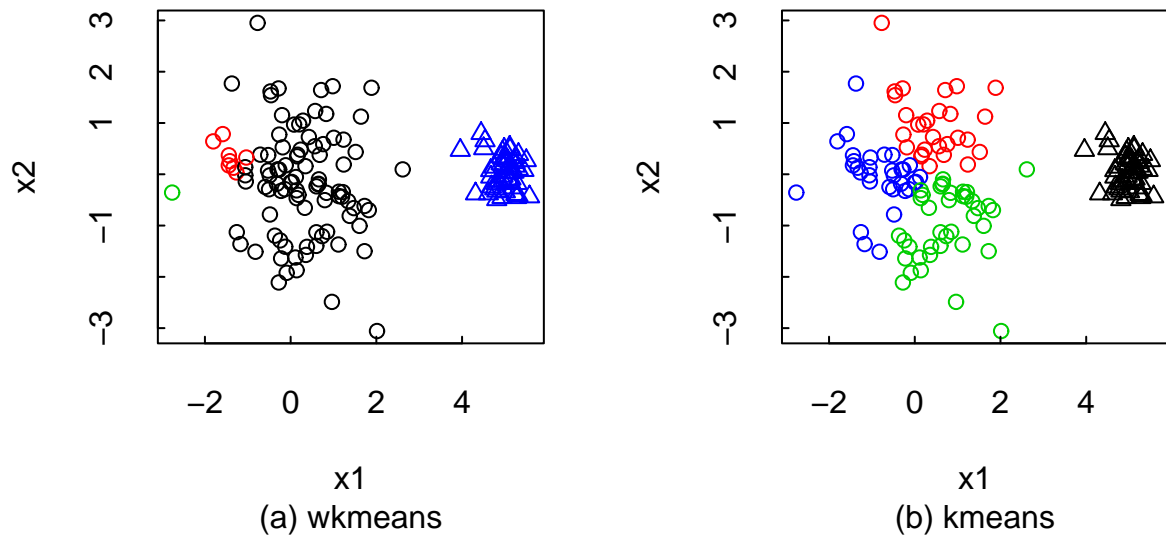


Figure 5.12: Classification results when the specified number of clusters is larger than the true cluster number in data: (a) our proposed method; (b) the k -means method.

Chapter 6

Summaries and Future Research

Cluster analysis is widely applied in various disciplines including biology, sociology, medical study and business. It has become an important multivariate analysis tool in many exploratory studies. However, although abundant researches on cluster analysis exist in the literature, none of them is convincingly acceptable, due to high complexity of real data sets. This dissertation focuses on two critical steps in cluster analysis: determining the number of clusters in a data set and developing a good clustering technique.

- Determining the number of clusters

An important step in cluster analysis is to determine the best estimate of the number of clusters in a data set, which has a deterministic effect on the clustering results. However, a completely satisfactory solution is not available in the literature. One aspect of my research is to tackle this best-number-of-clusters problem, which is specifically oriented at processing very complicated data that may contain multiple types of cluster structure. Potentially, our proposed methods will be applicable to any arbitrary research context where the cluster analysis is a suitable analytical tool. On the other hand, an interesting problem of recovering distinct patterns of changes in a particular feature across a series of experimental conditions presented in a data set has been discussed intensively in applications. An important example is to cluster objects (e.g. genes in a computational biology problem) based on their temporal profiles. We have been applying our methods of determining best-number-of-clusters successfully to microarray experimental data sets. We observe that our methods can be used to decide the number of different patterns and separate various patterns from each other in the

ways that the results can be interpreted very well.

In the current research, we propose a criterion for measuring the goodness-of-fit associated with the classification result given a specific number of clusters. Based on this criterion, two methods were developed to search for the optimal estimate of cluster number over a range of candidate values. One of the two methods can determine if the examined data is clustered or homogeneous, that is, if the data contain distinct clusters of observations or just came from one group. The other method will estimate the cluster number assuming that the data contain multiple clusters, that is, the number of clusters is more than 1. In addition, we propose a sequential type of clustering approach, called multi-layer clustering, by combining these two methods. This new clustering procedure is designed to detect clusters in complicated data sets. It has been successfully applied to cluster microarray gene expression data sets and it shows higher effectiveness than the one layer clustering approaches.

