UNSUPERVISED DATA MINING METHODS FOR FUNCTIONAL DATA ANALYSIS AND FEATURE SELECTION

by

PANAYA RATTAKORN

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2009

I delicate this dissertation to my father, Decha, mother, Dr. Amara,

and everyone in my family.

ACKNOWLEDGEMENTS

I would like to sincerely express my deepest gratitude to my supervising professor Dr. Seoung Bum Kim for his guidance, understanding, and most importantly, his patience during my doctoral study. I have been remarkably fortunate to have an advisor who gave me invaluable advice and knowledge to reach my goal. Without him, this dissertation would not been complete.

I am extremely grateful to my supervising committee members, Dr. Victoria C. P. Chen, Dr. Jay Rosenberger, and Dr. Yuan Bo Peng, for their guidance over the years and generously given their time and efforts on this dissertation. I would like to thank Dr. Kwok-Leung Tsui (from Georgia Institute of Technology) for his precious advice during our meeting at INFORMS and Arlington. I would also like to extend a sincere thanks to Dr. J. C. Chao for his insightful advices on neurostimulation data.

I would like to thank all COSMOS colleagues, especially Drs. Chivalai Temiyasathit, Thuntee Sukchorat, Prattana Punnakitikashem, Siriwat Visoldilokpun, Durai Sundaramoorthi, Huiyuan Fan, Prashant Tarun, Passakorn Phananiramai, Chatabush Roongrat, Ching-feng Lin, Poovich Phaladiganon, Panitarn Chongfuangprinya and Weerawat Jitpitaklert, for helpful discussions and friendships. My special thanks go to my friends: Sarinya and Jason Eberle, Apinan Kanyalak, Krirkkrit Sripaipan, Orn Sangtherapitikul, and many people from Meditation Center of

Texas, who always supports me during my hardships. I also would like to thank Luminant Power for funding my PhD study.

Most importantly, I wish to extend my warmest thanks to my parents who has encouraged me throughout this long journey. I deeply appreciate all of their love and support they have given me along the way. I could not have done it without them. Lastly, I am extremely indebted to my grandparents, aunts, uncles, brother, and cousins for their love and supports.

July 23, 2009

ABSTRACT

UNSUPERVISED DATA MINING METHODS FOR FUNCTIONAL DATA ANALYSIS AND FEATURE SELECTION

PANAYA RATTAKORN, PhD

The University of Texas at Arlington, 2009

Supervising Professor: Seoung Bum Kim

The objective of this dissertation is to develop new unsupervised data mining methods for functional data analysis and feature selection. Unsupervised learning is a modeling process that facilitates the extraction of implicit patterns and elicits the natural groupings within the dataset without using any information from the output (response) variable. This dissertation consists of two main parts: (1) unsupervised clustering approaches for functional data analysis and (2) unsupervised feature selection.

Functional data analysis has gained significant attention from a variety of disciplines. In this dissertation we propose an effective clustering procedure to categorize a number of profiles that are formed with nonlinear functions. The proposed clustering procedure first smoothes the data and then transforms the smoothed data to obtain their functional form. The coefficients of the function obtained from the preceding transformation step are used for clustering. Simulation studies under various scenarios indicated that our proposed clustering procedure correctly identified the true clusters and yielded better clustering results than a latent class cluster analysis, one of the existing clustering methods. Furthermore, the effectiveness of the proposed clustering procedure was demonstrated using real pain data in which the main objective is to characterize the responses of 144 spinal cord dorsal horn neurons with graded thermal stimuli that range from 37° C to 51° C in 2-degree C increments. The results showed that the proposed clustering strategy can successfully elicit natural grouping of the neurons with similar response patterns to graded thermal stimuli.

Feature selection has received considerable attention in various areas to select informative features and simplify the statistical model by achieving dimensional reduction. One of the widely used methods for dimensional reduction includes principal component analysis (PCA). Nevertheless, PCA suffers from the lack of interpretability with respect to the original feature because the reduced dimensions are linear combinations of a large number of original features. Traditionally, two or three dimensional loading plots provide information to identify important original features in the first few principal component dimensions. However, the interpretation of a loading plot is frequently subjective, particularly when the number of features is large. In this study, we proposed an unsupervised feature selection method that combines weighted principal components (PCs) with thresholding algorithm. The weighted PCs are obtained by weighted sum of first *k* PC of interest. Each of the *k* loading values in the weighted PC reflects the contribution of each individual feature. We also proposed the thresholding algorithm that identifies the significant features. Our experimental results with both the simulated and real datasets demonstrated the effectiveness of the proposed unsupervised feature selection method.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

As computer and database technology grows rapidly, vast amount of data are being generated with high complexity. The problems arise when the traditional data analysis methods are unable to handle these types of data. The data mining approach is employed to overcome these problems and can be viewed as a multidisciplinary joint effort from machine learning, information technology, and statistics. Recently, data mining has gained a great deal of attention from various disciplines including manufacturing, telecommunications, economic, gene expression studies, and biomedical studies and etc. Among those disciplines, functional data are commonly found because usually data are collected over long periods of time. In bioinformatics, the data often involve a large number of genes e.g. several thousands. These types of data often challenge analytical capabilities because their high dimensionality and their complexity. The major purpose of this dissertation is to present unsupervised data mining algorithms as a tool for functional data analysis and feature selection.

1.1 Functional Data Analysis

Functional data analysis is an analysis of curves or functions of data. The basic concept of functional data analysis is to consider the observed functions as a single objects rather than a sequence of individual observations (Ramsay and Silverman, 2005). Longitudinal data can be considered as functional data because the observations are collected as a function of time or other continuous variables. Functional data analysis differs from the traditional analysis because it uses the rates of change or derivatives of the curves to analyze and visualize data. Figure 1.1 illustrates example of functional data. The *y*-axis represents the observations that corresponding to the function of *x*.

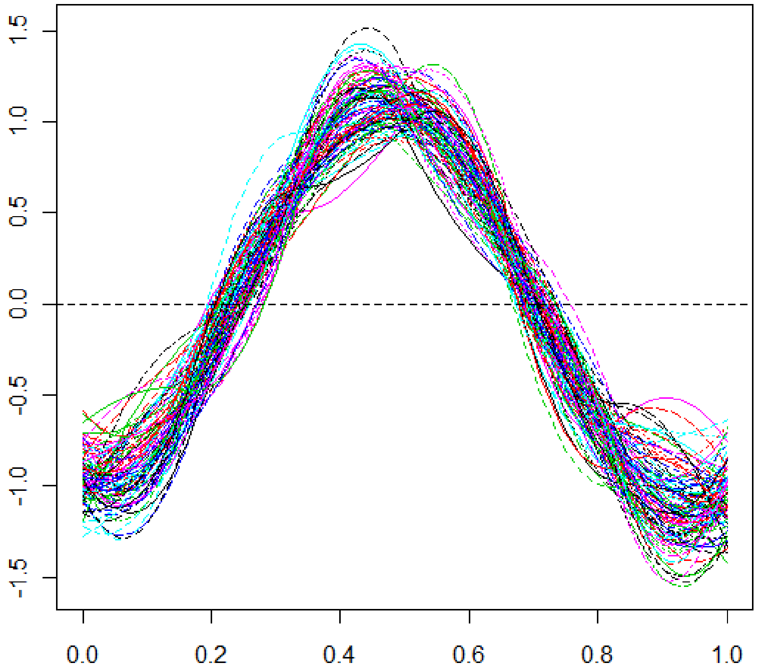


Figure 1.1 Illustration of functional data.

According to Ramsay and Silverman (2005), the objectives of functional data analysis are as follows: (i) to discover pattern or variation among data; (ii) to assist further analysis; (iii) to visualize data and highlight characteristics of data.

1.2 Data Mining

According to Wegman and Solka (2005), the definition of data mining can be defined as an extension process of exploratory data analysis to discover the unknown and unanticipated structure in the data. Data mining tools are commonly separated into two categories: supervised learning and unsupervised learning. Supervised approaches require both the input (predictors) variable and the output (response) variable, while unsupervised approaches rely solely upon the input (explanatory) variables.

1.2.1 Supervised Learning

Supervised learning is a data mining technique to obtain a model from a pair of input data and desired outputs. The characteristics of output variables are used to categorize supervised learning into either classification or regression problems. For classification problems, the output variable is a categorical variable. The task of classification is to assign existing class labels to the unknown observation. In the regression problems, the response is continuous variable so that the task is to predict the value of function for any valid predictors.

Linear regression models have been the most widely used approach for regression problems because of their simplicity and the models often provide an adequate and interpretable description of relationship between response and predictors.

The variable selection algorithms are used to obtain the model with the good sets of predictors. There has been several variable selection approaches developed for linear regression models. These methods are (1) forward selection, (2) backward elimination, (3) stepwise selection, and (4) best subsets selection (Larose, 2006).

1.2.2 Unsupervised Learning

Unsupervised approaches attempt to extract the information without using any response variables. Although visualization techniques elicit the natural groupings of the observations, the interpretation of graphical results is not necessarily straightforward. Clustering analysis is an unsupervised approach that systematically partitions the observations by minimizing within-group variations and maximizing between-group variations, then assigning a cluster label to each observation. Clustering analysis includes hierarchical and nonhierarchical methods.

Nonhierarchical clustering algorithms aim to group observations into *k* clusters. The number of *k* can be determined as a part of clustering procedure or can be specified in advance. The *k*-means clustering algorithm is one of the most popular of non hierarchical clustering methods (Kim et al., 2002). A brief summary of the *k*-means clustering algorithm is as follows. Starting with *k* seed points, each observation is assigned to one of the *k* seed points close to the observation, which creates *k* clusters. Then, seed points are replaced with the mean of the currently assigned clusters. This procedure is repeated with updated seed points until the assignments do not change. The *k*-means clustering algorithm depends on distance metrics and the number of clusters (*k*). A variety of distance metrics are available. Generally, the Euclidean distance is a widely used distance metric to analyze the multivariate data.

Principal component analysis (PCA) is one of the most widely used methods for unsupervised learning. The principal component analysis has two major proposes. The first objective is to reduce the data dimension and the second objective is to illustrate interpretation for the high dimensional data (Johnson and Wichern, 2001). The principal component is the linear combination of the original variables which transform into uncorrelated new variables. The transformed variables, called principal components (PCs) are uncorrelated, and generally, the first few PCs are sufficient to account for most of variability of the entire data. Thus, plotting the observations using these first few PCs facilitates the visualization of high-dimensional data.

A brief summary of the algorithm of PCA is as follows: Let $X' = \lfloor X_1, X_2, \cdots, X_p \rfloor$ have covariance matrix **Σ**, with eigenvector e_i' ($i = 1, \ldots, p$) corresponding to the eigenvalue λ_i , where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p$ (Johnson and Wichern, 2001). The principal component is the linear combination of the original variables which transform into uncorrelated new variables $Y_1, Y_2, Y_3, \ldots, Y_p$ as follows.

$$\begin{aligned} Y_1 &= e_1'X = e_{11}X_1 + e_{12}X_2 + \ldots + e_{1p}X_p \\ Y_2 &= e_2'X = e_{21}X_1 + e_{22}X_2 + \ldots + e_{2p}X_p \\ &\vdots \\ Y_p &= e_p'X = e_{p1}X_1 + e_{p2}X_2 + \ldots + e_{pp}X_p \end{aligned} \qquad (1.1)$$

In order to reduce the dimensionality of data, let $Y = [y_1, \quad y_2, \quad \ldots, \quad y_d]^T$, and $d < p$, we able to obtain the new variables y_j ($i = 1, \ldots, d$).

1.3 Applications

As previous mentioned, data mining has wide range of applications, including manufacturing, finance, telecommunications, bioinformatics, and neuronal studies. In manufacturing, data mining methods are widely implemented to predict the outcome of manufacturing process such as defective parts. In finance, credit analysis and prediction of loan payments are always critical to the business of financial firms. Data mining methods assist the firms to evaluate risk of their customers and also identify the important factors and eliminate irrelevant factors. In telecommunications, data mining aids providers to facilitate their churn detection activities and analyze fraudulent patterns and any unusual activities. In past decades, there has been an explosive growth in bioinformatics. Data mining methods support researchers to better understand the biological process e.g. molecular patterns, DNA and protein sequences. Combined with microarray technology, data mining methods have been applied to discover gene expression patterns and identify complex disease genes and biomarkers for disease diagnostic. In this dissertation, we explicitly focus on neurostimulation data and microarray gene expression data.

