A METHODOLOGY FOR SPATIAL AND TIME SERIES DATA MINING AND ITS APPLICATIONS

by

YOUNG-SEON JEONG

A dissertation submitted to the

Graduate School-New Brunswick

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Doctor of Philosophy

Graduate Program in Industrial and Systems Engineering

Written under the direction of

Professor Myong K. Jeong

And approved by

____

____

____

____

New Brunswick, New Jersey

May, 2011

ABSTRACT OF THE DISSERTATION

A Methodology for Spatial and Time Series Data Mining and Its Applications

By YOUNG-SEON JEONG

Dissertation Director:

Dr. Myong K. Jeong

In this dissertation, we present several methodologies for mining spatial and time-sequence data obtained in diverse domains. We first propose a new spatial randomness test and classification method for binary spatial data with specific application to the detection and identification of spatial defect patterns on semiconductor wafer maps. We present the generalized join-count (JC)-based statistic as an alternative approach, and derive a procedure to determine the optimal weights of JC-based statistics. In the proposed methodology, a spatial correlogram, which transforms binary spatial data into time-sequence data, is used as a novel feature to detect spatial autocorrelation and classify spatial defect patterns on the wafer maps.

Secondly, we propose a novel distance measure, denoted weighted dynamic time warping (WDTW), for time series classification and clustering problems. The dynamic time warping (DTW) algorithm has been extensively used as a distance measure in combination with the distance-based classifiers. However, the DTW algorithm ignores the relative importance of the phase distance between points in a time series, possibly leading to misclassification. Therefore, we propose a WDTW distance measure which does account for the relative importance of each point in terms of the phase distance between the time series points.

Thirdly, we propose a wavelet-based anomaly detection procedure to detect any possible process fault with time-sequence data that have some local variations even under normal working conditions. To handle the large number of parameters in both the mean and variance models, we have developed the wavelet-based mean and variance thresholding procedure to extract a few important wavelet coefficients that may explain local variations in the time domain.

Finally, we propose a kernel-based regression with lagged dependent variables. Kernel-based regression techniques are extensively used for exploring the nonlinearity of data in a relatively easy procedure involving the use of various kernel functions. However, the major drawback of current kernel-based regression techniques is their underlying assumption that there is no autocorrelation in the residuals of observations. To avoid this problem, we propose a kernel-based regression model with lagged dependent variables (LDVs), considering autocorrelations of both the response variables and the nonlinearity of data.

Acknowledgements

I would like to express my sincere appreciation to my advisor, Professor Myong K. Jeong, one of the most “Beautiful Minds” I ever met in my life, for his guidance and support throughout the course of my Ph.D. study. Professor Jeong was a patient and supportive mentor who taught me how to think critically and smartly and how to make my research look appealing and relevant. I am truly grateful to Professor Susan L. Albin, an exceptional researcher and teacher. Since the beginning of my studies at Rutgers University, her comments and help were certainly guiding me through this difficult journey. In addition, I must also thank my dissertation committee members, Professor Hoang Pham, Professor Wanpracha (Art) Chaovalitwongse and Professor Ying Hung for their expertise, support, intellectual insights and time.

I would like to especially thank my wife, Myungsun Hong, who always believed in me and encouraged me with unwavering love and support. I am also thankful to my sons, Munyoung and Daniel who make me cheer to live more sincerely and happily. Without my wife and sons, I could not have done this. I look forward to many, many happy years together.

Finally, I must thank my friends in the Rutgers University and my families in South Korea-- you helped me to finalize my goal, and importantly, you made my life most happy.

Table of Contents

CHAPTER 4 A Statistical Anomaly Detection Procedure for a Time Sequence Data with Local Variations

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Overview

Mining spatial and time series data has become increasingly important in various fields of research as it provides the means to extract meaningful information and other specific characteristics of the data. Examples of spatial and time series data mining are the daily pattern analysis of the Dow Jones Index (Alwan and Roberts 1988), spatial prediction of ozone concentration profiles (Temiyasathit *et al*. 2009), fault detection with sensing data in manufacturing systems (Lada *et al*. 2002), incident detection on freeways using time series traffic information (Jeong *et al*. 2010), and the prediction of traffic volumes on freeways based on known past events and location information (Zhao and Park 2004). However, complicated spatial and time series data with autocorrelated or dynamically changing patterns based on contributions from potential change events are associated with serious difficulties in dealing with these data in monitoring system processes. In this dissertation, we focus on the methodology used for spatial and time series data mining and its applications.

Spatial data can be characterized as topological, distance, and direction information organized by multidimensional spatial indexed structures (Cliff and Ord 1981). In addition, spatial data are usually binary in nature. If there are spatial autocorrelations in some areas, spatial patterns may exist. As an example of a specific application for mining binary spatial data, we investigate the procedure used to determine the presence of spatial autocorrelation and to classify spatial patterns on wafer maps. A wafer map is a graphical illustration of the locations of defective chips on a wafer. Defective chips are likely to exhibit a spatial dependence across the wafer map, which contains useful information on the fabrication process of integrated circuits (ICs). We have developed the spatial correlogram, which transforms binary spatial data into more informative time-sequence data. Based on the proposed spatial correlogram, we present a new spatial randomness test procedure for the detection of spatial autocorrelation and a classification method of spatial defect patterns on semiconductor wafer maps.

Time series classification and clustering is a classical problem in pattern recognition, with wide applications in the real world. Among the many algorithms used for time series classification and clustering problems, the nearest neighbor classifier with dynamic time warping (DTW) distance is one of the most extensively used approaches. However, the conventional DTW algorithm considers that all points in the time series are of equal value; hence, they are weighted equally whether or not there is a phase difference between two points. This disadvantage has led us to propose the weighted DTW (WDTW) technique, which weights nearer neighbor points more heavily depending on the distance between a reference point and a testing point. We show that the proposed WDTW is a generalized methodology of DTW and Euclidean distance that is dependent on the choices of weights of a phase difference. We also explore several mathematical properties of WDTW. To provide a clearer explanation of the rationale underlying the performance advantage of the proposed WDTW, we present a number of examples to graphically illustrate possible situations in which WDTW is clearly more effective than conventional DTW. The extensive experimental results reported here show that the proposed WDTW can achieve an improved accuracy compared to existing approaches, including DTW, Euclidean distance measure, and some variants of DTW.

Multiple sets of complicated time-sequence data have been generated in many engineering studies; these have been used for a multitude of purposes, including monitoring the quality of manufacturing processes. Due to the high dimensionality of the data, it is difficult to detect process change with time-sequence data, especially when there are systematic variations in local regions. As an alternative approach, we propose a wavelet-based anomaly detection procedure to detect a process fault with time-sequence data that display local variations under the normal working conditions. To deal with the large number of parameters in both the mean and variance models, we have developed an integrated mean and variance thresholding procedure that keeps the model simple and fits the data curves well. Guidelines are provided for selecting regularization parameters in the penalized likelihood used for parameter estimation. We have also developed process monitoring procedures for detecting process changes using the wavelet coefficients selected through the wavelet-based mixed effects model. Evaluation with real-life data sets shows that the proposed procedure performs better than several techniques extrapolated from methods based on single curve-based data reduction.

Kernel-based regression techniques, such as support vector machines for regression and kernel ridge regression (KRR), have been extensively used to explore the nonlinearity of data in a relatively easy procedure involving the use of various kernel functions. However, the major drawback of existing kernel-based regression techniques is their underlying assumption that there is no autocorrelation in the residuals of the observations. To avoid this problem, we propose here a kernel-based regression model with lagged dependent variables (LDVs) that considers both the autocorrelations of the response variables and the nonlinearity of data. We explore the nonlinear relationship between the response and both independent and past response variables using various kernel functions. In this specific case, it is difficult to apply existing kernel manipulations because of the LDVs. We derive the kernel ridge estimators with LDVs using a new mapping concept so that the nonlinear mapping does not have to be computed explicitly depending on kernel types. Also, the centering technique of the individually mapped data in the feature space is derived in order to consider an intercept term in KRR with LDVs. The experimental results show that the proposed approaches perform better than KRR or ridge regression, implying that the model can be used as a promising alternative when there are autocorrelations between dependent variables.

1.2 Thesis outline

This thesis is organized as follows. Chapter 2 presents the identification methodology of spatial defect patterns on binary spatial data with the specific application to a wafer map analysis in a semiconductor manufacturing process. A spatial correlogram is used to transform binary spatial data into informative time series data that can be used to detect the presence of spatial autocorrelations and classify defect patterns on the wafer map.

Chapter 3 proposes a weighted dynamic time warping algorithm (WDTW) for automatic time series classification and clustering. By considering different weight values that are dependent on the distance between a reference point and a test point in a sequence, the proposed WDTW has an enhanced accuracy in terms of time series classification and clustering problems. Chapter 4 presents a wavelet-based anomaly detection procedure by characterizing the variations of multiple curves at certain local regions. Chapter 5 proposes a kernel-based ridge regression with lagged dependent variables that considers both the autocorrelations of the response variables and the nonlinearity of data. Finally, Chapter 6 summarizes the research results and describes several research problems for future investigations.

CHAPTER 2

Detection of the Presence of Spatial Autocorrelations and Classification of Spatial Patterns with Binary Spatial Data

2.1 Introduction

Spatial autocorrelation and spatial pattern represents a correlation among the locations of spatial data and a consistent rule in the locations, respectively. If there is an autocorrelation in spatial data, it means that the locations are not independent of each other, but somehow linked systematically, i.e., the data are spatially dependent (Cliff and Ord 1981). Also if there are spatial autocorrelations in some areas, there may exist some spatial patterns. Those properties for spatial data are frequently encountered in ecological data, geographical data, environmental data, and even manufacturing data (Cliff and Ord 1981, Temiyasathit *et al*. 2009, Cunningham and Mckinnon 1998). In this section, we develop the procedure to detect the presence of spatial autocorrelation and classify spatial patterns for the binary spatial data with the specific example in the wafer map analysis.

A wafer is an elementary unit in semiconductor manufacturing. Several hundred integrated circuits (ICs) are simultaneously fabricated on a single wafer (Fenner *et al.* 2005). After the completion of IC fabrication, each chip is classified as either functional or defective. A wafer map is used to display the locations of defective ICs chips on the wafer. A wafer map is likely to exhibit a spatial dependence across the wafer. As explained in Hansen *et al*. (1997), defective chips commonly occur in clusters or display some systematic patterns. Such defect patterns contain useful information about manufacturing process conditions (Cunningham and McKinnon 1998). For example, uneven temperatures or chemical aging lead to spatial cluster on the wafer map. Clusters also can be the result of crystalline nonuniformity, photo-mask misalignment or particles caused by mechanical vibration. Stepper and/or probe malfunctioning and sawing imperfections also are major causes of repetitive patterns. Material shipping and handling also can leave a scratch on the wafer map (Cunningham and McKinnon 1998, Hansen and Tyregod 1998, Hansen *et al*. 1997, and Taam and Hamada 1993).

The defect patterns represented on the wafer map hold important information that can assist process engineers in their understanding of the ongoing manufacturing processes. Consequently, wafer maps have been widely used in the semiconductor industry for process monitoring and yield enhancement. Chen and Liu (2000) and Liu *et al.* (2002) developed intelligent systems that use wafer maps and wafer bin maps, respectively, to recognize defect spatial patterns and aid in the diagnosis of causes of failures. They adapted a neural network called as adaptive resonance theory network 1 (ART1) for this purpose. Hsieh and Chen (2004) developed an analytical structure made up of a fuzzy rule-based inference system to help identify defect spatial patterns. Tong *et al.* (2005) used the multivariate Hotelling T^2 control chart that indexes the number of defects and defect clusters as a way to monitor the wafer manufacturing process. The merit of this method is that it simultaneously monitors the number of defects and the presence of the cluster of defects.

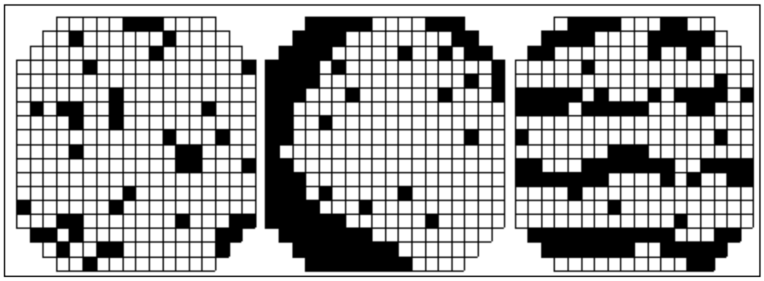
As a wafer gets larger, a spatial inhomogeneity frequently occurs. According to the literature (Bailey and Gatrell 1995), analysis of spatial inhomogeneity is also one of the promising approaches for detecting defective clustering. However, there is very little in the literature about the use of spatial correlogram to analyze defect patterns on the wafer map. This chapter proposes a new methodology based on spatial correlogram to detect the presence of spatial autocorrelations and classify defect patterns. This study is the first attempt to develop a methodology to detect spatial autocorrelation and to classify defect patterns automatically based on a spatial correlogram of a wafer map. After detecting the presence of defect patterns, dynamic time warping (DTW) is adopted to classify defect patterns into one of known patterns automatically. Spatial correlogram based on the proposed method is very robust to random noise, defect location, and defect size on the wafer map.

The remainder of this chapter is organized as follows. Section 2.2 generalizes a couple of join-count based statistics and explores their properties. Section 2.3 describes a spatial correlogram and proposes generalized joint-count based statistic with optimal weights. Section 2.4 contains a visual illustration that uses simulated and real life examples and presents a new spatial randomness test. In Section 2.5, we present the new automatic defect classification methodology and compare its performance with that of neural network. Section 2.6 presents conclusions and some future research topics.

2.2. Spatial Dependences in Wafer Map

2.2.1 Defect patterns and join-counts

The spatial patterns formed by defective chips are broadly categorized into three classes. The most elementary of these is the spatially random pattern. Hansen and Thyregod (1998) and Hansen *et al*. (1997) described this basic pattern by the spatially homogeneous Bernoulli process (SHBP).



(a) SHBP (b) Clustered effect (c) Repetitive pattern

Figure 2.1 Spatial patterns of wafer map

A constructed example of an SHBP wafer map is illustrated in Figure 2.1(a). However, as explained earlier, although this random pattern may be the most basic, the more commonly occurring pattern is spatially nonrandom. Figures 2.1(b) and (c) show wafer maps with a clustered effect and with a repetitive pattern, respectively.

As seen in Figure 2.1, the locations of defective chips are represented on the wafer map (chip-level). It is also possible to make a wafer map showing the locations of defects rather than chips (defect-level). However, this paper does not deal with defect-level wafer maps (see Cunningham and McKinnon (1998) and Jun *et al*. (1999) for defect-level wafer map analysis). The basic idea presented here is to deal with chip-level wafer maps by comparing how many functional chips are around a defective chip and how many defective chips are around a functional chip. This idea can be implemented using the join-count (JC) statistics that are explained below in detail.