There are several possibilities for extending the current research. First, in our research only the data consisting of continuous variables and the squared Euclidean distance, which is generally suitable in microarray data analysis, are considered. However, the multi-layer clustering approach is virtually designed for any arbitrary distance measure. For wider application of this method, it is necessary to study the behavior of this method given other types of data (categorical data and mixture of continuous and categorical data) and various choices of dissimilarity/similarity measures. Secondly, in multi-layer clustering, we mainly focused our research using the k-means clustering approach. It has been noticed that a decision about the cluster number may depend on the type of clustering algorithm utilized. So, it is worthwhile to examine the possibility of applying this sequential estimating method to different clustering algorithms other than the k-means method. Finally, the multi-layer analysis is actually a “top-to-bottom” divisive procedure and we may improve the accuracy of this approach by performing an additional “bottom-to-top” agglomerative type of examination of the detected clusters. In order to put such an idea into an applicable procedure, an agglomerative criterion needs to be determined and correspondingly, an efficient computational solution would be necessary.

- A new clustering criterion

The other aspect of my current research is to solve the commonly called “equal-size” problem with k-means clustering. The k-means method is one of the most popular clustering techniques used in practice. One drawback of this method, however, is

that it tends to equalize the number of observations assigned to each cluster without considering the shapes of different clusters. We propose a new clustering algorithm which is able to account for the presence of large discrepancy in sample sizes between clusters. The proposed algorithm maintains the simplicity of the k-means method in computation. At the current stage, advantages of the proposed method over k-means clustering have been demonstrated empirically using simulated data when dimension is low to moderate. We intend to proceed with this research in two directions in the future: first, the behavior of the proposed method will be examined given large-scale data since, in cluster analysis, most of the real data are measured on a large number of variables, such as microarray data (when clustering samples based on gene expression profiles); secondly, efforts will be put into deriving the asymptotic properties of our proposed algorithm so that the behavior of the two clustering methods can be compared theoretically.

- Summary

The multi-layer clustering approach provides a powerful tool for clustering complicated data sets. It not only functions as an efficient method of estimating the number of clusters, but also, by superimposing a sequential idea, improves the flexibility and effectiveness of any arbitrary existing one-layer clustering method. In the literature, such an idea of sequentially detecting cluster structure is not new, but it has never been formally developed into an applicable procedure. Another merit of our method is that it can disclose the natural hierarchical structure of clusters which may provide extra information for interpreting the clustering results. Successful implementation of multi-layer clustering can be expected when applied to large-scale data in other research fields beside computational biology.

K-means clustering is widely used in application mainly because of its simplicity in computation and interpretation. The new clustering algorithm proposed in our research overcomes a major shortcoming of the k -means method while maintaining the similar practical convenience. We expect that our method has equal or higher accuracy of classification in most applications than the k-means method.

Bibliography

- [1] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, and et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences USA*, 96:6745–6750, 1999.
- [2] M. R. Anderberg. *Cluster Analysis for Applications*. New York: Academic Press, 1983.
- [3] P. Arabie and J. D. Carroll. MAPCLUS: A mathematical programming approach to fitting the ADCLUS models. *Psychometrika*, 445:211–235, 1980.
- [4] G. H. Ball and D. J. Hall. A novel method of data analysis and pattern classification. Technical report, Stanford Research Institute, California, 1965.
- [5] J. D. Banfield and A. E. Raftery. Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49:803–821, 1993.
- [6] E. M. L. Beale. Euclidean cluster analysis. *Bulletin of the International Statistical Institute*, 43:92–94, 1969.
- [7] A. Ben-Dor and Z. Yakhini. Clustering gene expression patterns. *Proceedings of the Third Annual International Conference on Computational Molecular Biology*, pages 33–42, 1999.
- [8] H. Bensmail and J. J. Meulman. Model-based clustering with noise: Bayesian inference and estimation. Technical report, Department of Statistics, Operations, and Management Science, University of Tennessee, Tennessee, 2003.
- [9] J. C. Bezdek. Numerical taxonomy with fuzzy sets. *Journal of Mathematical Biology*, 1:57–71, 1974.