Microarray gene expression data contain thousands of genes reflecting the state of cell with different protein and mRNA compositions (Ding, 2002). Of the thousands of genes, there are only a small number of significant features. Thus, selecting of most important and relevant genes is an important task. Figure 1.2 illustrates example of microarray gene expression data. The unsupervised feature selection is applied in order to select a set of relevant genes.

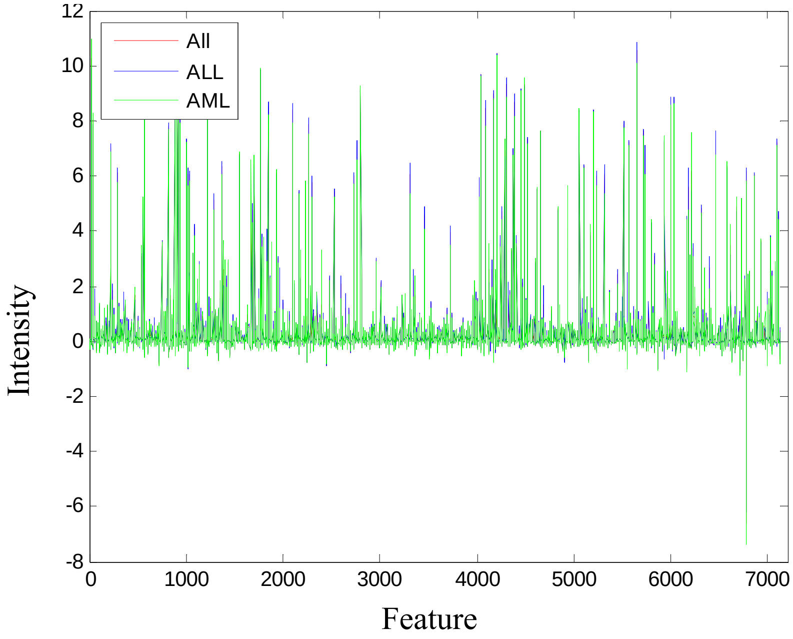


Figure 1.2 Illustration of microarray gene expression data.

1.4 Organize of this Dissertation

This dissertation begins with a brief introduction of data mining and biomedical signals in Chapter 1. Chapter 2 proposes a clustering strategy procedure for neuronal response profiles in graded thermal stimuli. Chapter 3 presents an unsupervised feature selection approach by using weighted principal component combines with thresholding algorithms. Finally, Chapter 4 discusses the future research directions.

CHAPTER 2

AN EFFECTIVE CLUSTERING PROCEDURE OF NEURONAL RESPONSE PROFILES IN GRADED THERMAL STIMULATION

2.1 Introduction

In recent years, functional data analysis has received considerable attention in a variety of fields of applied science in which collected data are formed with functions (Ramsay and Silverman, 2005). In the present study we apply functional data analysis to study how the dorsal horn neurons of a rat respond to graded heat stimuli. Pain is one of the most extensively studied neural systems. Pain can be evoked in the periphery by mechanical, thermal, and chemical stimuli. In the spinal cord, dorsal horn neurons play a critical role in receiving not only excitatory input from peripheral tissues (e.g., skin), but also descending inhibitory input from supraspinal structures (e.g., the brain stem). Thus, the response properties of these spinal dorsal horn neurons determine the final output of neural signals to the higher center. It is expected that the outcome of this analysis will help better classification of spinal dorsal horn neurons, which will be used to correlate the efficacy of the descending inhibition by electrical neurostimulation.

Neurostimulation has been efficiently used to reduce or block pain signals (Burchiel et al., 1996). A neurostimulation device over the surface of the spinal cord or over the primary motor cortex, for example, could deliver low levels of electrical current or heat directly to nerve fibers or neurons. Therapeutic studies have shown that when used on carefully selected chronic-pain patients, neurostimulation could significantly improved pain relief and reduce the need for narcotic medications (Burchiel et al., 1996). Neurostimulation has several significant advantages. First, it can be very effective, with few side effects, for certain conditions. Second, the implanted device can be controlled by patients or doctors with little risk of addiction or overdose. Third, the implant can be removed if it does not achieve the desired level of pain or symptomatic relief.

Several studies have been conducted to characterize the spinal cord dorsal horn neurons in a rat. Chung et al. (1986) and Owens et al. (1992) studied the response of spinal cord dorsal horn neurons to mechanical stimuli. Senapati et al. (2005) conducted electrical stimulation to induce inhibition of the responses of spinal cord dorsal horn neurons to noxious mechanical stimuli. Furthermore, a number of studies of spinal cord dorsal horn neurons were conducted to determine their response to graded heat stimuli (Kanui, 1985; Craig et al., 2001; Hayes and Rustioni 1981). The main conclusion from these studies has been that the responses of dorsal horn neurons increase as heat stimuli increase. Despite the number of studies conducted in this direction, few efforts have been made to characterize the response pattern of spinal cord dorsal horn neurons in *deeper* laminae to heat stimuli (Borzan et al., 2005). Borzan et al., (2005) used a latent class cluster analysis (LCCA) to categorize the dorsal horn neurons in the deep laminae of the rat in response to graded heat stimulation. They found five distinct response patterns in these neurons. LCCA is one of the model-based clustering methods that have been used to elicit the natural groupings of multivariate data. However, the responses of the dorsal horn neurons are functional observations in that the responses can be represented as a function of the temperature. Thus, the direct use of a clustering method for this functional dataset may lead to inefficient and unsatisfactory conclusions.

Indeed, the data often involve a number of complex nonlinear profiles that challenge analytical capabilities. To address this problem, we propose an effective clustering procedure to group a number of nonlinear profiles. In general bioinformatics research, a number of studies have used clustering methods, especially when gene expression data were the target, to discover patterns in functional data. Wakefield et al. (2002) proposed a model-based clustering method to efficiently cluster gene expression data. Schliep et al. (2003) suggested clustering genes based on hidden Markov models (HMM). They claim that HMM is robust to the noisy and frequently missing data and has an ability to handle cyclic and noncyclic biological time series. Luan and Li (2003) proposed a mixed-effects model with B-splines for clustering time-course gene expression data. Each of the aforementioned methods has its own advantages and disadvantages, and choosing among them depends on the purpose of the application. Recently, Serban and Wasserman (2005) developed CATS (clustering after transformation and smoothing) as a method to cluster a number of profiles and applied it to time-course gene expression data. The CATS method involves three required preprocessing steps before clustering analysis. These are transformation, smoothing, and screening, in that specific order. The purpose of the screening step in CATS is to filter out any constant profiles that may adversely affect the clustering. However, testing the constant profiles involves a statistical threshold that can be determined subjectively.

Further, although the authors claimed that the screening step in CATS is important, their simulation study found that the screening step brought no significant clustering improvement. In the present study we propose a modified version of CATS that does not require a screening step and efficiently clusters a number of nonlinear profiles collected for the heat stimulation study.

The remainder of this chapter is organized as follows. Section 2.2 describes a set of analytical methods, including smoothing, transformation, clustering, and clustering validity measures, used in the proposed clustering procedure. Section 2.3 presents simulation studies to investigate the performance of the proposed procedure and to compare it with one of the existing clustering methods. Section 2.4 presents a real application study to examine the response patterns of deep dorsal horn neurons to heat stimuli. Section 2.5 contains our concluding remarks.

2.2 Analytical Methods

The basic idea of our proposed clustering procedure is to conduct the clustering analysis after the data are smoothed and transformed, in that specific order. It should be noted that the order of smoothing and transformation is different from CATS. Detailed descriptions of our choices for smoothing and transformation methods are presented in the following subsections.

2.2.1 Smoothing

Generally, smoothing algorithms were used to remove noises and are presented in the data so as to highlight the real signals. A robust locally weighted regression (LOESS) method was used for smoothing a set of data points (x_i, y_i) for $i = 1, 2, \ldots, n$ in which the fitted value is the polynomial fit to the data (Cleveland, 1979). LOESS starts with a local polynomial fitting of a subset of data and then applies a robust fitting method to obtain the final fit. One of the parameters of LOESS is a smoothing factor that controls the degree of smoothness of the regression function. The larger the smoothing parameter is, the less sensitive it is to fluctuations in the data. The smoothing parameter can be determined by a cross-validation technique (Cleveland and Devlin, 1988). Another parameter in LOESS is the order of the polynomial function. Using a zero degree is the simplest computation that converts LOESS into a weighted moving average. Typically, smoothing parameter of one is sufficient to smooth the data (Cleveland, 1979).

2.2.2 Transformation

Having smoothed the data, we converted them into a functional form. In other words, we fitted the profiles of the smoothed data based on a set of basis functions that can be represented as follows:

$$x(t) = \sum_{k=1}^{K} c_k \phi_k(t), \tag{2.1}$$

where K is the number of basis functions, c_k are the coefficients, ϕ_k are polynomials, and t is the parameter (e.g., time).

Spline functions are commonly used for approximation of nonperiodic functional data (Ramsay and Silverman, 2005; Levitin et al., 2007). The B-spline basis is formed by joining polynomial functions of order K at fixed points, called knots. Let *L* be subintervals separated by the value of τ_i, where $l = 1,\ldots, L$-1. For each interval, *m* is the order of a polynomial. The spline function *S(t)* can be represented as follows:

$$S(t) = \sum_{k=1}^{m+L-1} c_k B_k(t,\tau), \tag{2.2}$$

where $B_k(t,\tau)$ is the value at *t* of the B-spline basis function as defined by the knot sequence τ, and c_k is the corresponding coefficient (Ramsay and Silverman, 2005).

Serban and Wasserman (2005) used the Fourier transform in their CATS method. The Fourier transform is efficient for the periodic data. However, the stimulation data from neuroscience often exhibit a nonperiodic pattern as shown in the latter part of this paper (please see Figure 2.3). Consequently, we propose here to use the B-spline function to transform the data.

2.2.3 Clustering Analysis

We performed clustering analysis to group the response profiles based on the coefficients of the B-spline functions. Clustering analysis systematically partitions the dataset by minimizing within-group variation and maximizing between-group variation and then assigning a cluster label to each observation (Xu and Wunsch II, 2005). The present study applied a *k*-means clustering algorithm to a set of the coefficients from the B-spline transform. A brief summary of the *k*-means clustering algorithm is as follows: Given *k* seed points, each observation is assigned to one of the *k* seed points close to the observation, a step that creates k clusters. Then, the seed points are replaced with the mean of the currently assigned clusters. This procedure is repeated with updated seed points until the assignments no longer change (Xu and Wunsch II, 2005). The k-means clustering algorithm depends on distance metrics and the number of clusters (k). A variety of distance metrics are available. Generally, the correlation distance is an appropriate choice to analyze the functional profile data because the correlation distance focuses on measuring the similarity in shapes between the two response profiles. As for determining the number of clusters, a number of methods are available. However, no consensus exists about which one is most satisfactory (Hastie et al., 2001). In the current study, we determined the appropriate number of k based on the opinions of domain experts.

An earlier study by Borzan et al. (2005) used a LCCA algorithm to cluster the same heat stimulation data used in the present study. LCCA is a model-based clustering approach based on finite mixture probability distributions. LCCA provides an estimate of the number of clusters in multivariate data with statistical confidence (Borzan et al., 2005). A brief explanation of LCCA is as follows: Let c be the number of clusters, n be the number of observations, and let π_i be the fraction of observations that belongs to cluster i (for $i = 1,\ldots, c$). The probability density function can be denoted by $f(x, \mu_i, \Sigma)$ where μ_i is the mean vector and Σ is the covariance matrix. The likelihood function can be computed as follows:

$$L(\theta_1,\ldots,\theta_c) = \prod_{j=1}^{n} \left\{ \sum_{i=1}^{c} \pi_i f(x_j; \mu_i, \Sigma_i) \right\}, \tag{2.3}$$

where $\theta_i = (\pi_i, \mu_i, \Sigma_i)$.

Likelihood ratio testing methodology was applied to acquire a good latent class model. We fit models starting with a one-cluster model and then added another cluster for each successive model. The difference between the log-likelihood with the *c*-1 clusters and with the *c* clusters model represents the amount of fit improvement associated with the *c* clusters model in comparison with the *c*-1 clusters model. In general, the difference in maximized log-likelihood can be tested with the standard chi-square distribution method for a nested model. However, if the model is not nested, a bootstrapping method can be used to obtain the test statistic (Borzan et al., 2005).