A join is formed when two chips are located in the neighborhood of each other. Let H denote a set of neighbors, and let n denote the total number of chips per wafer. The notation $(i, j) \in H$ implies that two chips i and j are neighbors. Therefore, under a certain neighborhood construction system, the number of possible joins is given by

$$c = \sum_{i<j} w_{ij}$$

where

$$w_{ij} = \begin{cases} 1, & (i, j) \in H \\ 0, & \text{elsewhere} \end{cases}$$

We have the following three types of join: 0-to-0 join (between functional chips), 0-to-1 join (between functional and defective chips), and 1-to-1 join (between defective chips). To discriminate between the three joins, we introduce an indicator variable for chip i as

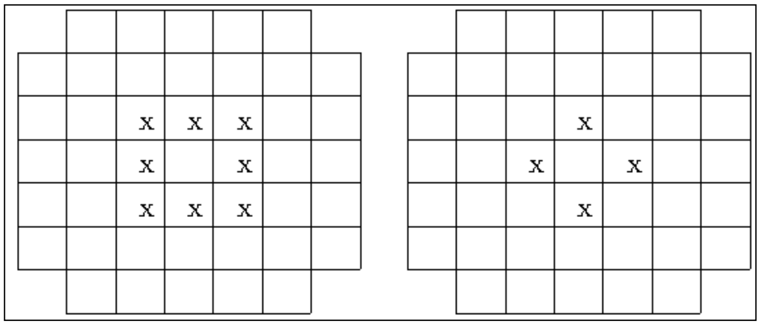
$$x_i = \begin{cases} 1, & \text{defective} \\ 0, & \text{functional} \end{cases}$$

Let c_{00}, c_{01} and c_{11} denote the numbers of 0-to-0, 0-to-1 and 1-to-1 joins, respectively. Then,

$$c_{00} = \sum_{i<j} w_{ij}(1-x_i)(1-x_j)$$
$$c_{01} = \sum_{i<j} w_{ij}(x_i - x_j)^2$$
$$c_{11} = \sum_{i<j} w_{ij}x_i x_j$$

By the definition of c_{00}, c_{01} and c_{11}, $c = c_{00} + c_{01} + c_{11}$.

In practice (c_{00}, c_{01}, c_{11}) depends on the neighborhood construction rule applied. The king-move neighborhood (KMN) and rook-move neighborhood (RMN) construction rules are the most popular. KMN is defined as the region in which the king can move on the chessboard as shown in Figure 2.2 (a). On the other hand, RMN can be defined as the region of one-step rook-moves as shown in Figure 2.2 (b) (Taam and Hamada 1993).



(a) King-move neighbors (b) Rook-move neighbors

Figure 2.2 Neighborhood construction rules

(Ramirez and Taam 2000, Taam and Hamada 1993)

2.2.2 Spatial analysis based upon join-counts

To measure spatially associative effects on the wafer map, Taam and Hamada (1993) proposed the following log odds ratio (LOR) by employing the KMN rule

$$LOR = \log \frac{c_{00} c_{11}}{(c_{01}/2)^2} .$$

Although LOR originally was used to measure the degree of association in a 2-by-2 contingency table, Taam and Hamada (1993) have insisted that it is also capable of investigating spatial patterns formed by process parameters. For example, an attraction of chips with identical characteristics produces a positive LOR value, whereas repulsion of those with different characteristics produces a negative LOR value. Therefore, a positive LOR indicates cluster patterns of defective chips on the wafer map while a negative LOR indicates a repetitive one. Small LOR values around zero can be interpreted as indicators of no evidence of spatial dependence. Based on numerical simulations, they also showed that the expectation of LOR is approximately independent of given yield. For statistical details about LOR, consult Agresti (1990). When an individual JC has a zero value, LOR is calculated using a correction term as follows:

$$LOR = \log \frac{(c_{00} + 0.5)(c_{11} + 0.5)}{(c_{01}/2 + 0.5)^2}$$

Hansen and Thyregod (1998) have described a LOR test as a procedure to identify the statistical significance of spatial patterns. Because the standard error of LOR is approximately

$$\hat{\sigma}_{LOR} = \sqrt{c_{00}^{-1} + c_{11}^{-1} + 4c_{01}^{-1}}$$

for a large sample size (Agresti 1990), the following test statistic was proposed under an SHBP null hypothesis.

$$Z_{LOR} = \frac{LOR}{\hat{\sigma}_{LOR}} \sim N(0,1)$$

Hansen and Thyregod (1998) concluded through numerical experiments that LOR test works well in detecting the presence of spatial dependences, but it is incapable of identifying spatial patterns. In addition, chi-squared statistic is also a measure of association in a 2-by-2 contingency table (Agresti 1990):

$$\chi^2 = c\left[\frac{c_{00}c_{11} - (c_{01}/2)^2}{(c_{00} + c_{01}/2)(c_{11} + c_{01}/2)}\right]^2 .$$

Hansen *et al.* (1997) developed a statistical hypothesis test to routinely monitor wafer maps and also presented an application tool using a classic *p*-chart. A monitoring statistic they used can be written as

$$T = \alpha_0 c_{00} + \alpha_1 c_{11}$$

where α_0 and α_1 are weights to be chosen after consideration of the degree of spatial clustering. However, it seems that this approach is still at the center of arguments of how to construct neighborhood rules and how to choose α_0 and α_1.

The contiguity ratio (CR) proposed by Moran is also known as a measure of spatial autocorrelation and is defined as follows (Cliff and Ord 1981):

$$CR = \frac{n\sum_{i \neq j} w_{ij}(x_i - \bar{x})(x_j - \bar{x})}{2c\sum(x_i - \bar{x})^2} .$$

After some mathematical manipulations, CR can be rewritten as

$$CR = (pc_{00} + qc_{11})/(cpq) - 1 ,$$

where $p = n_1/n$, $q = n_0/n$ and n_0, n_1 are, respectively, the numbers of functional and defective chips on wafer. If the weights in T are chosen as $(\alpha_0, \alpha_1) = (p, q)$, the monitoring statistic T can be rewritten as

$$T = cpq(CR + 1)$$

which indicates that T is equivalent to CR.

2.2.3 Numerical experiments for comparison of spatial clustering

In the Section 2.2.3, we conduct some numerical experiments to compare JC-based statistic: LOR, CR and χ^2. Table 2.1 shows these JC-based statistics depend on the defective rate *p* for a constructed set of 20x20-sized wafer maps. The values presented in the Table 2.1 are averaged over 100 runs at each level of *p*. In the experiments, LOR and CR are standardized using equations below, respectively.

$$\frac{LOR}{\hat{\sigma}_{LOR}}$$

and

$$\frac{CR - (1/(n-1))}{1/\sqrt{c}}.$$

As seen from Table 2.1, all statistics are relatively insensitive to the neighborhood construction rule. Moreover, they are not far away from 0 and 1, which are their respective means. Because there is little difference between the result of KMN rule and that of RMN rule, we choose the RMN rule as the neighborhood construction rule in the subsequent sections.

Table 2.1 Numerical comparison of LOR, CR, and χ^2 using SHBP wafer maps

(a) King-Move Neighborhood, c=1482						
p	c00	c01	c11	LOR	CR	χ^2
0.1	1200.89	266.32	14.79	0.05	0.01	0.87
0.2	947.04	476.69	58.27	-0.06	-0.18	1.03
0.3	723.35	625.29	132.36	-0.06	-0.16	0.82
0.4	532.40	715.10	234.50	-0.09	-0.22	1.11
0.5	367.44	744.31	370.25	-0.07	-0.17	0.96
0.6	235.10	714.69	532.21	-0.08	-0.19	1.19
0.7	133.98	621.29	726.73	0.04	0.07	1.13
0.8	57.69	477.29	946.45	-0.10	-0.27	0.93
0.9	14.90	266.04	1201.06	0.06	0.04	0.67
(b) Rook-Move Neighborhood, c=760						
0.1	615.86	139.45	7.69	0.17	0.04	0.98
0.2	485.77	244.33	29.90	-0.08	-0.12	0.82
0.3	371.50	320.74	67.76	-0.11	-0.12	0.92
0.4	272.30	367.99	119.71	-0.25	-0.25	1.13
0.5	189.08	380.90	190.02	-0.07	-0.07	0.90
0.6	120.69	366.40	272.91	-0.12	-0.12	1.28
0.7	68.62	318.81	372.57	0.05	0.04	1.47
0.8	29.66	244.80	485.54	-0.13	-0.17	0.79
0.9	7.76	136.26	615.98	0.20	0.07	0.92

2.3 Spatial Correlogram for Representation of Spatial Correlations

As pointed out by Hansen and Thyregod (1998) a single monitoring statistic is insufficient to represent a variety of widespread patterns across the wafer map. To overcome this drawback, this chapter proposes new approach to identify spatial patterns on the wafer map by using a spatial correlogram. A spatial correlogram represents the correlation between values of the same variable at different locations. Although spatial correlogram has been widely used in diverse fields of science such as geography, ecology, and the environment (Cliff and Ord 1981, Pierre and Louis 1998), to our knowledge, no study has reported the use of spatial correlogram for analysis of wafer maps. Spatial correlogram gives more useful information for the monitoring of defect patterns that appear on wafer maps because this technique can definitively describe spatial dependence, a phenomenon known as spatial autocorrelation, for spatial data (Bailey and Gatrell 1995). Based on illustrative examples, we will show that this new approach has the potential to outperform others that are based upon a single statistic.

Let $H(g)$ denote a set of *g*th-order neighbors, defined as chips that are g distant from each other. Consequently, $(i, j) \in H(g)$ implies that the two chips *i* and *j* are *g*th-order neighbors of one another. Moreover, $H(g)$ is also considered as a set of joins of which the length is equal to *g*. Join length corresponds exactly to the distance between the two chips involved. This study uses Manhattan distance which computes the distance from the chip $p_1(x_1, y_1)$ to the $p_2(x_2, y_2)$ as $d(p_1, p_2) = |x_1 - x_2| + |y_1 - y_2|$ to determine the distance between two chips because it is consistent to RMN rule used in this work.

If the distance between two chips is denoted by $d(i, j)$, $H(g)$ can be written as

$$H(g) = \{(i, j) \in W \mid d(i, j) = g\} \text{ for } g = 1, 2, \cdots, m$$

where *W* is a collection of all possible joins within the wafer map and m is a maximum length of join. Therefore, the number of *g*th-order joins is

$$c(g) = \sum_{i<j} w_{ij}(g),$$

where

$$w_{ij}(g) = \begin{cases} 1, & (i, j) \in H(g) \\ 0, & \text{elsewhere} \end{cases}$$

Based on several statistics, as we mentioned in Section 2.2, generalized JC-based statistic with *g*th-order neighbors as a measure of spatial autocorrelation can be written as follows:

$$T(g) = \alpha_0 f(c_{00}(g)) + \alpha_1 f(c_{11}(g))$$

where $c_{00}(g)$ and $c_{11}(g)$ are the number of *g*th-order 0-to-0 and 1-to-1 joins and $f(\cdot)$ stands for a monotonic function such as identity function or log function. We present the Lemma 2.1 to find the optimal weights that minimize the variance of *T*(*g*).

Lemma 2.1: For a generalized JC-based statistic with gth-order neighbors, i.e.

$$T(g) = \alpha_0 f(c_{00}(g)) + \alpha_1 f(c_{11}(g)),$$

the optimal weights that minimize the variance of *T*(*g*) when $f(\cdot)$ is identity function, are given as follows:

$$(\alpha_0, \alpha_1) = (p, q) \text{ subject to } \alpha_0 + \alpha_1 = 1.$$

Proof of Lemma 2.1

Given n_0 and n_1, the first and the second moments of $c_{00}(g)$ and $c_{11}(g)$, respectively, are obtained as follows (Cliff and Ord 1981, Hansen *et al*. 1997).

$$E(c_{00}(g)) = c(g) n_0^{(2)} / n^{(2)}$$

$$E(c_{11}(g)) = c(g) n_1^{(2)} / n^{(2)}$$

$$E(c_{00}^2(g)) = \sum_{k=1}^{4} b_k(g) n_0^{(k)} / n^{(k)}$$

$$E(c_{11}^2(g)) = \sum_{k=1}^{4} b_k(g) n_1^{(k)} / n^{(k)}$$

$$E(c_{00}(g) c_{11}(g)) = b_4(g) n_0^{(2)} n_1^{(2)} / n^{(4)}$$

where $c(g) = c_{11}(g) + c_{00}(g) + c_{01}(g)$, $b_1(g) = 0$, $b_2(g) = s_1(g)/4$, $b_3(g) = (s_2(g) - 2s_1(g))/4$, $b_4(g) = (s_0^2(g) - s_2(g) + s_1(g))/4$ and $m^{(k)} = \prod_{i=1}^{k}(m - i + 1)$ for a positive integer m. See Cliff and Ord (1981) to find $s_0(g)$, $s_1(g)$ and $s_2(g)$. Accordingly, we can have the expectation and the variance of $T(g)$ as

$$E(T(g)) = c(g)(\alpha_0 n_0^{(2)} + \alpha_1 n_1^{(2)})/n^{(2)}$$

and

$$\begin{aligned} Var(T(g)) = s_1(g)&\left[\alpha_0^2\left(\frac{n_0^{(2)}}{n^{(2)}} - 2\frac{n_0^{(3)}}{n^{(3)}} + \frac{n_0^{(4)}}{n^{(4)}}\right) + \alpha_1^2\left(\frac{n_1^{(2)}}{n^{(2)}} - 2\frac{n_1^{(3)}}{n^{(3)}} + \frac{n_1^{(4)}}{n^{(4)}}\right) + 2\alpha_0\alpha_1\frac{n_0^{(2)}n_1^{(2)}}{n^{(4)}}\right] \\ + s_2(g)&\left[\alpha_0^2\left(\frac{n_0^{(3)}}{n^{(3)}} - \frac{n_0^{(4)}}{n^{(4)}}\right) + \alpha_1^2\left(\frac{n_1^{(3)}}{n^{(3)}} - \frac{n_1^{(4)}}{n^{(4)}}\right) - 2\alpha_0\alpha_1\frac{n_0^{(2)}n_1^{(2)}}{n^{(4)}}\right] \\ + s_0^2(g)&\left[\alpha_0^2\left(\frac{n_0^{(4)}}{n^{(4)}} - \left\{\frac{n_0^{(2)}}{n^{(2)}}\right\}^2\right) + \alpha_1^2\left(\frac{n_1^{(4)}}{n^{(4)}} - \left\{\frac{n_1^{(2)}}{n^{(2)}}\right\}^2\right) - 2\alpha_0\alpha_1\left(\frac{n_0^{(2)}n_1^{(2)}}{n^{(4)}} - \frac{n_0^{(2)}n_1^{(2)}}{n^{(2)}n^{(2)}}\right)\right] \end{aligned}$$

respectively. Because $n_0^{(k)}/n^{(k)} \cong q^k$ and $n_1^{(k)}/n^{(k)} \cong p^k$ for a large value of n, the variance of T is approximately derived by

$$Var(T(g)) = s_1(g)p^2q^2(\alpha_0 + \alpha_1)^2 + (s_2(g) - 4s_0^2(g)/n)pq(\alpha_0 q - \alpha_1 p)^2$$

Minimizing the above equation subject to $\alpha_0 + \alpha_1 = 1$, we can find an optimum solution

$$(\alpha_0^*, \alpha_1^*) = (p, q).$$

Based on this result, the corresponding expectation and variance are respectively

$$E(T(g)) = c(g)pq$$

$$Var(T(g)) = c(g)p^2q^2.$$

Remark 2.1: In order to understand a variance reduction gain by the optimal choice of weights compared with $(\alpha_0, \alpha_1) = (0.5, 0.5)$, Figure 2.3 shows the relative efficiency (RE) of the optimal choice of weights. RE is computed as follows: $RE = \dfrac{\mathrm{var}[T(\alpha_0^*, \alpha_1^*)]}{\mathrm{var}[T(0.5, 0.5)]}$ where $\mathrm{var}[T(\alpha_0^*, \alpha_1^*)]$ is the variance of T with optimal weight values α_0^* and α_1^* and $\mathrm{var}[T(0.5, 0.5)]$ is the variance of T with weight values $\alpha_0 = 0.5$ and $\alpha_1 = 0.5$. Figure 2.3 indicates that the variance reduction gain with the optimum weights becomes large as wafer yield becomes higher when RMN is used. However, the difference becomes negligible as the size of the wafer increases.

