DATA MINING BASED THRESHOLD DEVELOPMENT FOR NOVELTY DETECTION

by

POOVICH PHALADIGANON

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 2012

To my father, Chairath Phaladiganon, my mother, Tidanoi Termsubsan,

and my brother, Chayunthorn Phaladiganon

ACKNOWLEDGEMENTS

I would like to thank many people for their guidance and support along my road to a doctorate. This dissertation would not have been possible without them. First of all, I would like to express my sincere gratitude to my supervising professors, Dr. Victoria C.P. Chen and Dr. Seoung Bum Kim, for their invaluable advice and knowledge for both my academic and personal life during my study at the University of Texas at Arlington (UTA). In addition, I want to thank my other committee members, Dr. Li Zeng and Dr. H. W. Corley, for their interest and helpful comments on this dissertation. Dr. Li Zeng also gave me an opportunity to teach in classes, which was a great experience that I very much enjoyed.

My additional thanks go to Dr. Seoung Bum Kim for his help in providing me with an opportunity to study in the Ph.D. program at UTA. He also arranged for me to visit in Korea three summers during my studies. I also want to thank all members in the Data Mining and Quality Management laboratory at Korea University, especially Jihoon Kang, Junghwan Son, and Jeonghun Kim for very good discussions and help during my visits. I would not have had such an enjoyable experience without you.

I would like to thank my COSMOS colleagues Bancha Ariyajunya, John Dickson, Narakorn Engsuwan, Aera Kim LeBoulluec, Piyush Kumar, Smriti Neogi, and Zirun Zhang for their friendship. My thanks go to COSMOS graduates Dr. Chivalai Temiyasathit, Dr. Panitarn Chongfuangprinya, Dr. Panaya Rattakorn, Dr. Surachai Chareonsri, and Dr. Passakorn Phananiramai as well.

My Special thanks go to Dr. Thuntee Sukchotrat, Dr. Weerawat Jitpitaklert, and Dr. Siriwat Visoldilokpun for useful discussions and their friendship. I want to

thank Dr. Nyle Spoelstra for helpful comments on this dissertation. Additionally, many thanks go to friends in Texas, Saranya Sukchotrat, Chotiya Sukchotrat, Thanat Thanapattum, Pisit Thanapattum, Primana Punnakitikashem, and Naruemol Detudompluet.

Finally, I want to thank my entire family for their unconditional love and unwavering support. My thanks go to my sisters, Yaowarej Mekratri, Kanyarat Phaladiganon, Atchara Mekratri, and Ujainee Phaladiganon. I want to thank my dad, Chairath Phaladiganon, and my mom, Tidanoi Termsubsan for their love, patience, advice, and encouragement. Without them, I could not have done it. I also had many good times during my study at UTA with my brother, Chayunthorn Phaladiganon. Thank you for always being with me. I love you all.

Thanks to all of you, I've "made it through!"

October 18, 2012

ABSTRACT

DATA MINING BASED THRESHOLD DEVELOPMENT FOR NOVELTY DETECTION

POOVICH PHALADIGANON, Ph.D.

The University of Texas at Arlington, 2012

Supervising Professor: Victoria C. P. Chen

The objective of this dissertation is to develop thresholds for novelty detection with applications to statistical process control (SPC). SPC is a widely used technique for improving process and product quality. The primary tool of SPC is a control chart that is used to monitor and detect abnormal processes. Traditional control chart techniques usually require a specific distributional assumption, typically the normal distribution, to establish their control limits. However, in modern manufacturing processes, the normality assumption is often violated. Novelty detection more generally seeks abnormal (or novel) patterns in data, and novelty detection techniques can be applied to control charts in SPC. This dissertation consists of three components.

First, a bootstrap-based threshold for detecting abnormal patterns in multivariate T^2 control chart is proposed. This approach can efficiently monitor a process when the distribution of observed data is nonnormal or unknown. The bootstrap method is a nonparametric technique that does not rely on the assumption of a parametric distribution of the observed data. Prior SPC literature only studies the bootstrap

technique to develop univariate control charts to monitor a single process, while in this dissertation, the bootstrap technique is integrated with multivariate control charts.

Second, principal component analysis (PCA)-based control charts have been widely used to address problems posed by high correlations by reducing dimensionality. However, an assumption that the data are normally distributed has limited the use of PCA control charts. In this dissertation, the bootstrapping threshold approach and a kernel density estimation approach are employed for threshold development to yield nonparametric PCA control charts that do not require any distributional assumptions for their construction.

In novelty detection, support vector data description (SVDD) is a one-class classification technique that constructs a boundary in order to differentiate novel from normal patterns. However, boundaries constructed by SVDD do not consider the density of the data. Data points located in low density regions are more likely to be novel patterns because they are remote from their neighbors. This study presents a density-focused SVDD (DFSVDD), for which its boundary considers both shape and the dense region of the data.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The capability to detect novel patterns in data, widely used in academia and science, is attracting increasing attention for various practical applications. Novelty detection is a widely used technique to identify novel patterns in data. The novel patterns targeted for identification are defined as a new or unusual occurrences in data that do not conform to normal patterns [1]. The basic idea of novelty detection is to differentiate the specified "normal" target class from all other classes. Furthermore, novelty detection can be used when one important class is undersampled. Although information on one of the classes in the data is typically abundant and easy to acquire, collection of data specific to the abnormal or novel conditions may be difficult because of their rarity and the expense of identifying and obtaining them. The need to record every example from the novel class to ensure that it represents all types of circumstance is a further complication.

For instance, the status of a machine operating normally, which would generate data referred to as the target class, can be seen daily in manufacturing. Data for this circumstance is easy and practical to measure and collect. In contrast, the data associated with a breakdown of the same machine, a situation referred to as a novelty, is rarely collected [2]. The representation of such a novelty also must cover all possible cases that can cause the machine's failure. Another typical example occurs in a medical diagnosis in which any type of disease must be detected. Information on healthy people in the target class is well sampled, but there is a lack of information on all types of diseases that constitute a novelty and depart from the norm. The

techniques of novelty detection have far-ranging potential application in such fields as detection of fraudulent credit transactions [3], intrusion into computer networks [4], and locations of mines in minefields [5].

As described earlier in the example of a machine breakdown, novelty detection can be used in quality control in terms of charting control problems. The main goal of a control chart is to detect abnormal behavior in a process. Furthermore, construction of control charts requires only data representing the target class. The main focus of this dissertation is development of thresholds for novelty detection for subsequent application to statistical process control, especially in problems of control charting.

Quality is a key factor in manufacturing success. Another factor, customers' purchasing of products and services, is mainly determined by the level of quality. To satisfy their customers, manufacturers try constantly to control and improve the quality of their products and services. It is the simple nature of things that no two products can be produced exactly the same. This is true because of variations in the process. The causes of these variations are of two types: common and special. Common causes of variation are a natural part of a process and cannot be removed or controlled. A process that varies only because of common causes is considered in control. In contrast, special causes lead to an excessive magnitude of variation in a process and render it out of control.

Statistical process control (SPC) contains a set of powerful tools used to improve processes to provide quality products and services to customers. SPC comprises seven major tools: histogram or stem-and-leaf plots, check sheets, Pareto charts, cause-and-effect diagrams, defect concentration diagrams, scatter diagrams, and control charts. This dissertation mainly focuses on a control chart used to monitor the performance of a process and product over time. Walter A. Shewhart of Bell Laboratories was the

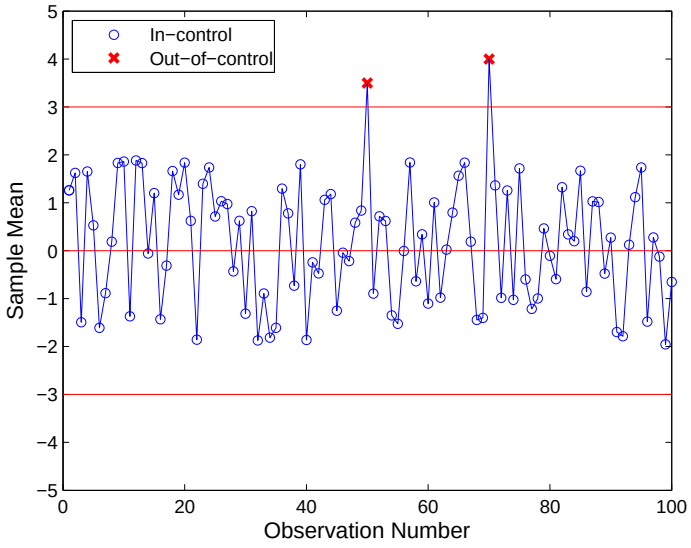


Figure 1.1 Shewhart $\bar{x}$ chart.

first to devise a control chart. His innovation can assist in reducing the variability inherent in processes and prevent the manufacture of defective products.

Several quality characteristics, such as thickness, diameter, and length, describe a product in a manufacturing process. Control charts are a graphical display of a quality characteristic [6]. A control chart is composed of two major components. The first is a monitoring statistic that is a function of a measurement of a quality characteristic taken from a process and used to plot the control chart. Some well-known examples of monitoring statistics are the sample mean ($\bar{x}$) and sample range (R). Control limits are the second component in control charts. These establish the threshold for a determination of whether or not a process is in control. When a process shifts out of control, the monitoring statistic falls outside the control limits and serves to indicate that the process is out of control. However, control charts sometimes falsely signal a change in the process. This situation is called a false alarm. Figure 1.1 illustrates a Shewhart $\bar{x}$ chart in which data were generated under

an assumption that the process is in control. However, two observations (50^{th} and 70^{th}) fall outside the upper control limit. These two points are identified as false alarms.

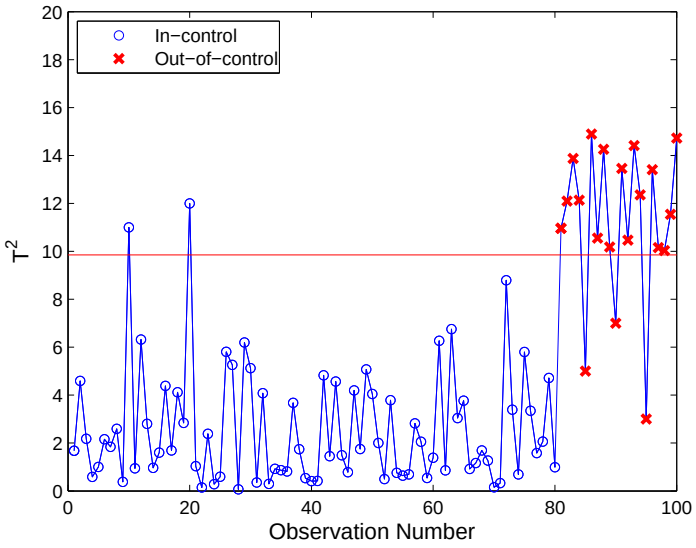


Figure 1.2 Multivariate Hotelling's T^2 chart.

Control charts can be categorized into two types: univariate and multivariate. Univariate control charts monitor a single quality characteristic. Figure 1.1 is an example of univariate Shewhart $\bar{\text{x}}$ chart widely used to monitor a process mean. Multivariate control charts monitor multiple quality characteristics in one chart. Figure 1.2 illustrates a multivariate Hotelling's T^2 control chart. These data were generated so that the first 80 observations are the actual in-control observations, but the last 20 are actual out-of-control observations. Figure 1.2 clearly shows that the process has shifted out of control.

Two types of error rates are commonly used to measure the performance of control charts. A Type I error rate (α), also referred as to the false alarm rate, is the

ratio of actual in-control observations that are incorrectly identified as out of control to the total number of actual in-control observations. A Type II error rate (β) is the actual out-of-control observations that are incorrectly identified as in control divided by the number of actual out-of-control observations. In Figure 1.2, observation numbers 10 and 20 are classified as Type I errors, and Type II errors can be recognized by observation numbers 85, 90, and 95. Another performance measure for control charts is the average run length (ARL), which is the average number of observations before an out-of-control signal occurs. In-control ARL (ARL_0) corresponds to the ARL before a false alarm occurs when the process is in control, and out-of-control ARL (ARL_1) is the ARL before occurrence of an out-of-control signal. In practice, an engineer specifies the ARL_0 so that few false alarms occur; however, a small ARL_1 is equally desirable for quick detection of an out-of-control situation.

1.1 Motivation and Contribution

The research in this dissertation was first motivated by the novelty detection aspect of SPC methods. Novelty detection techniques are applicable to control charting problems because both share similar detection objectives. Moreover, in novelty detection approaches the rare representative samples of the other classes (novelty) are difficult or impractical to acquire for model training. However, the construction of control charts requires only information about in-control (target) data. Apart from the process of training a model, a control chart needs only development of a threshold as a basis for decision making. However, thresholds (control limits) constructed for control charts require a parametric assumption. A certain distribution has to be known for mathematical derivation of a threshold formula.

Usually, traditional control charts assume that process data follow a certain probability distribution. In particular, multivariate T^2 control charts (T^2 charts)

are efficient and reliable when the process data are normally distributed (see Section 2.2). Under this assumption, T^2 statistics follow the F-distribution. The control limit of a T^2 chart is the upper quantile of the F-distribution with corresponding degrees of freedom. However, data from complex industrial processes often do not follow a specific distribution. This implies that the control limits obtained by the F-distribution may be inaccurate because the rate of false alarms is higher than desired. In this dissertation, a bootstrap approach that does not require any distributional assumptions is employed to establish control limits for T^2 charts under nonnormality.

The bootstrap method is a nonparametric estimation method that can be used for distributional approximation. This method helps to avoid a distributional assumption, such as normality, that is often a requirement of model construction. The general idea of the bootstrap method is to draw a random sample with replacement from the empirical distribution to obtain a bootstrap sample. This procedure is referred to as a resampling technique. Statistics of interest, such as mean, median, and mode, are calculated from each bootstrap sample. This process can be repeated many times to obtain the statistics of interest by using the power of computing. A large number of bootstrap samples are required for complex estimations [7].

Further, a large number of highly correlated quality characteristics may render traditional T^2 charts less effective at detecting shifts in the process because of an increased rate of false alarms. Principal component analysis (PCA)-based control charts have been used to overcome these problems (see Section 2.3). PCA is a statistical analysis technique primarily used for dimensional reduction. However, the distributional assumption of PCA-based control charts is that process data are normally distributed. In this dissertation, bootstrap and kernel density estimation (KDE) approaches have been implemented to construct control limits for PCA-based control charts when the data follow multivariate nonnormal distributions. Kernel

density estimation is a nonparametric method incorporated with kernel techniques that is used to estimate probability distributions.

Support vector data description (SVDD) is a one-class classification technique widely used for novelty detection. However, boundaries constructed from the traditional SVDD do not take into account the density of the data. Intuitively, the data points located in low density regions are most likely to be novelties because they are remote from their neighbors. This study presents a density-focused SVDD (DFSVDD) where its boundary considers both the shape and dense regions of the data.

1.2 Outline of this Dissertation

Chapter 2 gives a brief background of data mining techniques. Several novelty detection algorithms are also discussed in this chapter and classical univariate and multivariate control charts are briefly surveyed. In addition, this chapter contains a discussion of control charts in terms of novelty detection and the implementation of the bootstrap method for threshold development in control chart problems.

Chapter 3 presents a bootstrap-based T^2 multivariate control chart. The proposed bootstrap percentile approach is used to estimate the control limits of T^2 control charts when the data do not follow a multivariate normal distribution. The performance of the proposed bootstrap-based T^2 control charts is compared with the traditional T^2 and existing kernel density estimation (KDE)-based T^2 control charts based on in-control average run length.

Chapter 4 presents PCA-based bootstrap control charts for multivariate non-normal distributions. The existing PCA-based control charts, T^2_{PCA} and Q charts, are integrated into the bootstrap and KDE approaches to determine control limits. The nonparametric control limits obtained from the bootstrap and KDE approaches

are compared with the traditional PCA control charts in a simulation study that uses various types of multivariate nonnormal distributions.

Chapter 5 presents a method that was developed based on support vector data description (SVDD). The traditional SVDD includes the density of the data. The resultant method is called density-focused SVDD (DFSVDD). A simulation study demonstrates the comparative performance of DFSVDD and the traditional SVDD.

Chapter 6 summarizes this dissertation and presents future research.

CHAPTER 2

LITERATURE REVIEW

2.1 Data Mining

The rapid growth of data generation and the need to collect and analyze vast quantities of it has led to development of more sophisticated techniques for extracting potential information from these data. Data mining is defined as the process of discovering meaningful information in large data [8][9][10] and has been used in many areas of science, business, and industry. Data mining can be separated into two categories: supervised learning and unsupervised learning [11]. Supervised learning uses both input and output variables to create a model to predict or classify future output observations. Unsupervised learning uses only information from input variables. The goal of an unsupervised learning technique is to find hidden structure in the data without the benefit of information from output variables.

2.1.1 General Description of Data Mining Methods

Supervised learning uses two approaches, regression and classification [11]. Regression involves continuous output variables. The purpose of regression is to predict the response values of future observations. Classification is based on qualitative or categorical output variables. The goal is to create a model using existing labeled observations to classify unlabeled observations.

Linear regression models, one of the regression techniques, have been frequently used for prediction in regression problems because of their simplicity and interpretabil-

ity. Given $X = (X_1, X_2, \ldots, X_p)$ linear regression models describe the relationship between response (Y) and predictor variables (X) as follows:

$$Y = \beta_0 + \sum_{i=1}^{p} X_i \beta_i, \tag{2.1}$$

where least square estimation can be used to estimate the β parameters by minimizing the residual sum of squares. The goal of a linear regression model is to use single or multiple predictor variables to predict values of a response variable. However, a linear regression model may be misleading when the relationship between the response and predictor variables is nonlinear.

Classification methods constitute a decision boundary for the classification of unlabeled observations. Linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) are traditional classification techniques based on a multivariate normal distribution. LDA generates a linear decision boundary and assumes that different classes have the same covariance. In contrast, QDA determines the quadratic decision boundary and assumes that different classes have different covariances.

Decision tree models have been applied to many tasks because they can handle both classification and regression problems that, respectively, use categorical and continuous output variables. The tree models generate boundaries by partitioning the input variables' space into a set of rectangular regions. Typically, a top-down strategy is used to grow the tree model and then a tree pruning technique is performed to improve the prediction error rate. Decision tree models are flexible because they can apply to all types of data, ranging from data that is continuous to data that is a mix of categorical and numerical input variables. Furthermore, a decision tree model is easy to interpret because of the hierarchical structure that the model builds.

The k-nearest neighbor (kNN) method is a supervised learning technique that is considered as an instance-based learning method. The kNN algorithm can be

used in both classification and regression, depending on the type of output variables. kNN classification is suitable for categorical output variables, and kNN regression is suitable for continuous output variables. Generally, the kNN algorithm first uses distance measures to determine the k observations in a training data set that are closest to a new observation. Euclidean distance or Mahalanobis distance are typically used. In classification problems, the new observation is assigned to a class that has the largest majority vote among the k nearest points. In contrast, in regression problems the average output variables of the k nearest points is the predicted value of the new observation.

Support vector machine (SVM) learning is a statistical learning theory widely used in classification [12], and support vector regression is gaining popularity [13]. The SVM algorithm determines the optimal separating hyperplane in a high dimensional input variable space. SVM maximizes the margin and generalizability between classes by solving a convex optimization [14]. When the data are not linearly separable, SVM uses many types of kernel functions, such as polynomial, Gaussian, and sigmoid, to construct a nonlinear decision boundary.

Artificial neural network (ANN) models were motivated by biological learning systems that consist of a number of interconnected neurons [15]. ANN models are powerful for both regression and classification types of prediction, but they are difficult to interpret. The structure of an ANN model consists of input and output layers and typically one or more hidden layers in between. Each layer is composed of nodes that correspond to neurons. The number of nodes for the input and output layers is determined by the numbers of input and output variables, respectively, but the modeler must determine the number of nodes inside the hidden layer or layers. The model parameters specify weights on connections between nodes from one layer to the next. The ANN algorithm iteratively adjusts the weights during the training

process to find the optimal output that produces the minimum mean squared error. Activation functions convert the input from one node to the next based on the desired structure of the output from a node. For example, a step function can be used when classification is needed, but a continuous function is used in regression [16].

The other type of learning technique in data mining is unsupervised learning. Unsupervised learning techniques extract important information from data or identify patterns within it without regard for information from the output variables. Principal component analysis (PCA) is an unsupervised learning technique that is extensively used for dimensional reduction and visualization.

PCA is a multivariate analysis technique that extracts a new set of variables by projecting the original variables onto principal component space. The extracted variables, called PCs, are linear combinations of the original variables in which the coefficients of the linear combination can be obtained from the eigenvectors of the covariance (or correlation) matrix of the original data [17]. Geometrically, PCA rotates the axes of the original coordinate system to a new set of axes along the direction of maximum variability of the original data [18]. PCs are uncorrelated with each other, and the first few PCs can usually account for most of the information of the original data. Let $\mathbf{x} = [x_1, x_2, \ldots, x_p]^T$ represent a random vector of observations on p quality characteristics with the sample covariance matrix S. Based on the spectral decomposition that links the structure of a symmetric matrix to the eigenvalue-eigenvector pairs [19], the covariance matrix can be decomposed as follows:

$$\mathbf{U}^T\mathbf{S}\mathbf{U} = L, \tag{2.2}$$

where $\mathbf{U}$ is the matrix composed of eigenvectors in columns corresponding to the diagonal $\mathbf{L}$ matrix, which are the eigenvalues sorted in descending order, $l_1 > l_2 > \ldots > l_p > 0$. $\mathbf{U}$ is orthonormal eigenvectors such that $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. The principal component

technique transforms p correlated variables into new p uncorrelated variables by using the following equation:

$$Z = \mathbf{U}^T[\mathbf{x} - \bar{\mathbf{x}}]. \tag{2.3}$$

Determination of the appropriate number of PCs to retain can be subjective. In general, a scree plot that visualizes the proportion of variability of each PC can be used [20]. The first few PCs generally account for a large proportion of the variability in the data; this allows significant dimension reduction with little loss of information. However, PCA can be used effectively when data has a linear structure. In case of nonlinear structure, PCA can be implemented by incorporating a kernel technique into it [21].

Cluster analysis is a class of unsupervised learning methods that seek to discover underlying structure in data by grouping observations into homogeneous clusters. Observations within a cluster are considered to have higher similarity to each other than to observations assigned to different clusters. Cluster analysis can be separated into two types: nonhierarchical and hierarchical clustering. One of the most popular nonhierarchical clustering techniques is the k-means clustering algorithm [11]. The k-means clustering algorithm iteratively assigns new center points to form clusters until it achieves the minimum mean squared distance from each observation to its nearest center. However, the performance of the k-means algorithm depends on k, the user-specified number of clusters, and the choice of distance measures.

In hierarchical cluster analysis, distance measures still determine the clusters but are represented by a dendrogram. The advantage of this algorithm is that it does not require specification of the number of k clusters and a starting point. Hierarchical cluster analysis can be constructed according to two strategies: agglomerative and divisive methods. An agglomerative method constructs the dendrogram by starting

at the bottom in which each observation is set as an initial cluster. Then a selected pair of clusters is recursively merged into a single cluster. A divisive method is a top-down approach. All observations are used as an initial cluster and recursively split into two new clusters at each level. Further, distances between sets of observations can be determined by various linkage criteria, such as single, complete, and average linkage and by Ward's method [22].

As described in this section, data mining tools can be used for different tasks, depending on the application. Novelty detection, which is described in the next section, is viewed as a subset of data mining. The purpose of novelty detection is to detect abnormalities in the data.

2.1.2 Novelty Detection

Novelty detection is an approach to recognition of abnormal patterns among the normal patterns in data. In classical classification techniques, the predefined classes of training observations contain more than one category. Generally, a decision boundary in conventional classification problems is built from a binary class. That means the information on two classes is available to train a model. Once trained, a model determines the normalcy or novelty of the class of an unknown observation. However, building a classification model with the conventional classification algorithms requires a balance of classes from the collected data. Expense or time constraints can sometimes rule out gathering information on a class of data. To overcome such problems for novelty detection, one-class classification techniques can be used. The main purpose of one-class classification techniques is to construct a boundary to envelop the target class.

Several approaches have been proposed for detecting novelties in one-class classification tasks. The Gaussian density model is a well-known statistical model that

estimates mean and covariance from training observations. The basic assumption of the Gaussian density model is that the data observations are normally distributed. This model can be viewed as using the Mahalanobis distance. Any observations with larger distances than a threshold are classified as novelties. In order to relax the normality assumption, the Gaussian mixture model (GMM) is used. By specifying the parameters of the GMM, means and covariance, one can estimate distributions by using an expectation-maximization algorithm [23].

Parzen window estimation [24], also known as kernel density estimation, is a nonparametric technique to estimate the underlying probability density function of data. Each training observation serves as a center in a kernel function. To estimate the probability density of a new observation, the average distances from a new observation to all centers in a kernel function are calculated. A probability density value associated with a new observation is compared with a threshold to determine whether it is a normal or novel pattern. Although several existing kernels can be used for the Parzen window estimation, the Gaussian kernel function is commonly chosen in practice [23]. Further, the performance of the Parzen window is dependent on a smoothing parameter, which determines the smoothness of the estimation. Several approaches, such as maximum likelihood estimation, the normal reference rule, and Scott's rule, can be used to determine this smoothing parameter [25].