2.2.4. Clustering Validity Measures

To demonstrate the effectiveness of the clustering approach, the following misclustering rate φ can be defined for *N* profiles:

$$\varphi = \frac{1}{N}\sum_{j=1}^{N} I\left(C_j, \hat{C}_j\right), \tag{2.4}$$

where C_i and $\hat{C}_i$ represent, respectively, the true clustering label and the estimated clustering label of j th profile. *I* is an indicator function defined as follows:

$$I\left(C_j, \hat{C}_j\right) = \begin{cases} 0, & if \quad C_j = \hat{C}_j \\ 1, & if \quad C_j \neq \hat{C}_j \end{cases} \tag{2.5}$$

Thus, φ represents the fraction of the profiles that are incorrectly clustered among *N* profiles.

Another way to measure the quality of clustering is the silhouette width (Rousseeuw, 1987). The silhouette width for the *j*th observation is defined as follows:

$$S_i = \frac{(b_i - a_i)}{\max(a_i, b_i)}, \tag{2.6}$$

where a_i is the average dissimilarity to all other points in its own cluster and b_i is the average distance between the i^{th} observation and the other observations that are in the closest neighboring cluster. The silhouette value ranges from -1 to 1. The silhouette value closer to 1 indicates that its corresponding observation is well clustered, and the value closer to -1 indicates that its corresponding observation is not well clustered. The overall quality of clustering can then be determined from the average silhouette width by averaging S_i over all observations. A large average silhouette width indicates better clustering performance (Kaufman and Rousseeuw, 1990).

2.3 Simulation Study

A simulation study was conducted to examine the properties of the proposed clustering procedure and determine its effectiveness. Let Y_{it} be the *t*th observation on the *i*th profile (curve) at time t_{ij}, for $i = 1, ..., n$, $j = 1, ..., m$ where n is the number of observations, m is the number of time points, and $t_{ij} = j/m$. The simulated profiles were generated from the following models:

$$Y_{it} = f(t_{ij}) + \sigma\varepsilon_{ij}, \tag{2.7}$$

where ε_{ij} is normal distributed noise. The functions we considered are

$$f_1(t) = \left(\frac{2-5t}{2}\right),$$

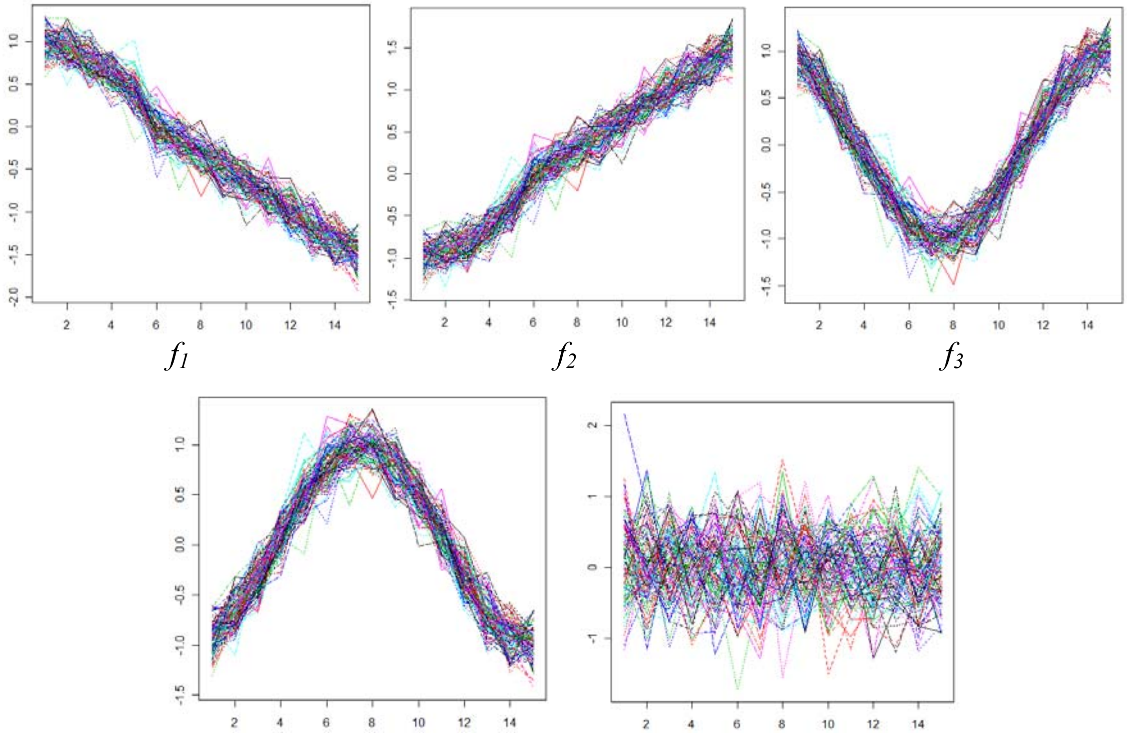
$$f_2(t) = -f_1(t),$$

$$f_3(t) = \cos(2\pi t),$$

$$f_4(t) = -f_3(t),$$

$$f_5(t) = 0\ .$$

We generated 100 profiles from each of five functions. It should be noted that the last function, f_5(t), generates constant profiles. To examine the effect, if any, of the amount of noise, we considered two sets of noise, low (σ = 0.5) and high (σ = 1). Figure 2.1 shows five true clusters of the simulated data.



f_1 f_2 f_3

f_4 f_5

Figure 2.1 Five true clusters of simulated data.

At this point, we compared our proposed clustering procedure with the following scenarios:

1. Original dataset (without smoothing and transformation).
2. Dataset with smoothing but without transformation.
3. Dataset with transformation but without smoothing.
4. Dataset with both smoothing and transformation.

The resultant validity measures of *k*-means clustering and LCCA for different scenarios (under a low noise level) are given in Table 2.1 and 2.2. In a comparison of the *k*-means clustering method with LCCA, the former performed significantly better than the latter across all scenarios. Among the different scenarios of smoothing and transformation, clustering analysis after both smoothing and transformation yielded better results than in the other cases, demonstrating the efficacy of the smoothing and transformation steps.

Table 2.1 Clustering validity measures of low-noise simulated data

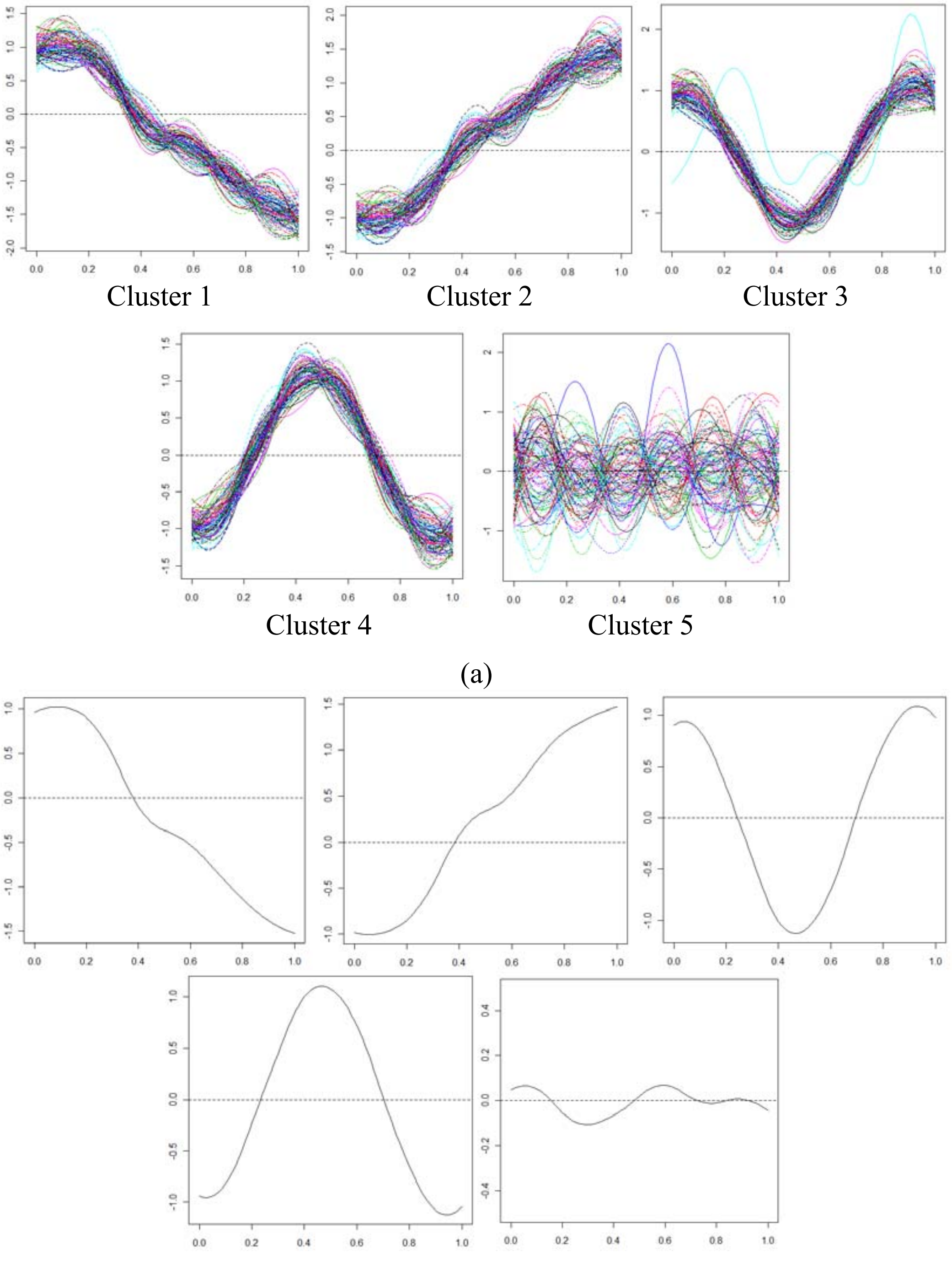
Method	Measurement	Actual	Smoothing	Transformation	Smoothed and Transformed
k-means	Misclustering Rate	0.132	0.150	0.004	0.002
	Silhouette Value	0.701	0.754	0.797	0.826
LCC	Misclustering Rate	0.400	0. 228	0.400	0.400
	Silhouette Value	0.289	0.283	0.438	0.465

Details of the clustering results from the proposed approach (i.e., clustering after smoothing and transformation) are displayed in Figure 2.2. These results show that the patterns of each of the five clusters are very similar to the patterns of the true clusters shown in Figure 2.1; this further demonstrates the usefulness of smoothing and transformation procedures to cluster the functional data. It should be noted that the proposed clustering procedure can properly isolate constant curves as a separate group.

We conducted the same experiment with the simulated data and a high noise level. The results also indicated that the *k*-means clustering analysis with smoothing and transformation produced the lowest misclustering rate and the highest silhouette values. Further, the results of the silhouette value showed that the *k*-means clustering method outperformed LCCA across all scenarios.

Table 2.2 Clustering validity measures of the high-noise simulated data

Method	Measurement	Actual	Smoothing	Transformation	Smoothed and Transformed
k-means	Misclustering Rate	0.282	0.296	0.386	0.116
	Silhouette Value	0.190	0.315	0.247	0.263
LCCA	Misclustering Rate	0.274	0.306	0.296	0.296
	Silhouette Value	0.012	-0.014	-0.067	-0.052



Cluster 1 Cluster 2 Cluster 3

Cluster 4 Cluster 5

(a)

(b)

Figure 2.2 Graphical representation of (a) *k*-means clustering results on simulated data after smoothing and transformation and (b) mean response profiles of each of five clusters.

2.4 Spinal Cord Dorsal Horn Responses to Graded Heat Stimuli

2.4.1 Data Description

We applied the proposed clustering approach to our previous set of data (Borzan et al. 2005), which contained the responses from 12 male rates to thermal stimuli applied to their deep spinal cord dorsal horn neurons. The response to thermal stimuli was measured by the number of action potentials per second minus background activity. This experiment involved eight different thermal stimuli that varied from 37° to 51°C in increments of 2 degrees C (i.e., 37, 39, 41, 43, 45, 47, 49, 51). By using a single cell recording, the responses were characterized for 147 neurons. Among these 147 neurons, we detected three neurons that had zero response to all graded temperatures; therefore, we decided to remove these neurons. Figure 2.3 shows the response profiles of 144 neurons to the gradients of eight thermal stimuli.