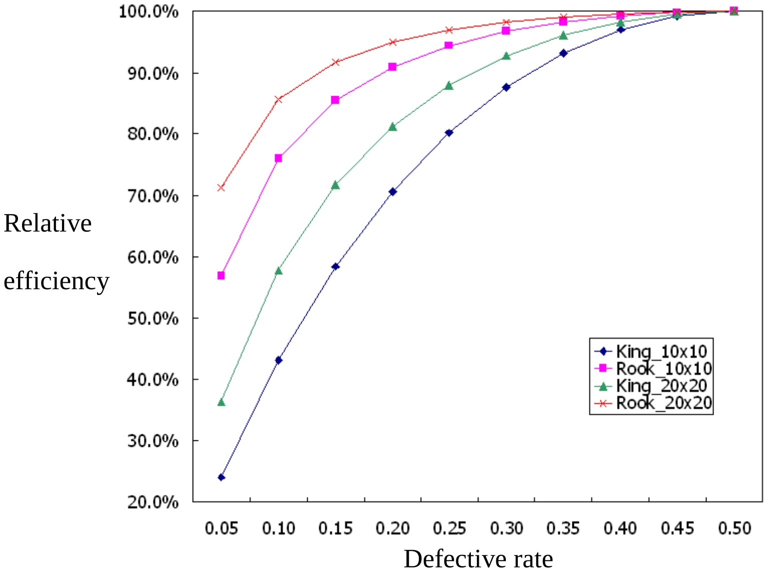


Figure 2.3 Relative efficiency of the optimum choice of weights compared with $(\alpha_0, \alpha_1) = (0.5, 0.5)$

Based on the Lemma 2.1, the *g*th-order *T*(*g*) can be simplified as follows:

$$T(g) = pc_{00}(g) + qc_{11}(g).$$

In addition, the mean and variance of statistic *T*(*g*) are given by the following equations (see the Proof of Lemma 2.1):

$$\mathrm{E}[T(g)] = c(g)pq$$

$$\mathrm{V}\left[T(g)\right] = c(g)\,p^2 q^2$$

Based on the central limit theorem, the standardized statistic *T*(*g*) approximates the standard normal distribution, i.e.,

$$Z_T(g) = \frac{T(g) - c(g)pq}{\sqrt{c(g)p^2q^2}} \ \sim\ N(0,1) \ \text{ as } c(g) \to \infty, \tag{2.1}$$

where *g* is *g*th-order neighbor and $c(g) = c_{00}(g) + c_{11}(g) + c_{01}(g)$.

2.4 Illustrative Case Study

This section presents simulated and real-life examples to illustrate the proposed approach.

2.4.1 Simulated examples

Before investigating spatial correlogram for defect pattern classification, we present formal randomness test that combines the test statistic with multiple spatial lags. There are several test statistics for spatial randomness testing such as LOR test and CR test as mentioned in Section 2.2. However, they are not applicable for spatial randomness test using multiple spatial lags because test statistic using multiple spatial lags should take into account all test values with different lags simultaneously. As we mentioned in Section 2.3, $Z_T(g)$ is approximately normally distributed when $c(g)$ is large. If we let $X_r = \left[Z_T(1), Z_T(2), \ldots, Z_T(r)\right]$, which is a collection of $Z_T(g)$ for the first *r* spatial lags, then X_r follows approximate multivariate normal distribution with mean of zero vector of length *r* and covariance $\sum_r$ under SHBP condition. For spatial randomness test using the first *r* spatial lags, we can use the following test statistic:

$$T_H{}^2(r) = \mathbf{x}_r{}'\hat{\Sigma}_r{}^{-1}\mathbf{x}_r\,,$$

where $\hat{\Sigma}_r$ is the estimated covariance matrix using *m* samples of wafer maps. Because the samples are individual observations (i.e., individual wafer maps), an approximate critical limit is given by (Montgomery 2005)

$$\text{CL} = \frac{r(m-1)}{m-r} F_{\alpha,r,m-r}\,.$$

Table 2.2 shows the comparison results of the proposed spatial randomness test for 150 SHBP wafer maps in terms of test accuracy with popular LOR and CR tests ($\alpha = 0.05$). We have used the total 150 of SHBP wafer maps for each defective rate ranging from 0.1 to 0.5. Overall, existing test procedures performed better for low defective rate (*p*=0.1) while our proposed procedure showed the improved performance for larger defective rates (*p*≥0.2). In our testing procedure, as defective rate (*p*) is getting larger, test statistic using larger spatial lags produced better accuracy. Therefore, 2~4 spatial lags is recommended for spatial randomness test for smaller defective rate (*p*≤0.3) whereas 5~7 is recommended for larger defective rates.

Table 2.2 Summary of spatial randomness testing for SHBP wafer maps

p	LOR	CR	$T_H^2(1)$	$T_H^2(2)$	$T_H^2(3)$	$T_H^2(4)$	$T_H^2(5)$	$T_H^2(6)$	$T_H^2(7)$	$T_H^2(8)$	$T_H^2(9)$	$T_H^2(10)$
0.1	98.0%	98.0%	96.0%	96.7%	96.7%	98.0%	95.3%	95.3%	98.0%	97.0%	96.0%	96.7%
0.2	96.7%	96.7%	97.3%	98.0%	96.0%	97.3%	97.3%	97.3%	97.7%	97.3%	94.7%	94.7%
0.3	92.0%	92.7%	94.7%	98.0%	98.0%	96.7%	96.0%	97.3%	96.0%	96.0%	94.0%	96.0%
0.4	95.3%	95.3%	96.7%	96.0%	98.0%	96.7%	97.7%	97.3%	96.7%	96.3%	94.1%	93.3%
0.5	92.7%	94.0%	96.7%	94.0%	94.0%	93.3%	96.0%	96.7%	97.3%	97.3%	94.7%	95.0%

However, as mentioned earlier, previous randomness test is insufficient to recognize a variety of widespread defect patterns across the wafer map. This study proposes new approach to detect spatial defect patterns using spatial correlogram. We construct 20x20-sized wafer maps as specified SHBP, cluster, circle, repetition, and mixed patterns. The generation of the simulated wafer maps is based on the previous literature (DeNicolao *et al.* 2003) except the SHBP. SHBP wafer maps are produced using random number generator. Figure 2.4 shows those simulated wafer maps used to create spatial correlograms in Figure 2.5.

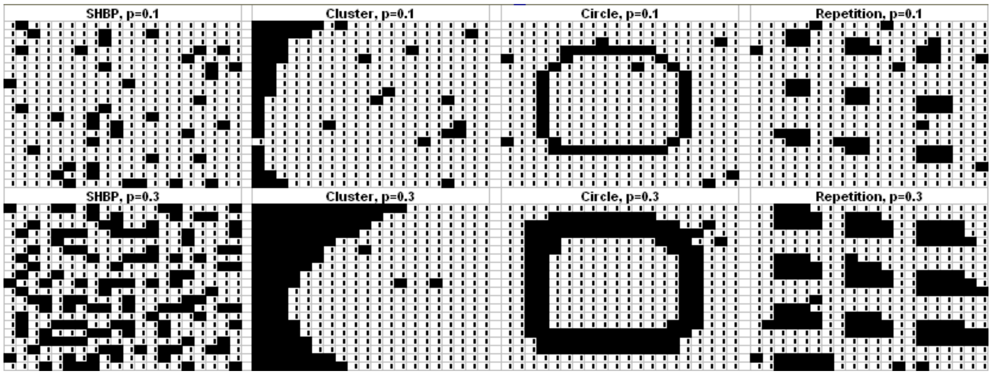
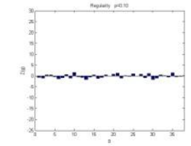
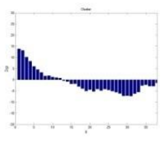
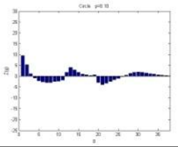
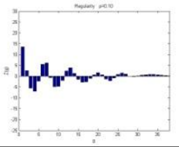
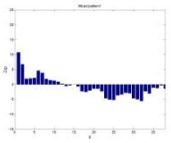
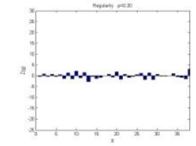
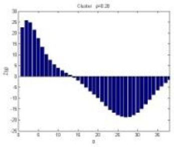
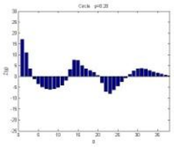
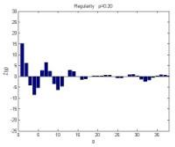
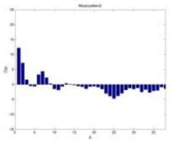
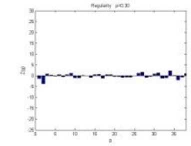
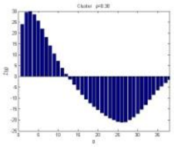
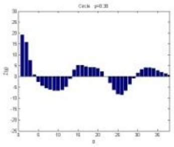
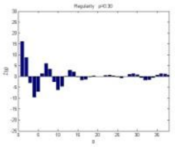
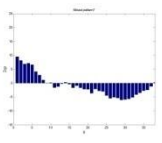
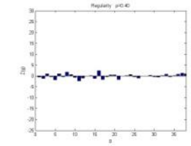
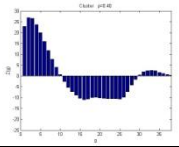
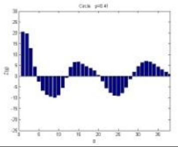
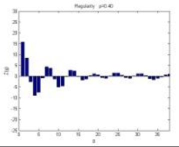
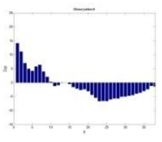
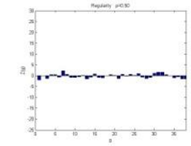
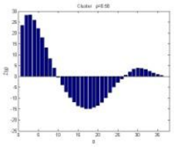
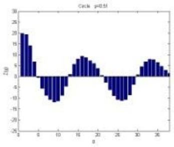
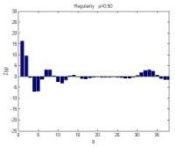
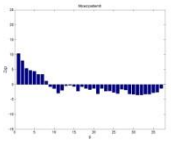


Figure 2.4 Simulated wafer maps

p	(a) SHBP	(b) Cluster	(c)Circle	(d) Repetition	(%)	(e) Mixed pattern*
0.1					S=10 C=10 R=10	
0.2					S=10 C=10 R=20	
0.3					S=10 C=20 R=20	
0.4					S=20 C=20 R=10	
0.5					S=20 C=20 R=20	

* S=SHBP, C=cluster pattern, R= repetitive pattern

Figure 2.5 Spatial correlograms of simulated wafer maps

Figure 2.5 illustrates the capability of spatial correlogram in order to discriminate among different spatial patterns. In Figure 2.5, the $Z_T(g)$ values computed from different defective rate p using Eq. (2.1) are displayed along the spatial lag g. In case of SHBP as shown in Figure 5(a), most of the $Z_T(g)$ values fluctuate around zero. SHBP has no unique shape of the correlogram. In case of the cluster patterns, $Z_T(g)$ smoothly changes along spatial lag g and its absolute values are relatively larger. The characteristic of a spatial correlogram of a circle pattern is that the absolute value of $Z_T(g)$ is also large like that of cluster pattern, but the circle pattern contains a soft cosine waveform. $Z_T(g)$ values of repetitive pattern are consistently small and no special pattern appears except for a frequent crossing around zero.

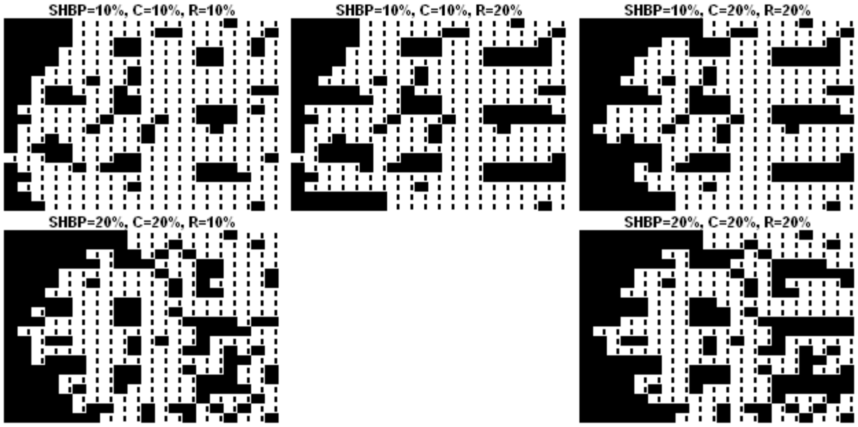


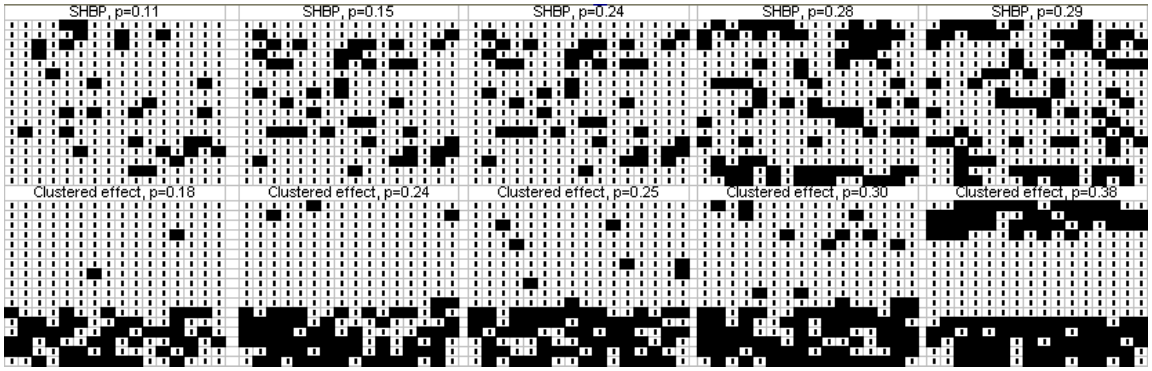
Figure 2.6 Wafer map with mixed effects (C: cluster pattern, R: repetitive pattern)

Finally, we attempt to simulate mixed effects by superimposing three types of wafer maps: SHBP, cluster pattern, and repetitive pattern. Figure 2.6 shows some examples of wafer maps with mixed effects of cluster pattern and repetitive pattern and their corresponding correlograms are shown in Figure 2.5(e). Interestingly, the distinctive patterns produced by the cluster and repetitive patterns are preserved under the superimposed models. It is indicated that global shape of spatial correlogram is similar to cluster's one and at the same time it locally contains the characteristic of repetitive pattern's one.

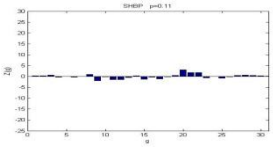
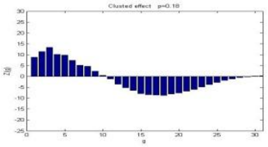
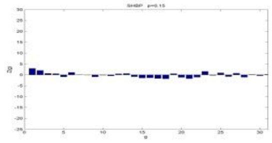
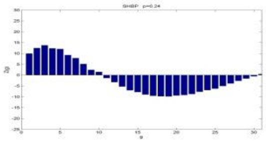
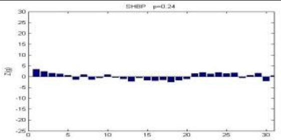
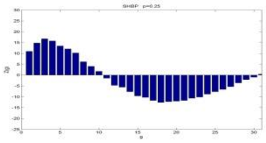
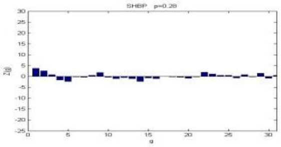
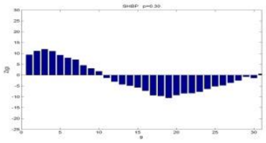
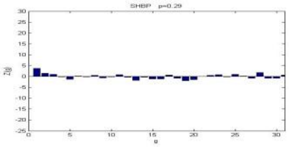
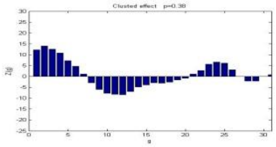
2.4.2 Real-life examples

Real-life wafer maps provided by a semiconductor manufacturing company were analyzed using the proposed approach. Each of wafer maps consists of 268 chips as shown in Figure 2.7 (a). The defective rate p in Figure 2.7 is calculated as followed: $p = \frac{b}{N}$ where b is the number of defective chips and N is the number of total chips on the wafer. In Figure 2.7 (b), it is observed that there is a distinction between spatial correlograms of cluster pattern and those of SHBP. In case of the clustered effects, $Z_T(g)$ changes smoothly along spatial lag g, and its absolute values are relatively larger. Not so with SHBP in which a frequent crossing around zero occurs.