Nearest neighbor data description (NNDD) uses information on local density for comparative distance purposes. The NNDD algorithm compares the distance from a new observation $\mathbf{z}$ to its nearest neighbor (NN($\mathbf{z}$)) with the distance from the NN($\mathbf{z}$) to its nearest neighbor of NN($\mathbf{z}$) [26]. To increase the robustness of the NNDD approach to noise, the k parameter can be introduced to select the number of neighbors among the given data points.

Other robust distance-based approaches also have been proposed to identify novelties in data. Harmeling et al. [27] introduced three distance-based novelty detection techniques: (1) distance to the k-th nearest neighbor, (2) average of distances to the k nearest neighbors, and (3) distance to the average of k nearest neighbors. The distance to the k-th nearest neighbor determines the radius of a hypersphere centered on an observation that then contains k neighbors. The average of distances to the k nearest neighbors, instead of considering only the farthest data point among the nearest neighbors, attempts to find distances from an observation to all of its k nearest neighbors and then take an average. Lastly, the distance to the average of k nearest neighbors is simply identified as a distance from an observation to the mean of its k nearest neighbors. Another useful distance-based approach for novelty detection is the minimum spanning tree (MST) proposed by Juszczak et al. [28]. The MST is represented by a graph, which consists of edges and vertices. The MST is an undirected and acyclic graph that connects all the vertices in such a way that the minimum total weight is obtained [29]. The distance from a new observation to the MST can be measured either from an edge or a vertex, either of which is chosen by the smallest distance criteria.

Several data mining methods from Section 2.1.1 can be used in novelty detection, including clustering, PCA, and SVM. Clustering algorithms seek to discover underlying structure in data by assigning target observations into different groups. As described earlier, the k-means clustering algorithm forms clusters by minimizing the mean squared error determined from each training observation to its nearest center [30]. For the purposes of novelty detection, the distance from a new observation to its nearest cluster can be evaluated as a novelty score and then compared with a threshold to determine if the observation is normal or novel. Using PCA, the residuals obtained from a remaining set of $p - k$ PCs are suggested for use in determining a

novelty. Because PCA only maximizes the information of the target data, the novelty loses more information when it is projected to a lower dimensional space than the target. SVM was originally developed for a two-class problems. However, Scholköpf et al. [31] extended the traditional SVM using a separating hyperplane to one-class problems of novelty detection. His one-class SVM constructs an optimal hyperplane that differentiates the targets from areas containing no data with a maximal margin [32].

Invention of support vector data description (SVDD) was inspired by the SVM. Instead of using a separating hyperplane to differentiate between targets and novelties, SVDD generates a hypersphere to enclose target observations with a minimum radius hypersphere. SVDD is a one-class classification technique, originally introduced by Tax and Duin [33] for novelty detection. SVDD produces a hyperspherical boundary constructed by a set of support vectors obtained by solving a convex optimization problem. Let $\mathbf{x}_i$, where $\mathbf{x}_i \in \mathbb{R}^p$ and $i = 1, 2, \ldots, N$ be the training observations. Let R^2 be the radius of the hypersphere and $\mathbf{a}$ be the center of the hypersphere. SVDD constructs a boundary by minimizing the volume of the hypersphere and maximizing the number of its enclosed training observations [34]. This can be summarized as follows:

$$\text{Minimize } R^2 + C \sum_{i=1}^{N} \xi_i, \tag{2.4}$$

with the constraint:

$$\|\mathbf{x}_i - \mathbf{a}\|^2 \leq R^2 + \xi_i, \tag{2.5}$$

where $\xi_i > 0$ is the slack variable that relaxes the constraint in order to allow $\mathbf{x}_i$ to be rejected from the hypersphere. By introducing a Lagrangian function, constraint (2.5) can be incorporated in (2.4) and the problem is modified into

$$L(R, \mathbf{a}, \alpha_i, \gamma_i, \xi_i) = R^2 + C \sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \alpha_i \{R^2 + \xi_i - (\|\mathbf{x}_i - \mathbf{a}\|^2)\} - \sum_{i=1}^{N} \gamma_i \xi_i, \tag{2.6}$$

where $\alpha_i \geq 0$ and $\gamma_i \geq 0$ are the Lagrange multipliers. Setting partial derivatives of L with respect to R, $\mathbf{a}$, and ξ_i to zero results in the following constraints:

$$\sum_{i=1}^{N} \alpha_i = 1,$$

$$\mathbf{a} = \sum_{i=1}^{N} \alpha_i \mathbf{x}_i,$$

$$C - \alpha_i - \gamma_i = 0.$$

By substituting these constraints to (2.6), the following optimization problem can be obtained:

$$\text{Maximize } \sum_i \alpha_i (\mathbf{x}_i \cdot \mathbf{x}_j) - \sum_{ij} \alpha_i \alpha_j (\mathbf{x}_i \cdot \mathbf{x}_j). \tag{2.7}$$

with the constraint:

$$0 \leq \alpha_i \leq C \tag{2.8}$$

$$\sum_{i=1}^{N} \alpha_i = 1, \tag{2.9}$$

where $i = 1, 2, \ldots, N$. By solving the optimization problem (2.7) with constraints (2.8) and (2.9), a set of α_i can be obtained. A training observation x_i corresponding to α_i satisfies the following:

$$\|\mathbf{x}_i - \mathbf{a}\|^2 < R^2 \Rightarrow \alpha_i = 0,$$

$$\|\mathbf{x}_i - \mathbf{a}\|^2 = R^2 \Rightarrow 0 < \alpha_i < C,$$

$$\|\mathbf{x}_i - \mathbf{a}\|^2 > R^2 \Rightarrow \alpha_i = C.$$

The observations $\mathbf{x}_i$ with $\alpha_i > 0$ are called support vectors (SVs). The relation between C and user-specified f, which is used to control the trade-off between the volume of hypersphere and the misclassified observations, can be represented as follows:

$$f = \frac{1}{NC}, \tag{2.10}$$

where N is the number of training observations. When no novelty is expected to be contained in the training data, C can be set equal to 1 [33]. One can obtain flexible boundaries by replacing the inner product with kernel functions. The following Gaussian kernel function is used throughout this work:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{S^2}), \tag{2.11}$$

where S can be changed to adjust the complexity of the SVDD boundary.

To determine if a testing observation $\mathbf{z}$ is a novelty, the distance D^2, between $\mathbf{z}$ and the center of the hypersphere $\mathbf{a}$, is used. That is, a testing observation is classified as a novelty if D^2 is greater than R^2.

$$D^2 = \|\mathbf{z}_i - \mathbf{a}\|^2 = K(\mathbf{z}, \mathbf{z}) - 2\sum_i \alpha_i K(\mathbf{z}, \mathbf{x}_i) + \sum_{ij} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j). \tag{2.12}$$

The effects from the parameters of SVDD (C and S) are illustrated in Figure 2.1 by using banana-shaped data generated through PRTools [35]. The SVDD boundaries were constructed from 200 target observations with different values for the parameters C and S. The parameter S increases from left to right, and the number of parameters C decreases from the top down. It can be determined from Figure 2.1 that given the same C ($C = 1$ and $C = 0.02$), the parameter S controls the smoothness of the boundary. A smaller value for S creates more complex SVDD boundaries, but a larger value yields smoother SVDD boundaries. As the parameter S becomes very large, the number of SVs used to describe SVDD boundaries decreases. Consequently, SVDD boundaries begin to resemble a regular hypersphere [33]. The parameter C controls the misclassification error rate in targets. Because the relationship between f and C is defined in Equation (2.10), when the parameter C decreases, the misclassification error rate in targets increases. That is, the target observations are rejected from the boundary of SVDD, which is shown from the top down in Figure 2.1, given

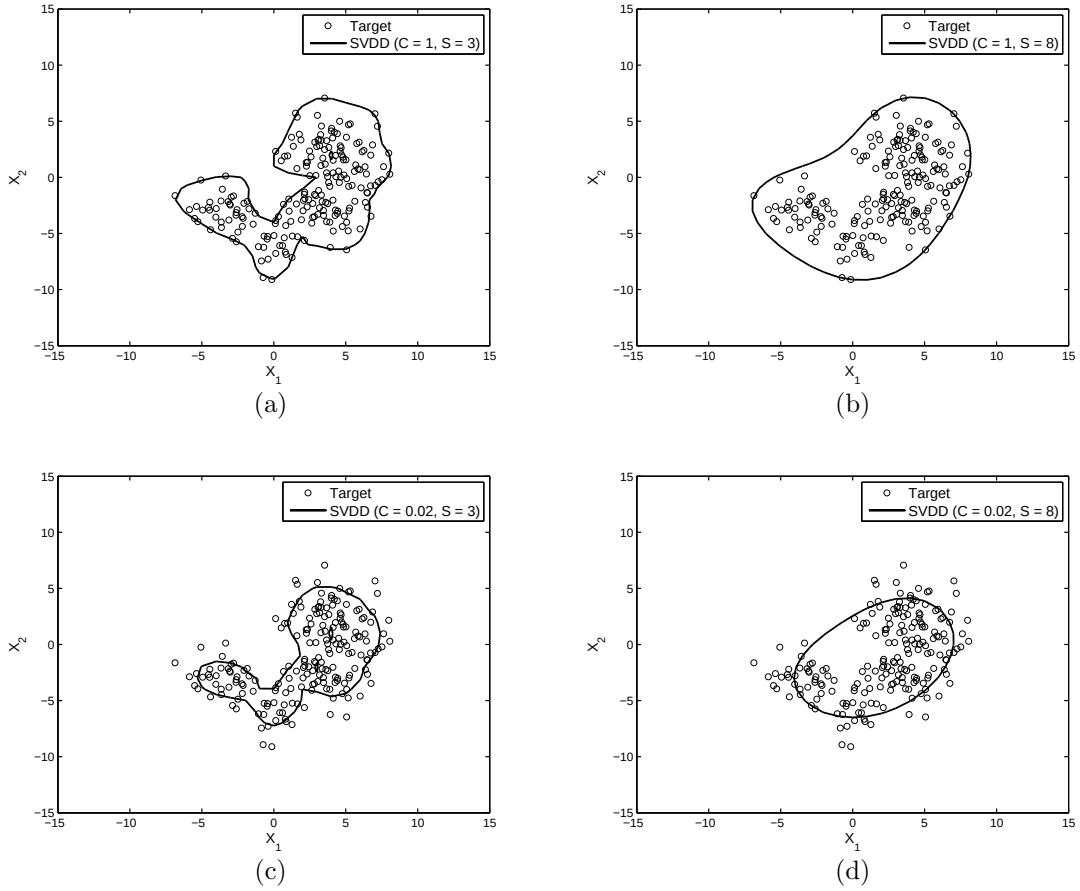


Figure 2.1 Boundaries of SVDD obtained from different values of parameters: (a) $C = 1$ and $S = 3$; (b) $C = 1$ and $S = 8$; (c) $C = 0.02$ and $S = 3$; (d) $C = 0.02$ and $S = 8$.

the same S ($S = 3$ and $S = 8$). Note that a hard-margin solution can be obtained when C is set to 1 because constraints (2.8) and (2.9) are always satisfied [34]. This implies that all the training data are included in the hypersphere (Figures 2.1(a) and 2.1(b)). Further, SVDD can be viewed as a one-class SVM when a Gaussian kernel function is applied to the construction of the classifier [26].

SVDD has been applied in a range of applications. Cho [36] used SVDD to detect abnormal observations in chemical and biological processes. Before modeling SVDD for process monitoring, orthogonal signal correction was used as a preprocess-

ing step to remove any unnecessary variation that exists in the data. In hyperspectral imagery, a certain object has a unique spectral signature representing its identity in an image. Banerjee et al. [37] applied SVDD in mine detection to distinguish between the mine and the background classes. Because of the scarcity of the novelty class (mines), the background images are used as the target class in which the mines were embedded.

The idea to use SVDD for pattern denoising was proposed by Park et al. [38]. The technique of SVDD-based pattern denoising uses a geodesic projection of noise data to obtain a denoised feature. The pre-image technique, which can reverse from the feature space back to the input space, is performed to recover the denoised pattern [39]. This technique was implemented in the application of handwritten digit data from the U.S. Postal Service digit database. One deceptive aspect of a novelty is its capability to act like data extracted from a normal operation. This happens because an error in data measurement can result in uncertainty in the characterization of data. The development of SVDD to handle uncertain data was proposed by Liu et al. [40]. The algorithm aims to reduce the impact of uncertainty in the data during the training process. A confidence score is assigned to each observation. A low confidence score indicates the observation is remote from the other observations. Assigning confidence scores diminishes the influence of remote observations on the construction of a SVDD.

Furthermore, the traditional SVDD was reformulated to incorporate local density from each training data point [41]. When a degree of local density was incorporated, the optimization problem becomes difficult to solve because it is no longer a quadratic programming problem. The use of SVDD can be extended to incorporate weighting as determined by a kernel possibilistic c-means algorithm (PCM) [42]. The kernel PCM assigns small weights to those observations likely to be novelties. The

purpose of the weights is to make the algorithm robust to the novelties represented in the data. This method can also be extended to multiclassification problems by adding a classification rule based on Bayesian decision theory. Zhang et al. [43] improved the weighted SVDD based on kernel PCM by incorporating novelties into the training process to obtain a better description of the data. The method was tested on data generated from rolling machinery fault emulation equipment.

SVDD can also be reformulated into two-class problems, called two-class SVDD, by introducing new constraints into the optimization problem [44]. In situations in which the training observations contain more than one class, the two-class SVDD approach constructs two hyperspherical boundaries to envelop the target observations in each class. Any observations that fall outside these boundaries are declared novelties.

2.2 Statistical Process Control

Statistical process control (SPC) is a widely used technique for monitoring and improving the performance of a process and the quality of products. One primary technique in SPC is a control chart. Control chart techniques can be viewed as a graphical display of the statistical hypothesis testing [45]. The main purpose of a control chart is the detection of an out-of-control signal so that process quality can be maintained and production of defective products prevented. Control charts consist of two major components: monitoring statistics and control limits. Monitoring statistics are used to represent quality characteristics of interest, such as temperature, pressure, and tensile strength, that are plotted over time. Control limits are used to determine whether the process is in control and are usually estimated from an assumed underlying distribution of the in-control monitoring statistics. If the value of a monitoring statistic exceeds the control limit, the corresponding observation is considered out of control. As a result, an appropriate action has to be taken to return

the process to an in-control state. Control charts can be categorized as univariate or multivariate. Univariate control charts were invented to monitor the quality of a single process variable, and multivariate control charts monitor multiple process variables.

2.2.1 Univariate Control Charts

The control chart was first proposed by Walter A. Shewhart [46]. The Shewhart $\bar{x}$ chart is a univariate control chart used to monitor a single quality characteristic of interest. The purpose of the $\bar{x}$ chart is to detect a shift in the process mean. Figure 1.1 in chapter 1 illustrates a Shewhart $\bar{x}$ chart in which all observations are inside the upper and lower control limits, representing an in-control process. However, the limitation of this type of $\bar{x}$ chart is that it may be inefficient in detecting small shifts in a process. The cumulative sum (CUSUM) [47] and the exponentially weighted moving average (EWMA) [48] control charts were developed to gain increased sensitivity in detecting small process shifts by using information from past and current observations.

The performance of control charts often relies on the underlying distribution of the quality characteristic. Traditional control charts assume that the distribution of a quality characteristic is normally distributed. When this assumption is violated, the performance of control charts can be degraded, leading to an increased rate of false alarms [6]. Several studies have investigated the effect of nonnormality on Shewhart $\bar{x}$ charts [49][50][51]. Further, Borror et al. [52] studied the robustness of EWMA control charts on nonnormal distributions. The results showed that an EWMA chart with a proper design can be used as an alternative to the Shewhart $\bar{x}$ chart.

2.2.2 Multivariate Control Charts

When the goal is to monitor a single quality characteristic, a univariate control chart, such as the Shewhart $\bar{\text{x}}$, is used. However, because of the advent of advanced technology, many modern industrial processes are characterized by numerous quality characteristics that are correlated with each other [53]. Reliance on univariate control charts when multivariate problems are involved may lead to unsatisfactory results such as an increased rate of false alarms [6][54]. Multivariate control charts that can monitor two or more quality characteristics have been used to compensate for the limitations of univariate control charts. The most widely used of these is Hotelling's T^2 control chart (T^2 chart), which can be considered a multivariate version of the Shewhart chart [55]. Hotelling's T^2 charts monitor T^2 statistics that measure the distance between an observation and a scaled-mean estimated from the in-control data. Suppose that a dataset contains n observations, and each observation is characterized by p process variables. Assuming that the dataset follows a multivariate normal distribution with an unknown μ and a covariance matrix Σ, the Hotelling's T^2 statistics can be calculated by the following equation:

$$T^2 = (\mathbf{x} - \bar{\mathbf{x}})^T \mathbf{S}^{-1} (\mathbf{x} - \bar{\mathbf{x}}), \tag{2.13}$$

where $\bar{\mathbf{x}}$ is a sample mean vector and $\mathbf{S}^{-1}$ is the inverse of a sample covariance matrix obtained from an in-control process. The control limits of T^2 can be computed by using procedures that will be discussed in subsequent sections.

2.2.2.1 F-Distribution

T^2 statistics follow the F-distribution with p and n-p degrees of freedom based on a multivariate normality assumption [6]. The control limits of the T^2 control chart(CL_{T^2}) can be determined by

$$CL_{T^2} = \frac{p(n+1)(n-1)}{n^2 - np} F_{(\alpha,p,n-p)}, \tag{2.14}$$

where n and p, respectively, are the number of observations and process variables. In other words, the $100\alpha\%$ upper percentile of an F-distribution is used as the control limit, where α is a Type I error rate (false alarm rate) that can often be specified by the user. A Type I error rate is estimated by the ratio of in-control observations that are incorrectly identified as out of control to the total number of in-control observations. The control limit thus established is used to monitor future observations. An observation is considered out of control if the corresponding T^2 statistic exceeds the control limit. Figure 1.2 in chapter 1 illustrates the multivariate Hotelling's T^2 chart in which the control limit is estimated based on the F-distribution. However, when the normality assumption about the data does not hold, a control limit based on the F-distribution may be inaccurate because a control limit determined this way can increase the rate of false alarms [56].

It is known that T^2 charts are efficient in detecting large shifts in mean. That is, T^2 charts are insensitive to small shifts. Multivariate CUSUM [57][58][59] and multivariate EWMA charts [60], which are non-Shewhart-type control charts, have been developed to overcome this problem. Traditional multivariate control charts require a normal distribution of the underlying process quality characteristic. However, as was previously mentioned, as a practical matter, the quality characteristics of a process are rarely normally distributed. Implementing traditional multivariate control charts on nonnormal data can generate more false alarms. Studies of multivariate EWMA

control charts have investigated properties of robustness to nonnormality [61][62]. The results indicate that multivariate EWMA control charts are robust to nonnormal data when a small weighting value is used to calculate the EWMA statistic.

2.2.2.2 Multivariate EWMA and CUSUM Charts

As mentioned previously, control limits based on the F-distribution may not yield reliable and accurate results unless the data follow the multivariate normal distribution [56]. A number of studies have been conducted to overcome the limitation posed by this parametric assumption in T^2 charts. Liu [63] proposed nonparametric control charts based on ranking depth. Data depth measures, without any distributional assumptions, how deep or how central a data point is within a data cloud. Simplicial depth and Mahalanobis depth were used to construct r, Q, and S charts, which are considered, respectively, the data depth versions of x, $\bar{x}$, and CUSUM charts. Qiu [64] proposed a nonparametric control chart based on a log-linear model. To determine the in-control joint distribution, a log-linear model was used. Then, a shift can be detected by a multivariate CUSUM procedure based on the log-linear model's estimate of the in-control distribution. Zou and Tsung [65] proposed a nonparametric control chart that incorporates a multivariate sign test with an EWMA scheme, called the multivariate sign EMWA (MSEWMA) chart, to detect process shifts. Unlike traditional MEWMA charts that use a mean vector and a covariance matrix, the MSEWMA charts use a multivariate affine-equivariant median and a corresponding transformation matrix estimated from in-control data.

2.3 Novelty Detection in Statistical Process Control

Data mining techniques have recently been integrated into control charting problems. In terms of detection, data mining techniques and control charts share a similar purpose. Hwang et al. [66] proposed multivariate control charts with artificial contrasts. They generated the out-of-control cases from independent uniform distributions, thus converting the traditional view of the control charting problem into a supervised learning technique. Sukchotrat et al. [67] used linear discriminant analysis (LDA) and k-nearest neighbors (kNN) to determine the probability of class (PoC) by making use of available out-of-control data. The PoC metric is used as a monitoring statistic and plotted over time. The control limits of PoC charts can be constructed by using bootstrapping. A PoC chart is considered a distribution-free control chart because its construction does not require any distributional assumptions. The experimental results in their study showed that when various types of nonnormal distributions were considered, the PoC charts performed better than T^2 charts.

A support vector machine (SVM) algorithm has been integrated in control charts to improve their monitoring performance when data are not normally distributed [68]. SVM-PoC control charts can be constructed by extracting the PoC values from a SVM algorithm. Bootstrapping is used to determine the control limit of a SVM-PoC chart. The study showed that the SVM-PoC chart outperformed the T^2, LDA-PoC and kNN-PoC charts under various nonnormal situations.

A k-linkage ranking (kLINK) algorithm determines the rank of a new observation relative to a set of in-control training observations. Low rankings indicate that observations are located in the dense areas of the in-control set, and high rankings indicate that the new observation is on the fringe or outside the in-control set. kLINK charts [69] calculate the Mahalanobis distance and use the linkage ranking rule to

determine a rank for a new observation. Then the rank of the new observation is used to calculate the PoC value. The control limits of the charts are determined by a user-specified α. The kLINK algorithm was compared with T^2 and ranking depth charts. The study results show that with various types of nonnormal distributions, kLINK charts provide lower type II error rates than T^2 and ranking depth charts.

2.3.1 Kernel Density Estimation

Kernel density estimation (KDE), also known as Parzen window estimation, is used to estimate the probability density of data in novelty detection as described in section 2.1.2. In addition to its use in novelty detection problems, KDE can determine a threshold (control limit) in a control chart. The control limit of the traditional T^2 control chart is accurate only with an assumption that the T^2 statistic follows an F-distribution. To relax the need for this assumption, Chou et al. [56] proposed a nonparametric approach that uses KDE to estimate the distribution of T^2 statistics. However, KDE is relatively complicated because before its full construction, it requires determination of several parameters. These include types of kernel functions, a smoothing parameter, and the number of spaced points. Moreover, the KDE-based T^2 control chart involves numerical integration to calculate the percentile value (i.e., control limit) of the estimated kernel distribution. Given n values of T^2 statistics $(T_1^2, T_2^2, \ldots, T_n^2)$ computed from the in-control observations, the distribution of the T^2 statistics can be estimated by the following kernel function:

$$\hat{f}_h(t) = \frac{1}{n}\sum_{i=1}^{n} K\left[\frac{(t - T_i^2)}{h}\right], \tag{2.15}$$

where K and h, respectively, are a kernel function and a smoothing parameter [56]. A number of kernel functions are available such as uniform, normal, triangular,

Epachenikov, quadratic, and cosines. Of these, the standard normal kernel function is most commonly used.

The control limit can be determined by a percentile of the estimated kernel distribution. That is, CL_{kernel} associated with $100 \cdot (1\text{-}\alpha)^{th}$ percentile can be calculated by

$$CL_{kernel} = \hat{f}_h(t)^{-1}(1-\alpha). \tag{2.16}$$

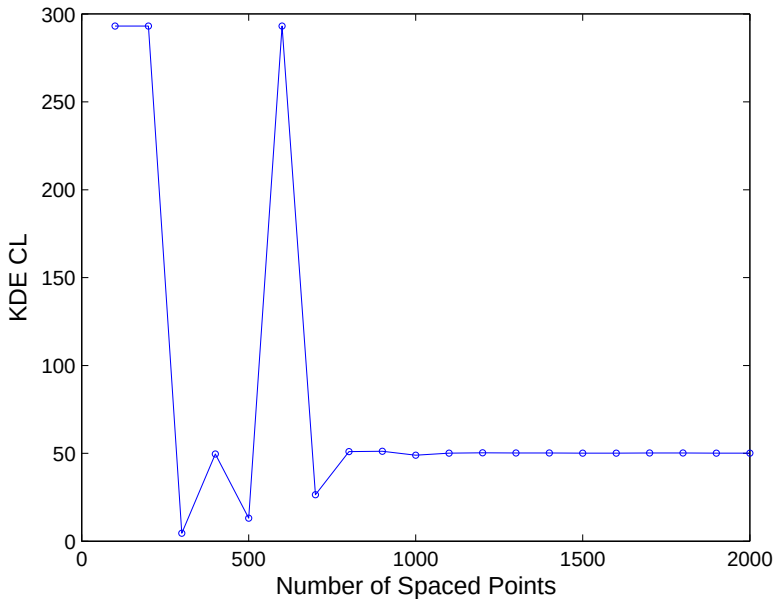


Figure 2.2 Control limits from KDE-based T^2 control charts with different numbers of spaced points.