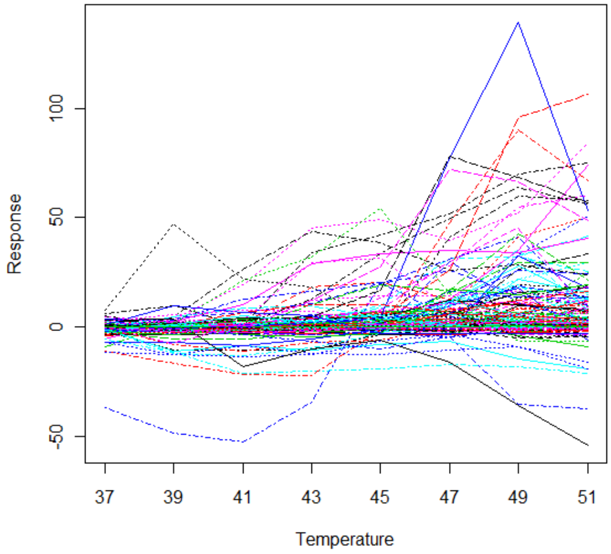


Figure 2.3 144 Neuronal response profiles to gradient thermal stimulation.

2.4.2 Smoothing and Transformation

We applied LOESS to our heat stimulation data. To determine the LOESS smoothing parameters (span), we plotted mean response profiles with different smoothing parameters (Figure 2.4). It can be seen that a span of 0.5 is well represented the fluctuation of our experimental data. As a consequence, we used a span of 0.5 in LOESS.

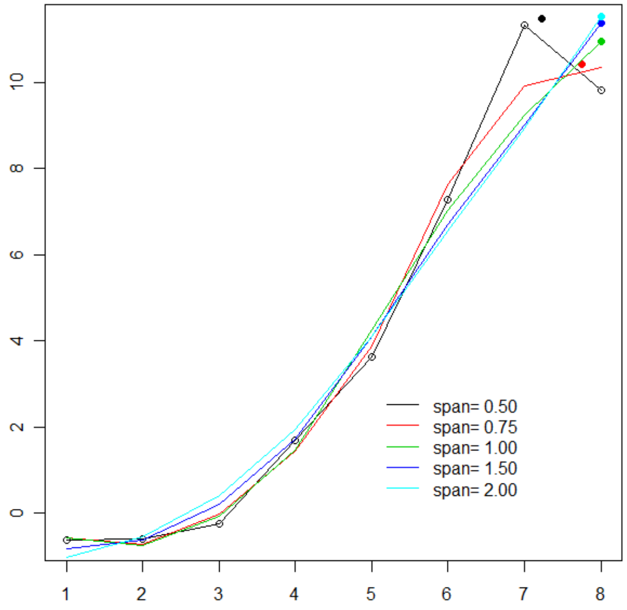
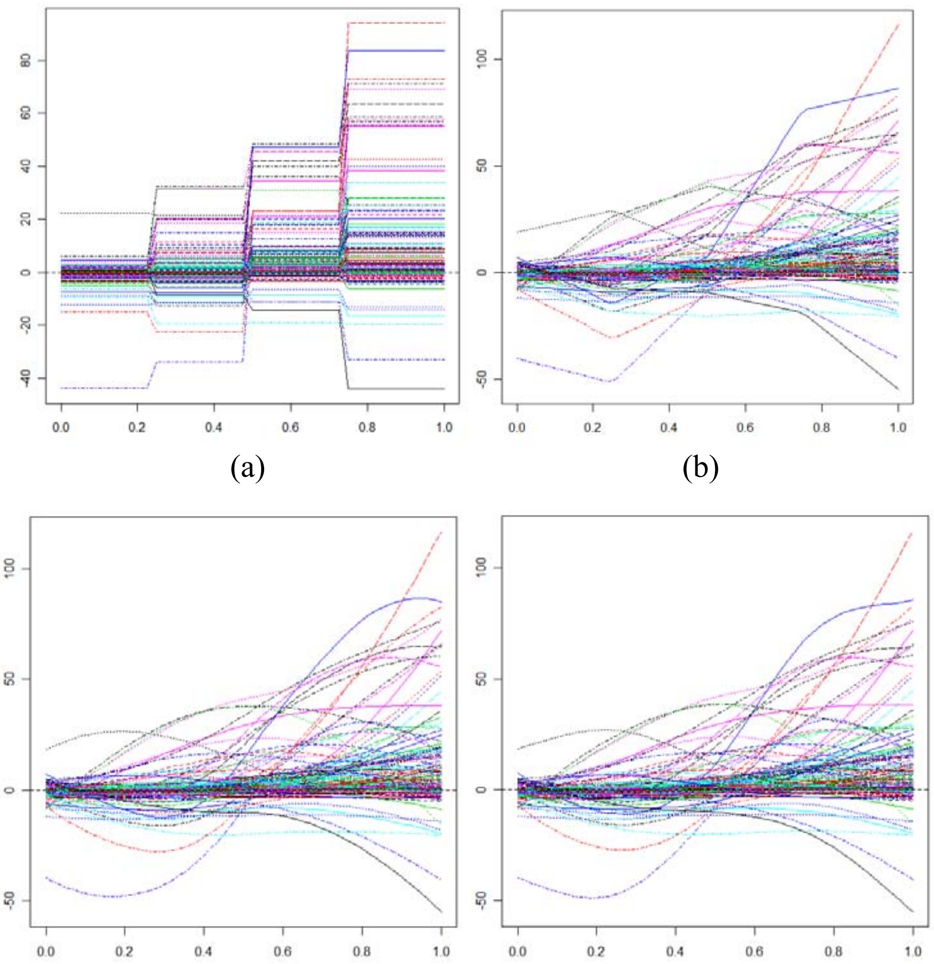


Figure 2.4 LOESS in a mean response profile with different smoothing parameters.

The smoothed data obtained from LOESS were transformed into functional form based on the B-spline function. The B-spline basis functions for different orders are shown in Figure 2.5, demonstrating that an order of three will represent the data well. Note the smoothing results between order = 3 and order = 4 are almost the same.



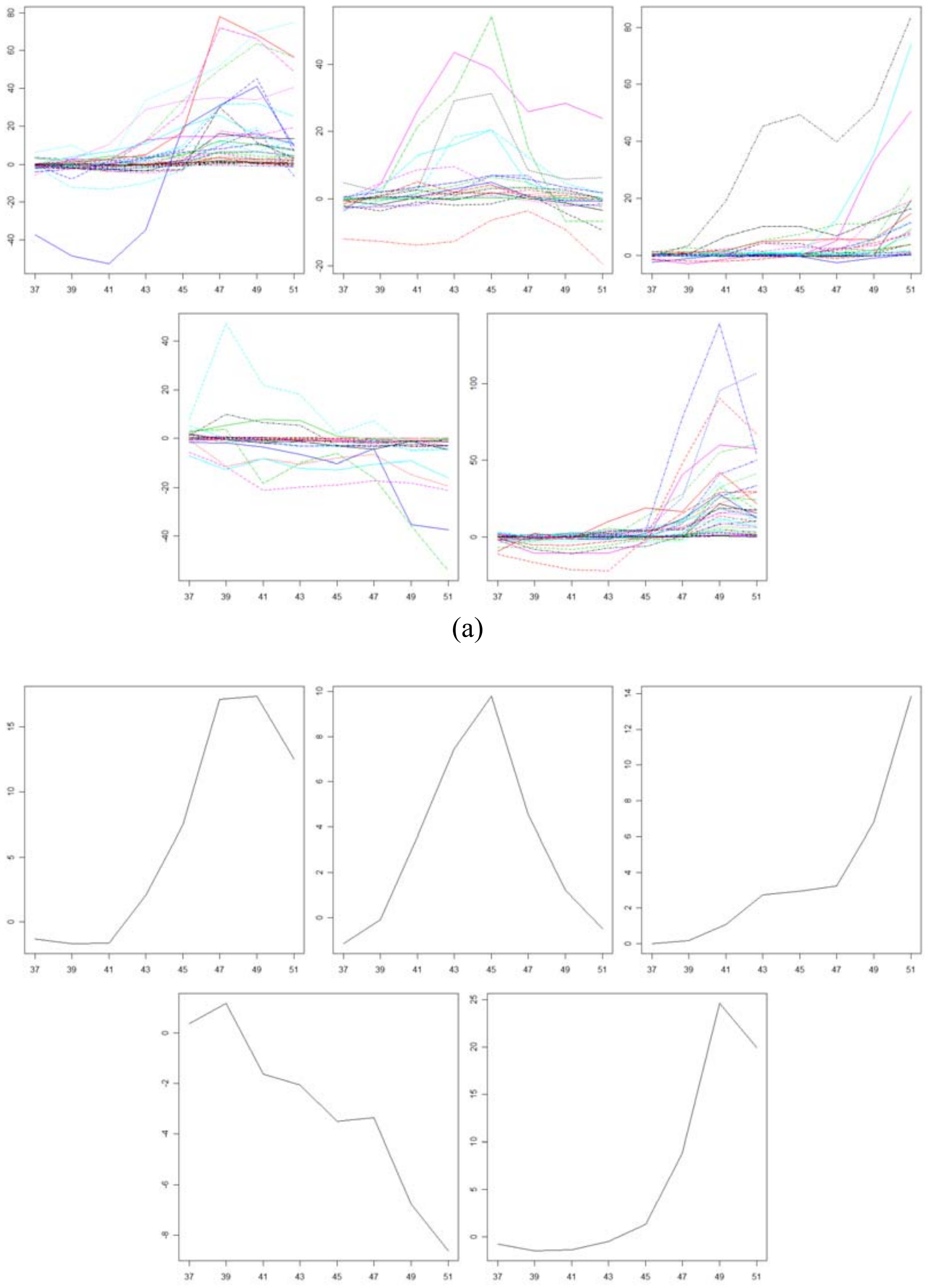
(a) (b)

(c) (d)

Figure 2.5 B-spline basis function plots with (a) order = 1, (b) order = 2, (c) order = 3, and (d) order = 4.

2.4.3 Functional Clustering Results

First, *k*-means clustering analysis with a correlation distance was conducted on all the original 144 profiles. We used *k*=5 as was done in the previous study by Borzan et al. (2005). Figure 2.6 shows the *k*-means clustering results of the original data and illustrates five distinct groups of neurons in which each group has a specific neuronal response profile of 34, 21, 29, 21, or 39.



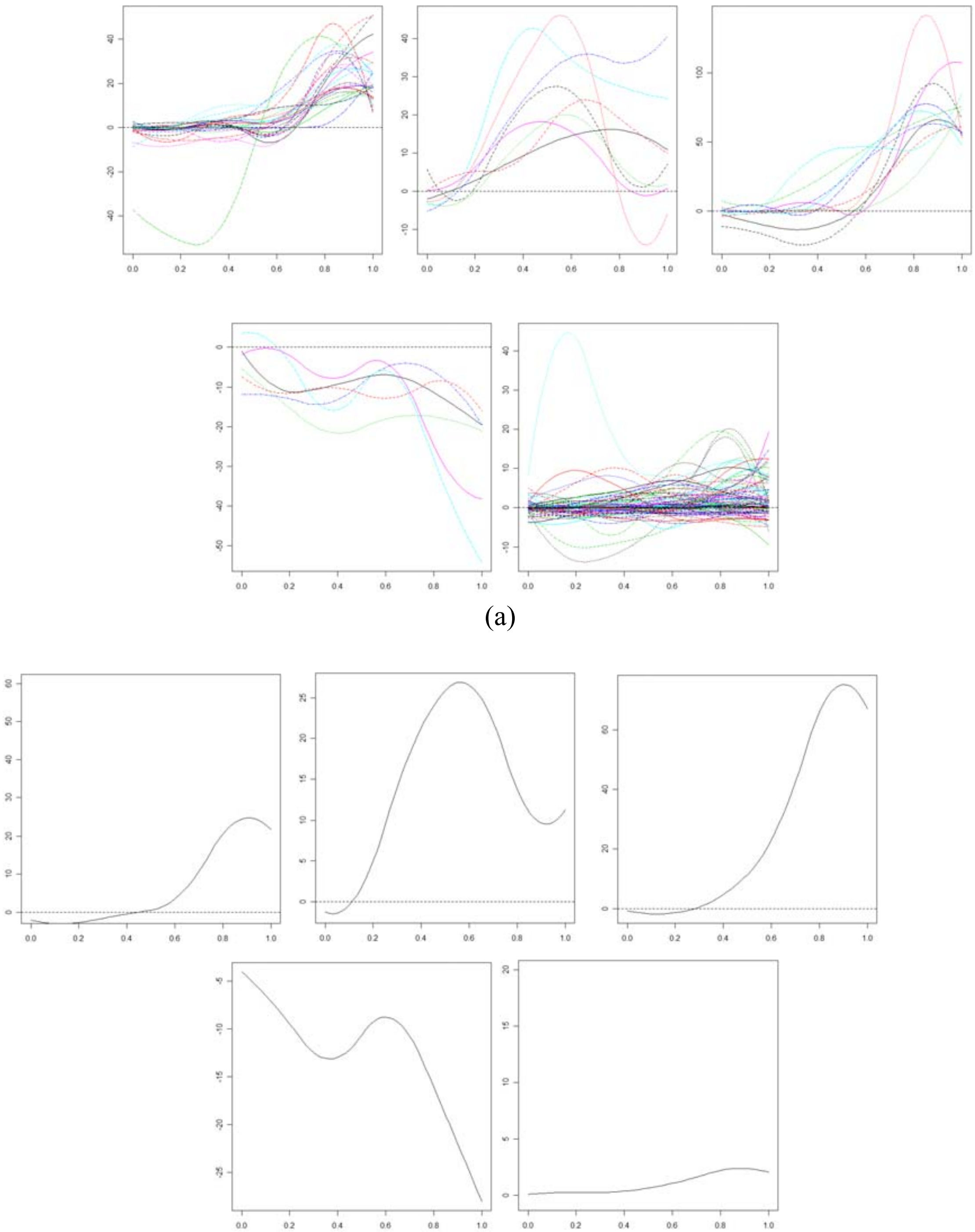
(a)

(b)

Figure 2.6 Graphical representation of (a) *k*-means clustering results on the original dorsal horn neuron data and (b) mean response profiles of each of five clusters.