Illustrative examples show that a spatial correlogram has the potential to identify defect patterns in semiconductor wafers. Information drawn by a spatial correlogram includes not only simple examination of spatial dependence for defective chips, but also recognition of a defect pattern in a wafer. Moreover, we can discover several benefits of our approach. A spatial correlogram is robust to defect location, robustness to defect size, and robustness to random noise. More details on the advantage of the proposed method will be explained in Section 2.5.



(a) Real-life wafer maps

p	SHBP	P	Cluster
0.11		0.18	
0.15		0.24	
0.24		0.25	
0.28		0.30	
0.29		0.38	

(b) Spatial correlograms

Figure 2.7 Real-life wafer maps and their corresponding spatial correlograms

2.5 Automatic Classification of Defect Patterns

This section presents new classification methodology based on dynamic time warping (DTW) using correlogram and compares its performance with that of popular neural network approach (Hsu and Chen 2007, Huang 2007, and Palma *et al*. 2005).

2.5.1 Review of dynamic time warping

As seen earlier, defect patterns of same class produce similar shapes of correlogram. In order to classify defect patterns based on correlogram, we have to first calculate the distance among different correlograms and then use the classification techniques that use distance measures. In our work, we use the 1-nearest neighbor classifier. However, since they are not aligned in the lag axis, linear mapping technique such as Euclidean distance that assumes ith point in one correlogram is aligned with the ith point in the other may produce higher misclassification rate. Figure 2.8 shows examples of classification based on Euclidean distance and DTW distance. To accurately classify each defect pattern using correlogram, non linear mapping technique is needed.

The classification based on DTW distance, which is popular in speech recognition applications, finds an optimal match between two sequences by allowing a non linear mapping of the one sequence to another by minimizing the distance between the two (Ratanamahatana and Keogh 2004a, 2004b).

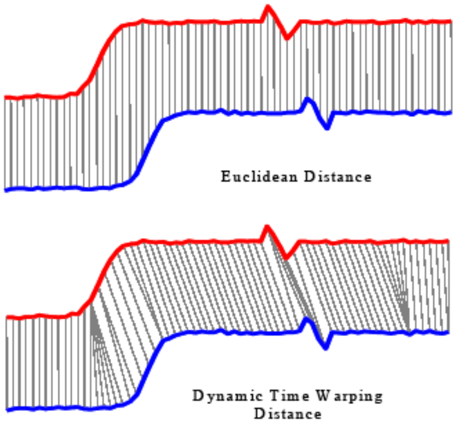


Figure 2.8 Classification of sequence based on Euclidean distance and DTW distance (Ratanamahatana and Keogh 2004a)

Let's suppose a sequence S of length m, $S = s_1, s_2, \ldots, s_i, \ldots, s_m$ and a sequence R of length n, $R = r_1, r_2, \ldots, r_j, \ldots, r_n$. We create n-by-m path matrix where the (i^{th}, j^{th}) element of matrix contains the distance between the two points s_i and r_j such as $d(s_i, r_j) = (s_i - r_j)^2$. The best match between these two sequences is the one for which there is the lowest distance path aligning the one sequence to the other. The optimal path is the path that minimizes the warping cost

$$DTW(S, R) = \min \sqrt{\sum_{k=1}^{K} w_k} \ ,$$

where w_k is the matrix element $(i, j)_k$ that also belongs to kth element of a warping path W, a contiguous set of matrix elements that represent a mapping between S and R (see Ratanamahatana and Keogh (2004a) and Ratanamahatana and Keogh (2004b) for detailed descriptions of DTW).

2.5.2 Experimental results

In order to evaluate the classification performance of the proposed algorithms, we generated a total of 400 wafer maps with 400 chips per wafer (20 by 20-sized map), i.e., 80 wafer maps for each of five patterns such as SHBP, circle, cluster, repetition and spot. We have eight level of random noise ranging from 0.05, 0.1, 0.15, …, 0.4. For each combination of noise level and pattern (total 8x5=40 combinations), we generated 10 wafer maps. Dataset {1} consists of wafer maps with the noise level of 0.05, dataset {2} with the noise level of 0.1, and so on. In this experiment, we divided 400 wafer maps into four different data sets as shown in Table 2.3.

Figure 2.9 presents typical four classes of defect patterns. We used the procedure proposed by DeNicolao *et al*. (2003) to generate the simulated data set. Based on our proposed spatial randomness test (α =0.05) using spatial lags 3, 98.8% SHBP wafer maps were accepted while all wafer maps with spatial defect patterns were rejected.

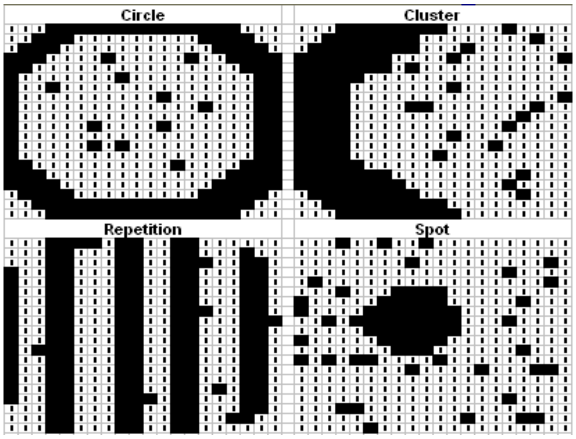


Figure 2.9 Typical defect patterns of wafer map

Four-fold cross validation (CV) is implemented for the comparison of classification accuracy of different procedures. Binary and multi-lags based supervised multilayer perceptron neural network (Huang 2007) is selected for comparison with the proposed method. The difference between the binary neural network and multi-lags neural network is the input vector. The type of input vector of binary neural network is "1" or "0" while that of multi-lags neural network is the $Z_T(g)$ values along spatial lag g. For instance, in case of wafer map with 20 by 20 size, binary neural networks have 400 (=20x20) binary values ("0" or "1") as input vector. On the other hand, multi-lags neural networks have a total 38 of $Z_T(g)$ as its input vector because a total number of spatial lag under rook-move neighborhood (RMN) rule of 20 by 20 sized wafer map is 38.

The architecture of neural network is composed as follows: 400 neurons in the input layer for binary neural network and 38 neurons for multi-lags neural network, single hidden layer with 10 neurons, and 1 output neurons. Tangent sigmoid function and linear transfer function are used for activation function in the hidden and output layer. On the other hand, multi-lags DTW utilizes a number of 38 of $Z_T(g)$ because a maximum number of spatial lag under RMN rule of 20 by 20 sized wafer map is 38 which shows best performance.

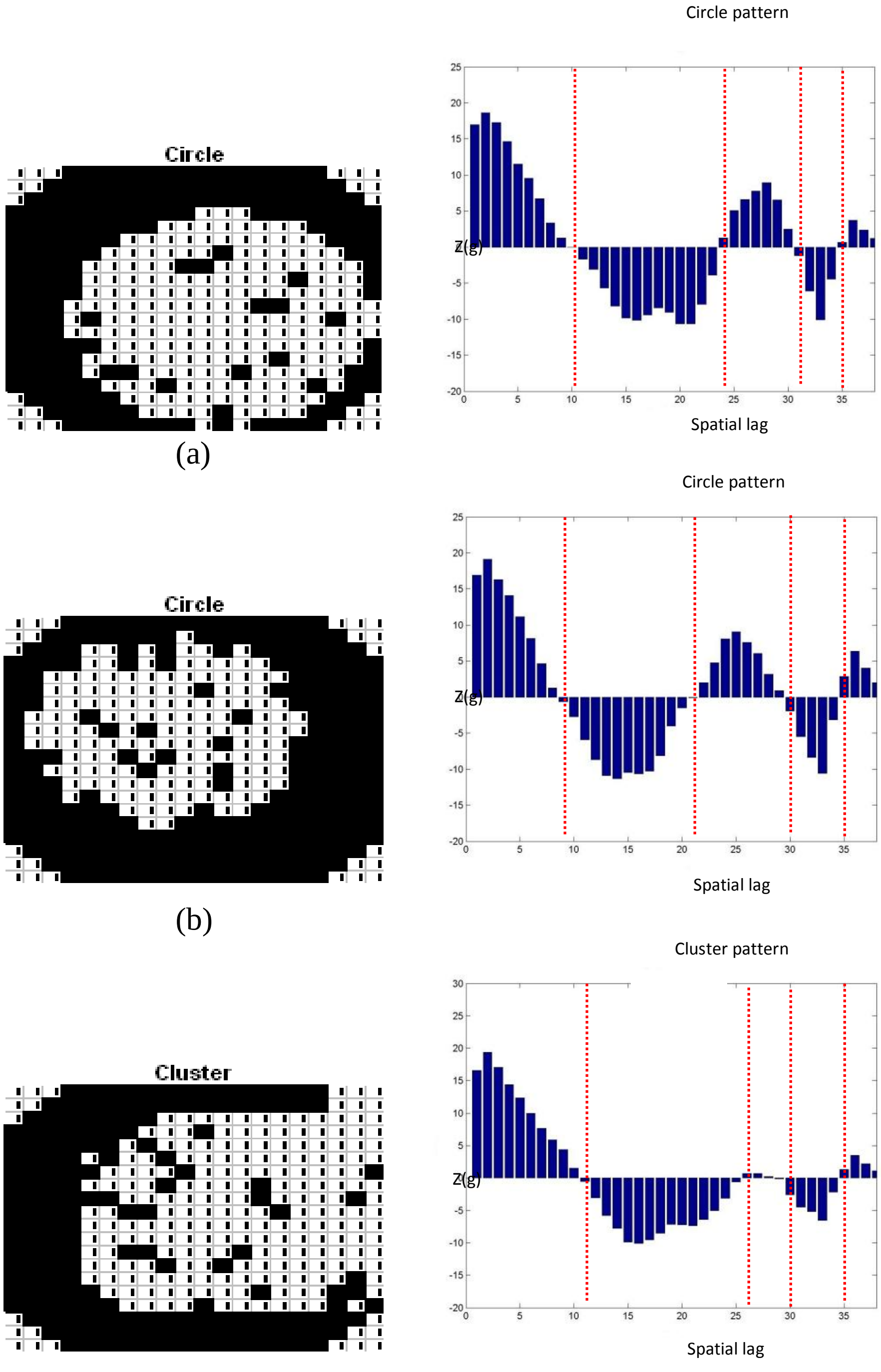
Table 2.3 Summary of classification performance

Testing set	Binary NNet	Multi-lags NNet	Euclidean distance	DTW
{1},{2}	81.3%	78.8%	92.5%	98.8%
{3},{4}	73.8%	71.3%	86.3%	92.5%
{5},{6}	58.8%	62.5%	77.5%	82.5%
{7},{8}	58.8%	71.3%	83.8%	88.8%
Average	68.2%	71.0%	85.0%	90.6%

Table 2.3 shows the accuracy of four procedures for both average and each fold of four-fold CV datasets. Overall, the proposed method is better than other ones. Especially, the accuracy of DTW outperforms that of Euclidean distance. The experimental results show that multi-lags based DTW is promising alternative for automatic defect classification of wafer map.

Figure 2.10 shows why DTW works to classify diverse defect types. Figure 2.10(a)-(c) show the testing wafer map, best matching wafer maps by DTW and Euclidean distance with their corresponding correlograms, respectively. The changes of defect location and size make some horizontal shift of correlograms. The classification based on DTW distance finds an optimal match between two correlograms by allowing a non linear mapping of the one correlogram to another by minimizing the distance between the two as shown in Figure 2.8.

DTW accurately classifies the new wafer map into circle pattern whereas Euclidean distance misclassifies it into cluster pattern. In specific, the distance between X_{new} and X_{circle} by DTW is $d^{DTW}_{(X_{new},X_{circle})}$=47.6 while the distance between X_{new} and $X_{cluster}$ by DTW is $d^{DTW}_{(X_{new},X_{cluster})}$=257.3, so the new wafer map is classified into circle pattern based on the 1-nearest neighbor classifier. On the other hand, the distance between X_{new} and X_{circle} by Euclidean distance is $d^{ED}_{(X_{new},X_{circle})}$=26.4 while the distance between X_{new} and $X_{cluster}$ by Euclidean distance is $d^{ED}_{(X_{new},X_{cluster})}$=19.1 , so the new wafer map is classified into cluster pattern.



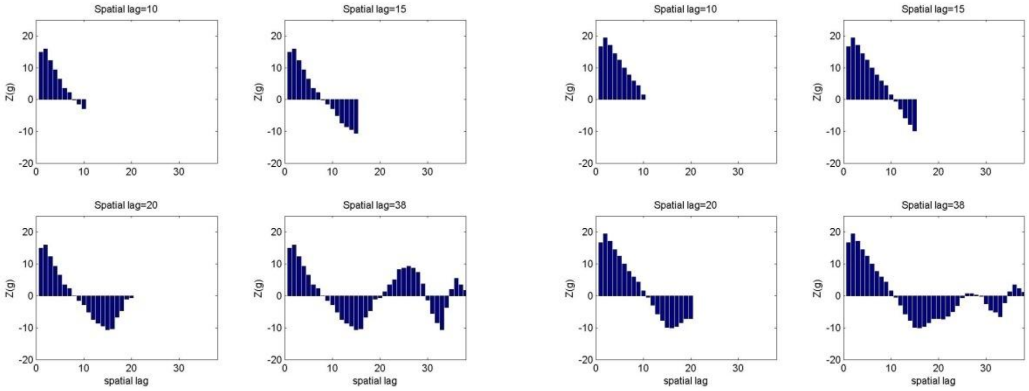
(a)

(b)

(c)

Figure 2.10 Classification results using DTW and Euclidean distance. (a) Testing wafer map and corresponding correlogram. (b) Best matching wafer map by DTW and corresponding correlogram. (c) Best matching wafer map by Euclidean distance and corresponding correlogram

For classification of spatial correlogram based on the 1-nearest neighbor classifier (or other classifiers using distance measures), it is important to compute the distance between two correlograms. However, some defect patterns cannot be clearly be discriminated using small number of spatial lags. For example, as shown in Figure 2.11, circle and cluster patterns cannot be clearly discriminated using smaller spatial lags while large number of spatial lags (≤30) clearly do.



(a) Circle pattern (b) Cluster pattern

Figure 2.11 Spatial correlogram of circle and cluster patterns

In order to explore an optimal spatial lags for the proposed classification method, Table 2.4 shows the classification accuracy of DTW with different spatial lags. As shown in Table 2.4, larger spatial lags are used, better classification accuracy is obtained. Therefore, for the classification purpose, full number of spatial lags is suggested for accurate classification.

Table 2.4 Classification accuracy of DTW with different spatial lags

Testing set	Lags=10	Lags=20	Lags=30	Lags=38
{1},{2}	81.3%	73.8%	98.8%	98.8%
{3},{4}	72.5%	82.5%	90.0%	92.5%
{5},{6}	61.3%	67.5%	76.3%	82.5%
{7},{8}	60.0%	72.5%	77.5%	88.8%
Average	68.8%	74.1%	85.7%	90.6%

To investigate the effectiveness of multi-lags based DTW, the details of classification performance of testing set {3, 4} are shown in the Table 2.5. As seen in Table 2.5, multi-lags based DTW misclassify circle pattern into cluster pattern or in opposite because in some cases, the distinction between spatial correlograms of two patterns is not clear due to high random noise. It can accurately classify repetitive and spot pattern regardless to random noise.