To calculate CL_{kernel}, a proper closed form that can be found in tables of integrals may be used. However, from a practical standpoint, it may not be efficient to use tables of integrals every time one wishes to calculate control limits. The trapezoidal rule [70], one of the numerical integration methods that approximates the value of a definite integral, can be used to calculate CL_{kernel}. The degree of approximation of

the trapezoidal rule depends on the number of space points (trapezoids). If the number of space points is large, the true integration result may not differ significantly from the result derived from the trapezoidal rule. Figure 2.2 shows control limits from KDE-based T^2 control charts with different numbers of spaced points when the dataset follows the multivariate lognormal distribution. The result shows that the values of the control limits fluctuated but stabilized after 1,000 spaced points.

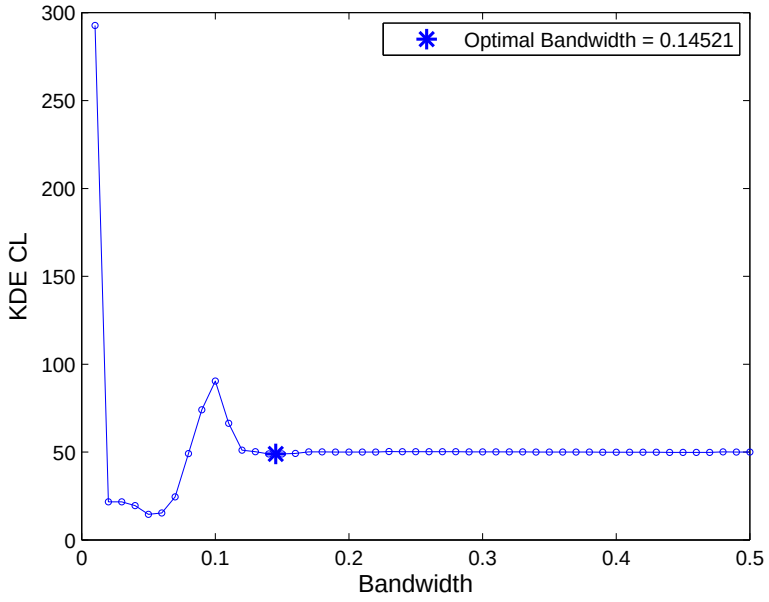


Figure 2.3 Control limits from KDE-based T^2 control charts with different bandwidths.

The accuracy of the estimates derived also depends on choosing an appropriate smoothing parameter capable of compromising between oversmoothness and undersmoothness of the estimated kernel distribution [71]. A number of methods are available to select an optimal smoothing parameter [72]. However, no consensus exists on the best method to satisfy all conditions. Figure 2.3 illustrates the control limits derived from a KDE-based T^2 control chart with different values of bandwidths

in situations in which the dataset follows a multivariate lognormal distribution. The asterisk shown in Figure 2.3 represents the optimal bandwidth obtained using a MATLAB Statistics Toolbox that uses an algorithm based on [73].

2.3.2 Existing Multivariate Control Charts based on Principal Component Analysis

Traditional multivariate control charts can be effective when only a few quality characteristics require monitoring. A large number of quality characteristics may degrade the ability of these charts to detect a shift in a process and also can lead to a multicollinearity problem [6][74]. Further, the process of calculating the T^2 statistic involves the inverse of the covariance matrix. When a large number of process characteristics are highly correlated, the covariance matrix becomes nearly singular, making it difficult to invert [53].

PCA is a useful multivariate statistical tool for handling this issue [53][75]. Integration of PCA and control chart techniques can improve the ability to detect faults early and detect changes in the covariance structure of the process variables [76]. Jackson [77] first presented the PCA technique for construction of T^2 charts that use the first k principal components (PCs). This chart is known as a T^2_{PCA} chart. Note that when all p PCs are used, T^2_{PCA} obtained by PCA process and conventional T^2 values are identical [78]. Q charts are another version of control charts based on PCA that can be constructed by using the residuals obtained from a remaining set of $p-k$ PCs [17].

Control charts based on PCA have been widely used in monitoring chemical processes. Ku et al. [74] developed PCA models that can handle static and dynamic processes by applying a multivariate autoregressive model to detect and isolate disturbances. Nijhuis et al. [79] used the first k PCs and $p-k$ PCs to develop a T^2 chart and a squared prediction error (SPE) chart, also as known the Q chart, to mon-

itor the capillary gas chromatography analysis of fatty acid composition in BCR162. They noted that when the number of PCs is underestimated, T^2_{PCA} charts, which use only first k PCs, may not be able to detect an out-of-control point caused by $p-k$ PCs. Hence, a Q chart is needed to detect shifts caused by the other PCs. Moreover, control charts based on PCA can deal with the problem of autocorrelation, which is a known hindrance to the detection performance of control charts [80]. Using the graph of the first two PCs for process monitoring is an effective way to detect an out-of-control signal in autocorrelated processes [81]. However, shifts that are orthogonal to the directions of the first two PCs cannot be detected. Monitoring the remaining PC components with any multivariate control chart or with a control chart based on residuals of the first two PCs (Q chart) is recommended.

As noted above, modern industrial processes often contain a large number of quality characteristics that are highly correlated with each other. This situation can affect the calculation of the T^2 statistic because a covariance matrix can be singular and thus cannot be inverted [76]. PCA overcomes this problem by using the first k PCs to compute the T^2 statistic. Note that because it is scale dependent, PCA should be performed based either on the covariance matrix of the normalized data or on the correlation matrix. We will briefly review the traditional PCA-based control charts presented by [82].

2.3.2.1 T^2-based PCA Charts

The T^2-based PCA chart (T^2_{PCA}) uses the first k PCs to build the control chart. The monitoring statistics of the T^2_{PCA} control chart can be obtained by using the following equation:

$$T^2_{PCA} = \sum_{i=1}^{k} \frac{z_i^2}{l_i}, \tag{2.17}$$

where the first k PCs are z_i, $i = 1, \ldots, k$, and l_i is the eigenvalue corresponding to the ith PC. This equation produces an ellipsoid in a PC coordinate system in which the directions can be represented by matrix $\mathbf{U}$ composed of eigenvectors [20]. Considering the largest eigenvalue, l_1, the direction of u_1 is represented as the major axis, and the directions of the remaining axes are defined by $u_2, u_3, \ldots, u_p$. Note that if k in Equation (2.17) is replaced with p, the T^2_{PCA} statistic obtained by the PCA process is equivalent to the traditional T^2 statistic.

Under the assumption that the data follow a multivariate normal distribution, the control limit of the T^2_{PCA} control chart can be computed as follows:

$$CL_{F-dist} = \frac{k(n+1)(n-1)}{n^2 - nk} F_{(\alpha,k,n-k)}, \tag{2.18}$$

where n and k, respectively, are the number of observations and the number of PCs retained, α is the false alarm rate, and $F_{(\alpha,k,n-k)}$ is the upper αth quantile of the F-distribution with k and $n-k$ degrees of freedom. If the T^2_{PCA} value exceeds the control limit, it can be concluded that the process is out of control.

2.3.2.2 Q Charts

The disadvantage of T^2_{PCA} control charts is that a shift in the process mean cannot be detected if the shift is orthogonal to the first k eigenvectors of covariance matrix S [81]. In this case, use of Q charts based on the residuals of PCs is suggested [17]. Q charts can be used to detect new observations that deviate from the hyperplane defined by the first k PCs [83]. The monitoring statistics of Q charts are simply the sum of the $p-k$ PCs and can be computed as follows:

$$Q = \sum_{i=k+1}^{p} z_i^2. \tag{2.19}$$

When the data follow the multivariate normal distribution, the control limit of the Q chart [17] can be computed by using the following equation:

$$CL_{Jackson} = \theta_1 \left[\frac{z_\alpha \sqrt{2\theta_2 h_0^2}}{\theta_1} + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_1^2} + 1 \right]^{1/h_0}, \tag{2.20}$$

where z_α is the upper αth quantile of the standard normal distribution,

$$\begin{aligned} h_0 &= 1 - 2\theta_1\theta_3/3\theta_2^2, \\ \theta_j &= \sum_{i=k+1}^{p} (l_i)^j. \end{aligned}$$

for j = 1,2,3

Another way to establish control limits for the Q chart is to use an approximate value based on the weighted chi-square distribution ($g\chi_h^2$) proposed by Box [84]. The g and h represent, respectively, the weight and the degrees of freedom of the weighted chi-square distribution that can be estimated by a matching moment estimation technique [85]. Consequently, estimates of g and h can be obtained by $\hat{g} = \nu/2m$ and $\hat{h} = 2m^2/\nu$, where m and ν are the sample mean and variance of the Q-statistic. The control limit based on $g\chi_h^2$ can be obtained from the following equation:

$$CL_{g\chi_h^2} = \frac{\nu}{2m} \chi^2_{2m^2/\nu,\alpha}, \tag{2.21}$$

where α is the false alarm rate.

2.3.3 Multivariate Control Charts based on One-class Classification Techniques

Conventional classification techniques usually require more than one class to train the classifier. However, collecting information on other classes (novelties) is sometimes difficult and expensive. Unlike conventional classification problems, one-class classification techniques require only information from the target class. This information can be easily collected from normal operation of the process. The purpose

of one-class classification is to separate the novelties from targets by constructing boundaries that aim to take the targets into account as inclusively as possible [26].

Support vector data description (SVDD) is one of the one-class classification techniques that generates a hypersphere to capture the target data. The goal of a SVDD algorithm is to maximize the number of target data points included in the hypersphere and to simultaneously minimize the volume of the hypershpere [34]. The kernel function can be incorporated into a SVDD algorithm to generate flexible boundaries. Commonly used kernel functions are Gaussian and polynomial kernel functions. Sun and Tsung [86] proposed kernel-distance-based multivariate control charts (K charts) that are developed from SVDD. K charts were compared with T^2 control charts, and the results show that K charts perform better than T^2 control charts for nonnormal distributions. Sukchotrat et al. [87] suggested using bootstrapping to construct the control limits for K charts.

Another approach to a one-class classification technique is k-nearest neighbors data description (kNNDD), described in section 2.1.2. The kNNDD algorithm uses a nearest neighbor algorithm to estimate the local density of the data [26]. Sukchotrat et al. [87] developed K^2 charts based on the kNNDD algorithm. The K^2 chart uses the average distance between the unknown observations and k nearest observations as monitoring statistics. The control limits of K^2 charts are established by bootstrapping. The K^2 chart was compared with the T^2 chart, and the results show that K^2 charts perform better than T^2 charts in various nonnormal situations.

Traditional control charts assume that the observations are independent over time. However, in some manufacturing processes, a serial relationship exists between observations [6]. That is, the consecutive observations of the process are correlated over time; when this occurs, it is referred to as to an autocorrelated process. An autocorrelated process can degrade the performance of traditional control charts by

decreasing the in-control average run length and increasing the false alarm rate [88]. Monitoring of the residuals has been the way SPC has traditionally addressed this issue. However, monitoring processes through the residuals leads to a model approximation in which some important information can be lost because the residuals are the difference between the fitted values and the original observations. Integrating one-class classification techniques into control charts has been used to monitor autocorrelated multivariate processes [89]. The present study has been conducted using the kNNDD algorithm as an alternate way to monitor autocorrelated multivariate processes.

2.4 Bootstrapping

2.4.1 General Description

The bootstrap technique is a simple and, therefore, attractive resampling technique initiated by Efron in 1979 [90]. The bootstrap method is considered a powerful tool that allows one to approximate the sampling distribution of a statistic for statistical inference. The overall purpose of the bootstrap method is to use the power of computing to avoid a complex derivation of an unknown distribution G. By resampling the observations from the distribution G, the empirical distribution $\hat{G}$ can be obtained. Although the bootstrap approach is considered computer-intensive, computing has become very powerful and inexpensive these days, facilitating the use of highly computational methods, like bootstrapping. The bootstrap method is often used to construct confidence regions, test hypotheses, construct prediction regions, and solve regression problems [91].

Let $X_1, X_2, \ldots, X_n$ be a random sample with n observations from an unknown distribution G. The bootstrap technique assigns probability $1/n$ to each observation

$X_1, X_2, \ldots, X_n$. By drawing a random sample with replacement from the original sample, the bootstrap technique creates a bootstrap sample $(X_1^*, X_2^*, \ldots, X_n^*)$ that follows the empirical distribution $\hat{G}$. The statistic of interest, $T_i^* = T(X_1^*, X_2^*, \ldots, X_n^*)$, can be calculated from each bootstrap sample. By repeating the bootstrap procedures for B times, B values of the statistics of interest, $T_1, T_2, \ldots, T_B$, can be obtained.

2.4.2 The Bootstrap Method in Statistical Process Control

Conventional control charts assume that the quality characteristic of the process follows a normal distribution. This is often untrue in real world problems, where violations of normality are often encountered, and the central limit theorem is insufficient to describe the normal distribution. To address the limitation posed by the distributional assumption underpinning traditional control charts, nonparametric (or distribution-free) control charts have been developed. In particular, many studies have focused on the construction of nonparametric control charts by using a bootstrap procedure. This procedure is favored because of its proven capabilities to effectively manage process data without making assumptions about their distribution. Bajgier [92] introduced a univariate control chart whose lower and upper control limits were estimated by using the bootstrap technique. However, when confronted by an unstable in-control process, Bajgier's control charts tend to generate a wide gap between the lower and upper control limits. Seppala et al. [93] proposed a subgroup bootstrap chart to compensate for the limitations of Bajgier's approach. The subgroup bootstrap chart uses residuals, which are the difference between the mean of j^{th} subgroup and each observation in j^{th} subgroup. The lower and upper control limits are determined by adding the mean of the residuals obtained by a bootstrap technique to the grand mean. Liu and Tang [94] proposed a bootstrap control chart that can monitor both independent and dependent observations. To monitor the mean of independent

processes, a general bootstrap method was used with samples of the subgroup data, and a moving block bootstrap proposed by Liu and Singh [95] was used to monitor the mean of dependent processes. Jones and Woodall [96] compared the performance of the above three bootstrap control charts in nonnormal situations and found that they did not perform significantly better than the traditional $\bar{x}$ chart in terms of in-control average run length (ARL_0). Further, Wu and Wang [97] constructed $\bar{x}$ and R charts using the bootstrap method to estimate control limits for a beta distribution with positive skewness. Their approach relied on bootstrapping the mean of each subgroup. ARL and type I errors were used as the performance measurements for these charts. The results showed that control limits determined by the bootstrap method improved the detection power of the charts.

Recently, Lio and Park [98] proposed a bootstrap control chart based on the Birnbaum-Saunders distribution. This chart performs better with data related to tensile strength and breaking stress data. Specifically, they proposed to use the parametric bootstrap technique to establish control limits for monitoring a specified percentile of the Birnbaum-Saunders distribution. They showed that their proposed parametric bootstrap method can accurately estimate the control limits for Birnbaum-Saunders percentiles. Further, Park [99] proposed median control charts whose control limits were determined via bootstrap techniques by estimating the variance of the sample median. The parametric bootstrap method was applied to construct limits for monitoring a small percentile of the Weibull distribution [100]. The performance of the charts was evaluated by in-control ARL (ARL_0) and out-of-control ARL (ARL_1) and compared with Padgett and Spurrier's Weibull control charts, which are Shewhart-type charts [101]. The results showed the bootstrap control charts for the Weibull distribution generated fewer false alarms than Pugett and Spurrier's Weibull control charts.

All of the methods discussed so far dealt with nonparametric situations in univariate processes. However, modern processes often involve a large number of process variables that are highly correlated with each other. Although univariate control charts can be applied to each process variable, this may lead to unsatisfactory results when multivariate problems are involved [54]. Polansky [102] provided a general framework for constructing control charts for both univariate and multivariate situations. The framework used the bootstrap technique to estimate a discrete distribution, used a density estimation method such as kernel density estimation to obtain a continuous distribution, and established the control limits by using a numerical integration approach.

Several studies have found increasing interest in the application of the bootstrap method to control charts. The main purpose in implementing the bootstrap technique is to find appropriate control limits when the process variables exhibit unknown or nonnormal distributions. However, as mentioned earlier, bootstrap control charts have only been used in univariate processes.

This dissertation extends the literature by developing the bootstrap approach for multivariate control charts in which the quality characteristics of a process are nonnormal (chapter 3), combining the PCA based charts with the bootstrap method (chapter 4), and modifying SVDD novelty detection to incorporate the dense region of the data (chapter 5). These approaches are fundamentally threshold development methods for novelty detection. However, chapters 3 and 4 are taken from the perspective of statistical process control.

CHAPTER 3

BOOTSTRAP-BASED T^2 MULTIVARIATE CONTROL CHARTS

As an alternative to KDE-based T^2 control chart described in Section 2.2, this chapter, develops a bootstrap-based T^2 control chart to establish the control limits of T^2 control charts when the observed process data are not normally distributed. The control limits of bootstrap-based T^2 control charts are calculated based on the percentile of T^2 statistics derived from bootstrap samples. The proposed bootstrap-based T^2 control chart is easy to implement because it requires neither specification of the parameters nor a procedure for numerical integration. The absence of these requirements makes the bootstrap-based T^2 control chart easier to use.

The remainder of this chapter is organized as follows. Section 3.2 presents simulation studies to evaluate the performance of the bootstrap-based T^2 control chart and compare it under various scenarios with traditional T^2 control charts and KDE-based T^2 control charts. Section 3.3 describes a case study undertaken to demonstrate the feasibility and effectiveness of the proposed bootstrap-based T^2 control chart in real situations. Section 3.4 contains some discussion.

3.1 The Bootstrap Percentile Approach

As discussed in chapter 2, KDE-based T^2 control charts require some effort to find the appropriate modeling parameters and to perform the numerical integration that calculates the area of the estimated kernel density. In particular, if the distribution is highly skewed (which is often the case), calculations of the area of the tail region become less accurate. To avoid these cumbersome tasks, a bootstrap approach

that can be considered an alternative to KDE (but is easier to use in practical applications) to establish the control limits for T^2 control charts when the dataset is not multivariate normally distributed. The bootstrap technique is one of the most widely used resampling methods to determine statistical estimates when the population distribution is unknown [103][90]. The bootstrap approach is more convenient than KDE as a way to establish control limits because it does not involve any modeling process in specifying the parameters.

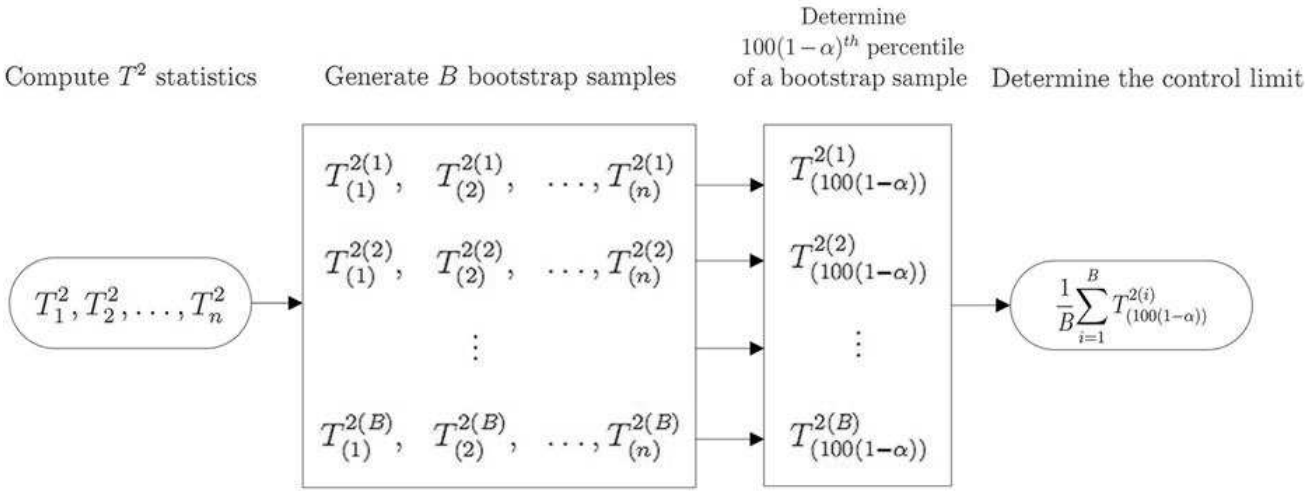


Figure 3.1 An overview of the bootstrap procedure in calculating the control limits in T^2 control charts.

Figure 3.1 illustrates an overview of the bootstrap procedure to calculate control limits, and it is summarized as follows:

1. Compute the T^2 statistics with n observations from an in-control dataset using (2.13).
2. Let $T_1^{2(i)}, T_2^{2(i)}, \ldots, T_n^{2(i)}$ be a set of n T^2 values from i^{th} bootstrap sample ($i = 1, \ldots, B$) randomly drawn from the initial T^2 statistics with replacement. In general, B is the large number (e.g., B > 1,000).

3. In each of B bootstrap samples, determine the 100·(1-α)th percentile value given a users-specified value α with a range between 0 and 1.
4. Determine the control limit by taking an average of B 100·(1-α)th percentile values ($\bar{T}^2_{100\cdot(1-\alpha)}$). Note that statistics other than the average can be used (e.g., median).
5. Use the established control limit to monitor a new observation. That is, if the monitoring statistic of a new observation exceeds $\bar{T}^2_{100\cdot(1-\alpha)}$, that specific observation is declared out of control.

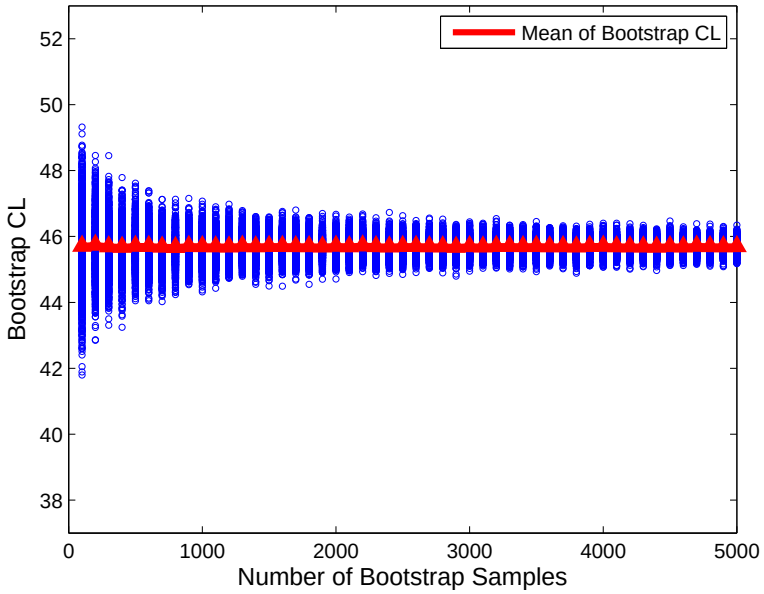


Figure 3.2 Control limits with different number of bootstrap samples.

Although the bootstrap procedure does not involve an explicit process to determine parameters, the number of bootstrap samples used may affect the determination of control limits. Figure 3.2 illustrates various bootstrap control limits as determined by different numbers of bootstrap samples from 100 to 5,000. For each number of

bootstrap samples, the control limit was calculated 1,000 times. The triangular in the figure indicates the average value of 1,000 control limits at each specified number of bootstrap samples. As expected, variability is greater when a small number of bootstrap samples are involved but stabilizes as the number of bootstrap samples increases. Determination of the appropriate number of bootstrap samples to use is not obvious. However, with reasonably large numbers of samples, the results vary little. The computational time required has been perceived as one of the disadvantages of the bootstrap technique, but this is not longer a significant issue because of the computing power currently available. Moreover, it is worth noting that the bootstrap tends to work better for statistics that are closer to being pivotal, such as the T^2 statistic. However, the bootstrap might not work so well if the process mean is the statistic for the control chart.

3.2 Simulation Study

3.2.1 Simulation Setup

Simulation studies were conducted to evaluate the performance of the proposed bootstrap-based T^2 control chart and to compare it with the traditional T^2 and KDE-based T^2 control charts. One thousand bootstrap samples ($B = 1,000$) were used in this experiment. For KDE-based T^2 control charts, the standard normal distribution was used as the kernel function.

To generate training data sets, 1,000 in-control observations ($n = 1,000$) were sampled based on multivariate normal (N), multivariate skew-normal (SN), multivariate lognormal ($LogN$), and multivariate gamma (Gam) distributions. Each dataset was characterized by three process variables. To simulate the multivariate normal, multivariate skew-normal, and multivariate gamma distributions, $\mu =$

$\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$. In the multivariate lognormal distribution, $\mu = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$ was used. Further, the following covariance matrix was used for the multivariate normal, multivariate skew-normal, and multivariate lognormal distributions:

$$\Sigma = \begin{bmatrix} 1.00 & 0.70 & 0.60 \\ 0.70 & 1.00 & 0.10 \\ 0.60 & 0.10 & 1.00 \end{bmatrix}.$$

In the multivariate skew-normal distribution, different degrees of skewness (λ) were considered so as to observe the effects of these differences on the performance of the control charts. The R package (www.r-project.org) was used to generate multivariate skew-normal data. For illustrative purposes, Figure 3.3 shows two-dimensional skew-normal distributions with degrees of skewness from zero to three. The zero degree skew-normal distribution shows the regular normal distribution without any skewness. However, this figure also shows that as the degree of skewness increases, the simulated data become more skewed. For more details about the multivariate skew-normal distribution, one can refer to Azzalini and Dalla Valle [104]. In the multivariate gamma distribution, the shape and scale of the parameters were specified as one.