The mean response profiles of each of the five clusters show the clear patterns of neuronal response profiles to graded thermal stimuli (Figure 2.6 (b)). The response patterns in Clusters 1, 3, and 5 showed an upward pattern, indicating that the responses of the neurons in these clusters increase as the heat stimuli increase. However, subtle differences were observed in neuron patterns among Clusters 1, 3, and 5. The neurons in Cluster 1 drop at about 47° C, while the neurons in Cluster 5 drop at about 49° C. The neurons in Cluster 4 monotonically decrease as the heat stimuli increase. The neurons in Cluster 2 show a pattern that looks like a Mexican hat (resembles a sombrero). The responses of the neurons gradually increase as the heat increases but decrease after their maximum responses at about 45° C. As for the neurons in Cluster 4, their response pattern shows a downward trend, indicating that these neurons respond inversely to temperature gradients. It should be noted that the *k*-means clustering analysis with the original data cannot successfully identify a group that contains a number of constant profiles. Instead, these constant profiles were distributed across all five clusters.

Next, the proposed clustering approach (clustering after smoothing and transformation) was applied to this dorsal horn neuron dataset. Figure 2.7 shows the *k*-means clustering results after smoothing and transformation. The number of neurons in Clusters 1, 2, 3, 4, and 5 is, respectively, 26, 8, 11, 6, and 93. The last group contains a set of constant profiles, although we can obviously see the one neuron that reacts only at a lower temperature.



(a)

(b)

Figure 2.7 Graphical representation of (a) *k*-means clustering results on the experimental data after smoothing and transformation and (b) mean response profiles for each of five clusters.

The neurons in Clusters 1, 2, and 3 show an upward trend in response to graded heat stimuli. The pattern in Clusters 1 and 3 looks similar, but the overall intensity of the neurons that belong to Cluster 1 is less than that of Cluster 3. The neurons in Cluster 2 react maximally to the heat stimuli at about 45° C and then decrease. The neurons in Cluster 4 inversely respond as heat stimuli increase. Note that the proposed clustering successfully separates the constant profiles.

Table 2.3 shows the silhouette values from the *k*-means and LCCA over four different scenarios. Here the misclustering rates were not reported because the information about the true clusters is unknown in real data. The *k*-means clustering method yielded larger silhouette values compared with LCCA. This justifies the ranking of the *k*-means clustering method as more appropriate than LCCA in functional data in which it is difficult to specify the parametric distribution of the data. Overall, the *k*-means clustering method after smoothing and transformation yielded the highest silhouette value, showing the usefulness and effectiveness of the proposed clustering procedure.

Table 2.3 Clustering validity measures (Silhouette Value) for the spinal dorsal horn data from *k*-means clustering and LCCA.

Silhouette Value	Actual	Smoothing Only	Transformation Only	Smoothed and Transformed
k-means clustering	0.385	0.560	0.484	0.660
LCCA	-0.170	-0.288	0.432	0.390

2.5 Conclusions

We have proposed an efficient clustering procedure to categorize a number of profiles that are formed with nonlinear functions. The basic idea of the proposed procedures is to cluster the profiles after smoothing and transformation. We performed simulation studies to examine the properties of the proposed procedure and demonstrate its effectiveness. Based on clustering validity measures, clustering analysis after smoothing and transformation outperformed clustering analysis without smoothing and transformation. Furthermore, the proposed clustering procedure outperformed LCCA, a traditional model-based clustering method. We have applied the proposed clustering procedure to real data in which the main objective is to characterize the response patterns of deep dorsal horn neurons to thermal stimulation gradients. Five distinct patterns for different dorsal neurons were found. In a comparison of the proposed clustering procedure with LCCA, the proposed procedure produced better clustering performance than LCCA.

CHAPTER 3

UNSUPERVISED FEATURE SELECTION USING WEIGHTED PRINCIPAL COMPONENTS

3.1 Introduction

One of the major challenges associated with high-dimensional data is to identify a subset of relevant features of interest. In recent years, feature selection/extraction has received considerable attention in various areas for which datasets with thousands of features are present. The main purpose of feature selection/extraction is to identify a subset of features that are most predictive or informative in a given a dataset. Successful implementation of feature selection/extraction simplifies the entire modeling process and thus reduces computational and analytical efforts.

It is important to distinguish between feature selection and feature extraction, although much of the literature fails to clearly distinguish between them (Jain et al., 2000). Feature selection is a process to select a subset of original features, and feature extraction creates new features through the transformation of the original features (Guyon and Elisseeff, 2003). Widely used feature extraction methods include principal component analysis (PCA) and partial least squares (PLS). PCA is an unsupervised feature extraction method in that the process depends solely upon the input variables, and does not take into account information from the output variable (Jolliffe, 2002). On the other hand, PLS is a supervised feature extraction in that the process takes into account both the input and output variables (Kim, 2008). In general, the first few transformed features obtained from PCA and PLS suffice to provide useful information in the original data. However, because these reduced dimensions from PCA and PLS are linear combinations of a large number of original features, their interpretation cannot be readily made and the extraction of meaningful information is cumbersome.

Interpretation problems posed by the transformation process in PCA and PLS can be overcome by using feature selection methods that simply pick the subset of original features. Feature selection methods can also be divided into supervised and unsupervised. Supervised feature selection methods use the information of an output variable to identify the best subset of given features in a dataset. Genetic algorithms have been successfully used as an efficient method of supervised feature selection for a high-dimensional spectral dataset (Cho et al., 2008; Davis et al., 2003). Moreover, supervised feature selection problems have been formulated by a multiple hypothesis testing procedure that controls the false discovery rate (Mei et al., 2009; Kim et al., 2008).

Despite extensive research in using the supervised/unsupervised feature extraction and supervised feature selection, relatively few attempts have been made to identify the important features by using unsupervised feature selection methods (Mao, 2005). Unsupervised feature selection methods usually have been divided into three categories — wrapper, filter, and hybrid approaches (Kim and Gao, 2006). The filter approach employs the general characteristics of the data to select a subset of the original data without using any clustering algorithms. In contrast, the wrapper approach necessitates the use of a predetermined clustering algorithm as evaluation criterion. The hybrid approach combines both the filter and wrapper approaches by using different evaluation criteria for each different state (Kim and Gao, 2006).

Dy and Brodley (2000) introduced a wrapper approach that uses an expectation-maximization (EM) clustering algorithm. Hastie et al. (2000) developed a gene-shaving method that used its first principal component to identify the best subsets of those features with large variations. Ding (2003) proposed a two-way ordering approach in which relevant genes were selected based on their similarity information.

Mao (2005) proposed a filter approach that sought to select a subset of original features by using principal components combined with an evaluation based on least square estimation (LSE). Motivated by Mao's idea, Kim and Gao (2006) developed a two-step hybrid approach. The first step is to subsets of features based on an LSE-based evaluation; the second is to apply a searching algorithm to obtain the best subsets that maximize clustering performance.

Although all of the existing unsupervised feature selection methods performed reasonably well within the limits of the situations for which they were designed, no consensus exists about which of them best satisfies all conditions. Moreover, most of the methods require a high computational load because they involve an extensive search procedure such as the forward selection or the backward elimination. Consequently, the methods based on a search algorithm are not relevant for identifying important features in high-dimensional dataset, often encountered in various applications in these days. In the present study, we propose a new unsupervised feature selection that combines the weighted principal components with a thresholding algorithm. To be specific, the contribution of each feature is represented by a loading value in a weighted principal component, and a thresholding algorithm based on a moving range-based control chart evaluates the significance of its contribution. The proposed method belongs to the filter category and is computationally efficient and easy to implement.

The remainder of this chapter is organized as follows. Section 3.2 presents the proposed unsupervised feature selection method. Section 3.3 presents the simulation study that examined the performance of the proposed method under various scenarios. Section 3.4 describes a case study developed to demonstrate the feasibility and effectiveness of the proposed method in real situations. Finally, Section 3.5 presents our concluding remarks.

3.2 The Proposed Unsupervised Feature Selection Approach

3.2.1 Weighted Principal Components

PCA is one of the most widely used multivariate data analysis techniques and is employed primarily for dimensional reduction and visualization (Jolliffe, 2002). PCA extracts a lower dimensional feature set that can explain most of the variability within the original data. The extracted features, PC_i's (Y_i) are each a linear combination of the original features with the loading values (α_{ij}, $i, j=1,2,\ldots,p$). The Y_i's can be represented as follows:

$$
\begin{aligned}
Y_1 &= \alpha_{11}X_1 + \alpha_{12}X_2 + \ldots + \alpha_{1p}X_p \\
Y_2 &= \alpha_{21}X_1 + \alpha_{22}X_2 + \ldots + \alpha_{2p}X_p \\
&\vdots \\
Y_p &= \alpha_{p1}X_1 + \alpha_{p2}X_2 + \ldots + \alpha_{pp}X_p
\end{aligned}
\tag{3.1}
$$

The loading values represent the importance of each feature in the formation of a PC. For example, α_{ij} indicates the degree of importance of the *j*th feature in the i^{th} PC. A two-dimensional loading plot (e.g., PC1 vs PC2 loading plot) may provide a graphical display for identification of important features in the first and second PC domains. However, the interpretation of a two-dimensional loading plot is frequently subjective, particularly in the presence of a large number of features. Moreover, in some situations, consideration of only the first few PCs may be insufficient to account for most of the variability in the data. Determination of the appropriate number of PCs (=*k*) to retain can be subjective. One can use a scree plot that visualizes the proportion of variability of each PC to determine the appropriate number of PCs (Johnson and Wichern, 2002).

If a PCA loading value for the *j*th original feature can be computed from the first *k* PCs, the importance of the *j*th feature can be represented as follows:

$$\omega_j = \sum_{i=1}^{k} \left|\alpha_{ij}\right| \pi_i \ , j=1, 2, \ldots , p, \tag{3.2}$$

where *k* is the total number of features of interest and π_i represents the weight of *i*th PC. The typical way to determine π_i is to compute the proportion of total variance explained by the *i*th PC. ω_j can be called a weighted PC loading for the feature *j*.

For illustration, Figure 3.1 displays a plot of ω_j s, computed from a simulated dataset that contains 1,000 features. A feature with a large value of ω_j indicates a significant feature. In the next section, we will present a systematic way to obtain a threshold that determines the significance of each feature.