Table 2.5 Detail of DTW performance of testing set {3, 4}

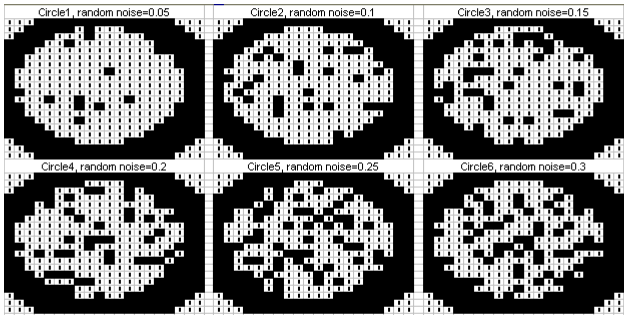
Type	Accuracy	Misclassification description
Circle	18/20 (90%)	Circle → Cluster
Cluster	16/20 (80%)	Cluster → Circle Cluster → Spot
Repetition	20/20 (100%)	None
Spot	20/20 (100%)	None

2.5.3 Test of robustness to noise, defect location, and defect size

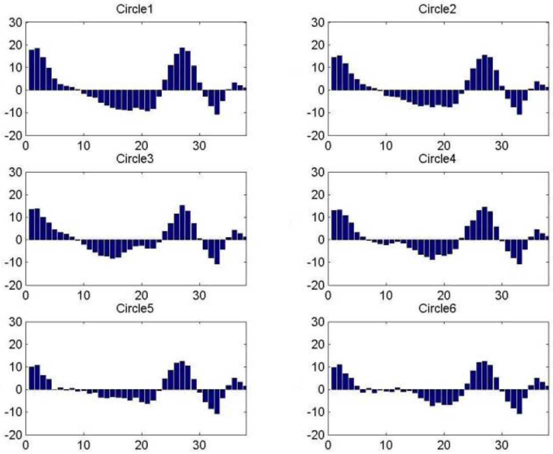
This section compares the multi-lags based DTW with existing algorithm such as binary neural networks in terms of the robustness to random noise, defect location, and defect size.

2.5.3.1 Robustness to random noise

Figure 2.12(a) shows same circle patterns with different random noise level ranging 0.05 to 0.3. Figure 2.12(b) presents correlograms of circle patterns in Figure 2.12(a). Each correlogram is similar regardless of random noise level.


Circle1, random noise=0.05
Circle2, random noise=0.1
Circle3, random noise=0.15
Circle4, random noise=0.2
Circle5, random noise=0.25
Circle6, random noise=0.3

(a) Wafer maps of circle patterns



(b) Spatial correlograms

Figure 2.12 Circle patterns with different levels of random noise and their corresponding spatial correlograms

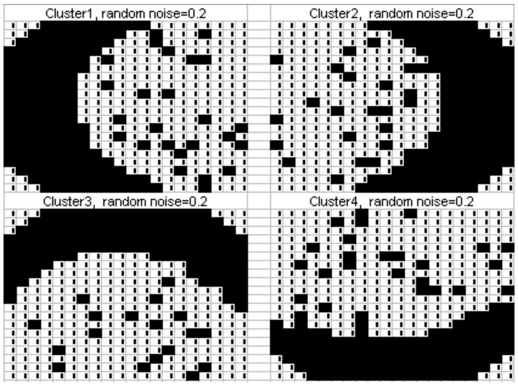
Table 2.6 summarizes the classification accuracy of circle patterns with different noise level from the experiments in Section 2.5.1. Our proposed procedure is robust to random noise whereas accuracy of other techniques decreases as noise level becomes large.

Table 2.6 Classification accuracy of circle pattern with different noise level

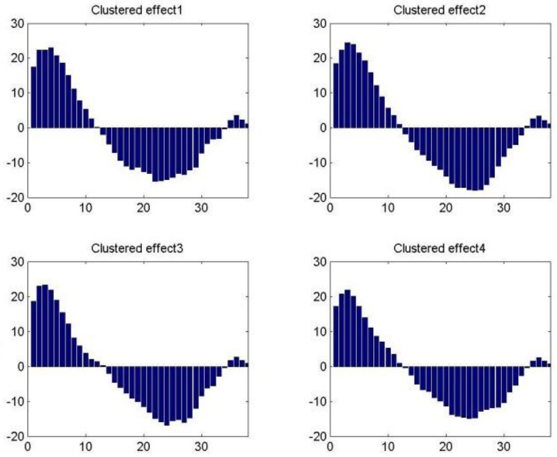
Random noise Level	Binary NNet	Multi-lags NNet	DTW
0.05 ~ 0.1	100%	100%	100%
0.15 ~ 0.2	95%	35%	90%
0.25 ~ 0.3	50%	35%	85%

2.5.3.2 Robustness to defect location

Figure 2.13(a) shows cluster patterns with different locations for a fixed random noise rate of 0.2 and their corresponding spatial correlograms are shown in Figure 2.13(b). The correlograms are almost same without regard to defect location.


Cluster1, random noise=0.2
Cluster2, random noise=0.2
Cluster3, random noise=0.2
Cluster4, random noise=0.2

(a) Wafer maps of cluster patterns



(b) Spatial correlograms

Figure 2.13 Cluster patterns with different defect locations and their corresponding spatial correlograms

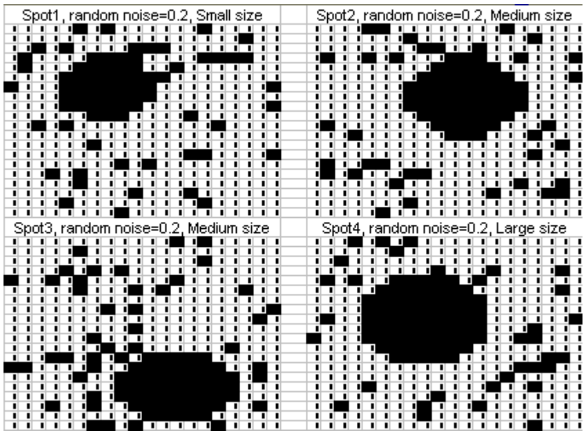
Table 2.7 summarizes the classification accuracy of cluster patterns with different defect locations from the experiments in Section 2.5.1. Our proposed procedure is robust to defect locations while other procedures show some classification errors for different defect locations.

Table 2.7 Classification accuracy of cluster pattern with different defect location

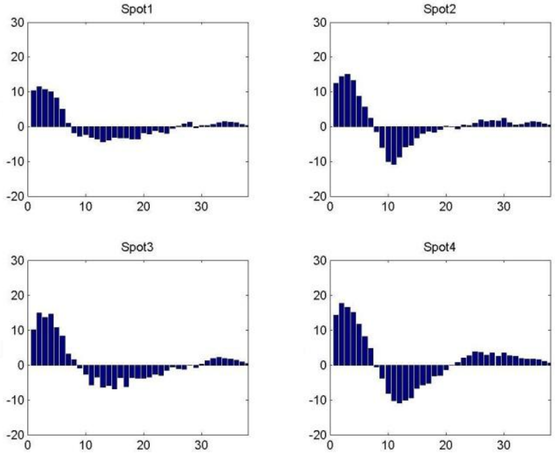
Defect location	Binary NNet	Multi-lags NNet	DTW
Right	80%	60%	100%
Left	60%	80%	100%
Up	80%	80%	100%
Down	60%	80%	100%

2.5.3.3 Robustness to defect size

Figure 2.14(a) shows spot pattern with different defect size for a random noise rate of 0.2 and their corresponding correlograms are shown in Figure 2.14(b). The shape of correlogram looks somewhat different, but unique characteristics of spot pattern such as three waves are preserved.


Spot1, random noise=0.2, Small size
Spot2, random noise=0.2, Medium size
Spot3, random noise=0.2, Medium size
Spot4, random noise=0.2, Large size

(a) Wafer maps of spot patterns



(c) Spatial correlograms

Figure 2.14 Spot patterns with different defect size and their corresponding spatial correlograms

Table 2.8 shows the classification accuracy of spot pattern with different defect size. Our proposed procedure shows slightly better robust performance to different defect size compared to neural network-based approaches.

Table 2.8 Classification accuracy of spot pattern with different defect size

Defect size	Binary NNet	Multi-lags NNet	DTW
Small	100%	70%	100%
Medium	90%	80%	100%
Large	70%	80%	90%

2.6 Concluding Remarks

Although an analysis of wafer map helps to better understand ongoing process problems, defect classification cannot be easily identified automatically. This chapter proposes a new methodology which incorporates spatial correlogram and DTW to detect the anomaly defect patterns and classify them into one of existing spatial defect patterns. The new spatial randomness test procedure based on spatial correlogram of a wafer map is used to detect anomaly defect patterns whereas the 1-nearest neighbor classifier using DTW distance with correlogram input is used to classify its corresponding anomaly defect type. Simulation studies show that the proposed methodology is generally more effective in detecting and classifying spatial defect patterns on the wafer map than those methods that use single lag. The experimental results show that our novel methodology is robust to random noise, defect location and defect size. Therefore, this study can be expected to make a contribution to the monitoring and diagnosis of IC manufacturing processes.

As for further study, there is a need to develop more advanced classification techniques of spatial patterns based on spatial correlogram. Also, we may extend our proposed approach to the wafer bin map that is more informative than the binary wafer map.

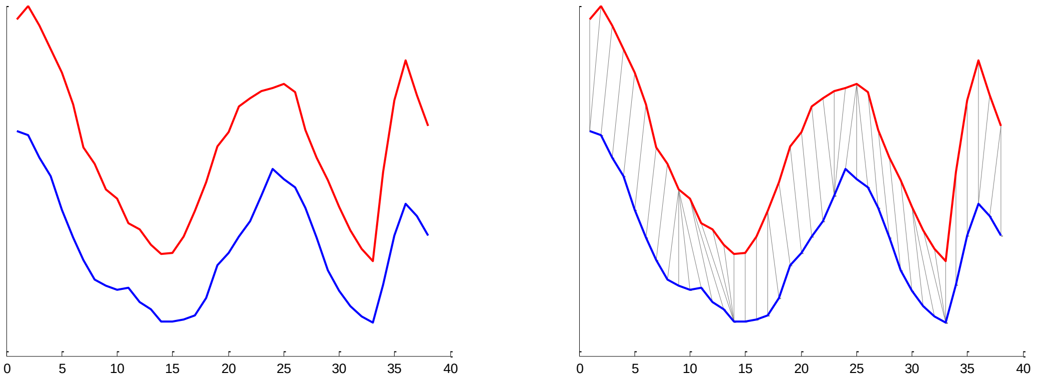
CHAPTER 3

Weighted Dynamic Time Warping for Time Series Classification and Clustering

3.1 Introduction

There has been a long-standing interest for time series classification and clustering in diverse applications such as pattern recognition, signal processing, biology, aerospace, finance, medicine, and meteorology (Dietrich *et al*. 2004, Eads *et al*. 2002, Jalba *et al*. 2005, Keogh and Ratanamahatana 2005, Lee *et al*. 2004, Nieeattrakul and Ratanamahatana 2007, Ubeyli 2008, Yu *et al*. 2007, Xi *et al*. 2006), and thus some notable techniques have been developed including nearest neighbor classifiers with a given distance measure, support vector machines, and neural networks (Eads *et al*. 2002, Guler and Ubeyli 2005, Ratanamahatana and Keogh 2004b). The nearest neighbor classifiers with dynamic time warping (DTW) has shown to be effective for time series classification and clustering because of its non-linear mappings capability (Itakura 1975, Nieeattrakul and Ratanamahatana 2007, Yu *et al*. 2007). The DTW technique finds an optimal match between two sequences by allowing a non-linear mapping of one sequence to another, and minimizing the distance between two sequences (Itakura 1975, Jalba *et al*.

2005, Keogh and Ratanamahatana 2005, Sakoe and Chiba 1978). The sequences are "warped" non-linearly to determine their similarity independent of any non-linear variations in the time dimension. The technique was originally developed for speech recognition, but several researchers have evaluated its application in other domains and have developed several variants such as derivative DTW (DDTW) (Keogh and Pazzani 2001, Rath and Manmatha 2003, Sakoe and Chiba 1978). Figure 3.1 shows the example of process of aligning two out of phase sequences by DTW.



(a) Two similar sequences, but out of phase (b) Alignment by DTW

Figure 3.1 Alignment of sequences based on DTW

The methodology for DTW is as follows. Assume a sequence *A* of length *m*, $A = a_1, a_2, \ldots, a_i, \ldots, a_m$ and a sequence *B* of length *n*, $B = b_1, b_2, \ldots, b_j, \ldots, b_n$. We create an *m*-by-*n* path matrix where the (i^{th}, j^{th}) element of matrix contains the distance between the two points a_i and b_j such that $d(a_i, b_j) = \left\| (a_i - b_j) \right\|_p$, where $\left\| \cdot \right\|_p$ represents the l_p norm. The warping path is typically subject to several constraints such as (Sakoe and Chiba 1978);

Endpoint constraint: The starting and ending points of warping path have to be the first and the last points of the path matrix, that is, u_1=(a_1,b_1) and u_k=(a_m,b_n).

Continuity constraint: The path can advance one step at a time. That is, when u_k=(a_i,b_j), u_{k+1}=(a_{i+1},b_{j+1}) where $a_i - a_{i+1} \leq 1$ and $b_i - b_{i+1} \leq 1$.

Monotonicity: The path does not decrease. That is, u_k=(a_i,b_j), u_{k+1}=(a_{i+1},b_{j+1}) where $a_i \geq a_{i+1}$ and $b_i \geq b_{i+1}$.

The best match between two sequences is the one with the lowest distance path after aligning one sequence to the other. Therefore, the optimal warping path can be found by using recursive formula given by:

$$DTW_p(A,B) = \sqrt[p]{\gamma(i,j)}$$

where $\gamma(i,j)$ is the cumulative distance described by:

$$\gamma(i,j) = \left|a_i - b_j\right|^p + \min\{\gamma(i-1,j-1), \gamma(i-1,j), \gamma(i,j-1)\}. \tag{3.1}$$

As seen from Eq. (3.1), given a search space defined by two time series sequences, DTW_p guarantees to find the warping path with the minimum cumulative distance among all possible warping paths that are valid in the search space. Thus, DTW_p can be seen as the minimization of warped l_p distance with time complexity of $O(mn)$. By restraining a search space using constraint techniques such as Sakoe-Chuba Band (Sakoe and Chiba 1978) and Itakura Parallelogram (Itakura 1975), the time complexity of DTW can be reduced. Figure 3.2 shows the warping matrix and optimal warping path between two sequences by DTW. In Figure 3.2, a band with width *w* is used to constrain the warping.

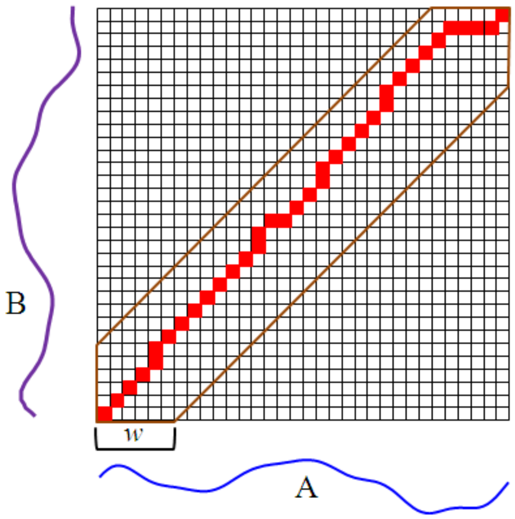


Figure 3.2 Warping matrix and optimal warping path by DTW

However, the conventional DTW calculates the distance of all points between two series with equal weight of each point regardless of the phase difference between a reference point and a testing point. This may lead to misclassification especially in applications such as image retrieval where the shape similarity between two sequences is a major consideration for an accurate recognition, thus neighboring points between two sequences are more important than others. In other words, relative significance depending on the phase difference between points should be considered.

Therefore, this paper proposes a novel distance measure, called the weighted dynamic time warping (WDTW), which weights nearer neighbors more heavily depending on the phase difference between a reference point and a testing point. Because WDTW takes into consideration the relative importance of the phase difference between two points, this approach can prevent a point in a sequence from mapping the further points in another one and reduce unexpected singularities, which are alignments between a point of a series with multiple points of the other series. Some practical examples will be presented to graphically illustrate possible situations where WDTW clearly is a better approach.