3.2.2 Simulation Results

3.2.2.1 Comparison of Control Limits

Two sets of 1,000 in-control observations were generated. The first set of 1,000 in-control observations was used to determine the control limits of the T^2 control charts from the F-distribution, KDE, and bootstrap percentile approaches. The second set of 1,000 in-control observations was monitored on the control charts, which were based on the control limits established by the first set of in-control observations.

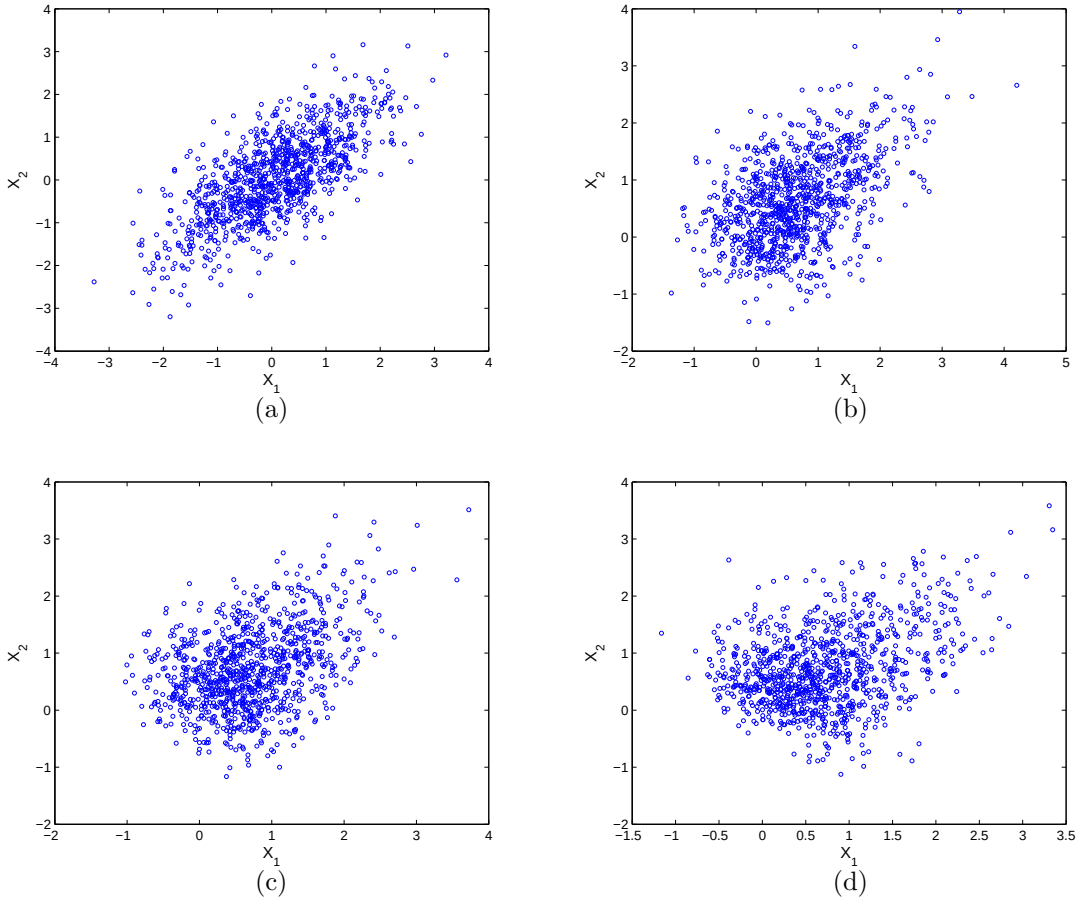


Figure 3.3 The multivariate normal and multivariate skew-normal distributions with different degrees of skewness in two dimensions: (a) normal distribution ($\lambda = 0$); (b) skew-normal distribution ($\lambda = 1$); (c) skew-normal distribution ($\lambda = 2$); (d) skew-normal distribution ($\lambda = 3$).

The control chart that produces a similar value for the actual false alarm rate and the assumed false alarm rate would be considered the better one.

Figure 3.4 displays T^2 control charts from the second set of 1,000 in-control observations that use the normal distribution in conjunction with four different degrees of skewness. The control limits were computed using the F-distribution, KDE, and bootstrap percentile approaches. The false alarm rate was specified as 0.01. As illustrated, all three approaches yielded similar control limits in the multivariate nor-

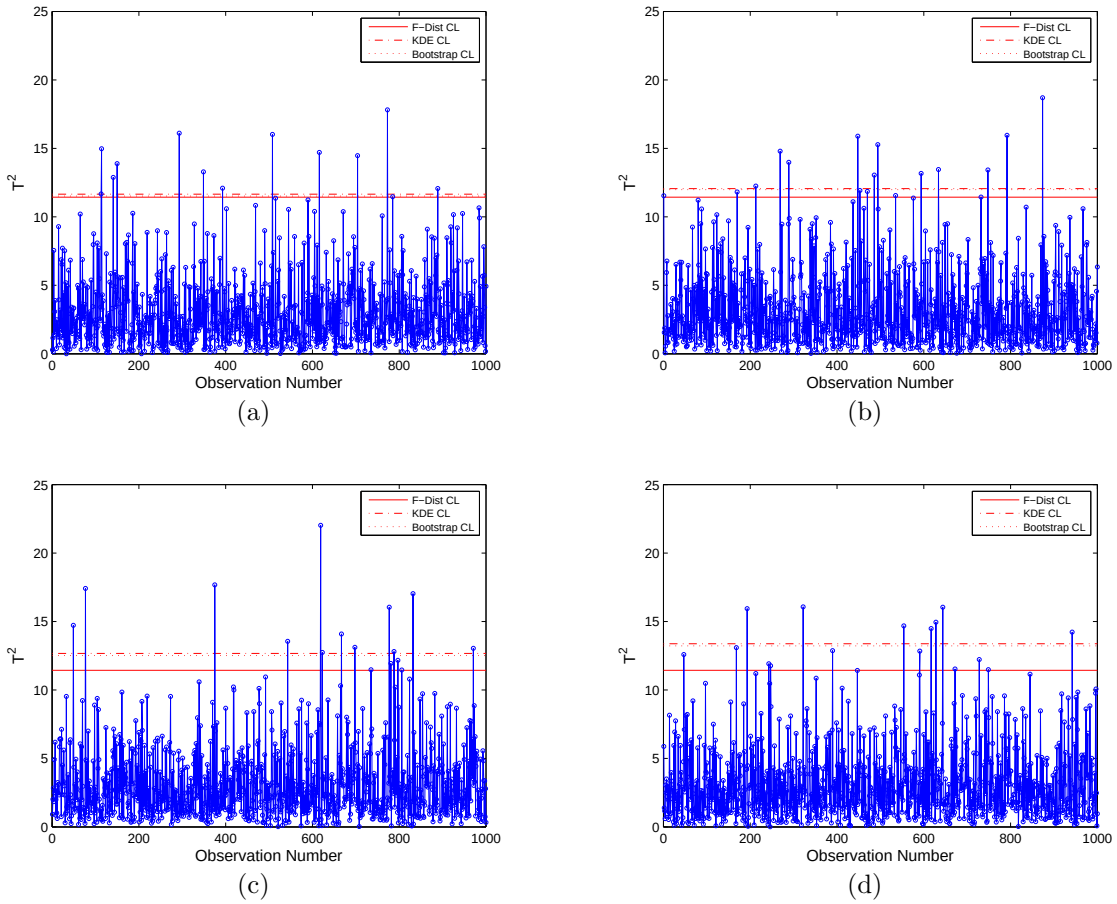


Figure 3.4 Control limits with $\alpha = 0.01$ established by the F-distribution, KDE, and the proposed bootstrap percentile under conditions of different degrees of skewness: (a) normal distribution ($\lambda = 0$); (b) skew-normal distribution ($\lambda = 1$); (c) skew-normal distribution ($\lambda = 2$); (d) skew-normal distribution ($\lambda = 3$).

mal distribution with zero skewness. As skewness (λ) increases, the control limits from the F-distribution tended to generate higher false alarm rates. However, the KDE and bootstrap percentiles controlled the assumed false alarm rates well. As can be seen from Figure 3.5, this behavior becomes much clearer in two nonnormal distributions (e.g., lognormal and gamma). In these, the false alarm rate was specified as 0.01 and generated three different control limits. The results clearly show that the F-distribution approach failed to control the assumed false alarm rate and

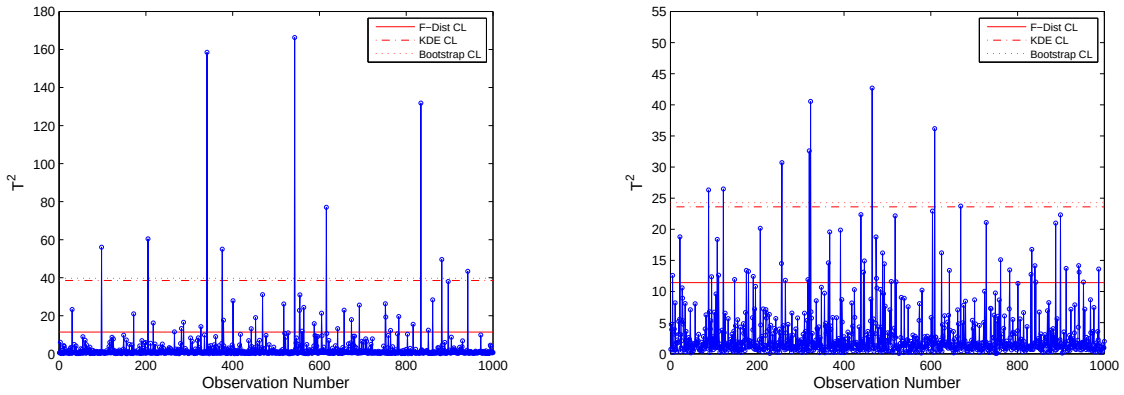


Figure 3.5 Control limits with $\alpha = 0.01$ established by the F-distribution, KDE, and the proposed bootstrap percentile on (a) multivariate lognormal distribution; (b) multivariate gamma distribution.

generated many false alarms. In contrast, the control limits determined by the KDE and bootstrap approaches produced similar values for the actual false alarm rate and for the assumed false alarm rate.

3.2.2.2 Comparison of In-Control Average Run Length

Average run length (ARL) is a performance measure that is widely used to evaluate control charts. In the present study, in-control ARL (ARL_0) was used to compare the performance of the control charts. ARL_0 is defined as the average number of observations required for the control chart to detect a change under the in-control process [105]. In this study, the average value of ARL_0 was calculated from 10,000 replications. Under the normality assumption, the actual ARL_0 of the T^2 control chart is expected to be the same as or close to the assumed ARL_0. Tables 3.1 $\sim$ 3.4 show the assumed ARL_0 values and the actual ARL_0 values as obtained by the F-distribution, KDE, and bootstrap percentile approaches in multivariate-normal situations in which different degrees of skewness were used. This figure shows

that across the different approaches the actual ARL_0 values are close to the assumed ARL_0 values when skewness is zero. However, as the degree of skewness increases, the differences between the actual ARL_0, as determined by the F-distribution and the desired ARL_0, increases. In contrast, KDE and the bootstrap percentile approaches in skew-normal situations generated similar actual and assumed ARL_0 results.

As with skew-normal situations, in the multivariate lognormal case (Table 3.5)and the multivariate gamma case (Table 3.6)the actual ARL_0 values from the KDE and bootstrap percentile approaches are close to the assumed ARL_0 values. Note that the average standard errors shown in parentheses in Tables 3.1 $\sim$ 3.6 are small enough to permit a meaningful conclusion.

Table 3.1 ARL_0 from the control limits established by using the F-distribution, KDE, and the bootstrap percentile approaches from 10,000 simulation runs based on the multivariate normal distribution (average standard errors are shown inside the parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
N	0.01	100.000	101.980	103.950	99.962
			(1.012)	(1.114)	(1.074)
	0.02	50.000	51.542	51.850	50.143
			(0.517)	(0.540)	(0.521)
	0.03	33.333	33.736	33.965	32.726
			(0.331)	(0.335)	(0.323)
	0.04	25.000	25.053	25.560	24.677
			(0.248)	(0.258)	(0.250)
	0.05	20.000	20.301	20.443	19.808
			(0.199)	(0.204)	(0.198)
	0.06	16.667	16.906	17.120	16.628
			(0.163)	(0.167)	(0.163)
	0.07	14.286	14.323	14.469	14.046
			(0.138)	(0.141)	(0.137)
	0.08	12.500	12.629	12.764	12.418
			(0.122)	(0.124)	(0.121)
	0.09	11.111	11.026	11.162	10.818
			(0.104)	(0.106)	(0.103)
	0.10	10.000	10.111	10.207	9.924
			(0.098)	(0.099)	(0.096)

Table 3.2 ARL_0 from control limits established by using the F-distribution, KDE, and bootstrap percentile approaches from 10,000 simulation runs based on the multivariate skew-normal distribution with $\lambda = 1$ (average standard errors are shown in parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
$SN(\lambda = 1)$	0.01	100.000	81.724	103.200	100.12
			(0.827)	(1.149)	(1.112)
	0.02	50.000	45.059	50.804	49.413
			(0.453)	(0.529)	(0.514)
	0.03	33.333	31.369	34.184	33.056
			(0.316)	(0.352)	(0.339)
	0.04	25.000	23.606	24.933	24.364
			(0.231)	(0.250)	(0.243)
	0.05	20.000	19.610	20.636	20.012
			(0.194)	(0.207)	(0.200)
	0.06	16.667	16.101	16.558	16.118
			(0.159)	(0.164)	(0.160)
	0.07	14.286	14.210	14.568	14.182
			(0.138)	(0.143)	(0.139)
	0.08	12.500	12.534	12.72	12.377
			(0.120)	(0.122)	(0.118)
	0.09	11.111	11.253	11.415	11.075
			(0.107)	(0.109)	(0.106)
	0.10	10.000	9.932	10.07	9.7786
			(0.096)	(0.098)	(0.095)

3.3 Case Study

The proposed bootstrap-based T^2 control chart was implemented by applying it as a case study to a real multivariate process in a power generation company. The ultimate goal of this case study was to develop an efficient monitoring and diagnostic tool for early detection of abnormal behavior and performance degradation in a power company. The dataset contains 2,000 observations collected over a period in which each observation was characterized by 18 process variables. Further, the power company's process experts confirmed that this dataset is in control and stable. To check its multivariate normality, the Royston's H test [106] was conducted. The p-

Table 3.3 ARL_0 from control limits established by using the F-distribution, KDE, and bootstrap percentile approaches from 10,000 simulation runs based on the multivariate skew-normal distribution with $\lambda = 2$ (average standard errors are shown in parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
$SN(\lambda = 2)$	0.01	100.000	71.901	101.45	98.821
			(0.719)	(1.090)	(1.051)
	0.02	50.000	41.846	51.772	50.562
			(0.423)	(0.532)	(0.520)
	0.03	33.333	29.587	33.850	32.944
			(0.294)	(0.343)	(0.331)
	0.04	25.000	23.089	25.483	24.897
			(0.228)	(0.256)	(0.250)
	0.05	20.000	19.019	20.370	19.894
			(0.185)	(0.200)	(0.195)
	0.06	16.667	16.137	16.944	16.512
			(0.157)	(0.166)	(0.163)
	0.07	14.286	13.977	14.467	14.146
			(0.136)	(0.143)	(0.139)
	0.08	12.500	12.338	12.642	12.277
			(0.120)	(0.124)	(0.120)
	0.09	11.111	11.007	11.141	10.859
			(0.106)	(0.108)	(0.105)
	0.10	10.000	10.018	10.074	9.806
			(0.096)	(0.097)	(0.094)

value, which measures the plausibility that the dataset follows multivariate normal distribution, was almost zero. This strongly indicates that this dataset does not come from the multivariate normal distribution.

Figure 3.6 shows the T^2 control charts whose control limits were estimated by the F-distribution, KDE, and proposed bootstrap approaches with a false alarm rate (α) of 0.01. As shown, the actual false alarm rates from both the KDE and bootstrap percentile approaches are 0.0095, which is similar to the assumed false alarm rate (0.01). On the other hand, the actual false alarm rate from the F-distribution is 0.052, resulting in a lower control limit and a higher false alarm rate. This demonstrates

Table 3.4 ARL_0 from the control limits established by using the F-distribution, KDE, and bootstrap percentile approaches from 10,000 simulation runs based on the multivariate skew-normal distribution with $\lambda = 3$ (average standard errors are shown in parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
$SN(\lambda = 3)$	0.01	100.000	69.605	103.410	101.050
			(0.692)	(1.106)	(1.076)
	0.02	50.000	39.696	50.037	48.928
			(0.402)	(0.591)	(0.509)
	0.03	33.333	28.982	33.824	33.02
			(0.287)	(0.347)	(0.338)
	0.04	25.000	22.744	25.183	24.668
			(0.229)	(0.256)	(0.251)
	0.05	20.000	18.712	20.090	19.612
			(0.185)	(0.197)	(0.193)
	0.06	16.667	15.961	16.834	16.404
			(0.156)	(0.164)	(0.160)
	0.07	14.286	14.064	14.661	14.273
			(0.137)	(0.143)	(0.139)
	0.08	12.500	12.214	12.479	12.193
			(0.119)	(0.121)	(0.118)
	0.09	11.111	11.360	11.553	11.254
			(0.110)	(0.112)	(0.108)
	0.10	10.000	10.078	10.143	9.899
			(0.096)	(0.097)	(0.094)

the effectiveness of the proposed bootstrap-based T^2 control chart in a real situation in which the process does not follow the multivariate normal distribution.

3.4 Discussion

This study proposed a bootstrap approach as a way to determine the control limits of a T^2 control chart when the observations do not follow a normal distribution. KDE is an existing method used to establish the control limits of T^2 control charts in nonnormal situations. It is important to emphasize again that the purpose of the present study is not to outperform the KDE approach. Rather, the bootstrap approach is an alternative to KDE-based T^2 control chart for dealing with nonnor-

Table 3.5 ARL_0 from control limits established by using the F-distribution, KDE, and bootstrap percentile approaches from 10,000 simulation runs based on the multivariate lognormal distribution (average standard errors are shown in parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
$LogN$	0.01	100.000	20.457	108.680	105.780
			(0.208)	(1.251)	(1.142)
	0.02	50.000	17.505	52.392	51.512
			(0.172)	(0.562)	(0.527)
	0.03	33.333	15.850	34.085	33.917
			(0.155)	(0.349)	(0.342)
	0.04	25.000	14.724	25.300	25.119
			(0.143)	(0.272)	(0.251)
	0.05	20.000	13.831	20.179	20.127
			(0.137)	(0.201)	(0.200)
	0.06	16.667	13.177	16.833	16.731
			(0.130)	(0.167)	(0.163)
	0.07	14.286	12.660	14.54	14.492
			(0.126)	(0.145)	(0.145)
	0.08	12.500	12.085	12.646	12.547
			(0.117)	(0.123)	(0.121)
	0.09	11.111	11.417	11.106	10.981
			(0.111)	(0.108)	(0.106)
	0.10	10.000	11.307	10.163	10.075
			(0.110)	(0.099)	(0.098)

mal situations. Nevertheless, the proposed bootstrap-based T^2 chart is a model-free approach and thus easier to implement without recourse to a strong statistical background. The simulation study showed that the proposed bootstrap-based T^2 control charts outperformed the traditional T^2 control charts in both skew-normal and non-normal cases and were comparable in ARL performance with the KDE-based T^2 control charts. With normally distributed data, all three approaches produced comparable ARL performance. This result clearly indicates that the proposed bootstrap-based control chart is efficient in both normal and nonnormal situations. Further, the proposed control chart was used to monitor a real multivariate process in a power generation company and obtained results consistent with the simulation study. The fundamental value of the present study includes the integration of the bootstrap

Table 3.6 ARL_0 from control limits established by using the F-distribution, KDE, and bootstrap percentile approaches from 10,000 simulation runs based on the multivariate gamma distribution (average standard errors are shown in parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
Gam	0.01	100.000	20.939	105.650	103.050
			(0.207)	(1.164)	(1.114)
	0.02	50.000	16.995	50.437	50.169
			(0.162)	(0.518)	(0.514)
	0.03	33.333	14.572	33.472	33.338
			(0.141)	(0.341)	(0.338)
	0.04	25.000	13.253	25.142	25.147
			(0.129)	(0.249)	(0.250)
	0.05	20.000	12.215	20.214	20.105
			(0.119)	(0.201)	(0.200)
	0.06	16.667	11.158	16.675	16.563
			(0.107)	(0.163)	(0.162)
	0.07	14.286	10.452	14.210	14.132
			(0.100)	(0.139)	(0.138)
	0.08	12.500	10.054	12.604	12.537
			(0.097)	(0.122)	(0.121)
	0.09	11.111	9.298	11.041	10.982
			(0.089)	(0.109)	(0.108)
	0.10	10.000	8.91	9.965	9.909
			(0.086)	(0.097)	(0.096)

method with traditional Hotelling's T^2 control charts to extend their applicability in nonnormal situations.

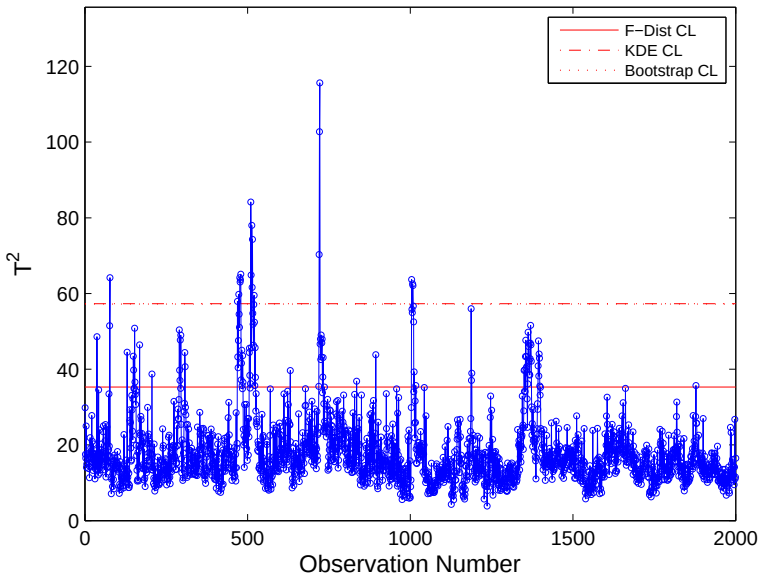


Figure 3.6 Control limits established by the F-distribution, KDE, and proposed bootstrap percentile approach on the real dataset.

CHAPTER 4

PRINCIPAL COMPONENT ANALYSIS-BASED CONTROL CHARTS FOR MULTIVARIATE NONNORMAL DISTRIBUTION

In this chapter, the PCA-based control chart methodologies described in chapter 2 are extended to handle nonnormal processes. Recall that existing PCA control charts and their applications are all based on the assumption that the control limits of the T^2_{PCA} and Q charts can be derived based on the normality assumption. The goal of the research in this chapter is to develop a nonparametric way to establish control limits in PCA control charts. To be specific, bootstrap and kernel density estimation (KDE) approaches are used to establish control limits for the T^2_{PCA} and Q charts when the data depart from the multivariate normal distribution.

The chapter is organized as follows. Section 4.1 presents nonparametric approaches such as bootstrapping and KDE to establish control limits. Section 5.2 contains simulation studies to evaluate the performance of the proposed nonparametric PCA control charts and compare them with parametric PCA control charts. Finally, Section 5.3 contains some discussion.

4.1 Proposed PCA-Based Control Charts for Multivariate Nonnormal Distributions

As explained in chapter 2, obtaining reliable control limits for the existing PCA-based control charts (T^2_{PCA} and Q charts) requires a multivariate normality assumption. In this chapter, PCA-based control charts are developed for situations where the data follow nonnormal distributions. The main idea of the proposed control

charts is to use nonparametric approaches to establish control limits of PCA-based control charts. The detail procedure will be discussed in the subsequent sections.

4.1.1 Combination of PCA and Kernel Density Estimation

When the distribution of the data is nonnormal or unknown, nonparametric density estimation can be used to determine the control limits for T^2_{PCA} and Q charts. Given n observations of the monitoring statistic $(X_1, X_2, \ldots, X_n)$ determined from in-control data, the distribution of the monitoring statistics can be estimated by the following kernel function:

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K\left[\frac{(x - X_i)}{h}\right], \tag{4.1}$$

where K is called a kernel so that $\int K dt = 1$. Usually, the standard normal distribution is selected as a kernel [25]. A bandwidth, h is a smoothing parameter, which balances the trade-off between oversmoothing and undersmoothing. A small value for h creates noise in an estimate, and a large value provides a smoother estimate. Too large a value of h may lead to oversmoothing, where important structure in the data is lost. Although a number of methods are available to determine the appropriate parameter h, no consensus exists on the best method to satisfy all conditions [72]. In this study, the normal reference rule is used: $h = (4/3)^{1/5}\sigma n^{-1/5}$ [71]. To fit long-tailed distributions and outliers, a robust estimate for σ can be calculated as $\sigma = median\{|X_i - \tilde{X}|\}/0.6745$, where $\tilde{X}$ denotes the median of the data [73]. Once the distribution of the monitoring statistics is estimated by KDE, the control limits can be determined by the $100 \cdot (1-\alpha)th$ percentile of $\hat{f}_h(x)$. In the present study the trapezoidal rule, one of the approximate techniques for calculating the definite integral [70] is used to calculate the control limit. Note that the trapezoidal rule depends on the number of spaced points. If the number of spaced points is large, the integra-

tion result may not differ much from the true integration results. In the present study, 5,000 spaced points were used to obtain accurate control limits. The experimental results to determine the appropriate parameters will be shown in Section 5.2.3.