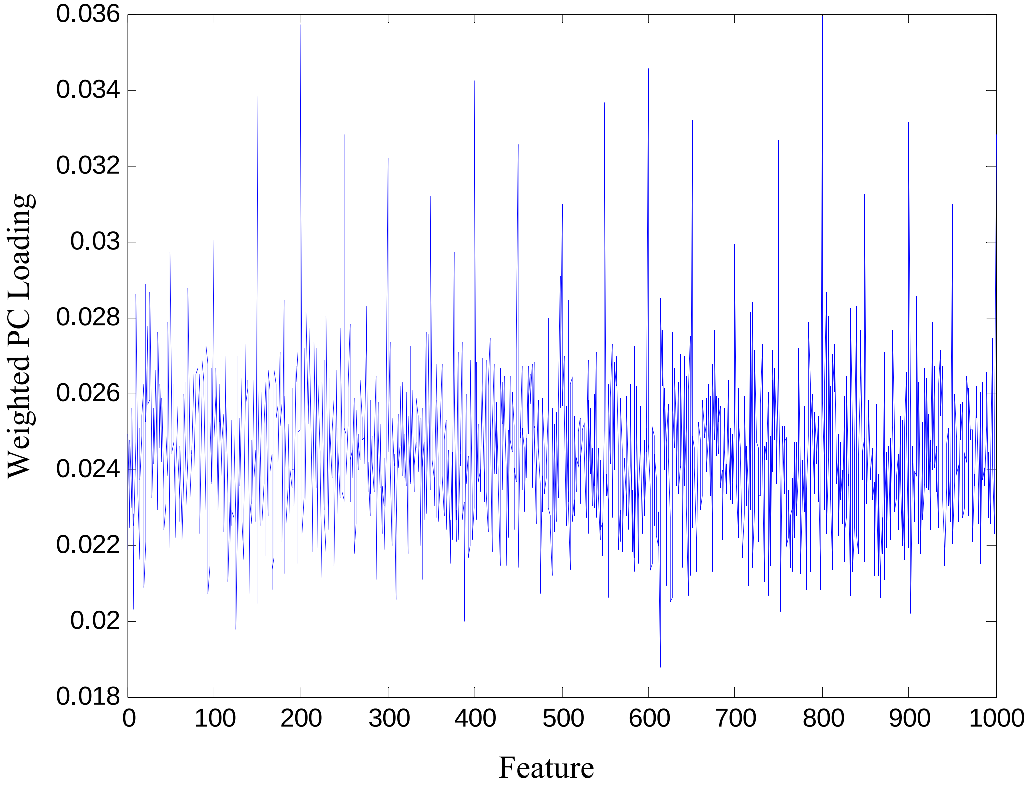


Figure 3.1 Weighted PC loading values of individual features.

3.2.2 Moving Range-Based Thresholding Algorithm

We propose a moving range-based thresholding algorithm as a way to identify the significant features from the weighted PC loadings discussed in the previous section. The main idea of a moving range-based thresholding algorithm comes from a moving average control chart that has been widely used in quality control (Vermaat et al. 2003). A control chart provides a comprehensive graphical display for monitoring the performance of a process over time so as to keep the process within control limits (Woodall and Montgomery, 2001). A typical control chart comprises monitoring statistics and the control limit. When the monitoring statistics exceed (or fall below) the control limit, an alarm is generated so that proper remedial action can be taken. A moving range control chart is useful when the sample size used for process monitoring is one. Moreover, the average moving range control charts perform reasonably well when the observations deviate moderately from the normal distribution (Vermaat et al. 2003).

In our problem, we can consider the weighted PC loading values as the monitoring statistics. Thus, we plot these loading values on the moving range control chart and identify the significant features when the corresponding weighted PC loading exceeds the control limit (threshold). Given a set of the weighted PC loading values for individual features $(\omega_1, \omega_2, ..., \omega_p)$, the threshold (γ) can be calculated as follows (Vermaat et al. 2003):

$$\gamma = \overline{\omega} + \Phi^{-1}(1-\alpha)\frac{\sqrt{\pi}}{2} * \sigma , \tag{3.3}$$

where $\overline{\omega} = \frac{1}{p}\sum_{i=1}^{p}\omega_i$, Φ^{-1} is the inverse standard normal cumulative distribution function, and α is the Type I error rate that can be specified by the user. The range of α is between 0 and 1. In typical moving range control charts, σ can be estimated by $\overline{MR}$, calculated by the average of the moving ranges of two successive observations.

$$\overline{MR} = \frac{|\omega_1 - \omega_2| + |\omega_2 - \omega_3| + \ldots + |\omega_{p-1} - \omega_p|}{p-1} \tag{3.4}$$

However, in our feature selection problems, because the weighted PC loading values for individual features $\omega_1, \omega_2, ..., \omega_p$ are not ordered, we cannot simply use (4). To address this issue, we propose a different way of computing the $\overline{MR}$ that can properly handle a set of unordered observations. Given the fact that there is no specific order of observations $\omega_1, \omega_2, ..., \omega_p$, they are randomly reshuffled, and $\overline{MRs}$ are recalculated. Therefore, for B=1,000, we obtain a set of $\overline{MRs}$ $\overline{MR}_{(1)}, \overline{MR}_{(2)}, ..., \overline{MR}_{(B)}$. The $\overline{MR}$ for unordered observations is calculated by

$$\overline{MR}^* = \frac{1}{B}\sum_{j=1}^{B} \overline{MR}_{(j)}\ , \tag{3.5}$$

Finally, the threshold of the proposed feature selection method can be obtained by the following equation:

$$\gamma = \overline{\omega} + \Phi^{-1}(1-\alpha)\frac{\sqrt{\pi}}{2}\overline{MR}^*. \tag{3.6}$$

A feature is reported as significant if the corresponding weighted PC loading exceeds the threshold γ.

3.2.3 Feature Validity Measures

We used the sensitivity and specificity as performance measures (Altman and Bland, 1994). Sensitivity and specificity can be expressed as follows:

$$\text{Sensitivity} = \frac{TP}{TP + FN}, \tag{3.6}$$

$$\text{Specificity} = \frac{TN}{TN + FP}, \tag{3.7}$$

where *TP* is the number of true positives (number of true significant features identified), *TN* is the number of true negatives (number of true insignificant features identified), *FN* is the number of false negatives, and *FP* is the number of false positives. In short, sensitivity is the proportion of true positives correctly identified by the procedure. Specificity is the proportion of true negatives correctly identified. The range of both sensitivity and specificity is between 0 and 1. The method that produces the largest sensitivity and specificity scores would be considered the better method.

3.3 Simulation Study

3.3.1 Simulated Data

A simulation study evaluated the performance of the proposed method and compared it with other algorithms under various scenarios. Table 3.1 shows a summary of the simulated data used in this study.

Table 3.1 Summary of the simulated data

Data	Number of features	Number of observations	Number of classes	Number of true significant features	Mean shift
Scenario 1	500	200	2	10	5σ
Scenario 2	100	200	2	10	3σ
Scenario 3	1000	200	2	10	1σ
Scenario 4	1000	400	4	100	1σ, 2σ, 3σ
Scenario 5	1000	400	4	20	5σ, 10σ, 20σ
Scenario 6	1000	400	4	20	0.5σ, 1σ, 2σ
Scenario 7	3000	200	2	100	0.5σ
Scenario 8	3000	200	2	20	2σ
Scenario 9	3000	200	2	300	1σ, 2σ, 3σ
Scenario 10	7000	200	2	300	1σ, 2σ, 3σ

Each scenario contains the number of observations, the number classes, the number of true significant features, and different degrees of shifts in the mean. Specifically, the simulated data in Scenarios 1 ~ 3 contain two class datasets in which the covariance matrix of each class is the identity matrix ($\Sigma_1 = \Sigma_2 = I$). The mean of Class 1 equals zero, and the mean of Class 2 equals the mean of Class 1 plus the shift in mean as shown in the last column of Table 3.1. Other scenarios can be explained similarly.

3.2 Simulation Results

Table 3.2 presents the number of identified features, sensitivity, specificity, and computational time (CPU time) in the 10 simulation scenarios. The experiments were conducted on an Intel® Core™2 Duo @ 2.2 GHz computer with 2 GB memory. We compared the proposed weighted PC loading method with the LSE method (Mao, 2005), one of the existing unsupervised feature selection methods. In the LSE method, a subset of significant features was selected based on error reduction after adding additional features. The error reduction function was calculated based on PCs that were obtained from the PCA in the complete data. A sequential forward selection strategy was then used to determine a subset of significant features (Mao, 2005).

The results showed that across all simulation scenarios, the sensitivities and specificities of the proposed method were all one, implying that our method successfully detected all the true significant features. The LSE method also yielded sensitivity and specificity results comparable with the proposed method. However, the LSE method tended to identify more numbers of features than the number of true significant features. More important, the LSE method imposes a high computational load compared with the proposed method. In particular, faced with more than 3,000 features, the LSE method takes a significant amount of time to identify the significant ones.

Table 3.2 Number of identified features, sensitivity (Se), specificity (Sp), and CPU time on 10 scenarios

Scenario	Method	# of true significant features	# of identified features	Se	Sp	CPU Time (Sec.)
1	LSE	10	11	1.000	0.998	36.02
	WPC + MR (α = 0.01)		10	1.000	1.000	4.38
	WPC + MR (α = 0.10)		10	1.000	1.000	4.38
2	LSE	10	11	1.000	0.998	7.62
	WPC + MR (α = 0.01)		10	1.000	1.000	1.49
	WPC + MR (α = 0.10)		10	1.000	1.000	1.49
3	LSE	10	11	1.000	0.998	65.16
	WPC + MR (α = 0.01)		10	1.000	1.000	10.33
	WPC + MR (α = 0.10)		19	1.000	0.991	10.33
4	LSE	100	94	0.940	1.000	1663
	WPC + MR (α = 0.01)		100	1.000	1.000	10.73
	WPC + MR (α = 0.10)		100	1.000	1.000	10.73
5	LSE	20	21	1.000	0.998	185.35
	WPC + MR (α = 0.01)		20	1.000	1.000	10.91
	WPC + MR (α = 0.10)		20	1.000	1.000	10.91
6	LSE	20	21	1.000	0.998	176.89
	WPC + MR (α = 0.01)		20	1.000	1.000	10.97
	WPC + MR (α = 0.10)		20	1.000	1.000	10.97
7	LSE	100	94	0.940	1.000	3830
	WPC + MR (α = 0.01)		100	1.000	1.000	64.82
	WPC + MR (α = 0.10)		100	1.000	1.000	64.82
8	LSE	20	20	1.000	1.000	391.23
	WPC + MR (α = 0.01)		20	1.000	1.000	59.59
	WPC + MR (α = 0.10)		47	1.000	0.991	59.59
9	LSE	300	287	0.957	1.000	47882
	WPC + MR (α = 0.01)		300	1.000	1.000	60.85
	WPC + MR (α = 0.10)		300	1.000	1.000	60.85
10	LSE	300	378	1.000	0.988	215810
	WPC + MR @ α = 0.01		300	1.000	1.000	275.42
	WPC + MR @ α = 0.10		300	1.000	1.000	275.42

3.4 Experiments with Real Data

In addition to the simulation study, we used three real datasets (Wisconsin diagnostic breast cancer, wine, and leukemia microarray) to demonstrate the effectiveness of the proposed weighted PC loading method. These datasets are available on the UCI database (http://archive.ics.uci.edu/ml/), and their summary is shown in Table 3.3.

Table 3.3 Summary of real datasets

Data	Number of features	Number of observations	Number of classes
Wisconsin diagnostic breast cancer	30	569	2
Wine	13	178	3
Leukemia	7129	72	2

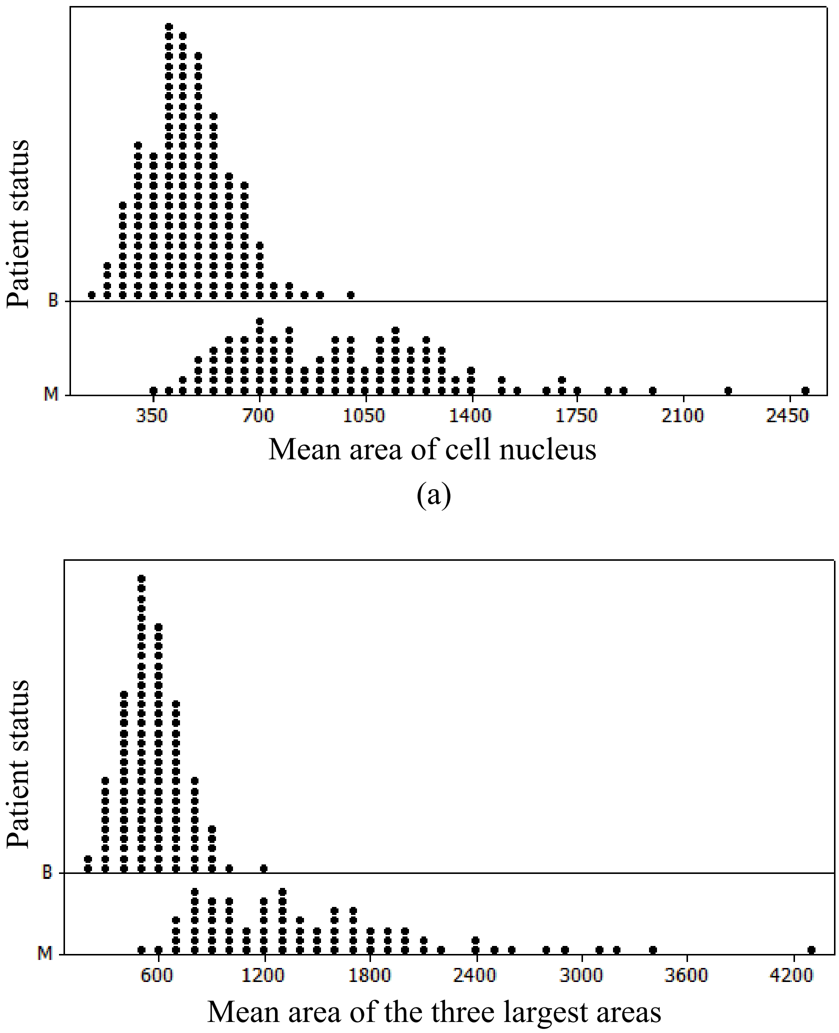
We evaluated the performance of the proposed method and compared it with the Baseline Case and the LSE method. The Baseline Case represents the use of all features for comparison. Table 3.4 shows feature selection results, classification accuracy derived from a classification algorithm, and CPU time on the real datasets. Classification accuracy is defined as the number of observations correctly classified divided by the total number of observations. To compute classification accuracy, we used a support vector machines (SVM) algorithm, one of the most widely used classification methods (Shawe-Taylor and Cristianini, 2000). The SVM classification accuracy reported here is the average value ± standard deviation from 10-fold cross validation. Note that specificity and sensitivity were not reported here; their omission is because information about the true clusters is unknown in real data. Moreover, we did not report CPU time for the Baseline Case because this case does not involve any feature selection process but instead uses all of the features.