In addition, a new weight function, called the modified logistic weight function, is proposed to assign weights as a function of the phase difference between a reference point and a testing point. The proposed weight function extends the properties of logistic function to enhance the flexibility of setting bounds on weights. By applying different weights to adjacent points, the proposed algorithm can enhance the detection of similarity between series.

Finally, we extend the proposed idea to other variants of DTW such as derivative dynamic time warping (DDTW) and propose the weighted version of DDTW (WDDTW). We compare the performances of our proposed methods with other popular approaches using public datasets available through UCR Time Series Data Mining Archive (Keogh *et al*. 2006) for both time series classification and clustering problems. The experimental results show that the proposed procedures achieve improved accuracy for time series classification and clustering problems.

This remainder of the paper is organized as follows. In Section 3.2, we review some related literatures on times series classification and its methodologies. Section 3.3 explains the rationale of the advantage of the proposed idea. In Section 3.4, we describe the proposed WDTW and the modified logistic weight function for the automatic time series classification. The experimental results are presented and discussed in Section 3.5. The paper ends with concluding remarks and future works in Section 3.6.

3.2 Related Works

As a result of the increasing importance of time series classification in diverse fields, lots of algorithms have been proposed for different applications. Husken *et al*. (2003) utilized recurrent neural networks for time series classification and Guler *et al*. (2005) presented the wavelet-based adaptive neuro-fuzzy inference system model for classification of ectroencephalogram (EEG) signals. Rath *et al*. (2003) used DTW for word image matching and compared the performance of DTW with other popular techniques, including affine-corrected Euclidean distance mapping, the shape context algorithm, and correlation using sum of squared differences. Gullo *et al*. (2009) developed a time series representation model, called Derivative time series Segment Approximation (DSA), which combines the notions of derivative estimation, segmentation and segment approximation, for supporting accurate and fast similarity detection in time series data. Eads *et al*. (2002) introduced a hybrid classification algorithm that employs evolutionary computation for feature extraction, and a support vector machine for classification with the selected features. They tested their algorithm on a lightning classification task using data acquired from the Fast On-orbit Recording of Transient Events (FORTE) satellite.

In the area of new distance measures for time series classification and clustering, Keogh and Pazzani (2005) proposed a modification of DTW, called Derivative Dynamic Time Warping (DDTW), which transforms an original sequence into a higher level feature of shape by estimating derivatives. By preventing the production of unexpected singularities, DDTW has showed promising results for several special cases such as (1) two sequences differ in the Y-axis as well as X-axis, (2) cases in which there are local differences in the Y-axis, for instance, a peak in one sequence may be higher that the corresponding peak in the other sequences.

However, DDTW retains the assumption that all points in the sequence are weighted equally; that is, it is possible that a point of a series may be matched with further neighboring points of the other series, generating a similar problem as DTW. With a similar concept to DDTW, Xie and Wiltgen (2010) recently proposed an adaptive feature based dynamic time warping, which was designed to align two sequences with local and global features of each point in a sequence instead of its value or derivative.

3.3 Rationale for the Performance Advantages of WDTW

In this section, we will present the rationale underlying the proposed WDTW with practical examples to graphically illustrate situations where WDTW shows better performance than conventional DTW. The first example deals with automatic classification of defect patterns on semiconductor wafer maps. Figure 3.3 (a)-(d) show four common classes of defect patterns on wafer maps. Jeong *et al*. (2008) presented the effectiveness of using spatial correlograms (i.e., time series data) as new features for the classification of wafer maps instead of original binary input variables for each pixel where 1 represents the defective chip (black color) and 0 indicates the good chip (white color). Figure 3.3 (e)-(h) show the corresponding spatial correlograms of Figure 3.3 (a)-

(d), respectively. In correlograms, X-axis represents the spatial lags and Y-axis indicates their corresponding statistic value.

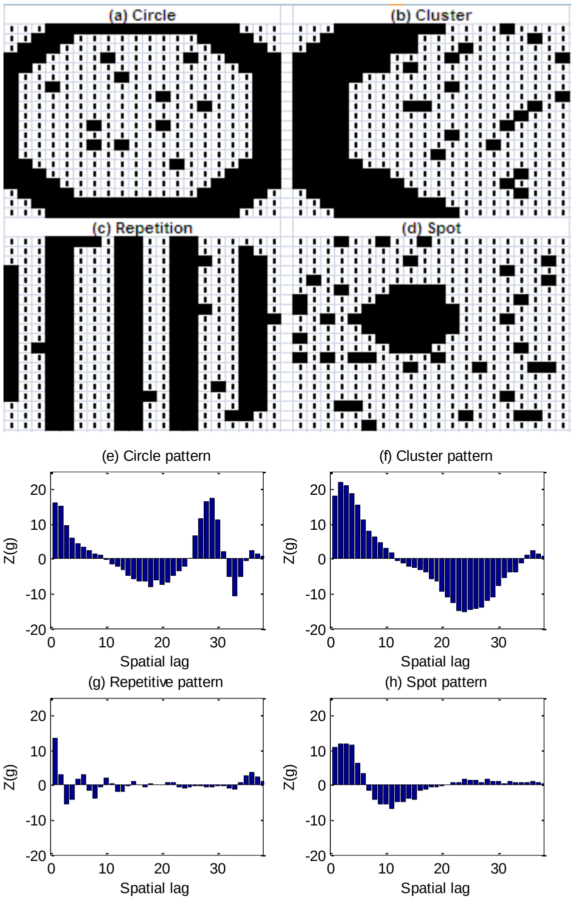


Figure 3.3 Typical defect patterns on wafer map and their corresponding correlograms

The correlogram plots the standardized value of $T(d)$ over the spatial lag d where $T(d)$ is given as follows for a given defective rate (p) (Jeong *et al*. 2008).

$$T(d) = pc_{00}(d) + (1-p)c_{11}(d) ,$$

where $c_{00}(d)$ and $c_{11}(d)$ represents the total number of normal (0)-to-normal (0) chip and defective (1)- to-defective (1) chip joins at a lag d for a given wafer map, respectively (for more details, see the (Jeong *et al*. 2008 and 2009)). Higher value of T(d) means that defective chips or good chips exist together at lag d. Figure 3.4 shows the definition of neighbors (or joins) at lag d under a Rook-move neighborhood (RMN) construction rule. In Figure 3.4, the black square represents a reference chip and red lines indicate neighboring chips (i.e. neighbors of a reference chip) with spatial lag d=1. Similarly, blue lines present neighboring chips with spatial lag d=2.

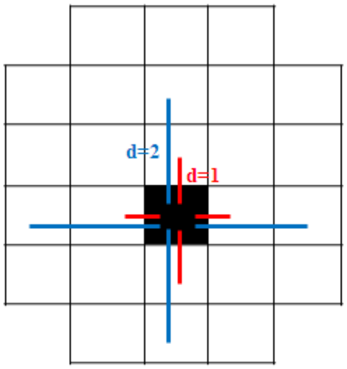
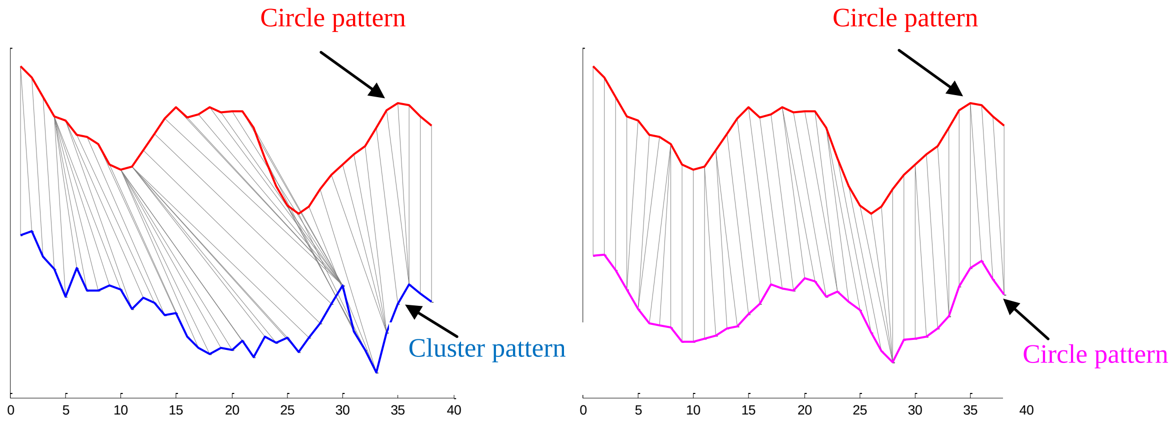


Figure 3.4 RMN neighborhood construction rules

If $T(d)$ is large, the neighbors at distance d from a reference defective chip (normal chip) include more defective chips (normal chips) than expected. If $T(d)$ is small, a reference defective chip (normal chip) tends to have normal chips (defective chips) as its neighbor at distance d. For example, in case of a cluster defect pattern, correlogram in Figure 3.3 (b), shows larger value of $T(d)$ for the 1^{st} - 5^{th} lags, meaning that at those distances, defective chips are clustered at certain areas. From 20th – 30th lag, statistic value is a large negative, indicating that at that distance, defective chips (normal chips) are joined with normal chips (defective chips). Thus, the comparison of statistic value at the ***same*** lag (or ***neighboring*** lags) between two correlograms (or sequences) is more meaningful when they are compared for defect pattern classifications and WDTW may choose higher value of g where g is the control parameter for the penalization level in weighting function. The higher g value, the more penalizing to points with higher phase difference to determine the optimal weights (see Section 3.4 for the detailed introduction of weight function).

Figure 3.5 and Figure 3.6 show the classification results of a new observation in testing data using DTW and WDTW, respectively. The red line represents a new time series data that should be classified into one of classes, and blue and pink lines represent the training dataset. Figure 3.5 (a) shows the result of alignment using DTW, showing the nearest distance among training dataset. The distance is 41.31. Figure 3.5 (b) shows the result of alignment using DTW, showing the second nearest distance among training dataset. The distance is 41.82. In case of DTW, some points in circle sequence (testing data, red line) are matched with further points in cluster sequence, distorting a minimum distance. Thus, a new testing sequence, which should be classified into a circle class, is misclassified into a clustering class. However, as shown in Figure 3.6, our proposed distance measure accurately classifies testing circle pattern into a same class because it penalizes more a point with higher phase difference between points, in other words, by preventing a point in a sequence from matching further points in another one. Note that for this case study, the optimal parameter g value for WDTW, which was optimized using the validation data set, was found to be 0.4, implicating much more penalizing for further points to increases the classification accuracy because the matching between points with same or neighboring lags is more meaningful for the classification of defect patterns.



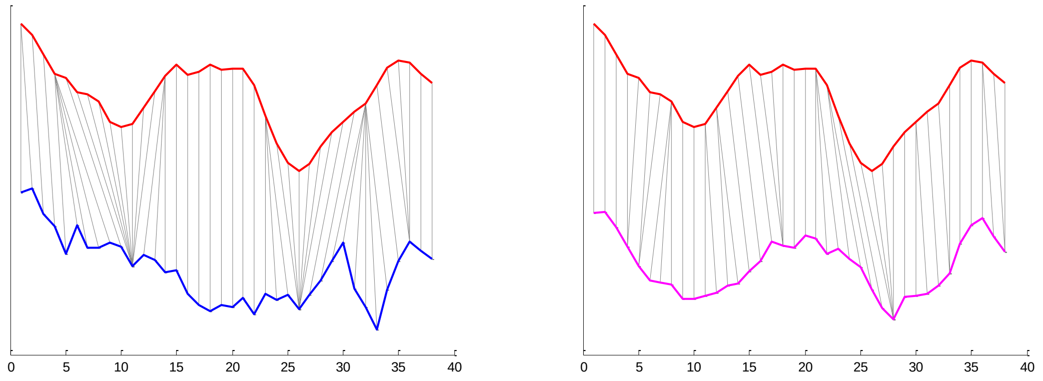
(a)

(b)

(a) Circle pattern (a new observation in testing data, red line) vs. Cluster pattern (an observation with the minimum distance using DTW in training data, blue line), DTW distance=41.31

(b) Circle pattern (a new observation in testing data, red line) vs. Circle pattern (an observation with the second minimum distance using DTW in training data, pink line), DTW distance=41.82

Figure 3.5 Alignment results generated by DTW



(a)

(b)

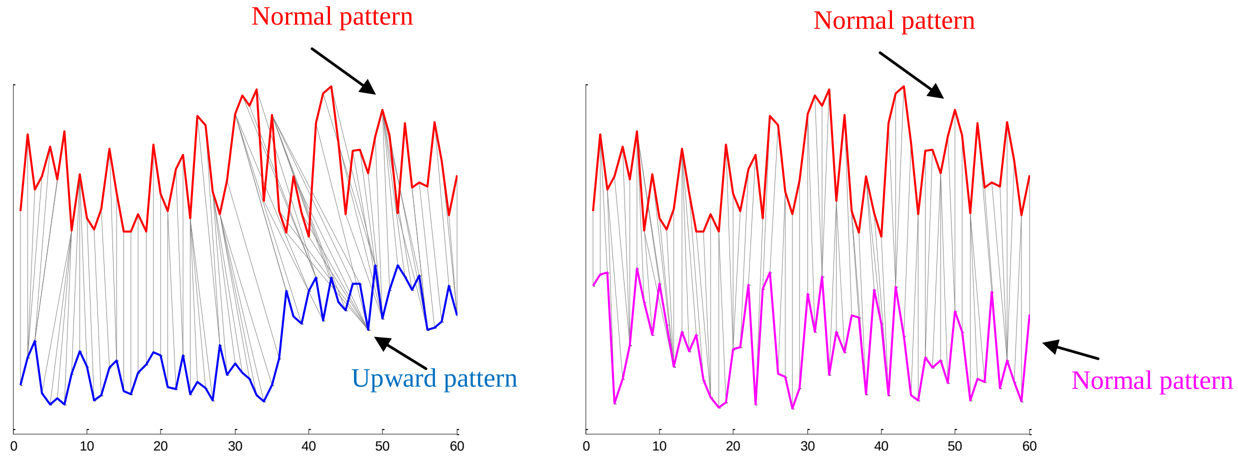
(a) Circle pattern (a new observation in testing data, red line) vs. Cluster pattern (an observation that showed the minimum distance using DTW in training data, blue line); WDTW distance=0.16

(b) Circle pattern (a new observation in testing data, red line) vs. Circle pattern (an observation with the minimum distance using WDTW in training data, pink line); WDTW distance=0.03

Figure 3.6 Alignment results generated by WDTW (*g*=0.4)

The second motivating example considers time series from "UCR Time Series Data Mining Archive." The data consists of six classes (Normal, Cycle, Increasing trend, Decreasing trend, Upward shift, and Downward shift). Figures 3.7 and 3.8 represent the alignments generated by DTW and WDTW, respectively. The red line indicates a new observation (in the test data) which is a "Normal" pattern, and blue and pink line represents "Upward shift" and "Normal" pattern in the training data, respectively. In order to correctly classify a given sequence, a point in the series should be matched with nearer neighbors of the other series because all sequences in the same class have similar shape. As shown in Figure 3.7, which shows the alignment by DTW, DTW maps a point in the red sequence to the points with further distance in the blue sequence. This alignment certainly does not have a positive impact on the similarity evaluation of these two sequences even though they have a minimum DTW distance between them. For example, Figure 3.7 (a) presents the alignments by DTW between Normal (a new observation in the testing data, red line) and Upward shift (training data, blue line) with 17.4 of DTW distance while Figure 3.7 (b) shows the alignments by DTW between Normal (a new observation in the testing data, red line) and Normal (training data, pink line) with 18.6 of DTW distance. Thus, DTW selects Upward shift sequence as the best match for a new sequence of Normal class, causing a misclassification. Meanwhile, Figure 3.8 (a) presents the alignment by WDTW between Normal (a new observation in the testing data, red line) and Upward shift (training data, blue line) with 0.134 of WDTW distance while Figure 3.8 (b) shows the alignment by WDTW between Normal (a new observation in the testing data, red line) and Normal (training data, pink line) with

0.123 of WDTW distance, correctly classifying Normal sequence. For WDTW, parameter *g* value was optimized using validation data set and was set to 0.3 in this case.