- [10] D. A. Binder. Bayesian cluster analysis. *Biometrika*, 65:31–38, 1978.
- [11] R. K. Blashfield. Mixture model tests of cluster analysis: Accuracy of four agglomerative hierarchical methods. *Psychological Bulletin*, 83:377–385, 1976.
- [12] T. Calinski and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3:1–27, 1974.
- [13] J. G. Campbell, C. Fraley, F. Murtagh, and A. E. Raftery. Linear flaw detection in woven textiles using model-based clustering. *Pattern Recognition Letters*, 18:1539–1548, 1997.
- [14] J. W. Carmichael, L. A. Gorge, and R. S. Julius. Finding natural clusters. *Systematic Zoology*, 17:144–150, 1968.
- [15] J. W. Carmichael and P. H. A. Sneath. Taxometric maps. *Systematic Zoology*, 18:402–415, 1969.
- [16] R. B. Cattell and M. A. Coulter. Principles of behavioural taxonomy and the mathematical basis of the taxonome computer program. *British Journal of Mathematical and statistical Psychology*, 19:237–269, 1966.
- [17] Y. Cheng and G. M. Church. Biclustering of expression data. *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 93–103, 2000.
- [18] R. M. Cormack. A review of classification. *Journal of the Royal Statistical Society A*, 134:321–367, 1971.
- [19] R. M. M. Crawford and D. Wishart. A rapid multivariate method for the detection and classification of groups of ecologically related species. *Journal of Economics*, 55:505–524, 1976.
- [20] A. Dasgupta and A. E. Raftery. Detecting features in spatial point processes with clutter via model-based clustering. *Journal of the American Statistical Association*, 93:294–302, 1998.
- [21] S. Datta and S. Datta. Comparisons and validation of statistical clustering techniques for microarray gene expression data. *Bioinformatics*, 19:459–466, 2003.

- [22] N. E. Day. Estimating the components of a mixture of normal distributions. *Biometrika*, 56:463–474, 1969.
- [23] G. De Soete. Optimal variable weighting for ultrametric and additive tree clustering. *Quality and Quantity*, 20:169–180, 1986.
- [24] G. De Soete. OVWTRE: A program for optimal variable weighting for ultrametric and additive tree clustering. *Journal of Classification*, 5:101–104, 1988.
- [25] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood for incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39:1–38, 1977.
- [26] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.
- [27] S. Dudoit and J. Fridlyand. A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome Biology*, 3, 2002. research0036.1-0036.21.
- [28] A. W. F. Edwards and L. L. Cavalli-Sforza. A method for cluster analysis. *Biometrics*, 21:362–375, 1965.
- [29] M. B. Eisen, P. T. Spellmann, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences USA*, 95:14863–14868, 1998.
- [30] B. S. Everitt, S. Landau, and M. Leese. *Cluster Analysis*. Oxford University Press Inc., New York, NY, 4 edition, 2001.
- [31] R. A. Fisher. Multiple measurements in taxonomic problems. *Annals of Eugenics*, VII:179–188, 1936.
- [32] J. L. Fleiss and J. Zubin. On the methods and theory of clustering. *Multivariate Behavioral Research*, 4:235–250, 1969.
- [33] E. W. Forgy. Cluster analysis of multivariate data: Efficiency vs. interpretability of classification. *Biometrics*, 21:768–769, 1965.
- [34] E. B. Fowlkes and C. L. Mallows. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78:553–584, 1983.

- [35] C. Fraley and Raftery. A. E. How many clusters? Which clustering method? Answers via model-based cluster analysis. *Computer Journal*, 41:578–588, 1998.
- [36] C. Fraley and A. E. Raftery. Model-based clustering, discriminant analysis and density estimation. *Journal of the American Statistical Association*, 97:611–631, 2002.
- [37] H. P. Friedman and J. Rubin. On some invariant criteria for grouping data. *Journal of the American Statistical Association*, 62:1159–1178, 1967.
- [38] F. D. Gibbons and F. P. Roth. Judging the quality of gene expression-based clustering methods using gene annotation. *Genome Research*, 12:1574–1581, 2002.
- [39] I. Gitman and M. D. Levine. An algorithm for detecting unimodal fuzzy sets and its application as a clustering technique. *IEEE Transactions on Computers*, 19:583–593, 1970.
- [40] Sherman B. T. Hosack D. A. Yang J. Baseler M. W. Lane H. C. Glynn, D. Jr. and R. A. Lempicki. DAVID: Database for Annotation, Visualization, and Integrated Discovery. *Genome Biology*, 4(5): 3, 2003.
- [41] R. Gnanadesikan, J. R. Kettenring, and S. L. Tsao. Weighting and selection of variables for cluster analysis. *Journal of Classification*, 12:113–136, 1995.
- [42] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, and et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.
- [43] A. D. Gordon. A review of hierarchical classification. *Journal of the American Statistical Association A*, 150:119–137, 1987.
- [44] A. D. Gordon. *Classification*. Chapman & Hall/CRC, Boca Raton, FL, 2 edition, 1999.
- [45] J. Hartigan. *Clustering Algorithms*. New York: Wiley, 1975.
- [46] T. Hastie, R. Tibshirani, M. B. Eisen, A. Alizadeh, R. Levy, L. Staudt, W. C. Chan, D. Botstein, and P. Brown. ‘Gene shaving’ as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology*, 1:research0003.1–research0003.21, 2000.