4.1.2 Combination of PCA and Bootstrapping

The bootstrap technique is a widely used resampling method that does not require any distributional assumptions about the data [90, 103]. Bootstrapping assigns a probability of $1/n$ to each observation. Let $\{X_1, X_2, \ldots, X_n\}$ be the original sample with the underlying distribution G. The bootstrap technique generates a set of bootstrap samples by drawing B times with replacement from the original sample $\{X_1, X_2, \ldots, X_n\}$. A bootstrap sample can be denoted by $\{X_1^*, X_2^*, \ldots, X_n^*\}$. The statistic of interest can be computed from the bootstrap sample: $T_i = T(X_1^*, X_2^*, \ldots, X_n^*)$. By repeating the procedure B times, $T_1, T_2, \ldots, T_B$ can be obtained.

To construct control limits, in general, let $Y_1, Y_2, \ldots, Y_n$ with n observations be the monitoring statistics from in-control data. The monitoring statistics are drawn with replacement for B times. Therefore, the B sets of bootstrap samples are obtained as follows:

$$\begin{array}{ccccc} Y_1^{*(1)}, & Y_2^{*(1)}, \cdots, & Y_n^{*(1)} \\ Y_1^{*(2)}, & Y_2^{*(2)}, \cdots, & Y_n^{*(2)} \\ \vdots & \vdots & \vdots \\ Y_1^{*(B)}, & Y_2^{*(B)}, \cdots, & Y_n^{*(B)}. \end{array}$$

In each of B bootstrap samples, the $100 \cdot (1-\alpha)th$ percentile value is determined, where α is the user-specified value with a range between 0 and 1. The control limit can be computed by taking an average of the B percentile values, although any statistics can be used. Figure 4.1 illustrates an overview of the bootstrap procedure

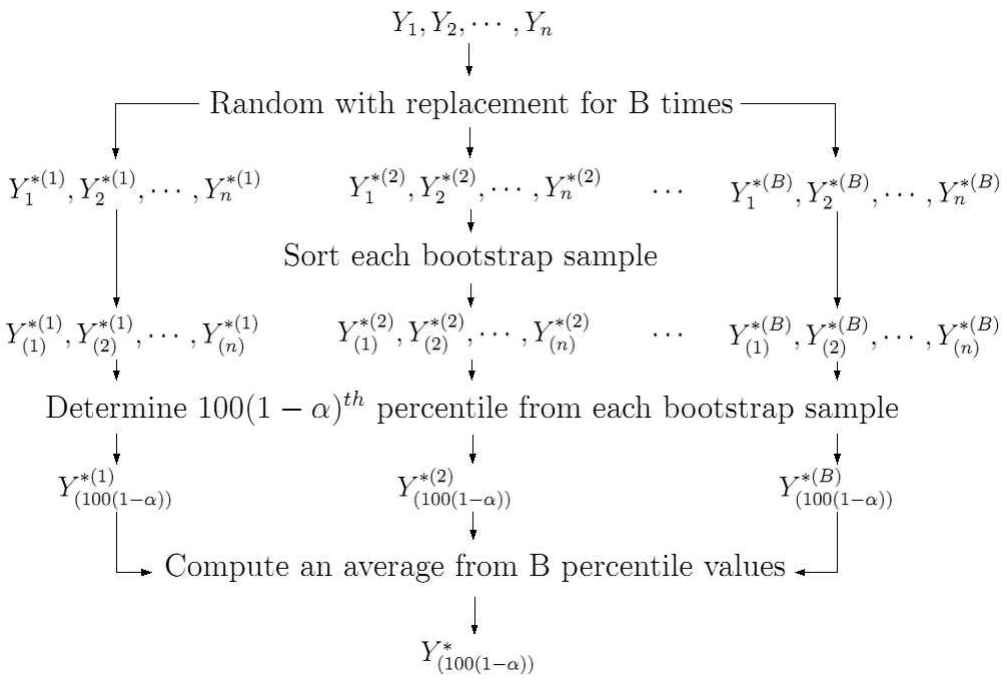


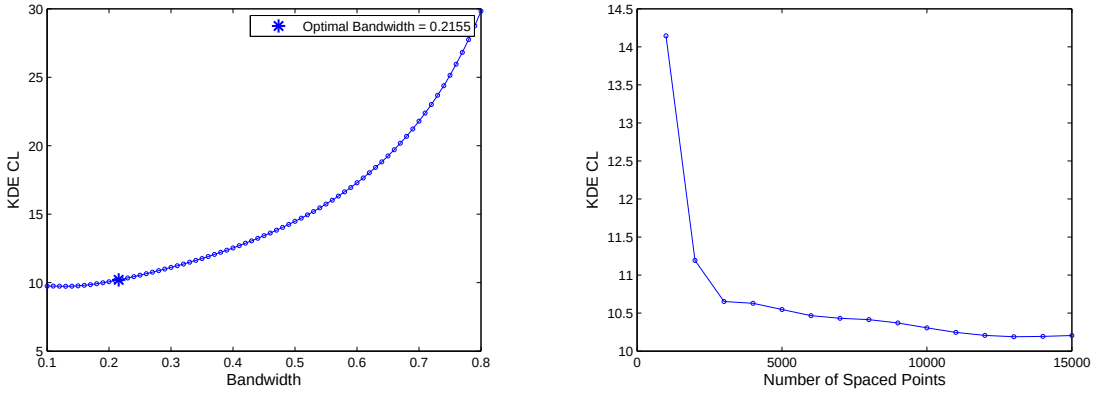
Figure 4.1 The bootstrap procedure in calculating control limits for control charts.

to calculate control limits. The established control limit can be used to monitor future observations. That is, if the new monitoring statistic exceeds the control limit, then it is declared out of control.

4.2 Simulation Study

4.2.1 Simulation Setup

Simulation studies were conducted using MATLAB (MathWorks, Natick, MA) to evaluate the performance of the proposed nonparametric PCA control charts and compare their performance to traditional parametric PCA control charts. To generate training data sets, 1,000 in-control observations (i.e., $n = 1,000$) were sampled based on multivariate normal (N), multivariate gamma (Gam), and multivariate t (t) distributions. A set of in-control observations was characterized by eight process variables with a zero mean vector. Further, a correlation matrix was estimated using



(a) (b)

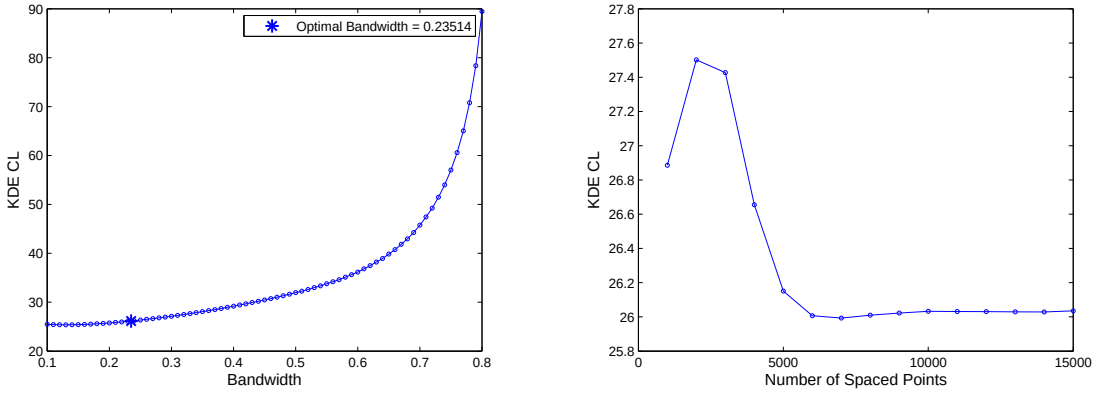
Figure 4.2 The effect of bandwidth and the number of spaced points on control limits based on multivariate normal distribution: (a) control limits from KDE approach-based T^2_{PCA} control charts with different bandwidths; (b) control limits from KDE approach-based T^2_{PCA} control charts with different numbers of spaced points.

eight physical measurements of 305 girls taken from a dataset used by [82, p. 160] to illustrate the PCA process:

$$\Sigma = \begin{bmatrix} 1.00 & 0.85 & 0.80 & 0.86 & 0.47 & 0.40 & 0.30 & 0.38 \\ 0.85 & 1.00 & 0.88 & 0.83 & 0.38 & 0.33 & 0.28 & 0.42 \\ 0.80 & 0.88 & 1.00 & 0.80 & 0.38 & 0.32 & 0.24 & 0.34 \\ 0.86 & 0.83 & 0.80 & 1.00 & 0.44 & 0.33 & 0.33 & 0.36 \\ 0.47 & 0.38 & 0.38 & 0.44 & 1.00 & 0.76 & 0.73 & 0.63 \\ 0.40 & 0.33 & 0.32 & 0.33 & 0.76 & 1.00 & 0.58 & 0.58 \\ 0.30 & 0.28 & 0.24 & 0.33 & 0.73 & 0.58 & 1.00 & 0.54 \\ 0.38 & 0.42 & 0.34 & 0.36 & 0.63 & 0.58 & 0.54 & 1.00 \end{bmatrix}.$$

The PCs that explain approximately 80 percent of the total variation were retained to calculate a T^2_{PCA} chart and the remaining variation for a Q chart.

Figure 4.2 shows the process to determine the parameters (bandwidth (h) and number of spaced points) for KDE when the dataset follows a multivariate normal



(a) (b)

Figure 4.3 The effect on control limits of bandwidth and the number of spaced points, based on multivariate gamma distribution: (a) control limits from KDE approach-based T^2_{PCA} control charts with different bandwidths; (b) control limits from KDE approach-based T^2_{PCA} control charts with different numbers of spaced points.

distribution. The asterisk in Figure 4.2(a) represents the optimal bandwidth obtained from the normal reference rule. With optimal bandwidth, the results show that the values of control limits begin to stabilize after 4,000 spaced points (Figure 4.2(b)). Similar results in the multivariate gamma distribution can also be illustrated (Figures 4.3). Using the optimal bandwidth, the values of the control limits start stabilizing after the elbow point in Figure 4.3(b).

In the bootstrap method, 1,000 bootstrap samples ($B = 1,000$) were used. For the multivariate gamma and multivariate t distributions, it was found that 1,000 in-control observations were too few to estimate a reliable control limit for the Q chart through the bootstrap method. Thus, the number of in-control observations was increased to 5,000.

4.2.2 Simulation Results

4.2.2.1 Comparison of Control Limits

Two sets of 1,000 in-control observations were generated. The first set of 1,000 in-control observations was used to determine the control limits of T^2_{PCA} control charts using the F-distribution, KDE, and bootstrap approaches. The second set of 1,000 in-control observations was used to evaluate the performance of the control charts. Figure 4.4 represents T^2_{PCA} control charts from the second set of 1,000 in-control observations. The false alarm rate was specified at 0.01. The control chart that yields similar values for both the specified false alarm rate and the actual rate would be considered the better one.

Figure 4.4(a) shows that all three approaches produced comparable control limits for the multivariate normal distribution. However, under the multivariate gamma distribution (Figure 4.4(b)) and the multivariate t distribution (Figure 4.4(c)), the control limits from the F-distribution generated higher false alarm rates. Conversely, the bootstrap and KDE approaches yielded similar values between the specified and actual false alarm rates. Note that in the case of the multivariate t distribution, the KDE approach could not obtain control limits because estimation of the control limit required too many spaced points.

Similar results were obtained from the Q charts. Figure 4.5 shows the control limits of Q charts that were constructed from the $CL_{Jackson}$, weighted χ^2 ($g\chi^2_h$), KDE, and bootstrap approaches. In Figure 4.5(a), under the multivariate normal distribution, all four approaches yielded similar control limits. The differences in control limits become clear when the parametric approaches were used and compared with nonparametric approaches for the multivariate gamma distribution (Figure 4.5(b)). The $CL_{Jackson}$ yielded the highest actual false alarm rate, and the $g\chi^2_h$, KDE, and

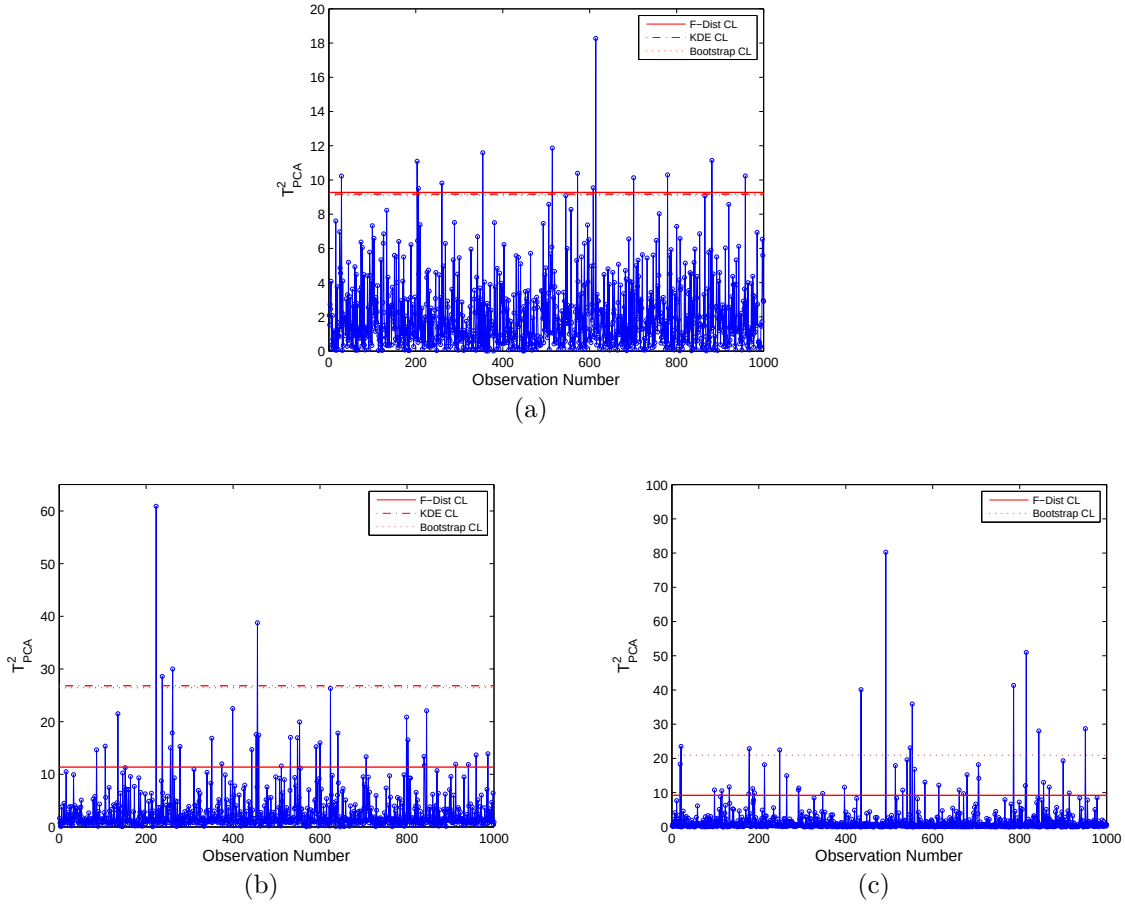
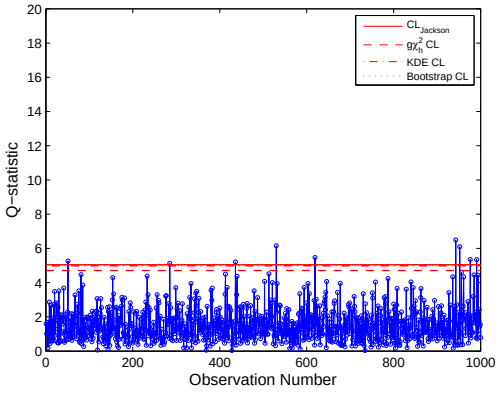


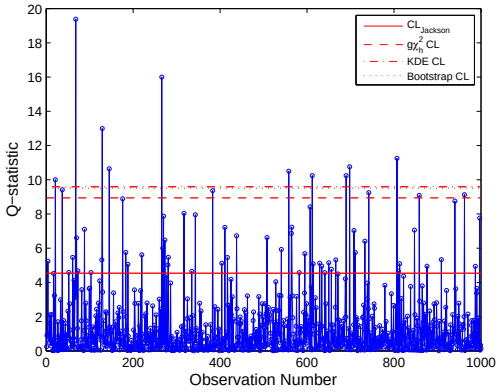
Figure 4.4 Control limits of the T^2_{PCA} control charts established by the F-distribution, KDE, and bootstrap approaches on (a) multivariate normal distribution; (b) multivariate gamma distribution; (c) multivariate t distribution ($\alpha = 0.01$).

bootstrap approaches were robust to the nonnormal distribution and able to control the false alarm rate. However, Figure 4.5(c) illustrates that the $g\chi^2_h$ approach does not control the false alarm rate well, but the bootstrap approach does in the case of the multivariate t.

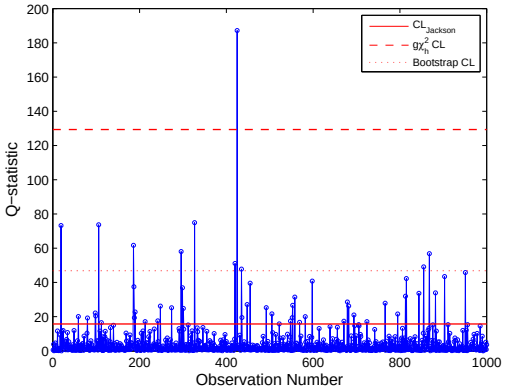
It should be noted that the KDE approach failed to determine control limits in case of the multivariate t because the distribution of the Q-statistic was highly skewed. It was found that using KDE to estimate a highly skewed distribution re-



(a)



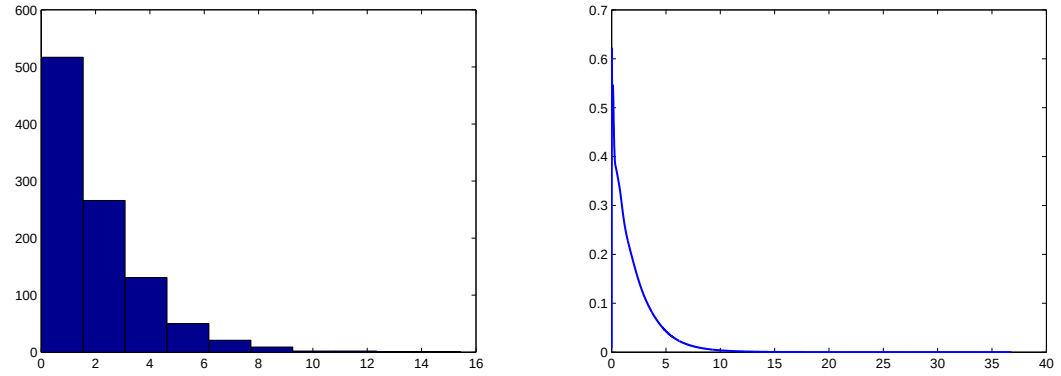
(b)



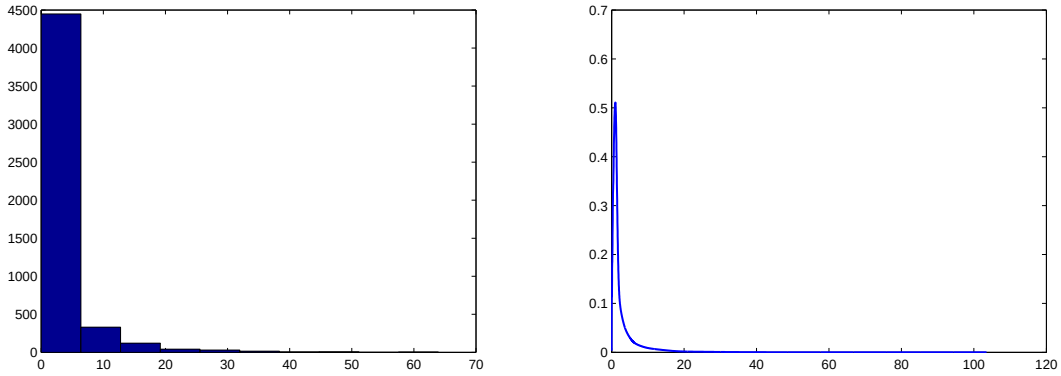
(c)

Figure 4.5 Control limits of Q control charts established by the F-distribution, weighted χ^2, KDE, and bootstrap approaches on (a) multivariate normal distribution; (b) multivariate gamma distribution; (c) multivariate t distribution ($\alpha = 0.01$).

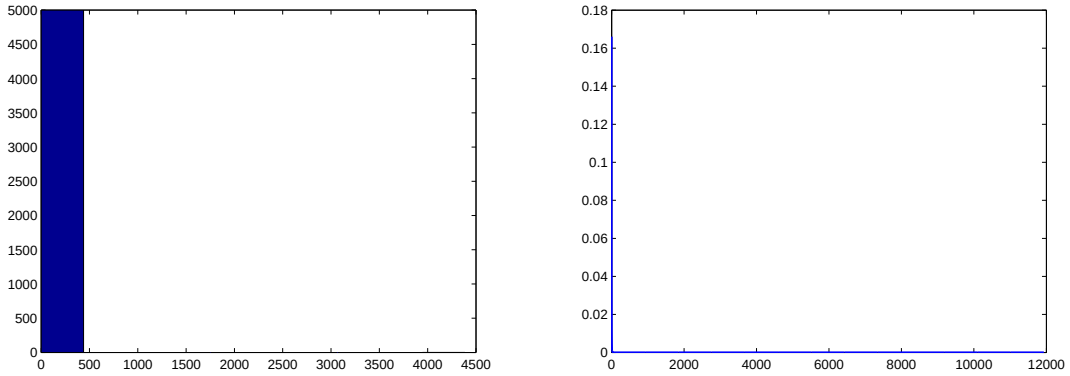
quires such a large number of spaced points that its implementation may not be practical. Figures 4.6 and 4.7 illustrate the histogram and the KDE plots of the T^2_{PCA} statistic and Q-statistic, obtained from the multivariate normal, multivariate gamma, and multivariate t distributions. The results show that the KDE approach cannot accurately estimate the distribution of the T^2_{PCA} statistic or the Q-statistic obtained from the multivariate t distribution because the distributions of these two monitoring statistics are highly skewed (Figures 4.6(c) and 4.7(c)).