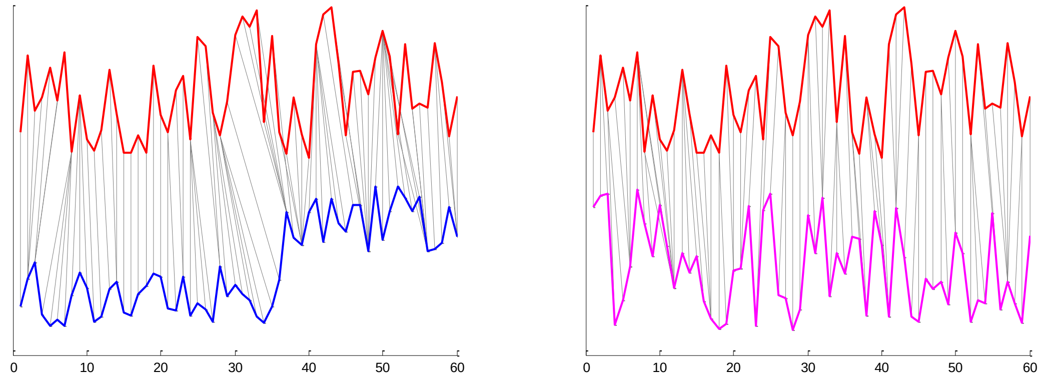
(a)

(b)

(a) Normal (a new observation in testing data, red line) vs. Upward shift (an observation with the minimum distance using DTW in training data, blue line), DTW distance=17.4

(b) Normal (a new observation in testing data, red line) vs. Normal (an observation with the second minimum distance using DTW in training data, pink line), DTW distance=18.6

Figure 3.7 Control chart pattern alignments generated by DTW



(a)

(b)

(a) Normal (a new observation in testing data, red line) vs. Upward shift (an observation that showed the minimum distance using DTW in training data, blue line); WDTW distance=0.134

(b) Normal (a new observation in testing data, red line) vs. Normal (an observation with the minimum distance using WDTW in training data, pink line); WDTW distance=0.123

Figure 3.8 Control chart pattern alignments generated by WDTW (*g*=0.3)

3.4 Proposed Algorithm for Time Series Classification

This section presents the proposed WDTW measure and a new weighting function, so called modified logistic weight function (MLWF) for time series data.

3.4.1 Weighted dynamic time warping

As mentioned earlier, the standard DTW calculates the distance of all points with equal penalization of each point regardless of the phase difference. The proposed WDTW penalizes the points according to the phase difference between a test point and a reference point to prevent minimum distance distortion by outliers. The key idea is that if the phase difference is low, smaller weight is imposed (i.e., less penalty is imposed) because neighboring points are important, otherwise larger weight is imposed.

In the WDTW algorithm, when creating the *m*-by-*n* path matrix, the distance between the two points a_i and b_j is calculated as $d_w(a_i, b_j) = \left\| w_{|i-j|}(a_i - b_j) \right\|_p$ where $w_{|i-j|}$ is a positive weight value between the two points a_i and b_j. The proposed algorithm implies that when we calculate the distance between a_i in a sequence *A* and b_j in a sequence *B*, the weight value will be determined based on the phase difference $|i - j|$. In other words, if the two points a_i and b_j are near, smaller weights can be imposed. Thus, the optimal distance between the two sequences is defined as the minimum path over all possible paths as follows:

$$WDTW_p(A, B) = \sqrt[p]{\gamma^*(i, j)} \tag{3.2}$$

where $\gamma^*(i,j) = \left|w_{|i-j|}(a_i - b_j)\right|^p + \min\{\gamma^*(i-1, j-1), \gamma^*(i-1, j), \gamma^*(i, j-1)\}$.

Based on the classical analysis of l_p spaces, we present the following Propositions that show some mathematical properties of WDTW such as WDTW$_p$ distance decreases monotonically as p increases and the opposite can be obtained under the specific condition on the measured space.

Proposition 3.1 For $0 < p < q \le \infty$, $WDTW_p(a_i, b_j) \ge WDTW_q(a_i, b_j)$

Proposition 3.2 For $0 < p < q \le \infty$, $WDTW_p(s_i, r_j) \le (2n-2)^{(1/p)-(1/q)} WDTW_q(s_i, r_j)$, where n is the length of the two sequences.

Proof of Proposition 3.1:

By classical analysis of l_p spaces, for $0 < p < q \le \infty$, we obtain that $\|\mathbf{x}\|_p \ge \|\mathbf{x}\|_q$ where **x** is a sequence. Let ***a*** and ***b*** denotes the sequence with same length, respectively. Given the two aligned sequences $\mathbf{a}^*$ and $\mathbf{b}^*$, it is true $\left\|\mathbf{a}^* - \mathbf{b}^*\right\|_p \ge \left\|\mathbf{a}^* - \mathbf{b}^*\right\|_q$ so that $\left\|\mathbf{w}(\mathbf{a}^* - \mathbf{b}^*)\right\|_p \ge \left\|\mathbf{w}(\mathbf{a}^* - \mathbf{b}^*)\right\|_q$ due to $\mathbf{w} > 0$. Therefore, $WDTW_p(\mathbf{a}^*, \mathbf{b}^*) \ge WDTW_q(\mathbf{a}^*, \mathbf{b}^*)$.

Proof of Proposition 3.2:

By classical analysis of l_p spaces, given **x** sequence with *n* length, $\|\mathbf{x}\|_p \le (n)^{(1/p)-(1/q)}$ $\|\mathbf{x}\|_q$ for $0 < p < q \le \infty$. In addition, the length of a minimal warping path in DTW is at most 2*n*-2 when *n*>1 (Lemire 2009). Given the two aligned sequences $\mathbf{a}^*$ and $\mathbf{b}^*$, it is true

$\left\|\mathbf{a}^* - \mathbf{b}^*\right\|_p \le (2n-2)^{(1/p)-(1/q)} \left\|\mathbf{a}^* - \mathbf{b}^*\right\|_q$. Thus, $\left\|\mathbf{w}(\mathbf{a}^* - \mathbf{b}^*)\right\|_p \le (2n-2)^{(1/p)-(1/q)}$ $\left\|\mathbf{w}(\mathbf{a}^* - \mathbf{b}^*)\right\|_q$ due to $\mathbf{w} > 0$. Therefore, $WDTW_p(\mathbf{a}^*, \mathbf{b}^*) \le (2n-2)^{(1/p)-(1/q)}$ $WDTW_q(\mathbf{a}^*, \mathbf{b}^*)$.

Given the lengths of two sequences are *m* and *n*, respectively, the time complexity of WDTW is the same as DTW, which is *O*(*mn*). There are weight factors to a distance calculation in WDTW, but each cell in an *m*-by-*n* path matrix should be filled in with the same time. Also, the best distance measure is related to the selection of *p* because $WDTW_p$ can be seen as the minimization of the warped l_p weighed distance. Even though optimal *p* depends on applications, l_1 and l_2 are usually good choices to classify time series data set (Lemire 2009, Morse and Patel 2006).

3.4.2. Modified logistic weight function

The next issue is how to systematically assign weight as a function of the phase difference between two points. In this section, we present our proposed modified logistic weight function (MLWF). One of the most popular classical symmetric functions that use only one equation is the logistic function. However, the standard form of logistic function is not flexible in setting bounds on weights. Therefore, in this paper, we propose modified logistic weight function (MLWF), which extends the properties of logistic function.

The weight value $w_{(i)}$ is defined as

$$w_{(i)} = \left[\frac{w_{\max}}{1 + \exp(-g * (i - m_c))} \right] \tag{3.3}$$

where i=1, … ,m, m is the length of a sequence and m_c is the midpoint of a sequence. $w_{\max}$ is the desired upper bound for the weight parameter, and g is an empirical constant that controls the curvature (slope) of the function; that is, g controls the level of penalization for the points with larger phase difference. The value of g could range from zero to infinity, but we investigate the characteristics of MLWF for four special cases. The characteristics of these four cases are summarized as follows: (1) Constant weight: This is the case in which all points are given the same weight. This can be achieved when g=0. (2) Linear weight: This is applicable to cases in which the weight is linearly proportional to the extent of the distance. This is the case when g=0.05, then the value of $w_{(i)}$ is nearly a linearly increasing relationship. (3) Sigmoid weight: Different sigmoid pattern can be achieved using different values of g. For example, the weight function follows a sigmoid pattern when g=0.25. (4) Two distinct weights: In this case, the first one-half is given one weight and the second one-half is given another weight. This is possible when g=3. The pictorial representations of the different weights for these g values are shown in Figure 3.9. Figure 3.9 also shows that the profile for MLWF is symmetric around the midpoint (m_c) of the total length of a sequence. The m and $w_{\max}$ are set to 100 and 1, respectively. It has been shown that a linear weighting profile and a sigmoidal pattern of weighting profile can be obtained by setting g=0.05 and g=0.25, respectively. Setting g=3 results in two distinct weights.

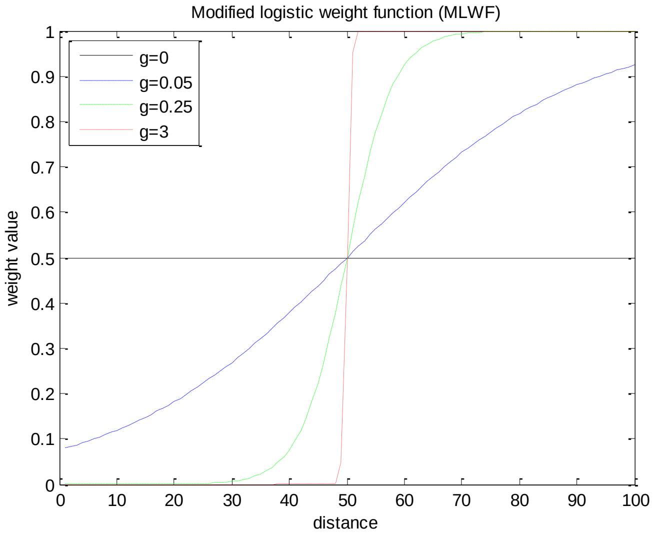


Figure 3.9 The pictorial representations of MLWF with different values of *g*

Remark 3.1 Conventional DTW and Euclidean distance measures are special cases of the proposed WDTW. For example, when $w_{|i-j|}$ is constant, i.e., *g*=0 in MLWF, with regard to phase $|i-j|$, WDTW is equivalent to DTW. However, as $w_{|i-j|}$ becomes smaller, i.e., *g* becomes larger, for the points in nearer phase $|i-j|$, WDTW will be closer to Euclidean distance because it does not allow non-linear alignments of one point to another. By choosing the appropriate *g* value, WDTW can achieve improved performance in diverse situations.

Remark 3.2 Based on our empirical study, the range of optimal *g* is distributed from 0.01 to 0.6. Smaller *g* means the less penalty for further points in the sequence, thus WDTW performance is similar to DTW. For example, in case of the signals with common initial phase shift, smaller penalty (or g) will be selected. For larger *g*, WDTW considers higher penalty for further points, leading to a similar performance of Euclidean distance.

3.4.3 Weighted derivative dynamic time warping (WDDTW)

The proposed weighted concept can be extended to variants of DTW. In this subsection, we extend the proposed idea to derivative dynamic time warping (DDTW) (Keogh and Pazzani 2001), which is one popular variant of DTW, and propose the weighted version of DDTW (WDDTW). Because DTW may try to explain variability in the Y-axis by warping the X-axis, this may lead to the unexpected singularities, which are alignments between a point of a series with multiple points of the other series, and unintuitive alignments. In order to overcome those weaknesses of DTW, DDTW transforms the original points into the higher level features, which contain the shape information of a sequence. The estimate equation for transforming data point a_i in the sequence A is given by (Keogh and Pazzani 2001),

$$D_A(d_i^a) = \frac{(a_i - a_{i-1}) + ((a_{i+1} - a_{i-1})/2)}{2}, \qquad 1 < i < m$$

where *m* is the length of sequence A. Because the first and last estimates are not defined, it is considered that $d_1^a = d_2^a$ and $d_m^a = d_{m-1}^a$.

The weighted version of DDTW is given as follows:

$$WDDTW_p(D_A, D_B) = \sqrt[p]{\xi^*(i, j)} \tag{4}$$

where $\xi^*(i, j) = \left| w_{|i-j|}(d_i^a - d_j^b) \right|^p + \min\left\{ \xi^*(i-1, j-1), \xi^*(i-1, j), \xi^*(i, j-1) \right\}$, and D_A and D_B are the transformed sequences from sequence A and B, respectively.

3.5 Experiment Results

3.5.1 Performance comparison for time series classification

In this section, we perform extensive experiments to verify the effectiveness of the proposed algorithm for time series classification and clustering. All datasets, which include real-life time series, synthetic time series, and generic time series, come from different application domains and are obtained from "UCR Time Series Data Mining Archive" (Keogh *et al*. 2006). For the detailed descriptions of the datasets, please see Ratanamahatana and Keogh (2004a, 2004b).

Euclidean distance, conventional DTW, and DDTW techniques are selected for comparison with the proposed algorithm. In addition, for comparison with state-of-art for time series similarity search, we implement the Longest Common Subsequence (LCSS), which is one of the popular methods for time series similarity because of its robustness to noise (Vlachos *et al*. 2002). LCSS measure has two parameters, δ and ε, which should be optimized using validating data set. The constant δ, which is usually set to less than 20 % of the sequence length, controls the window size in order to match a given point from one sequence to a point in another sequence. The constant ε, where $0 < \varepsilon < 1$, is the matching threshold (please refer to Vlachos *et al*. (2002) in details). In this paper, we use 1-nearest neighbor classifier because the 1-nearest neighbor classifier with DTW showed very competitive performance and has been widely used for time series classification (Xi *et al*. 2006).

For WDTW, two parameters should be fixed prior to the evaluation of testing performance. Different $w_{\max}$ does not affect its performance, thus, we set $w_{\max}$ to 1 in this work. In addition, because an optimal *g* value is different depending on the application domains, we choose the optimal *g* value using the validation data set after we divide the given data set into training, validating, and testing sets.

Table 3.1 shows the classification accuracy of the four different procedures for each dataset. In this work, the error rate is calculated as follows;

$$\text{Error rate} = \frac{(\text{total number of testing data}) - (\text{total number of correctly classified data})}{(\text{total number of testing data})}$$

As seen in Table 3.1, our proposed distance measures, WDTW and WDDTW, clearly outperform standard DTW, DDTW, and LCSS measures. In most of cases, the accuracies of WDTW and WDDTW is better (or equal in a few cases) than those of DTW and DDTW. In addition, we can see that depending on the application domains, DDTW results in better accuracy than DTW. The experimental results indicate that our proposed procedures are quite promising for automatic time series classifications in diverse applications. Note that when *g* becomes smaller, the error rate for WDTW becomes similar to that of DTW.