- [47] R. J. Hathaway and J. C. Bezdek. Recent convergence for the fuzzy c-means clustering algorithms. *Journal of Classification*, 5:237–247, 1988.
- [48] S. Hohmann and W. H. Mager. *Yeast Stress Responses*. Springer: Berlin, 2003.
- [49] L. Hubert. Monotone invariant clustering procedure. *Psychometrika*, 38:47–62, 1973.
- [50] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, pages 193–218, 1985.
- [51] A. K. Jain and R. Dubes. *Algorithms for Clustering Data*. Englewood Cliffs, NJ: Prentice-Hall, 1988.
- [52] R. C. Jancey. Multidimensional group analysis. *Australian Journal of Botany*, 14:127–130, 1966.
- [53] J. O. Katz and E. L. Rohlf. Function point cluster analysis. *Systematic Zoology*, 22:295–301, 1973.
- [54] L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data. An Introduction to Cluster Analysis*. Wiley-Interscience, New York, 1990.
- [55] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59–69, 1982.
- [56] T. Kohonen. *Self-Organizing Maps: Second Extended Edition, Springer Series in Information Sciences*. Springer-Verlag, Berlin, 1997.
- [57] W. J. Krzanowski and Y. T. Lai. A criterion for determining the number of groups in a data set using sum of squares clustering. *Biometrics*, 44:23–34, 1988.
- [58] G. L. Liu. *Introduction to Combinatorial Mathematics*. McGraw Hill, New York, 1968.
- [59] P. Macnaughton-Smith, W. T. Williams, M. B. Dale, and L. G. Mockett. Dissimilarity analysis: a new technique of hierarchical sub-division. *Nature*, 202:1034–1035, 1964.
- [60] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1:281–297, 1967.
- [61] F. H. C. Marriott. Practical problems in a method of cluster analysis. *Biometrics*, 27:501–514, 1971.

- [62] F. H. C. Marriott. Optimization methods of cluster analysis. *Biometrika*, 69:417–421, 1982.
- [63] G. J. McLachlan and K. E. Basford. *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, New York, 1988.
- [64] G. J. McLachlan, R. W. Bean, and D. Peel. A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, 18:413–422, 2002.
- [65] G. W. Milligan. An examination of the effect of six types of error perturbation on fifteen clustering algorithms. *Psychometrika*, 45:325–342, 1980.
- [66] G. W. Milligan. A Monte Carlo study of thirty internal criterion measures for cluster analysis. *Psychometrika*, 46:187–199, 1981.
- [67] G. W. Milligan. A study of the comparability of external criteria for hierarchical cluster analysis. *Multivariate Behavioral Research*, 21:441–458, 1986.
- [68] G. W. Milligan. A validation study of a variable weighting algorithm for cluster analysis. *Journal of Classification*, 6:53–71, 1989.
- [69] G. W. Milligan. Clustering validation: results and implications for applied analyses. *Clustering and Classification*, pages 341–375, 1996.
- [70] G. W. Milligan and M. C. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50:159–179, 1985.
- [71] G. W. Milligan and M. C. Cooper. Methodological review: Clustering methods. *Applied Psychological Measurement*, 11:329–354, 1987.
- [72] G. W. Milligan and M. C. Cooper. A study of variable standardization. *Journal of Classification*, 5:181–204, 1988.
- [73] R. Moronna and P. M. Jacovkis. Multivariate clustering procedures with variable metrics. *Biometrics*, 30:499–505, 1974.
- [74] S. Mukherjee, E. D. Feigelson, G. J. Babu, F. Murtagh, C. Fraley, and A. E. Raftery. Three types of gamma ray bursts. *Astrophysical Journal*, 508:314–327, 1998.
- [75] P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