(a)

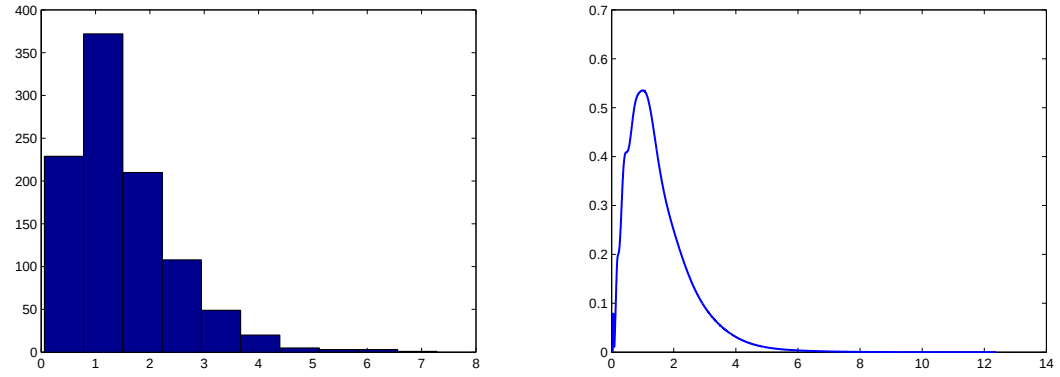


(b)

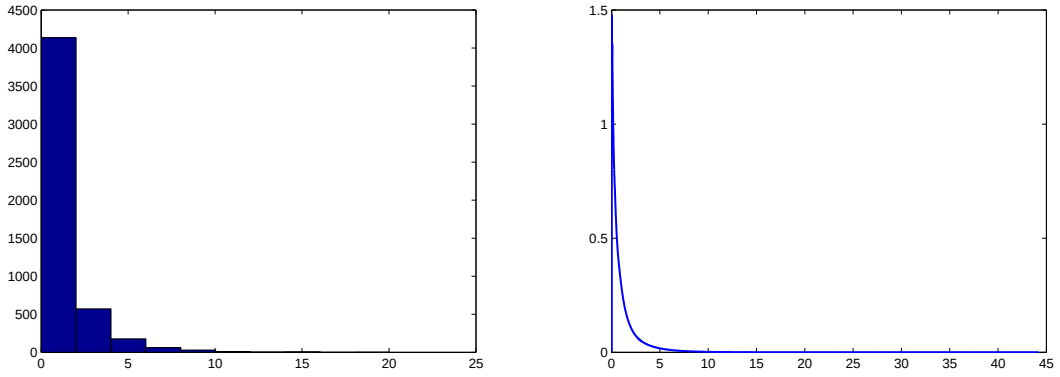


(c)

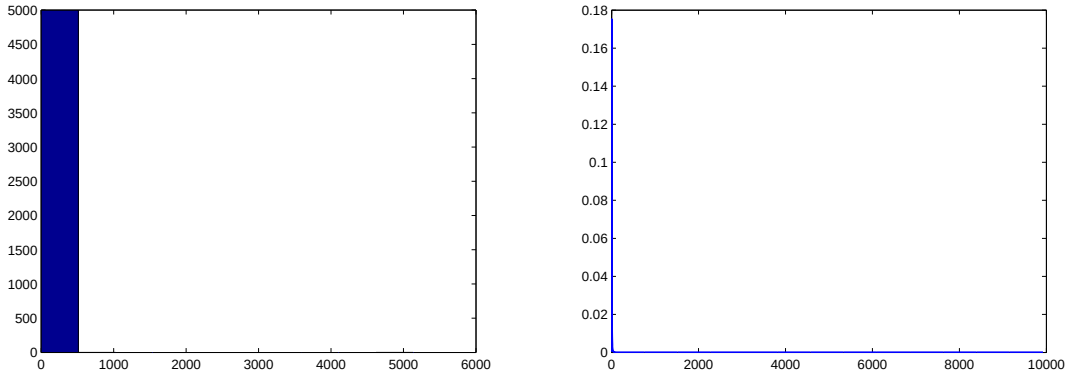
Figure 4.6 Histogram and kernel density estimation plots using T^2_{PCA} statistics calculated from (a) multivariate normal distribution; (b) multivariate gamma distribution; (c) multivariate t distribution.



(a)



(b)



(c)

Figure 4.7 Histogram and kernel density estimation plots using Q-statistics calculated from (a) multivariate normal distribution; (b) multivariate gamma distribution; (c) multivariate t distribution.

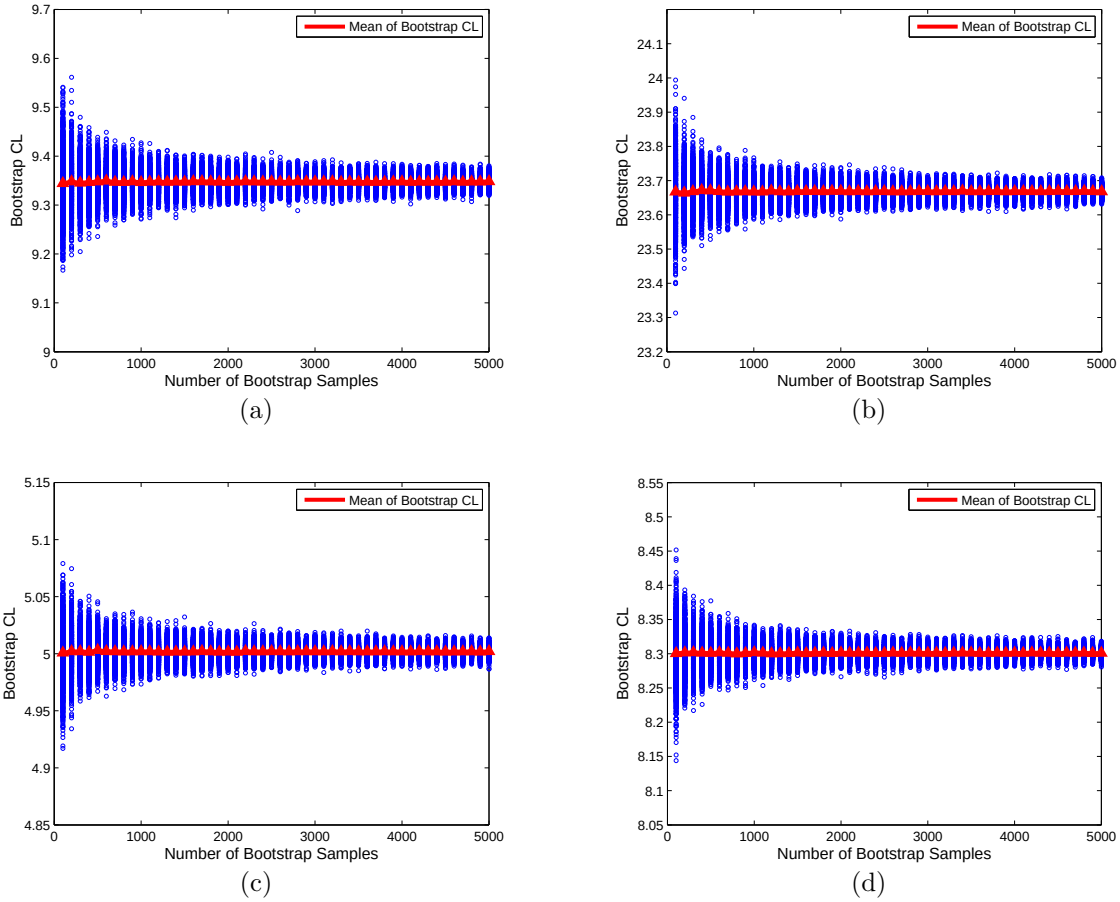


Figure 4.8 Control limits with different numbers of bootstrap samples: (a) control limits of the T^2_{PCA} chart from a normal distribution; (b) control limits of the Q chart from a normal distribution; (c) control limits of the T^2_{PCA} chart from a gamma distribution; (d) control limits of the Q chart from a gamma distribution.

The KDE approach also requires considerable effort to find the appropriate parameters, such as bandwidth and number of spaced points, and requires numerical integration to calculate the area under the estimated density. A highly skewed distribution diminishes the accuracy of the values of the control limits estimated by the KDE approach. On the other hand, the bootstrap approach, the other nonparametric approach presented in this study, overcomes this issue and establishes control limits. It involves only one parameter, the number of bootstrap samples, which may affect

the calculation of control limit values. Figure 4.8 illustrates the values of the control limits of the T^2_{PCA} and Q charts in multivariate normal and gamma distributions as determined by various numbers of bootstrap samples ranging from 100 to 5,000. The control limits were calculated 1,000 times for each bootstrap sample. The line in the middle represents the average values of 1,000 control limits at each bootstrap sample. As the number of bootstrap sample increases, the variability decreases and stabilizes. The bootstrap approach has been considered a computationally intensive technique when a large number of bootstrap samples are used. However, computing power now routinely available can compensate for this issue.

4.2.2.2 Comparison of In-Control Average Run Length

Average run length (ARL) is the most widely used performance measurement for control charts. This study focuses on the in-control ARL (ARL_0), which is defined as the average number of observations required until an out-of-control signal is detected under the in-control process [105]. In this study, the value of ARL_0 was calculated from 10,000 simulations. Under the normality assumption, the difference between the specified ARL_0 and the actual ARL_0 is expected to be close to zero. Table 4.1 shows the control limits obtained by the F-distribution, KDE, and bootstrap approaches for the T^2_{PCA} control chart under the multivariate normal distribution situation. The results show that all three approaches yielded an actual ARL_0 close to the specified ARL_0. As in the cases of multivariate gamma and multivariate t distribution, the actual ARL_0 values from the bootstrap approach are near the specified ARL_0 values (Table 4.2, Table 4.3). However, the actual ARL_0 values obtained from the F-distribution and the specified ARL_0 values are different under nonnormal situations. As noted earlier, the actual ARL_0 values cannot be obtained for the

t distribution because of the large number of spaced points required for a skewed distribution.

Similar to the T^2_{PCA} control chart, for the Q charts ARL_0 was compared for weighted χ^2 ($g\chi^2_h$), KDE, and bootstrap. Table 4.4 shows the results of different control limits under the multivariate normal distribution. The results show that across the different approaches, the actual ARL_0 values are close to the specified ARL_0 values. However, the $g\chi^2_h$ method tends to yield an actual ARL_0 that is lower than the specified ARL_0. As for the gamma distribution, Table 4.5 shows that the actual ARL_0 values from the KDE and bootstrap approaches are close to the assumed ARL_0 values. It is interesting to see that the $g\chi^2_h$ approach performs nearly as well for ARL_0 as KDE and bootstrapping. However, the $CL_{Jackson}$ approach produced actual ARL_0 values that are too small. This will lead to many false alarms. Table 4.6 shows the results of the multivariate t distribution. It can be observed that the $CL_{Jackson}$ and $g\chi^2_h$ methods produced an actual ARL_0 unlike the specified ARL_0, but the bootstrap method produced ARL_0 values similar to the actual and specified cases.

4.3 Discussion

Nonparametric PCA control charts were presented that can be used effectively in many modern processes when a number of highly correlated quality characteristics are present and multivariate normality cannot be assumed. Two nonparametric approaches were proposed – KDE and bootstrapping – to determine control limits for T^2_{PCA} and Q charts in nonnormal situations. The simulation showed that the proposed nonparametric PCA control charts performed better than traditional parametric PCA control charts in terms of ARL_0 performance.

It was found from the simulation that the KDE approach cannot produce accurate control limits when the monitoring statistics are highly skewed. This approach fails in this situation because of the large number of spaced points required to estimate the tail area of the distribution. When the monitoring statistics are skewed, the bootstrap approach is more effective for constructing control limits.

Both KDE and bootstrapping are nonparametric approaches. However, the actual implementation of KDE is relatively complicated because it requires determination of several parameters (smoothing, number of spaced points, kernel types) for its full construction. In contrast, bootstrapping is a model-free approach that does not require determination of many parameters and is easy to implement.

Table 4.1 ARL_0 from the T^2_{PCA} chart using control limits established by using the F-distribution, KDE, and bootstrap approaches from 10,000 simulation runs based on multivariate normal distribution (average standard errors are shown inside the parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
N	0.01	100.000	101.790	105.510	101.110
			(1.023)	(1.139)	(1.082)
	0.02	50.000	50.835	51.412	50.010
			(0.512)	(0.533)	(0.518)
	0.03	33.333	34.172	34.323	33.334
			(0.336)	(0.342)	(0.332)
	0.04	25.000	25.354	25.526	24.887
			(0.250)	(0.253)	(0.247)
	0.05	20.000	20.436	20.483	19.966
			(0.198)	(0.198)	(0.194)
	0.06	16.667	16.732	16.881	16.448
			(0.165)	(0.167)	(0.163)
	0.07	14.286	14.262	14.455	14.115
			(0.138)	(0.140)	(0.137)
	0.08	12.500	12.613	12.797	12.494
			(0.119)	(0.122)	(0.119)
	0.09	11.111	11.134	11.230	10.994
			(0.105)	(0.107)	(0.104)
	0.10	10.000	10.176	10.296	10.066
			(0.098)	(0.099)	(0.097)

Table 4.2 ARL_0 from the T^2_{PCA} chart using control limits established by using the F-distribution, KDE, and bootstrap approaches from 10,000 simulation runs based on multivariate gamma distribution (average standard errors are shown inside the parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
Gam	0.01	100.000	19.728	102.810	102.020
			(0.194)	(1.051)	(1.040)
	0.02	50.000	16.127	50.474	50.315
			(0.157)	(0.498)	(0.498)
	0.03	33.333	13.869	33.100	32.987
			(0.133)	(0.323)	(0.320)
	0.04	25.000	12.297	24.716	24.690
			(0.118)	(0.245)	(0.245)
	0.05	20.000	11.443	20.148	20.104
			(0.110)	(0.198)	(0.197)
	0.06	16.667	10.938	16.826	16.769
			(0.103)	(0.161)	(0.160)
	0.07	14.286	10.186	14.498	14.453
			(0.097)	(0.139)	(0.139)
	0.08	12.500	9.742	12.767	12.740
			(0.092)	(0.124)	(0.124)
	0.09	11.111	9.156	11.156	11.121
			(0.086)	(0.106)	(0.106)
	0.10	10.000	8.873	10.143	10.111
			(0.084)	(0.096)	(0.096)

Table 4.3 ARL_0 from the T^2_{PCA} chart using control limits established by using the F-distribution, KDE, and bootstrap approaches from 10,000 simulation runs based on multivariate t distribution (average standard errors are shown inside the parentheses)

Case	α	Desired ARL_0	F-dist	KDE	Bootstrap
t	0.01	100.000	32.815	N/A	102.94
			(0.592)	(N/A)	(1.034)
	0.02	50.000	27.563	N/A	50.486
			(0.600)	(N/A)	(0.497)
	0.03	33.333	22.811	N/A	32.937
			(0.408)	(N/A)	(0.329)
	0.04	25.000	21.106	N/A	25.305
			(0.512)	(N/A)	(0.245)
	0.05	20.000	19.367	N/A	19.962
			(0.487)	(N/A)	(0.200)
	0.06	16.667	18.067	N/A	16.654
			(0.479)	(N/A)	(0.160)
	0.07	14.286	16.321	N/A	14.334
			(0.327)	(N/A)	(0.138)
	0.08	12.500	14.726	N/A	12.499
			(0.249)	(N/A)	(0.118)
	0.09	11.111	14.558	N/A	11.053
			(0.383)	(N/A)	(0.104)
	0.10	10.000	13.596	N/A	10.191
			(0.285)	(N/A)	(0.098)

Table 4.4 ARL_0 from the Q chart using control limits established by using the $CL_{Jackson}$, weighted χ^2 ($g\chi^2_h$), KDE, and bootstrap approaches from 10,000 simulation runs based on the multivariate normal distribution (average standard errors are shown inside the parentheses)

Case	α	Desired ARL_0	$CL_{Jackson}$	$g\chi^2_h$	KDE	Bootstrap
N	0.01	100.000	113.970	83.141	103.990	99.455
			(1.150)	(0.857)	(1.110)	(1.053)
	0.02	50.000	53.689	44.229	50.304	48.379
			(0.543)	(0.453)	(0.533)	(0.512)
	0.03	33.333	34.810	30.755	33.428	32.355
			(0.343)	(0.307)	(0.337)	(0.326)
	0.04	25.000	25.886	23.520	25.073	24.161
			(0.256)	(0.234)	(0.251)	(0.240)
	0.05	20.000	20.549	19.258	20.127	19.465
			(0.202)	(0.190)	(0.200)	(0.194)
	0.06	16.667	17.184	16.415	16.902	16.325
			(0.169)	(0.163)	(0.166)	(0.161)
	0.07	14.286	14.486	14.063	14.437	13.954
			(0.140)	(0.137)	(0.140)	(0.135)
	0.08	12.500	12.551	12.388	12.551	12.121
			(0.121)	(0.120)	(0.122)	(0.117)
	0.09	11.111	11.185	11.128	11.190	10.833
			(0.107)	(0.106)	(0.107)	(0.104)
	0.10	10.000	10.046	10.066	10.116	9.788
			(0.096)	(0.098)	(0.098)	(0.095)

Table 4.5 ARL_0 from the Q chart using control limits established by using the $CL_{Jackson}$, weighted χ^2 ($g\chi^2_h$), KDE, and bootstrap approaches from 10,000 simulation runs based on the multivariate gamma distribution (average standard errors are shown inside the parentheses)

Case	α	Desired ARL_0	$CL_{Jackson}$	$g\chi^2_h$	KDE	Bootstrap
Gam	0.01	100.000	18.918	92.173	99.087	97.722
			(0.188)	(0.938)	(1.005)	(0.990)
	0.02	50.000	14.437	52.652	49.363	49.009
			(0.140)	(0.524)	(0.492)	(0.488)
	0.03	33.333	11.992	37.929	33.437	33.161
			(0.116)	(0.375)	(0.333)	(0.330)
	0.04	25.000	10.575	28.850	24.683	24.542
			(0.101)	(0.286)	(0.242)	(0.241)
	0.05	20.000	9.691	23.289	19.735	19.610
			(0.092)	(0.227)	(0.192)	(0.191)
	0.06	16.667	8.814	19.754	16.565	16.489
			(0.082)	(0.192)	(0.161)	(0.160)
	0.07	14.286	8.257	16.873	14.235	14.170
			(0.079)	(0.164)	(0.139)	(0.139)
	0.08	12.500	7.749	14.913	12.479	12.404
			(0.072)	(0.146)	(0.121)	(0.120)
	0.09	11.111	7.360	13.342	11.258	11.188
			(0.069)	(0.128)	(0.108)	(0.107)
	0.10	10.000	6.943	12.106	10.144	10.085
			(0.065)	(0.117)	(0.098)	(0.097)

Table 4.6 ARL_0 from Q chart using control limits established by using the $CL_{Jackson}$, weighted χ^2 ($g\chi^2_h$), KDE, and bootstrap approaches from 10,000 simulation runs based on the multivariate t distribution (average standard errors are shown inside the parentheses)

Case	α	Desired ARL_0	$CL_{Jackson}$	$g\chi^2_h$	KDE	Bootstrap
t	0.01	100.000	21.434	258.240	N/A	98.890
			(0.208)	(2.969)	(N/A)	(0.992)
	0.02	50.000	18.041	133.340	N/A	49.489
			(0.176)	(1.409)	(N/A)	(0.486)
	0.03	33.333	15.818	82.174	N/A	33.273
			(0.157)	(0.857)	(N/A)	(0.330)
	0.04	25.000	14.777	56.243	N/A	25.416
			(0.146)	(0.578)	(N/A)	(0.251)
	0.05	20.000	13.413	39.639	N/A	19.951
			(0.130)	(0.406)	(N/A)	(0.193)
	0.06	16.667	12.677	29.619	N/A	16.838
			(0.126)	(0.311)	(N/A)	(0.165)
	0.07	14.286	11.859	22.369	N/A	14.273
			(0.116)	(0.242)	(N/A)	(0.138)
	0.08	12.500	11.123	17.515	N/A	12.267
			(0.107)	(0.191)	(N/A)	(0.118)
	0.09	11.111	10.645	13.948	N/A	11.027
			(0.100)	(0.155)	(N/A)	(0.103)
	0.10	10.000	10.283	11.462	N/A	10.029
			(0.100)	(0.132)	(N/A)	(0.097)

CHAPTER 5

DENSITY-FOCUSED SUPPORT VECTOR DATA DESCRIPTION METHOD

Support vector data description (SVDD) method described in chapter 2 has gained much attention from many researchers. However, traditional SVDD aims to form the boundary to capture as many observations in the target class as possible, but fails to take into account the density of the data. Logically, the data points near dense regions should be considered as targets while extreme points should be considered outliers. In this chapter, a density-focused SVDD (DFSVDD) method is proposed. The goal of this DFSVDD to take into account the density of the data when constructing the boundary. Specifically, there are two distance measures combined in this method. The first distance measure is the original kernel distance calculated by the SVDD method. The second measure relies on the support vectors, but considers how close each support vector is to the observations. This enables a measure of density because support vectors near a dense region will be close to more observations. Overall, the kernel distance is used to identify shape and then the density distance measure is used to give higher weight to data points in more dense regions.

As mentioned in chapter 2, the misclassification error rate for targets can be adjusted by changing the parameter C. By using a smaller C value, one can reject more target observations. Given the same values of the parameters S and C, Figure 5.1 displays the boundaries of SVDD and DFSVDD. Both were constructed from 200 observations generated by the multivariate gamma distribution. Here the gamma distribution is chosen because it contains both low and high density regions. Figure

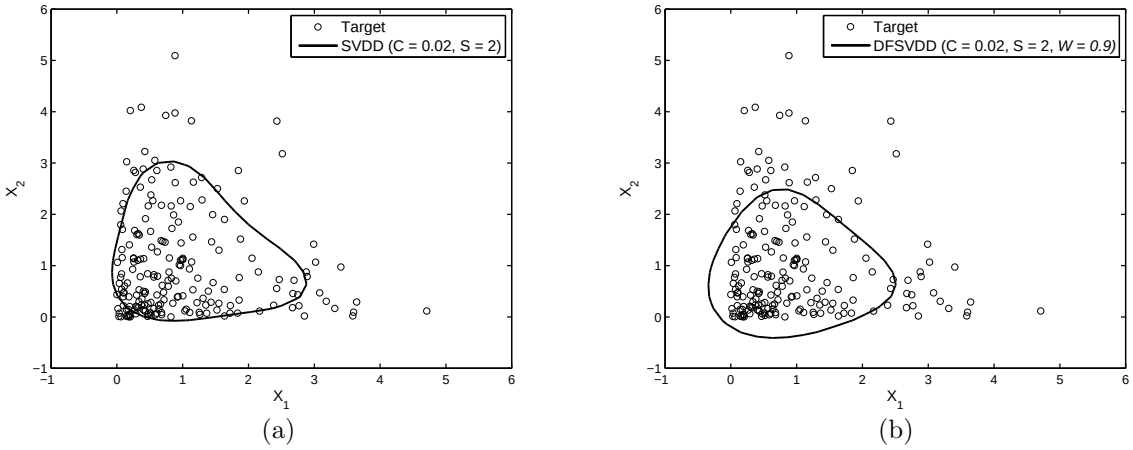


Figure 5.1 Boundaries of (a) SVDD and (b) DFSVDD.

5.1(a) shows that for SVDD with $C = 0.02$ some portions of the target observations that are located on the edge of high density region are located *outside* of the control boundary. Intuitively, the data points that are in the region of high density should not be considered novelties. To address this limitation of SVDD, DFSVDD, shown in Figure 5.1(b), establishes the boundary by taking into account the density of the data. It can be seen that the data points in the high density remain within the boundary of DFSVDD. The detail description of DFSVDD is discussed in the next section.

5.1 Density-focused SVDD (DFSVDD)

Conventional SVDD constructs a hypersphere to include the data and uses this hypersphere to classify a testing observation as either a target or a novelty. Data in real-world applications may not be evenly scattered because different degrees of density in each region can be represented. Data points that are in regions of low density are more likely to be novelties because they are remotely located from their neighbors. However, conventional SVDD disregards the density of the data when con-

structing its boundary. To address this issue, the present study proposes DFSVDD, which utilizes the support vectors (SVs) to measure the denseness of the data. The main idea of DFSVDD is to combine conventional SVDD with the density measure to create a new novelty score. The boundary from DFSVDD can be determined by solving the following optimization problem:

$$\text{Maximize } \sum_i \alpha_i K(\mathbf{x}_i, \mathbf{x}_j) - \sum_{ij} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j). \tag{5.1}$$

with the constraints:

$$0 \leq \alpha_i \leq C \tag{5.2}$$

$$\sum_{i=1}^{N} \alpha_i = 1, \tag{5.3}$$

where $i = 1, 2, \ldots, N$. The solution to the optimization yields $\alpha_1, \alpha_2, \ldots, \alpha_N$, one for each of the training observations. The observations that produce $0 < \alpha_i < C$ are the SVs on the boundary (SV^{bnd}). The distance from SV^{bnd} to the training observations is calculated by the following kernel distance:

$$\|\phi(SV_j^{bnd}) - \phi(\mathbf{x}_i)\|^2 = K(SV_j^{bnd}, SV_j^{bnd}) - 2K(SV_j^{bnd}, \mathbf{x}_i) + K(\mathbf{x}_i, \mathbf{x}_i), \tag{5.4}$$

where $j = 1, 2, \ldots, n$ indexes the SV^{bnd} observations and $i = 1, 2, \ldots, N$ indexes the training observations. This kernel distance can be used to quantify the density of the data near the SV^{bnd}. If there are many data points located near SV_i^{bnd}, the total distances to SV_i^{bnd} becomes smaller, indicating that the SV_i^{bnd} is near a region of high density. After calculating the kernel distances from each SV_i^{bnd} to all observations, these distances are normalized by using the following equation:

$$\alpha_j^{density} = \frac{\sum_i \frac{1}{\|\phi(SV_j^{bnd}) - \phi(\mathbf{x}_i)\|^2}}{\sum_j \sum_i \frac{1}{\|\phi(SV_j^{bnd}) - \phi(\mathbf{x}_i)\|^2}} \tag{5.5}$$

where $\sum_j \alpha_j^{density} = 1$. Division by zero is possible when $\mathbf{x}_i$ is located on SV^{bnd}, yielding a kernel distance of zero. To avoid this situation, we assign that data point a distance equal to the smallest kernel distance among other observations. We replace α_j in Equation (2.12) with $\alpha_j^{density}$ to obtain the following density-focused distance measure:

$$D^2_{density} = K(\mathbf{z}, \mathbf{z}) - 2\sum_i \alpha_i^{density} K(\mathbf{z}, \mathbf{x}_i) + \sum_{ij} \alpha_i^{density} \alpha_j^{density} K(\mathbf{x}_i, \mathbf{x}_j). \tag{5.6}$$

Finally, a hybrid measure combines $D^2_{density}$ with D^2 (Equation (2.12)) is obtained by the following equation:

$$H = W \cdot D^2 + (1 - W) \cdot D^2_{density}, \tag{5.7}$$

such that $W \in (0, 1)$. When W is set to 1, the proposed method becomes the conventional SVDD.