Table 3.4 Comparison of unsupervised feature selection methods on the Wisconsin diagnostic breast cancer, wine, and leukemia datasets

Data	Method	No. of Identified Features	SVM Classification Accuracy (%)	CPU Time (Second)
Wisconsin Diagnostic Breast Cancer	Baseline	30	97.37 ± 2.89	-
	LSE	12	95.59 ± 3.39	2.226
	WPC + MR (α = 0.01)	2	92.27 ± 1.47	1.979
	WPC + MR (α = 0.05)	2	92.27 ± 1.47	1.979
Wine	Baseline	13	97.19 ± 4.72	-
	LSE	3	98.86 ± 2.41	1.363
	WPC + MR (α = 0.01)	1	93.79 ± 2.35	0.717
	WPC + MR (α = 0.05)	1	93.79 ± 2.35	0.717
Leukemia	Baseline	7129	88.68 ± 1.59	-
	LSE	-	-	-
	WPC + MR (α = 0.01)	384	87.03 ± 1.34	274.830
	WPC + MR (α = 0.05)	457	90.29 ± 1.15	274.830

In the Wisconsin breast cancer data, our proposed method identified the smallest number of significant features but produced classification accuracy comparable to the Baseline Case and the LSE method. In order to explore more about this outcome, Figure 3.2 shows the dot plots of two features identified by our proposed method according to the status of patient (malignant, benign). These two features are the mean area of the cell nucleus and the mean of the three largest area values. These features clearly distinguished between benign and malignant samples. Further, we also generated dot plots of two features that the LSE method identified but the proposed method did not (Figure 3.3). These features are the mean of texture and standard error of perimeter. It can be seen that these feature could not clearly differentiate between benign and malignant samples.

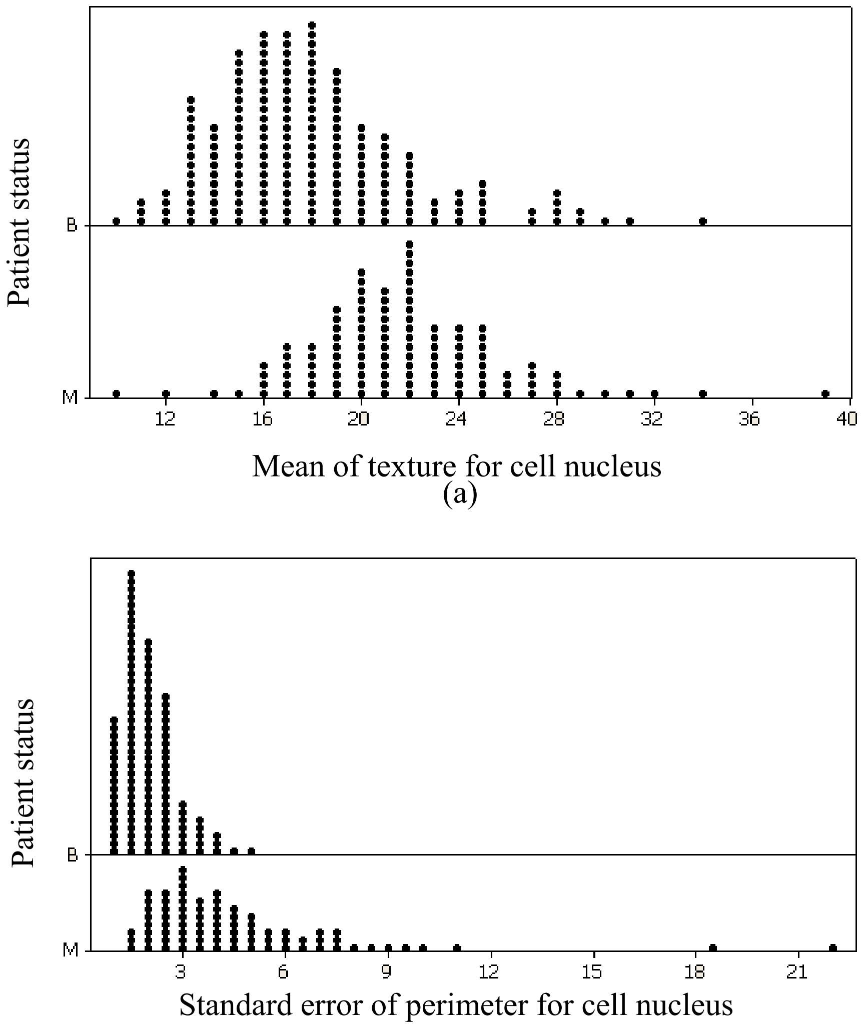
In the wine data, the proposed weighted PC loading method algorithm identified only one significant (proline) feature out of 13. This one-feature result of SVM classification is not significantly worse in terms of accuracy than the three-feature performance of the LSE method (Table 3.4). Figure 3.4 displays the dot plot of the "proline" feature by the type of wine. A clear distinction can be observed between the first and the second and third types. However, this proline may not be a good feature for distinguishing between the second and third types of wine. Figure 3.5 shows a dot plot of the feature (alkalinity of ash) identified by the LSE method but not by the weighted PCA loading method. There is an overlapping among the samples, indicating that the feature, alkalinity of ash may not play a significant role in discriminating the type of wine.



(a)

(b)

Figure 3.2 Dot plots of two significant features for Wisconsin diagnostic breast cancer data. (a) mean area of cell nucleus, (b) mean of the three largest areas feature. The features were identified by both the proposed method and the LSE method according to patient types (malignant, benign).



(a)

(b)

Figure 3.3 Dot plots of two significant features for Wisconsin diagnostic breast cancer data. (a) mean of texture for cell nucleus, (b) standard error of perimeter feature. The features were identified by only the LSE method (not by the proposed method) according to patient types (malignant, benign).

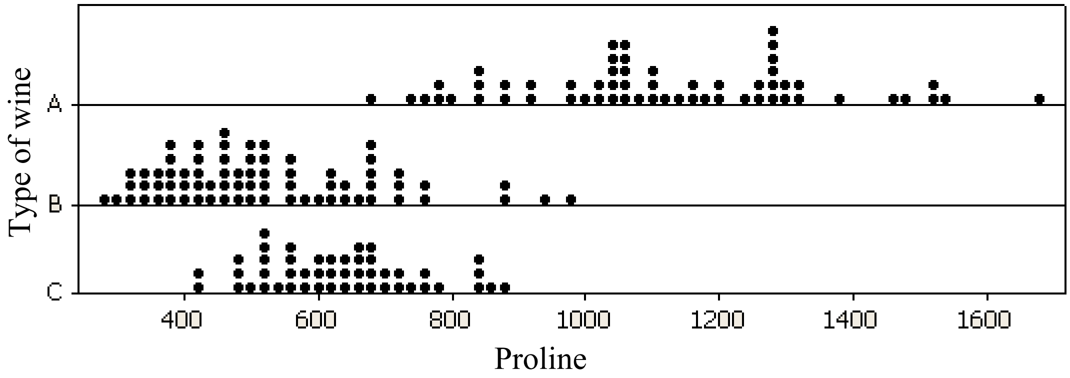


Figure 3.4 A dot plot of the proline feature by the type of wine. The feature was identified by the proposed method.

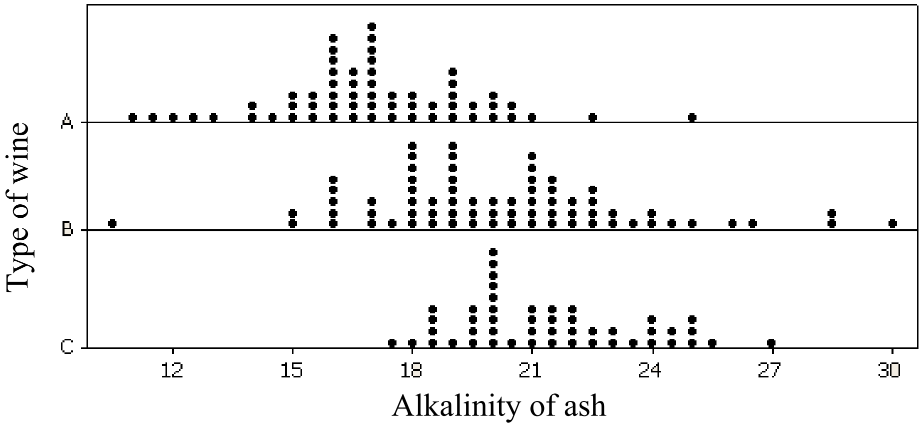
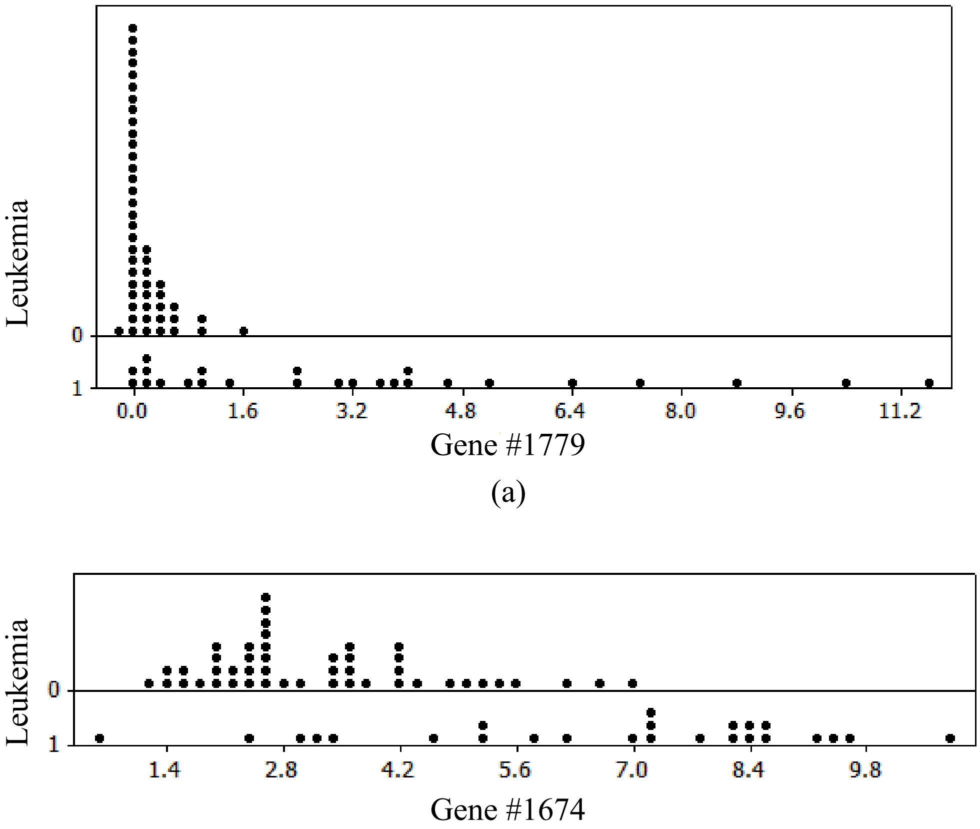


Figure 3.5 A dot plot of the significant feature (alkalinity of ash) identified by only the LSE method (not by the proposed method) according to wine types.