- [76] A. J. Scott and M. J. Symons. Clustering methods based on likelihood ratio criteria. *Biometrics*, 27:387–397, 1971.
- [77] W. Sha, A. Martins, V. Shulaev, and P. Mendes. A time course study of oxidative stress response in *S. cerevisiae* cultures exposed to cumene hydroperoxide. *To be submitted*, 2006.
- [78] R. N. Shepard and P. Arabie. Additive clustering: Representations of similarities as combinations of discrete overlapping properties. *Psychological Review*, 86:87–123, 1979.
- [79] R. C. Singleton and W. Kautz. Minimum squared error clustering algorithm. Technical report, Stanford Research Institute, Stanford, CA, 1965.
- [80] P. H. Sneath and R. R. Sokal. *Numerical taxonomy*. San Francisco: Freeman, 1973.
- [81] C. A. Sugar and G. M. James. Finding the number of clusters in a dataset: an information-theoretic approach. *Journal of the American Statistical Association*, 98:750–763, 2003.
- [82] M. J. Symons. Clustering criteria and multivariate normal mixtures. *Biometrics*, 37:35–43, 1981.
- [83] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. S. Lander, and T. R. Golub. Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proceedings of the National Academy of Sciences USA*, 96:2907–2912, 1999.
- [84] R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of data clusters via the gap statistic. *Journal of the Royal Statistical Society B*, 63:411–423, 2001.
- [85] M. Vichi. On a flexible and computationally feasible divisive clustering technique. *Rivista di Statistica Applicata*, 18:199–208, 1985.
- [86] D. Voges, P. Zwickl, and W. Baumeister. The 26S proteasome: a molecular machine designed for controlled proteolysis. *Annual Review of Biochemistry*, 68:1015–1068, 1999.
- [87] N. Wang and A. E. Raftery. Nearest neighbor variance estimation (NNVE): Robust covariance estimation via nearest neighbour cleaning (with discussion). *Journal of the American Statistical Association*, 97:994–1019, 2002.

- [88] J. H. Ward. Hierarchical groupings to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244, 1963.
- [89] R. Wehrens, L. Buydens, C. Fraley, and A. E. Raftery. Model-based clustering for image segmentation and large datasets via sampling. Technical report, University of Washington, Department of Statistics, 2003.
- [90] R. Wehrens, A. Simonetti, and L. Buydens. Mixture-modeling of medical magnetic resonance data. *Journal of Chemometrics*, 16:1–10, 2002.
- [91] W. T. Williams and J. M. Lambert. Multivariate methods in plant ecology, Association analysis in plant communities. *Journal of Ecology*, 54:427–445, 1959.
- [92] D. Wishart. Mode analysis. In A. J. Cole, editor, *Numerical Taxonomy*. Academic Press, New York, 1969.
- [93] D. Wishart. An improved multivariate mode-seeking cluster method. 1973. Paper presented at Royal Statistical Society General Applications Section and Multivariate Study Group Conferences.
- [94] W. H. Wolberg and O. L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences, USA*, 87:9193–9196, 1990.
- [95] M. A. Wong. A hybrid clustering method for identifying high-density clusters. *Journal of the American Statistical Association*, 77:841–847, 1982.
- [96] M. A. Wong and T. Lane. A k th nearest neighbour clustering procedure. *Journal of the Royal Statistical Society, B*, 45:362–368, 1983.
- [97] M. A. Woodbury and K. G. Manton. A new procedure for the analysis of medical classification. *Methods of Information in Medicine*, 21:21–220, 1982.
- [98] M. Yan and K. Ye. Determining the number of clusters using the weighted gap statistic. *Submitted*.
- [99] K. Y. Yeung, C. Fraley, A. Murua, A. E. Raftery, and W. L. Ruzzo. Model-based clustering and data transformations for gene expression data. *Bioinformatics*, 17:977–987, 2001.

- [100] P. Zwickl, D. Voges, and W. Baumeister. The proteasome: a macromolecular assembly designed for controlled proteolysis. *Philosophical Transactions of the Royal Society of London. Series B. Biological Sciences*, 354:1501–1511, 1999.

Vita

Mingjin Yan was born in 1978 in Gansu, China. She entered Fudan University in China in 1995 and obtained her Bachelor's degree in Statistics and Probability in 1999. In 2001, Mingjin Yan was enrolled in the Master's program in Statistics at Virginia Polytechnic Institute and State University. She graduated with her Master's degree in Statistics in 2002 and continued to pursue her Ph.D. degree in Statistics. She received her Ph.D. in Statistics in December 2005.