To examine the effects of parameters in DFSVDD, we constructed the boundaries with various values of C, S, and W from 200 target observations that follow a multivariate gamma distribution. The empirical threshold was used to determine the decision boundaries of DFSVDD by setting to reject approximately 10% of the target observations. From left to right column, Figure 5.2 shows that the boundaries of DFSVDD, given the same C, became smoother when S was increased because the number of SVs used to describe the boundary decreases [34]. Figure 5.2 also displays (from top to bottom) when the values of S were fixed, the boundaries of DFSVDD exhibited less complexity and became closer to a hypersphere when the parameter C was decreased. Further, DFSVDD consists of two pieces of information combined using the additional parameter W as shown in Equation (5.7). The values of the parameter W equal to 0.85 and 0.95 were used. When W is set equal to 0.95, it means that DFSVDD considers less information in the dense regions than smaller W.

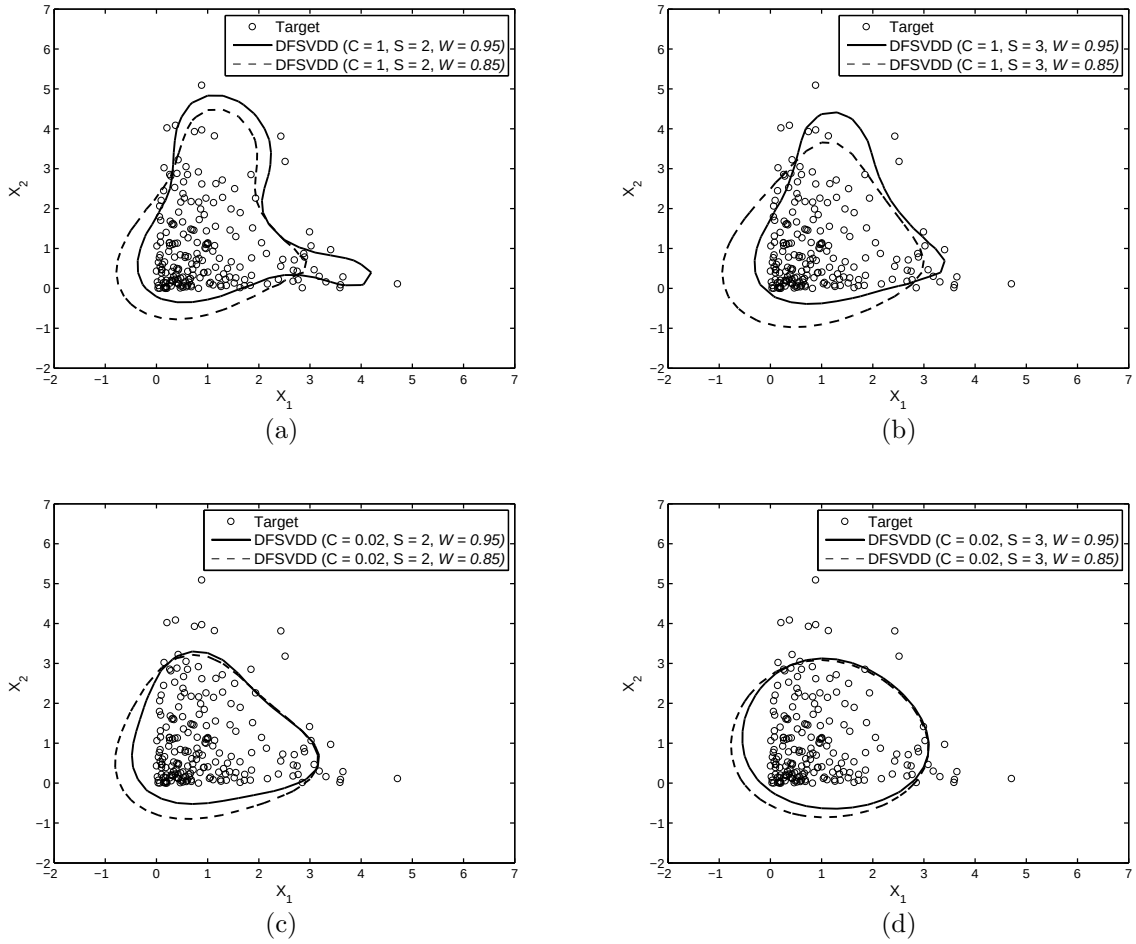


Figure 5.2 Boundaries of DFSVDD obtained from different values of the parameters and weighting factors $W = 0.85$ and 0.95: (a) $C = 1$ and $S = 2$; (b) $C = 1$ and $S = 3$; (c) $C = 0.02$ and $S = 2$; (d) $C = 0.02$ and $S = 3$.

Therefore, when the value of W was set at 0.85 as compared to higher value of $W = 0.95$, the boundaries extended the shape in the direction of the high density region, as illustrated in Figure 5.2.

5.2 Simulation Study

5.2.1 Simulation Setup

Simulation studies were conducted with MATLAB (MathWorks, Natick, MA) to examine the performance of the proposed density-focused support vector data description (DFSVDD) and to compare it with the traditional support vector data description (SVDD). A set of 200 training observations was generated from a multivariate gamma distribution. For a testing set, 200 targets and 40 novelties were generated based on the same distribution as the training set. Note that novelties were created by adding one standard deviation to the mean from the set of training observations.

The parameter C, used to control the volume of the hypersphere, was specified as 0.02, 0.04, 0.1, and 1 for both SVDD and DFSVDD. The parameter S in the Gaussian kernel function, used to control the complexity of the boundary, was set to 1, 2, 3, and 4 for both methods. Furthermore, the additional parameter W, a weighting factor to combine two distances, in the DFSVDD was set at 0.7, 0.8, 0.9, 0.95, 0.97, 0.98, and 0.99 to represent the different weighting of information in the dense regions. To determine a decision boundary for the proposed method, an empirical threshold was used. However, a bootstrap approach [90][103] can be an alternate way to estimate a threshold for DFSVDD. Note that DDtools [107] was used to perform the traditional SVDD method.

5.2.2 Performance Measurement

A receiver operating characteristic (ROC) curve [108] is a well-known graphical approach designed to evaluate the performance of novelty detection. The ROC graph

displays a plot between the true positive rate (TPR) and false positive rate (FPR) along the y-axis and x-axis, respectively. The TPR and FPR are defined as follows:

$$\text{TPR} = \frac{\text{Targets correctly classified}}{\text{Total number of targets}}, \tag{5.8}$$

$$\text{FPR} = \frac{\text{Novelties incorrectly classified}}{\text{Total number of novelties}}. \tag{5.9}$$

In the first experiment, the area under the ROC curve (AUC) is used as the standard approach to compare performances of different classifiers [109]. A value of AUC can be obtained by integrating the values of TPR and FPR. The result of the integration can yield any value from 0 to 1. A classifier with a larger value of AUC indicates better performance.

In the second experiment, we measured the performance between traditional SVDD and the proposed DFSVDD in terms of the actual TPR value. This experiment is required in order to examine the boundaries between SVDD and DFSVDD when new observations occur around the edge of a multivariate gamma distribution. We generated 200 observations that follow the multivariate gamma distribution as a training set. A testing set containing 100 observations was generated from a multivariate normal distribution with a mean and a standard deviation set at 0 and 0.5, respectively. As mentioned, this testing set aims to focus on the dense region in the multivariate gamma distribution. The parameter C was set to 0.025, 0.05, 0.1, 0.4, 0.5, 0.7, and 0.9 for both SVDD and DFSVDD. The parameter S was set to 2 for both methods. For DFSVDD, the additional parameter W was set equal to 0.9. The actual value of TPR is used as the performance measurement. The method that provides a larger TPR is considered better.

5.2.3 Simulation Results

5.2.3.1 Detecting Power

In this study, the artificial data were generated from the multivariate gamma distribution, which can represent dense and sparse regions in the data. The average AUC was used as the performance measure. The average value of AUC was evaluated from 100 simulations. The method that yields a larger average AUC would be considered the better one.

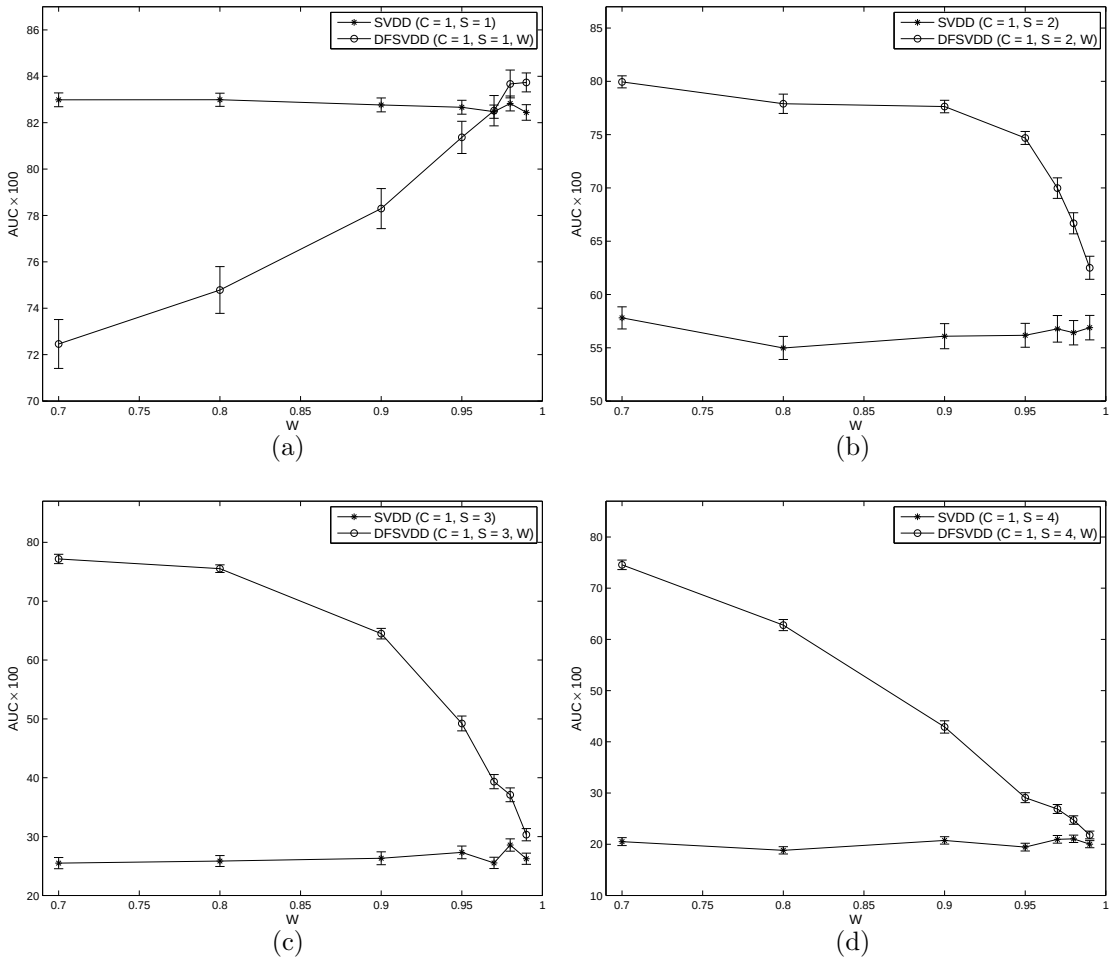


Figure 5.3 Average AUC×100 from SVDD and DFSVDD when $C = 1$: (a) $S = 1$; (b) $S = 2$; (c) $S = 3$; (d) $S = 4$.

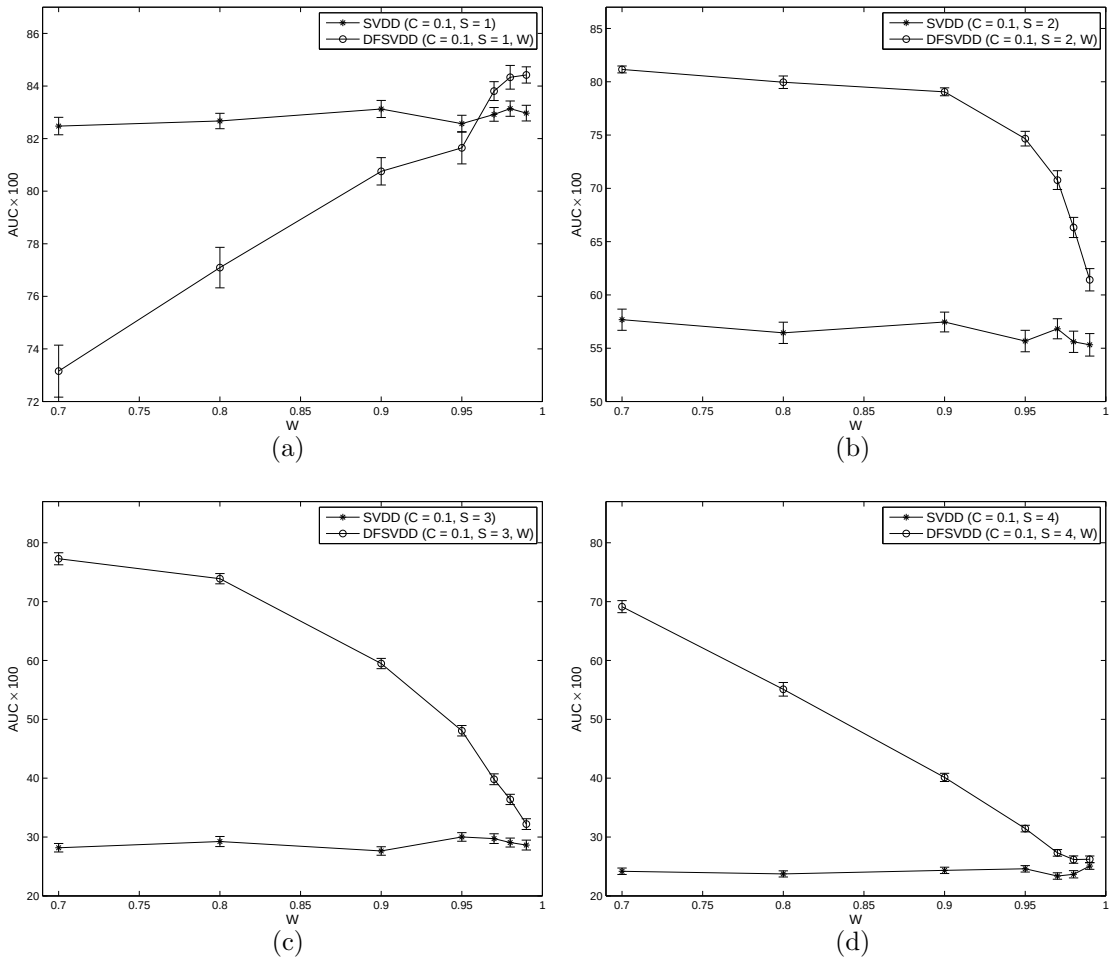


Figure 5.4 Average AUC×100 from SVDD and DFSVDD when $C = 0.1$: (a) $S = 1$; (b) $S = 2$; (c) $S = 3$; (d) $S = 4$.

Figures 5.3 $\sim$ 5.5 illustrate the average AUC from SVDD and DFSVDD with different parameter settings. The additional parameter W in DFSVDD was varied from 0.7 to 0.99. In order to produce fair performance comparisons, SVDD was also evaluated for each W parameter setting in DFSVDD. Figure 5.3(a) shows that DFSVDD with high W values produced larger values of average AUC than the conventional SVDD, when S is equal to 1. The results indicate that performance can be improved by using a lower weight on dense regions. On the other hand, when smaller

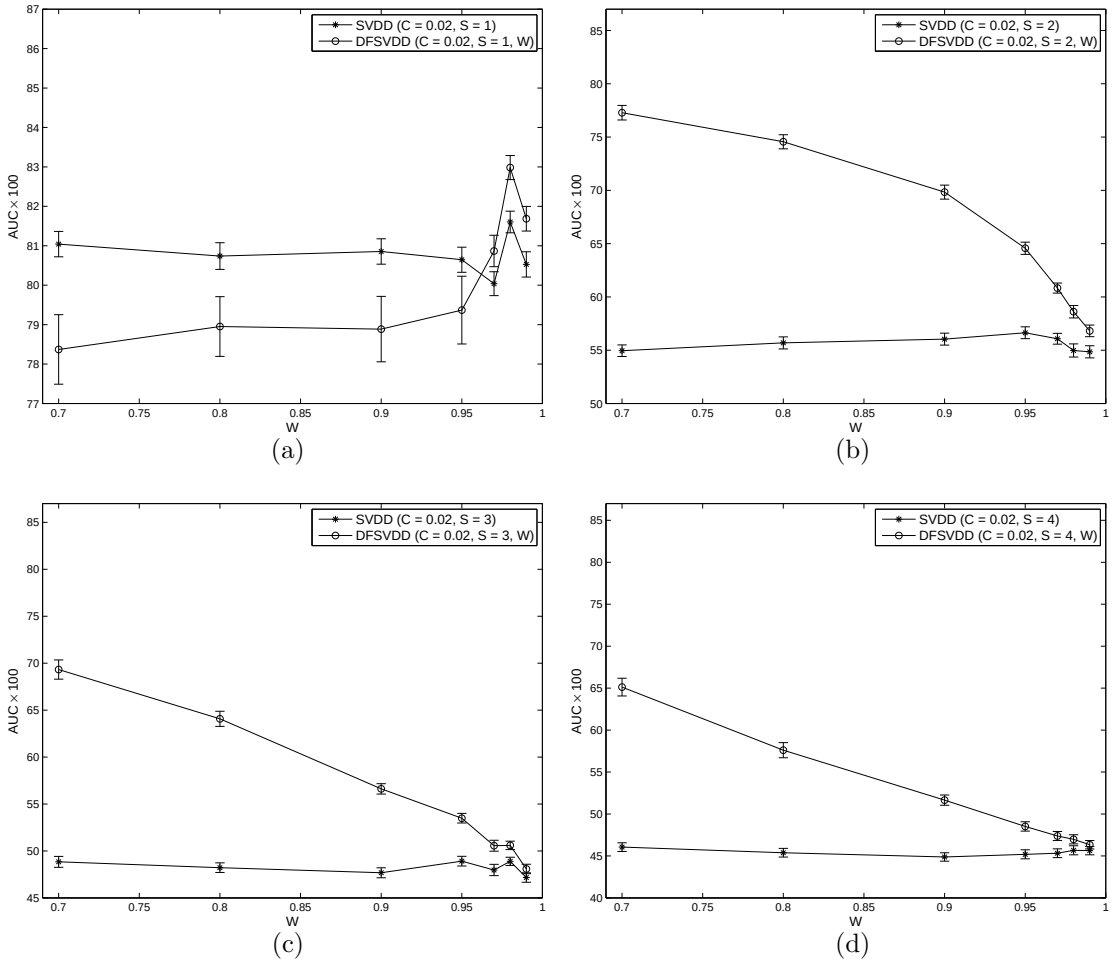


Figure 5.5 Average AUC×100 from SVDD and DFSVDD when $C = 0.02$: (a) $S = 1$; (b) $S = 2$; (c) $S = 3$; (d) $S = 4$.

values of W were applied, the performances of DFSVDD became worse because it yielded smaller values of average AUC than SVDD.

Figure 5.6 demonstrates the comparison of boundaries between SVDD and DFSVDD with the parameters C and S equal to 1. The additional parameter W was set to 0.7 and 0.99 for the proposed DFSVDD. The corresponding boundary of DFSVDD with $W = 0.7$ represents the situation when the small weighting factor is used, while $W = 0.99$ represents the corresponding boundary of DFSVDD when

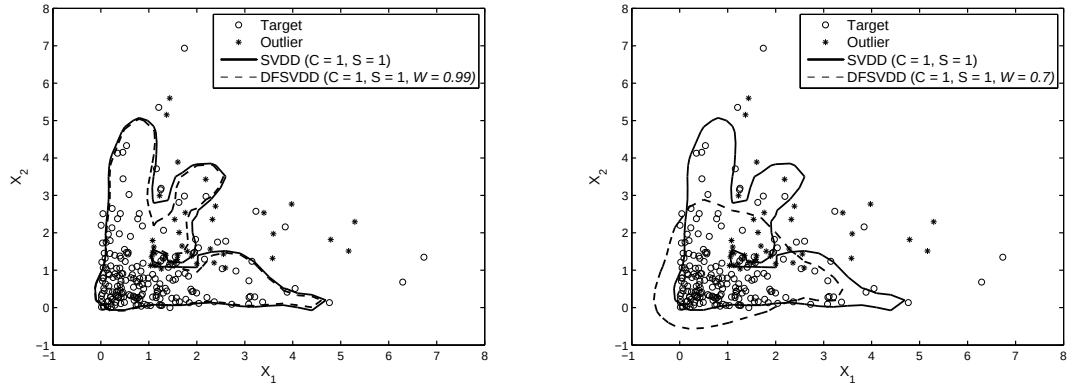


Figure 5.6 Boundaries of SVDD and DFSVDD obtained from different parameters: (a) $C = 1$, $S = 1$, and $W = 0.99$; (b) $C = 1$, $S = 1$, and $W = 0.7$.

using a large weighting factor. The decision boundaries for both approaches were established from 200 target observations and adjusted to produce the same TPR. The FPR can then be measured by generating 40 novelties. Figure 5.6(a) represents the case when the large parameter W ($W = 0.99$) was used in DFSVDD. The corresponding boundary generated from DFSVDD a yielded smaller value of FPR (FPR = 0.28) than the conventional SVDD (FRP = 0.35). Because of the influence of the parameter W, the boundary from DFSVDD tends to move in the direction of the dense region. Consequently, DFSVDD constructed a tighter boundary than SVDD, while it still maintained the complexity of the boundary. The novelties located in the middle of the figure were correctly classified by DFSVDD.

In contrast, when the parameter W was set to 0.7 as shown in Figure 5.6(b), the value of FPR obtained from the proposed DFSVDD method increased to 0.6. Because the influence of parameter W, a less complex boundary was constructed. Furthermore, the boundary established from DFSVDD moved towards the dense re-

gion in the data when the parameter W was reduced. As a result, there were more novelties misclassified as targets by DFSVDD than by SVDD.

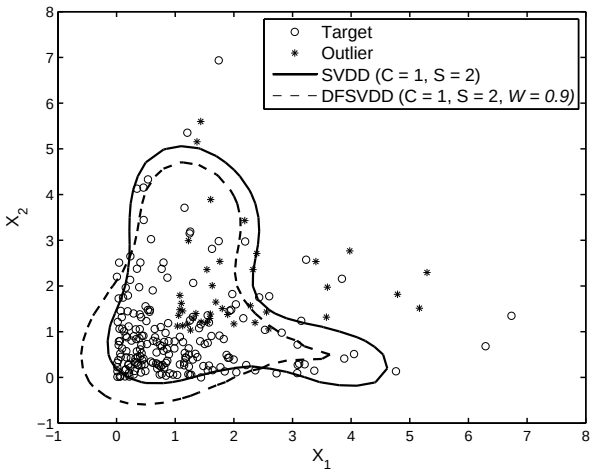


Figure 5.7 Boundaries of SVDD and DFSVDD obtained from $C = 1$, $S = 2$, and $W = 0.90$.

When the parameter S was increased to 2, 3, and 4, DFSVDD with all weighting factors performed better than SVDD. As shown in Figures 5.3(b), 5.3(c), and 5.3(d), the proposed DFSVDD can achieve larger average AUC values than SVDD. To represent this circumstance, Figure 5.7 illustrates the boundaries of SVDD and DFSVDD with the parameters $C = 1$ and $S = 2$. For DFSVDD, the parameter W was set to 0.9. It can be seen that the corresponding boundary generated by DFSVDD produces a lower FPR (FPR = 0.65) than the traditional SVDD (FPR = 0.78), given the same TPR. This indicates that DFSVDD can distinguish novelties from targets better than SVDD because lower error rates can be achieved. By varying the parameter C as shown in Figures 5.4 and 5.5, similar results can be obtained. The vertical lines on each plot show that the standard deviations are small enough to draw a significant conclusion.

Table 5.1 Average AUC of SVDD and DFSVDD over different values of the parameter C

		$C = 1$	$C = 0.1$	$C = 0.02$	Avg.
SVDD	$S = 1$	82.84	82.86	80.77	82.16
	$S = 2$	56.94	56.24	55.52	56.23
	$S = 3$	26.56	29.09	48.23	34.63
	$S = 4$	20.20	24.23	45.37	29.93
DFSVDD	$S = 1$	78.12	79.32	79.54	78.99
	$S = 2$	74.36	75.06	68.60	72.67
	$S = 3$	59.26	58.45	60.03	59.25
	$S = 4$	49.20	47.46	56.43	51.03

Furthermore, we can compare the performances of SVDD and DFSVDD by taking an average of the average AUC values along the parameter C. Table 5.1 shows that SVDD is slightly better than the proposed method when the parameter S is set equal to one. Because of the effect of the parameter S shown in Figure 5.6, SVDD with $S = 1$ can generate more complex boundaries than DFSVDD. As a result, there are fewer novelties incorrectly classified as targets. However, DFSVDD outperforms the traditional SVDD when a larger S is used. This demonstrates that DFSVDD is more effective than SVDD when there is a dense region represented in the data.