In the microarray leukemia data, our proposed method with $\alpha = 0.05$ identified 457 features as significant out of 7,129 and produced an even better result than the Baseline Case (Table 3.4). The performance of the LSE method is not reported here because it requires a significant amount of time (more than 48 hours), which of itself is enough to disqualify it as a valid competitor with our proposed method.

Figure 3.6 presents dots plots of two features identified as significant from our proposed algorithm by the status of patients.



(a)

(b)

Figure 3.6 Dotplots for leukemia data of (a) gene #1779 and (b) gene #1674 by type of leukemia.

3.5 Conclusions

We have presented a new method of unsupervised feature selection for identification of important features in high-dimensional datasets. The proposed method combines PCA techniques and a moving range-based thresholding algorithm. We first obtained the weighted PC, which can be calculated by the weighted sum of the first *k* PCs of interest. Each of the *k* loading values in the weighted PC reflects the contribution of each individual feature. To identify the significant features, we proposed a moving-range thresholding algorithm. Features are considered to be significant if the corresponding weighted PC loadings exceed the threshold obtained by a moving-range thresholding algorithm. Our experimental results with both simulated and real datasets demonstrated that the proposed method could successfully detect the true significant features. Moreover, compared with LSE, which is one of the existing methods of unsupervised feature selection, the proposed method requires significantly lesser computational loads and thus can efficiently handle high-dimensional datasets.

Our study extends the application scope of both the PCA and control chart techniques. We hope that the procedure discussed here stimulates further investigation into development of better procedures for problems of unsupervised feature selection.

CHAPTER 4

SUMMARY AND FUTURE WORKS

In this dissertation, we presented unsupervised clustering approaches for functional data analysis. Simulation studies under various scenarios indicated that our proposed clustering procedure correctly identified the true clusters and yielded better clustering results than a latent class cluster analysis, one of the existing clustering methods. The proposed clustering strategy can successfully elicit natural grouping of the neurons with similar response patterns to graded thermal stimuli. In chapter 3, we proposed an unsupervised feature selection method that combines weighted principal components (PCs) with thresholding algorithm. Our experimental results with both the simulated and real datasets demonstrated that the proposed method could successfully detect the true significant features.

To further demonstrate our clustering strategy for neurostimulation data, we would like to extend our research by including other datasets such as NMR/NIR spectra. The comparison study with other performance measures should be considered.

Based on our previous work in Chapter 3 "Unsupervised Feature Selection using Weighted Principal Component", the purpose of our study is to propose a procedure for identifying a set of significant features. In the present study, all variables in simulated data and real data are continuous variables. Although, performance of our model was quite good, the result might not be hold when there is a mixture of categorical variables. To further the development of this proposed method, we would like to extend our research by using different data including categorical variables.

REFERENCES

1. Altman D.G. and Bland J.M. (1994), "Diagnostic tests. 1: Sensitivity and specificity," BMJ, 308, 1552.

2. Borzan J., LaGraize, S.C., Hawkins, D.L., and Peng, Y.B., (2005), “Dorsal horn neuron response patterns to graded heat stimuli in the rat,” Brain Research, 1045(1-2), 72–9.

3. Burchiel, K.J., et al., (1996), “Prospective, multicenter study of spinal cord stimulation for relief of chronic back and extremity pain,” Spine, 21(23), 2786–2794.

4. Cadima, J. F. and Jolliffe, I. T. (1995), “Loadings and correlations in the interpretation of principal components,” Journal of Applied Statistics, 22(2), 203–214.

5. Cadima, J.F. and Jolliffe, I. T. (2001), “Variable Selection and the Interpretation of Principal Subspaces,” Journal of Agricultural, Biological, and Environmental Statistics, 6(1), 62–79.

6. Cho, H.-W., Kim, S.B., Jeong, M., Park, Y., Ziegler, T. R., and Jones, D. P. (2008), “Genetic algorithm-based feature selection in high-resolution NMR spectra,” Expert Systems with Applications, 35, 967–975.

7. Chung, J.M., Surmeier, D.J., Lee, K.H., Sorkin, L.S., Honda, C.N., Tsong, Y., and Willis, W.D., (1986), “Classification of primate spinothalamic and somatosensory thalamic neurons based on cluster analysis,” Journal of Neurophysiology, 56 (2), 308– 327.

8. Cleveland, W.S. (1979), “Robust Locally Weighted Regression and Smoothing Scatterplots,” Journal of the American Statistical Association, 74(368), 829–836.

9. Cleveland, W.S. and Devlin, S.J. (1988), "Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting," Journal of the American Statistical Association, 83(403), 596–610.

10. Craig, A. D., Krout, K., and Andrew, D. (2001), “Quantitative response characteristics of thermoreceptive and nociceptive lamina I spinothalamic neurons in the cat,” Journal of Neurophysiology, 86(3), 1459–1480.

11. Dash, M. and Liu, H. (1997), "Feature Selection for Classification," Intelligent Data Analysis: An International Journal, 1(3), 131–156.

12. Dash, M., Liu, H., and Yao, J. (1997), "Dimensionality Reduction of Unsupervised Data," Proceedings Ninth IEEE International Conference on Tools with AI (ICTAI '97), 532–539.

13. Davis, R.A., Charlton, A.J., Oehlschlager, S. and Wilson, J.C. (2006), "Novel feature selection method for genetic programming using metabolomic H1 NMR data," Chemometrics and Intelligent Laboratory Systems, 81, 50–59.

14. Ding, C. (2003), "Unsupervised feature selection via two-way ordering in gene expression analysis," Bioinformatics, 19(10), 1259–1266.

15. Dy, J. and Brodley, C. (2000), "Feature subset selection and order identification for unsupervised learning," Proceedings of the 17th International Conference on Machine Learning, 247–254.

16. Guyon, I. and Elisseeff, A. (2003), "An introduction to variable and feature selection," Journal of Machine Learning Research, 3, 1157–1182.

17. Handl, J. and Knowles, J. (2006), "Feature subset selection in unsupervised learning via multiobjective optimization," International Journal on Computational Intelligence Research, 2(3), 217–238.

18. Hastie, T., Tibshirani, R., Eisen, M., Alizadeh, A., Levy, R., Staudt, L., Chan, W., Botstein, D., and Brown, P., (2000) "'gene shaving' as a method for identifying distinct sets of genes with similar expression patterns," Genome Biology, 1(2), 0003.1– 0003.21.

19. Hastie, T., Tibshirani, R., and Friedman, J., (2001), The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, New York.

20. Hayes, N.L. and Rustioni, A., (1981), "Descending projections from brainstem and sensorimotor cortex to spinal enlargements in the cat," Experimental Brain Research, 41(2), 89–107.

21. Jain, A.K., Duin, R. P.W., and Mao J. (2000), "Statistical pattern recognition: a review," IEEE Transactions on Pattern Analysis & Machine Intelligence, 20, 4–37.

22. Jansen, J. J., Hoefsloot, H. C. J., Boelens, H. F. M., Greef, J. V. D., and Smilde, A. K. (2004) "Analysis of longitudinal metabolomics data," Bioinformatics, 20, 2438–2446.

23. Johnson, R. A. and Wichern, D. W. (2001) Applied multivariate statistical analysis, 5th Edition, Prentice-Hall, Inc., New Jersey.

24. Jolliffe, I. T. (2002) Principal Component Analysis, Springer-Verlag, New York.

25. Kanui, T.I., (1985), "Thermal inhibition of nociceptor-driven spinal cord neurones in the rat," Pain, 21(3), 231–240.

26. Kaufman, L. and Rousseeuw, P.J., (1990), Finding Groups in Data: An Introduction to Cluster Analysis, Wiley, New York.

27. Kim, S.B. (2008), "Features extraction and selection in high-dimensional spectral data," Encyclopedia of Data Warehousing and Mining (2nd Edition), J. Wang editor, IGI Global, Pennsylvania, 863–869.

28. Kim, S.B., Chen, V. C. P., Park, Y., Ziegler, T. R., and Jones, D. P. (2008) "Controlling the false discovery rate for features selection in high-resolution NMR spectra," Statistical Analysis and Data Mining, 1, 57–66.

29. Kim, Y. and Gao, J. (2006) "Unsupervised gene selection for high dimensional data," Proceedings of IEEE Symposium of Bioinformatics and Bioengineering (IEEE BIBE), 227–232.

30. Larose, Daniel T. (2006) Data Mining Methods and Models, John Wiley & Sons, New Jersey.

31. Levitin, D. J., Nuzzo, R.L., Vines, B.W., and Ramsay, J.O., (2007), "Introduction to Functional Data Analysis," Canadian Psychology, 48(3), 135–155.

32. Luan, Y. and Li, H., (2003), "Clustering of time-course gene expression data using a mixed-effects model with B-splines," Bioinformatics, 19(4), 474–482.

33. Mao, K. Z. (2005) "Identifying Critical Variables of Principal Components for Unsupervised Feature Selection," IEEE Transactions on Systems, Man, and Cybernatics - Part B: Cybernatics, 35 (2), 339–344.

34. Mei, Y., S.B. Kim, K.-L. Tsui (2009), "Identification of major metabolite features in high-resolution NMR spectra using linear-mixed effects models," Expert Systems with Applications, 36, 4703–4708.

35. Montgomery, D.C. (2004) Introduction to Statistical Quality Control, 5th Edition, Wiley, New York.

36. Morita, M., Sabourin, R., Bortolozzi, F., and Suen, C. Y. (2003), "Unsupervised feature selection using multi-objective genetic algorithms for handwritten word

recognition," Proceedings of the Seventh International Conference on Document Analysis and Recognition, 2, 666–671.

37. Owens, C.M., Zhang, D., and Willis, W.D., (1992), "Changes in the response states of primate spinothalamic tract cells caused by mechanical damage of the skin or activation of descending controls," Journal of Neurophysiology, 67(6), 1509– 1527.

38. Ramsay, J.O. and Silverman, B.W., (2005), Functional Data Analysis, Springer, New York.

39. Rousseeuw, P.J., (1987), "Silhouettes: graphical aid to the interpretation and validation of cluster analysis," Journal of Computing Applied Mathematics, 20, 53–65.

40. Schliep, A., Schönhuth, A., and Steinhoff, C., (2003), "Using hidden Markov models to analyze gene expression time course data," Bioinformatics, 19(Suppl.), i255–i263.

41. Senapati, A.K., Lagraize, S.C., Huntington, P.J., Wilson, H.D., Fuchs, P.N., Peng. Y.B. (2005), "Electrical stimulation of the anterior cingulate cortex reduces responses of rat dorsal horn neurons to mechanical stimuli," Journal of Neurophysiology, 94(1), 845–51.

42. Serban, N. and Wasserman, L., (2005), "CATS: Clustering After Transformation and Smoothing," Journal of the American Statistical Association, 100(471), 990-999.

43. Shawe-Taylor, J. and Cristianini, N. (2000), Support Vector Machines and Other Kernel-Based Learning Methods, Cambridge University, New York.

44. Vermaat, M. B., Ion, R. A., Does, R. J. M. M., and Klaassen, C. A. J. (2003) "A comparison of Shewhart individuals control charts based on normal, non-parametric, and extreme-value theory," Quality and Reliability Engineering International, 19, 337–353.

45. Wakefield, J., Zhou, C., and Self, S. (2002), "Modelling Gene Expression Data over Time: Curve Clustering with Informative Prior Distributions," Bayesian Statistics 7, Proceedings of the Seventh Valencia International Meeting, 2003.

46. Woodall, W.H. and Montgomery, D.C. (1999), "Research issues and ideas in statistical process control," Journal of Quality Technology, 31, 376–386.

47. Xu, R. and Wunsch II, D., (2005), "Survey of Clustering Algorithms," IEEE Transactions on Neural Network, 16(3), 645–678.

BIOGRAPHICAL INFORMATION

Panaya Rattakorn graduated from Assumption University with a B.S. in Information Technology. In 1999, she received her Master degree in Management Information System from Chulalongkorn University, Bangkok. In 2005, she completed a M.S. degree in Logistics from the University of Texas at Arlington (UTA), where she has continued pursuing a Ph.D. degree and working as a graduate research assistant in Luminant's emission project. Her primary research interest is to develop data mining methods and their applications to bioinformatics and neurostimulation problems. Her work covers the following areas: feature selection and functional data analysis. She is a member of the Institute for Operations Research and the Management Sciences (INFORMS) and Institute of Industrial Engineers (IIE).