5.2.3.2 The True Positive Rate in the Dense Region

In contrast to the conventional SVDD, DFSVDD considers the dense region in the data and extends its boundary in the direction of the dense regions. In this study, the focus is on the correct classification of targets in a dense region near the boundary, comparing DFSVDD to SVDD in terms of the actual TPR. The parameter settings were described in Section 5.2. The empirical threshold was specified for both methods to generate the same TPR. To measure their performances, the actual TPR is calculated from the testing set over 100 simulations. A larger TPR indicates a better classification performance of the method in the dense region.

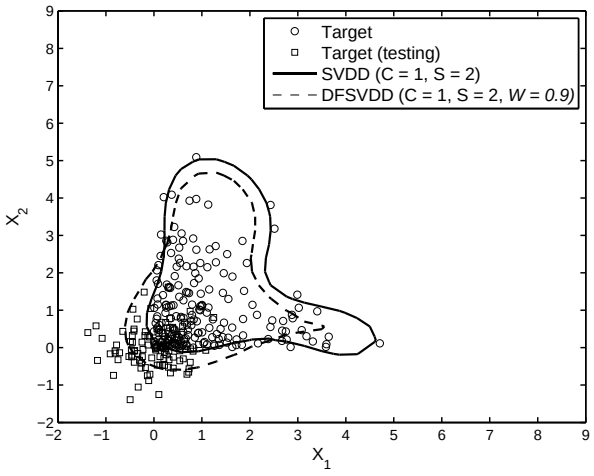


Figure 5.8 Comparison of the capturing ability of the dense region in target between SVDD and DFSVDD.

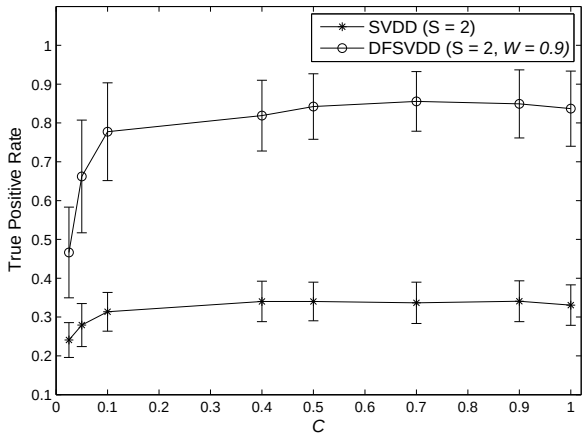


Figure 5.9 The actual true positive rate between SVDD and DFSVDD.

Figure 5.9 shows that the proposed DFSVDD yields higher actual TPR than SVDD for every value of the parameter C. The results indicate that DFSVDD effectively classifies the targets in the dense region, while some portion of target observations are rejected by SVDD. This leads to lower actual TPR values for conventional SVDD. The vertical lines in the plot represent the standard errors from the simulation.

5.3 Discussion

We have presented a density-focused SVDD method that can be effectively used for novelty detection. However, the boundary of SVDD excludes the data points on the edge of dense regions. The data points that are near the dense regions should be accepted by the classifier. At the same time, the data points that lie in the sparse region should be rejected from the boundary. We propose a density-focused SVDD (DFSVDD) that considers the shape as well as the dense region of the data. Two distance measures – kernel distance and density distance – are determined to construct a boundary for DFSVDD. The kernel distance can be obtained from the regular SVDD algorithm and the density distance is estimated from the support vectors obtained by a quadratic optimization of the conventional SVDD. An extra parameter W is introduced to combine these two distance measures and to control the trade-off between the shape and the density of the data. The simulation study shows the overall performance of DFSVDD is better than the conventional SVDD in terms of average AUC values. This study considers only the two-dimensional case. Results in higher dimensions will have to be the subject of further study.

CHAPTER 6

SUMMARY AND FUTURE DIRECTIONS

This dissertation focuses on the development of thresholds for novelty detection with application to SPC, especially in control charts. In Chapter 3, we proposed a bootstrap-based T^2 multivariate control chart. A bootstrap method is a nonparametric technique that does not require any distributional assumptions on the data. In a simulation study, the proposed bootstrap-based T^2 control charts performs better than the traditional T^2 and existing kernel density estimation (KDE)-based T^2 control charts. In Chapter 4, a principal component analysis (PCA)-based bootstrap control chart is proposed to improve the estimation of a control limit when the data do not follow a multivariate normal distribution. The existing PCA-based control charts, T^2_{PCA} and Q charts, are integrated with the bootstrap and KDE approaches to determine control limits. A simulation study indicates that the nonparametric control limits constructed using bootstrapping and KDE show superior performance over the traditional PCA control charts under different types of multivariate nonnormal distributions. Finally, in Chapter 5, the study of support vector data description (SVDD) indicates that SVDD does not take the density of the data into account when constructing its boundary. A density-focused SVDD (DFSVDD) is proposed to overcome this limitation of SVDD by combining two distance measures, the kernel distance and the density distance. A simulation study shows the higher effectiveness of DFSVDD over the traditional SVDD when the data contain a dense region.

In future work, the idea of combining boundaries for novelty detection will be explored. This has been shown to improve detection performance [110]. An individual

decision may result in a biased opinion. Additional opinions are required to increase the possibility of making the right decision. This techniques is known as an ensemble of classifiers that uses several unique classifier techniques to create a multiple classifier system [111]. A subject of further study is applying an ensemble technique to create multiple classifier systems for control charts. Such systems can assist in determining a proper and accurate decision as to whether or not a process is out of control. For instance, a classical control chart typically requires certain assumptions to establish the chart. This parametric chart can serve as a single decision-maker. When the data deviate from the assumption, another type of a control chart, such as those based on the nonparametric methods in this dissertation, can be combined with other charts resulting in an increase in detection performance. It will be useful to study if the ensemble aspect will yield improvements in SPC.

REFERENCES

[1] M. Markou and S. Singh, "Novelty detection: a review-part 1: statistical approaches," *Signal Processing*, vol. 83, no. 12, pp. 2481–2497, 2003.

[2] P. Hayton, B. Schölkopf, B. Tarassenko, and P. Anuzis, "Support vector novelty detection applied to jet engine vibration," in *Advances in Neural Information Processing Systems*, 2001, pp. 946–952.

[3] R. J. Bolton and D. J. Hand, "Statistical fraud detection: a review," *Statistical Science*, vol. 17, no. 3, pp. 235–249, 2002.

[4] D.-Y. Yeung and C. Chow, "Parzen-window network intrusion detectors," in *16th International Conference on Pattern Recognition*, vol. 4, 2002, pp. 385–388.

[5] S. Byers and A. E. Raftery, "Nearest-neighbor clutter removal for estimating features in spatial point processes," *Journal of the American Statistical Association*, vol. 93, no. 442, pp. 577–584, 1998.

[6] D. C. Montgomery, *Introduction to Statistical Quality Control*, 6th ed. New York, NY: Wiley, 2008.

[7] M. R. Chernick, *Bootstrap Methods: A Guide for Practitioners and Researchers*, 2nd ed. Hoboken, NJ: Wiley, 2008.

[8] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. San Francisco, CA: Morgan Kaufmann, 2005.

[9] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," AI *Magazine*, vol. 17, no. 3, pp. 37–53, 1996.

[10] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy, "Mining data streams: a review," *SIGMOD Rec.*, vol. 34, no. 2, pp. 18–26, 2005.

[11] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY: Springer, 2001.

[12] V. N. Vapnik, *Statistical Learning Theory*. New York, NY: Wiley, 1998.

[13] N. Cristianini and J. Shawe-Taylor, *Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, UK: Cambridge University Press, 2000.

[14] S. Abe, *Support Vector Machines for Pattern Classification*. London, UK: Springer, 2000.

[15] T. M. Mitchell, *Machine Learning*. New York, NY: McGraw-Hill, 1997.

[16] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ: Prentice Hall, 1999.

[17] J. Jackson and G. Mudholkar, "Control procedures for residuals associated with principal component analysis," *Technometrics*, vol. 21, no. 3, pp. 341–349, 1979.

[18] A. Webb, *Statistical Pattern Recognition*, 2nd ed. West Sussex, UK: Wiley, 2002.

[19] G. Strang, *Linear Algebra and Its Applications*, 3rd ed. Orlando, FL: Harcourth Brace Jovanovich, 1988.

[20] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, 6th ed. Upper Saddle River, NJ: Prentice Hall, 2007.

[21] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *Artificial Neural Networks ICANN'97*, W. Gerstner, A. Germond, M. Hasler, and J.-D. Nicoud, Eds. Springer Berlin/Heidelberg, 1997, vol. 1327, pp. 583–588.

[22] P. Giudici and S. Figini, *Applied Data Mining for Business and Industry*, 2nd ed. West Sussex, UK: Wiley, 2009.

[23] C. M. Bishop, *Pattern Recognition and Machine Learning*, 2nd ed. New York, NY: Springer, 2006.

[24] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[25] W. L. Martinez and A. R. Martinez, *Computational Statistical Handbook with* MATLAB, 2nd ed. Boca Raton, FL: Chapman and Hall/CRC, 2007.

[26] D. M. J. Tax, "One-class classification: Concept-learning in the absence of counter-examples," Ph.D. dissertation, Delft University of Technology, 2001.

[27] S. Harmeling, G. Dornhege, D. M. J. Tax, F. Meinecke, and K.-R. Müller, "From outliers to prototypes: ordering data," *Neurocomputing*, vol. 69, no. 1, pp. 1608–1618, 2006.

[28] P. Juszczak, D. M. J. Tax, E. Pekalska, and R. P. W. Duin, "Minimum spanning tree based one-class classifier," *Pattern Recognition*, vol. 72, no. 7-9, pp. 1859–1869, 2009.

[29] R. Graham and P. Hell, "On the history of the minimum spanning tree problem," *Annals of the History of Computing*, vol. 7, no. 1, pp. 43–57, 1985.

[30] R. O. Duda, P. E. Hart, and D. G. Stok, *Pattern Classification*, 2nd ed. New York, NY: Wiley, 2000.

[31] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt, "Support vector method for novelty detection," in *Advances in Neural Information Processing Systems*, 2000, pp. 582–588.

[32] S. S. Khan and M. G. Madden, "A survey of recent trends in one class classification," *Artificial Intelligence and Cognitive Science*, vol. 6206, pp. 188–197, 2010.

[33] D. M. J. Tax and R. P. W. Duin, "Support vector domain description," *Pattern Recognition Letters*, vol. 20, no. 11-13, pp. 1191–1199, 1999.

[34] ——, "Support vector data description," *Machine Learning*, vol. 54, no. 1, pp. 45–66, 2004.

[35] R. P. W. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, and D. M. J. Tax, "PRTools4: The MATLAB toolbox for pattern recognition," Delft University of Technology, Netherlands, 2007. [Online]. Available: http://www.prtools.org

[36] H. Y. Cho, "Data description and noise filtering based detection with its application and performance comparison," *Expert Systems with Applications*, vol. 36, no. 1, pp. 434–441, 2009.

[37] A. Banerjee, P. Burlina, and R. Meth, "Fast hyperspectral anomaly detection via SVDD," in IEEE *International Conference on Image Processing*, 2007, pp. IV –101 –IV –104.

[38] J. Park, D. Kang, J. Kim, J. T. Kwok, and I. W. Tsang, "SVDD-based pattern denoising," *Neural Computation*, vol. 19, no. 7, pp. 1919–1938, 2007.

[39] J. T. Kwak and I. W. Tsang, "The pre-image problem in kernel methods," IEEE *Transactions on Neural Networks*, vol. 15, no. 6, pp. 1517–1525, 2004.

[40] B. Liu, Y. Xiao, L. Cao, Z. Hao, and F. Deng, "SVDD-based outlier detection on uncertain data," *Knowledge and Information Systems*, pp. 1–22, 2012.

[41] K. Lee, D.-W. Kim, D. Lee, and K. H. Lee, "Improving support vector data description using local density degree," *Pattern Recognition*, vol. 38, no. 10, pp. 1768–1771, 2005.

[42] Y. Zhang, Z.-X. Chi, and K.-Q. Li, "Fuzzy multi-class classifier based on support vector data description and improved PCM," *Expert Systems with Applications*, vol. 36, no. 5, pp. 8714–8718, 2009.

[43] Y. Zhang, X.-D. Liu, F.-D. Xie, and K.-Q. Li, "Fault classifier of rotating machinery based on weighted support vector data description," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7928–7932, 2009.

[44] G. Huang, H. Chen, Z. Zhou, F. Yin, and K. Guo, "Two-class support vector data description," *Pattern Recognition*, vol. 44, no. 2, pp. 320–329, 2011.

[45] W. H. Woodall, "Controversies and contradictions in statistical process control," *Journal of Quality Technology*, vol. 32, no. 4, pp. 341–350, 2000.

[46] W. A. Shewhart, *Economic Control of Quality of Manufactured Product*. Princeton, NJ: Van Nostrand Press, 1931.

[47] E. S. Page, "Cumulative sum charts," *Technometrics*, vol. 3, no. 1, pp. 1–9, 1961.

[48] J. M. Lucas and M. S. Saccucci, "Exponentially weighted moving average control schemes: properties and enhancements," *Technometrics*, vol. 32, no. 1, pp. 1–12, 1990.

[49] E. G. Schilling and P. R. Nelson, "The effect of non-normality on the control limits of $\bar{x}$ charts," *Journal of Quality Technology*, vol. 8, pp. 183–188, 1976.

[50] N. Balakrishnan and S. Kocherlakota, "Effects of nonnormality on $\bar{x}$ charts: single assignable cause model," *Sankhyā: The Indian Journal of Statistics, Series B*, vol. 48, no. 3, pp. 439–444, 1986.

[51] S. A. Yourstone and W. J. Zimmer, "Non-normality and the design of control charts for averages," *Decision Sciences*, vol. 23, no. 5, pp. 1099–1113, 1992.

[52] C. M. Borror, D. C. Montgomery, and G. C. Runger, "Robustness of the EWMA control chart to non-normality," *Journal of Quality Technology*, vol. 31, no. 3, pp. 309–316, 1999.

[53] R. L. Mason and J. C. Young, *Multivariate Statistical Process Control with Industrial Applications*. Philadelphia, PA: American Statistical Association and Society for Industrial and Applied Mathematics, 2002.

[54] S. Bersimis, S. Psarakis, and J. Panaretos, "Multivariate statistical process control charts: an overview," *Quality & Reliability Engineering International*, vol. 23, no. 5, pp. 517–543, 2006.

[55] H. Hotelling, *Multivariate Quality Control*, ser. Techniques of Statistical Analysis, C. Eisenhart, M. W. Hastay, and W. A. Wallis, Eds. New York, NY: McGraw-Hill, 1947.

[56] Y.-M. Chou, R. L. Mason, and Y. J. C., "The control chart for individual observations from a multivariate non-normal distribution," *Communications in Statistics–Simulation and Computation*, vol. 30, no. 8-9, pp. 1937–1949, 2001.

[57] R. B. Crosier, "Multivariate generalizations of cumulative sum quality-control schemes," *Technometrics*, vol. 30, no. 3, pp. 291–303, 1988.

[58] W. H. Woodall and M. M. Ncube, "Multivariate CUSUM quality control procedures," *Technometrics*, vol. 27, pp. 285–292, 1985.

[59] J. D. Healy, "A note on multivariate CUSUM procedures," *Technometrics*, vol. 29, no. 4, pp. 409–412, 1987.

[60] C. A. Lowry, W. H. Woodall, C. W. Champ, and S. E. Rigdon, "A multivariate exponentially weighted moving average control chart," *Technometrics*, vol. 34, no. 1, pp. 46–53, 1992.

[61] Z. G. Stoumbos and J. H. Sullivan, "Robustness to non-normality of the multivariate EWMA control chart," *Journal of Quality Technology*, vol. 41, no. 13, pp. 260–276, 2002.

[62] M. C. Testik, G. C. Runger, and C. M. Borror, "Robustness properties of multivariate EWMA control charts," *Quality & Reliability Engineering International*, vol. 19, no. 1, pp. 31–38, 2003.

[63] R. Y. Liu, "Control charts for multivariate processes," *Journal of the American Statistical Association*, vol. 90, no. 432, pp. 1380–1387, 1995.

[64] P. Qiu, "Distribution–free multivairate process control based on log–linear modeling," *IIE Transactions*, vol. 40, no. 7, pp. 664–677, 2008.

[65] C. Zou and F. Tsung, "A multivariate sign EWMA control chart," *Technometrics*, vol. 53, no. 1, pp. 84–97, 2011.

[66] W. Hwang, G. Runger, and E. Tuv, "Multivariate statistical process control with artificial contrasts," *IIE Transactions*, vol. 39, no. 6, pp. 659–669, 2007.

[67] T. Sukchotrat, S. B. Kim, K. L. Tsui, and V. C. P. Chen, "Integration of classification algorithms and control chart techniques for monitoring multivariate processes," *Journal of Statistical Computation and Simulation*, vol. 81, no. 12, pp. 1897–1911, 2011.

[68] P. Chongfuangprinya, S. B. Kim, S. K. Park, and T. Sukchotrat, "Integration of support vector machines and control charts for multivariate process monitoring," *Journal of Statistical Computation and Simulation*, vol. 81, no. 9, pp. 1157–1173, 2011.

[69] H. M. Bush, P. Chongfuangprinya, V. C. P. Chen, T. Sukchotrat, and S. B. Kim, "Nonparametric multivariate control charts based on a linkage ranking algorithm," *Quality & Reliability Engineering International*, vol. 26, no. 7, pp. 663–675, 2010.

[70] R. L. Burden and J. D. Faires, *Numerical Analysis*, 7th ed. Boston, MA: Brooks Cole, 2001.

[71] B. Silverman, *Density Estimation for Statistics and Data Analysis*, 5th ed. London, UK: Chapman & Hall/CRC, 1986.

[72] S. J. Sheather and M. C. Jones, "A reliable data-based bandwidth selection method for kernel density estimation," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 53, no. 3, pp. 683–690, 1991.

[73] A. W. Bowman and A. Azzalini, *Applied Smoothing Techniques for Data Analysis*. London, UK: Oxford University Press, 1997.

[74] W. Ku, R. H. Storer, and C. Georgakis, "Disturbance detection and isolation by dynamic principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 30, pp. 179–196, 1995.

[75] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. New York: Springer, 2002.

[76] T. Kourti, "Application of latent variable methods to process control and multivariate statistical process control in industry," *International Journal of Adaptive Control and Signal Processing*, vol. 19, pp. 213–246, 2005.

[77] J. E. Jackson, "Quality control methods for several related variables," *Technometrics*, vol. 1, no. 4, pp. 359–377, 1959.

[78] T. Kourti and J. F. MacGregor, "Multivariate SPC methods for process and product monitoring," *Journal of Quality Technology*, vol. 28, no. 4, pp. 409–428, 1996.

[79] A. Nijhuis, S. De Jong, and B. G. M. Vandeginste, "Multivariate statistical process control in chromatography," *Chemometrics and Intelligent Laboratory Systems*, vol. 38, pp. 51–62, 1997.

[80] A. A. Kalgonda and S. R. Kulkarni, "Multivariate quality control chart for autocorrelated processes," *Journal of Applied Statistics*, vol. 31, pp. 317–328, 2004.

[81] C. M. Mastrangelo, G. C. Runger, and D. C. Montgomery, "Statistical process monitoring with principal components," *Quality & Reliability Engineering International*, vol. 12, no. 3, pp. 203–210, 1996.

[82] J. E. Jackson, *A User's Guide to Principal Components*. New York, NY: Wiley, 1991.

[83] A. Ferrer, "Multivariate statistical process control based on principal component analysis (MSPC-PCA): some reflections and case study in an autobody assembly process," *Quality Engineering*, vol. 19, no. 4, pp. 311–325, 2007.

[84] G. E. P. Box, "Some theorems on quadratic forms applied in the study of analysis of variance problem: effect of inequality of variance in one-way classification," *The Annals of Mathematical Statistics*.

[85] P. Nomikos and J. F. MacGregor, "Multivariate SPC charts for monitoring batch processes," *Technometrics*, vol. 37, no. 1, pp. 41–59, 1995.

[86] R. Sun and F. Tsung, "A kernel-distance-based multivariate control chart using support vector methods," *International Journal of Production Research*, vol. 41, no. 13, pp. 2975–2989, 2003.

[87] T. Sukchotrat, S. B. Kim, and F. Tsung, "One-class classification-based control charts for multivariate process monitoring," *IIE Transactions*, vol. 42, no. 2, pp. 107–120, 2009.

[88] D. C. Montgomery and C. M. Mastrangelo, "Some statistical process control methods for autocorrelated data," *Journal of Quality Technology*, vol. 23, no. 3, pp. 179–204, 1991.

[89] S. B. Kim, W. Jitpitaklert, and T. Sukchotrat, "One-class classification-based control charts for monitoring autocorrelated multivariate processes," *Communications in Statistics - Simulation and Computation*, vol. 39, no. 3, pp. 461–474, 2010.

[90] B. Efron, "Bootstrap methods: another look at the jackknife," *The Annals of Statistics*, vol. 7, no. 1, pp. 1–26, 1979.

[91] C. Léger, D. N. Politis, and J. P. Romano, "Bootstrap technology and applications," *Technometrics*, vol. 34, no. 4, pp. 378–398, 1992.

[92] S. M. Bajgier, "The use of bootstrapping to construct limits on control charts," in *Proceedings of the Decision Science Institute*, 1992, pp. 1611–1613.

[93] T. Seppala, H. Moskowitz, R. Plante, and J. Tang, "Statistical process control via the subgroup bootstrap," *Journal of Quality Technology*, vol. 27, no. 2, pp. 683–690, 1995.

[94] R. Y. Liu and J. Tang, "Control charts for dependent and independent measurements based on bootstrap methods," *Journal of the American Statistical Association*, vol. 91, pp. 1694–1700, 1996.

[95] R. Y. Liu and K. Singh, *Moving Blocks Bootstrap and Jackknife Capture Weak Dependence*, ser. Exploring the Limits of Bootstrap, R. LePage and L. Billard, Eds. New York, NY: Wiley, 1992.

[96] L. A. Jones and W. H. Woodall, "The performance of bootstrap control charts," *Journal of Quality Technology*, vol. 30, no. 4, pp. 362–375, 1998.

[97] Z. Wu and Q. Wang, "Bootstrap control charts," *Quality Engineering*, vol. 9, no. 1, pp. 143–150, 1996.

[98] Y. L. Lio and C. Park, "A bootstrap control chart for Birnbaum-Saunders percentiles," *Quality & Reliability Engineering International*, vol. 24, no. 5, pp. 585–600, 2008.

[99] H. I. Park, "Median control charts based on bootstrap method," *Communications in Statistics-Simulation and Computation*, vol. 38, pp. 558–570, 2009.

[100] M. D. Nichols and W. J. Padgett, "A bootstrap control chart for Weibull percentiles," *Quality & Reliability Engineering International*, vol. 22, no. 2, pp. 141–151, 2006.

[101] W. J. Padgett and J. D. Spurrier, "Shewhart-type charts for percentiles of strength distributions," *Journal of Quality Technology*, vol. 22, no. 4, pp. 283–288, 1990.

[102] A. Polansky, "A general framework for constructing control charts," *Quality & Reliability Engineering International*, vol. 21, no. 6, pp. 633–653, 2005.

[103] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*. Boca Raton, FL: Chapman & Hall/CRC, 1993.

[104] A. Azzalini and A. Dalla Valle, "The multivariate skew-normal distribution," *Biometrika*, vol. 83, no. 4, pp. 715–726, 1996.

[105] W. H. Woodall and D. C. Montgomery, "Research issues and ideas in statistical process control," *Journal of Quality Technology*, vol. 31, no. 4, pp. 376–386, 1999.

[106] J. P. Royston, "Some techniques for assessing multivariate normality based on the shapiro-wilk *W*," *Applied Statistics*, vol. 32, no. 2, pp. 121–133, 1983.

[107] D. M. J. Tax, "DDtools, the data description toolbox for MATLAB," Delft University of Technology, Netherlands, May 2012, vesrion 1.9.1. [Online]. Available: http://prlab.tudelft.nl

[108] C. E. Metz, "Basic principles of ROC analysis," *Seminars in Nuclear Medicine*, vol. 8, no. 4, pp. 283–298, 1978.

[109] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

[110] R. Polikar, "Ensemble based systems in decision making," *Circuits and Systems Magazine,* IEEE, vol. 6, no. 3, pp. 21–45, 2006.

[111] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms.* Hoboken, NJ: Wiley, 2004.

BIOGRAPHICAL STATEMENT

Poovich Phaladiganon received his B.S. degree in Management Technology from Sirindhorn International Institute of Technology (SIIT), Thammasat University in 2006. In 2008, he completed his M.S. degree in Industrial and Manufacturing Systems Engineering from the University of Texas at Arlington (UTA), where he has continued his studies for a Ph.D. degree and worked as a graduate research assistant and a graduate teaching assistant. His current research interests are multivariate statistical process control and novelty detection. He is a member of the Center of Stochastic Modeling, Optimization & Statistics (COSMOS) at UTA and the Institute for Operations Research and Management Science (INFORMS).