Table 3.1 Summary of classification performance

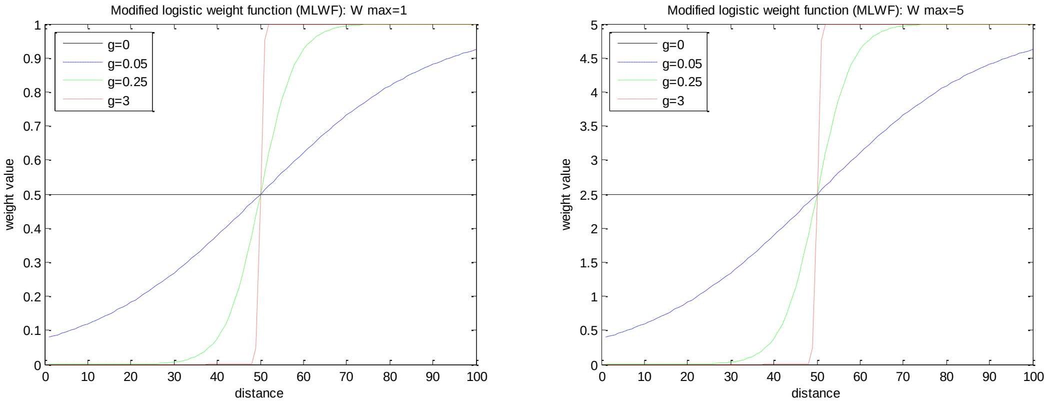
Data Name	Number of classes	Size of training set	Size of validating set	Size of testing set	Time series length	Error rates					
						ED*	DTW	WDTW (g)	DDTW	WDDTW (g)	LCSS (δ^*, ε)
Synthetic Control	6	300	150	150	60	0.153	0.007	**0.002** (0.3)	0.433	0.433 (0.01)	0.033 (5, 0.6)
Gun-Point	2	50	75	75	150	0.093	0.080	0.040 (0.2)	**0**	**0** (0.1)	0.027 (6, 0.1)
CBF	3	30	450	450	128	0.136	**0.002**	**0.002** (0.08)	0.418	0.418 (0.01)	0.004 (6, 0.3)
Face (all)	14	560	845	845	131	0.319	0.258	0.257 (0.01)	0.144	**0.131** (0.1)	0.300 (2, 0.1)
OSU Leaf	6	200	121	121	427	0.438	0.388	0.372 (0.6)	0.116	**0.091** (0.01)	0.231 (11, 0.2)
Swedish Leaf	15	500	313	312	128	0.218	0.210	0.138 (0.03)	0.115	**0.096** (0.6)	0.122 (5, 0.2)
50Words	50	450	228	227	270	0.352	0.317	**0.194** (0.1)	0.330	0.216 (0.1)	0.255 (6, 0.1)
Trace	4	100	50	50	275	0.240	**0**	**0** (0.01)	**0**	**0** (0.01)	0.100 (2, 0.2)
Two Patterns	4	1000	1000	3000	128	0.09	**0**	**0** (0.01)	0.002	0.003 (0.1)	0.002 (14, 0.1)
Wafer	2	1000	1000	5164	152	0.005	0.004	**0.002** (0.3)	0.023	0.006 (0.1)	0.004 (3, 0.5)

Face (four)	4	24	44	44	350	0.182	0.136	0.136 (0.1)	0.273	0.250 (0.1)	**0.023** (2, 0.1)
Lightning-2	2	60	31	30	637	0.200	**0.100**	**0.100** (0.1)	0.367	0.133 (0.03)	0.167 (4, 0.1)
Lightning-7	7	70	37	36	319	0.472	0.222	**0.200** (0.1)	0.278	0.228 (0.1)	0.277 (5, 0.3)
ECG	2	100	50	50	96	0.180	0.180	**0.140** (0.5)	0.220	0.160 (0.6)	0.16 (2, 0.2)
Adiac	37	390	196	195	176	0.390	0.390	0.364 (0.1)	0.426	**0.333** (0.4)	0.569 (3, 0.1)
Yoga	2	300	1000	2000	426	0.174	0.165	0.165 (0.1)	0.176	0.175 (0.1)	**0.141** (4, 0.1)
Fish	7	75	88	87	463	0.184	0.1379	0.126 (0.01)	0.126	**0.023** (0.1)	0.057 (6, 0.1)
Beef	5	30	15	15	470	0.600	0.600	0.600 (0.2)	0.400	**0.333** (0.1)	0.800 (1, 0.1)
Coffee	2	28	14	14	286	0.200	0.133	0.133 (0.01)	0.071	**0** (0.4)	0.2667 (1, 0.4)
Olive Oil	4	30	15	15	570	0.188	**0.188**	**0.188** (0.01)	0.313	0.313 (0.01)	0.857 (1,0.3)

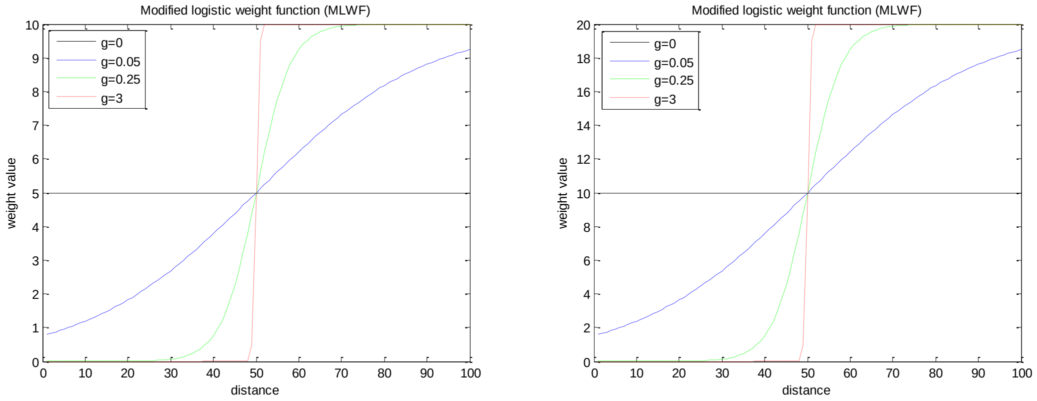
*ED: Euclidean distance, δ : % of sequence length

3.5.2. Effect of parameter values in WDTW

For WDTW, two parameters should be considered prior to the evaluation of testing performance. The $w_{\max}$, which is used to set the maximum of weight values, does not influence on the accuracy of experimental results in this study because weight is positive and $w_{\max}$ represents the full scale of weights in MLWF. For example, Fig. 10 presents the MLWF with different $w_{\max}$ values. Regardless of $w_{\max}$ value, MLWF retains its shape, implying that MLWF assigns weights with constant ratios to points in a sequence.



(a) $w_{\max} = 1$ (b) $w_{\max} = 5$



(c) $w_{\max} = 10$ (d) $w_{\max} = 20$

Figure 3.10 MLWF with different value $w_{\max}$

In addition, WDTW should choose the optimal *g* value depending on the application domains. Figure 3.11 shows the effect of *g* to the error rates of the validation data for the "Swedish Leaf" data set. "Swedish Leaf" data set was split into a training set of 500 samples, a validation set of 313 samples, and a test set of 312 samples. As shown in Figure 3.11, at the beginning, as *g* value increases, error rate decreases because nearer points are heavily weighed so that it is highly possible that sequence with a similar shape is chosen with minimum distance. However, as *g* value increases continuously, error rate increases after reaching the minimum error rate (0.115) because too large *g* value does not allow non-linear alignments of one point to another. In order words, WDTW with large *g* value will achieve similar performance to Euclidean distance measure as shown in Table 3.1. This example indicates that WDTW can adjust the level of penalization of the phase difference on each point by using different *g* values depending on applications.

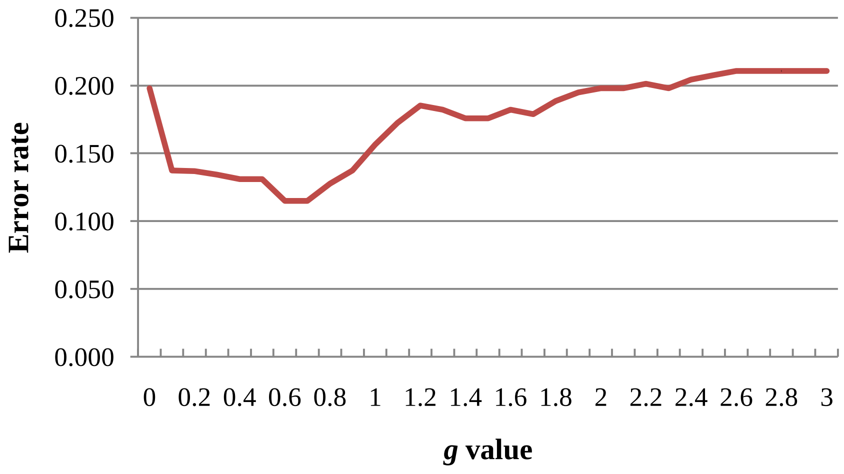


Figure 3.11 Effect of *g* to the error rates of validation data for the "Swedish Leaf" data

3.5.3 Performance comparison for time series clustering

Since WDTW is essentially a distance measure that can be generally used with different data mining tasks that consider the distance between two observations, we can extend the applications of WDTW to different tasks such as a clustering problem. Following the procedures of several literatures (Keogh and Lin 2005, Nieeattrakul and Ratanamahatana 2007, Yu *et al*. 2007), which presented DTW-based *K*-means method for time series clustering; we compare the performance of WDTW with that of DTW. As evaluation measures for validating a clustering quality, we used entropy and F-measure for external cluster validity and average within-cluster-distance (the intra-cluster compactness) and average between-cluster-distance (the inter-cluster separation) for internal cluster validity (Lu *et al*. 2008, Zhao *et al*. 2010).

Given data set belonging to *I* classes and partitioning them into *J* clusters using clustering algorithms, let *n* be the size of data set, n_i be the size of class *i*, n_j be the size of cluster *j*, and n_{ij} be the number of data belonging to both class *i* and cluster *j*. Then, Entropy and F-measure can be calculated as follows (Lu *et al*. 2008)

$$Entropy = \sum_{j=1}^{J} \frac{n_j}{n} \left(-\sum_{i=1}^{I} P(i,j) \log_2 P(i,j) \right)$$

$$F-measure = \sum_{i=1}^{I} \frac{n_i}{n} \max_{0<j<J} \left[\frac{2 \times R(i,j) \times P(i,j)}{R(i,j) + P(i,j)} \right]$$

where $R(i,j) = \frac{n_{ij}}{n_i}$, $P(i,j) = \frac{n_{ij}}{n_j}$. The lower the value of entropy, the higher the clustering quality, on the contrary, the higher the value of F-measure, the better the clustering quality. For internal cluster criteria, average within-cluster-distance (d_{ave_within}) and average between-cluster-distance (d_{ave_bet}) are calculated by (Keogh and Lin 2005)

$$d_{ave\_within} = \frac{1}{K * N_i} \sum_{i=1}^{K} \sum_{j=1}^{N_i} d(C_i, X_j)$$

$$d_{ave\_bet} = \frac{1}{M} \sum_{i=1}^{K} \sum_{j>i}^{K} d(C_i, C_j)$$

where $M = \sum_{m=1}^{K-1} m$ is the number of pairs of cluster centers, $d(C_i, X_j)$ is the distance between time series *j* in the cluster *i* and the cluster center of cluster *i*, and $d(C_i, C_j)$ is the distance between cluster centers of cluster *i* and cluster *j*. In addition, *K* and N_i the number of clusters and the number of items in cluster *i*, respectively. The smaller the value of average within-cluster-distance, the more compact each cluster, and the bigger the value of average between-cluster-distance, the more separate the clusters.

Table 3.2 shows the clustering results of 8 data sets out of 20 data sets. The cluster validity measures in Table 3.2 present the average values of 5 runs with the same data set. As for the value of *g* for WDTW, we used the selected value in Table 3.1 instead of optimizing it for a clustering purpose. As shown in Table 3.2, in most cases, WDTW outperforms both Euclidean distance and DTW even though we did not optimize the value of *g* for WDTW in terms of both external and internal cluster validity measures. Even though we used only datasets that have either small number of observations or low dimension of an input vector due to the limitation of computational time, similar conclusion can be made for the remaining datasets.

Table 3.2 Summary of clustering performance

Data Name	Number of classes	Data size	length	External cluster validity						Internal cluster validity					
				Entropy			F- measure			Average within-cluster-distance			Average between-cluster-distance		
				ED*	DTW	WDTW	ED*	DTW	WDTW	ED*	DTW	WDTW	ED*	DTW	WDTW
Gun-Point	2	200	150	1.012	0.999	**0.336**	0.5	0.505	**0.886**	3.989	3.865	**3.797**	7.223	7.384	**7.549**
Trace	4	200	275	1.807	**1.621**	**1.621**	0.482	**0.588**	**0.588**	4.399	**4.391**	4.806	15.969	**18.080**	17.901
Face (four)	4	112	350	0.925	**0.877**	0.916	0.758	**0.797**	0.778	13.566	13.653	**12.108**	11.957	12.021	**16.274**
Lighting 2	2	121	637	0.953	0.943	**0.868**	0.579	0.595	**0.612**	20.112	**18.112**	18.693	8.297	14.335	**16.566**
ECG	2	200	96	0.807	0.807	**0.752**	0.737	0.737	**0.769**	5.809	4.909	**4.461**	2.533	7.523	**8.079**
Beef	5	60	470	1.916	1.917	**1.906**	0.503	0.504	**0.542**	0.394	0.384	**0.354**	1.667	1.878	**2.069**
Coffee	2	56	286	0.891	**0.719**	**0.719**	0.631	**0.773**	**0.773**	35.769	34.817	**32.722**	82.319	79.539	**83.561**
Olive Oil	4	60	570	1.319	1.235	**1.214**	0.636	0.669	**0.685**	0.079	0.079	**0.053**	0.126	0.125	**0.183**

*ED: Euclidean distance

3.6 Concluding Remarks

A new automatic time series classification methodology, weighted dynamic time warping (WDTW), is proposed to accurately classify time series dataset in diverse applications. Compared with the conventional DTW, the proposed algorithm considers near neighbor points to be more important than others by applying more weights. In addition, a novel weighting function, called modified logistic weight function (MLWF), is developed to systematically assign weights depending on the distance among time series points.

The extensive experimental results using datasets from diverse applications show that the proposed WDTW with optimal weights have great potential for accuracy improvement of time series classification. As a part of further research, because the effectiveness of the proposed WDTW is the focus of this work, our proposed algorithm would be combined with the some of the pruning techniques such as LB_Keogh and warping-window-DTW to reduce computational time for much longer time series datasets.

CHAPTER 4

A Statistical Anomaly Detection Procedure for a Time Sequence Data with Local Variations

4.1 Introduction

A time sequence (series) data or curve, has been popularly used to monitor the quality of processes. Examples of time sequence data used in applications include investigation of a biomarker in early colon carcinogenesis (Morris *et al*. 2003), modeling of functional sulfur dioxide samples for environmental monitoring (Castro *et al*. 2005), development of SPC procedures for monitoring semiconductor manufacturing quality (Kang and Albin 2000), prediction of wood properties using high-dimensional spectral data (Fang *et al*. 2010), and modeling of acoustic emission signals to improve nano-machining process quality (Ganesan *et al*. 2003).

Wavelets has been used to model the time sequence data as a preprocessing technique. Existing wavelet models for time sequence data mostly focus on analysis of of *single data curve* (see Donoho and Johnstone (1994) and Jeong *et al.* (2006a) and references therein). A typical model assumed in these studies is $y(t) = f(t) + z(t)$, where the mean function $f(t)$ can be modeled by a sum of wavelet coefficients multiplied by their wavelet bases as shown in Eq. (4.1) in Section 4.2. Usually, errors at different time points are assumed to be independent and identically distributed as normal with mean zero and a constant variance σ^2. However, this model does not work well to describe ***local*** variations in the multiple time-sequence data that will be discussed next.

For example, in Figure 4.1, the center has more variations than the two sides in the stamping process. In case of Antenna data as shown in Figure 4.4 (b), if one examines the variations of data patterns closer to the two sides, the curve-to-curve variations are larger than the variations of data closer to the center. When the variation pattern is changed, process engineers need to investigate its cause. Figure 4.1 shows the three classes of multiple curves that represent one normal-condition data (Class 1) and two fault-condition data with 24 samples in each set (Jin and Shi 2001). Modeling the curve-to-curve variation for those three classes of products helps in understanding of stamping process behavior.

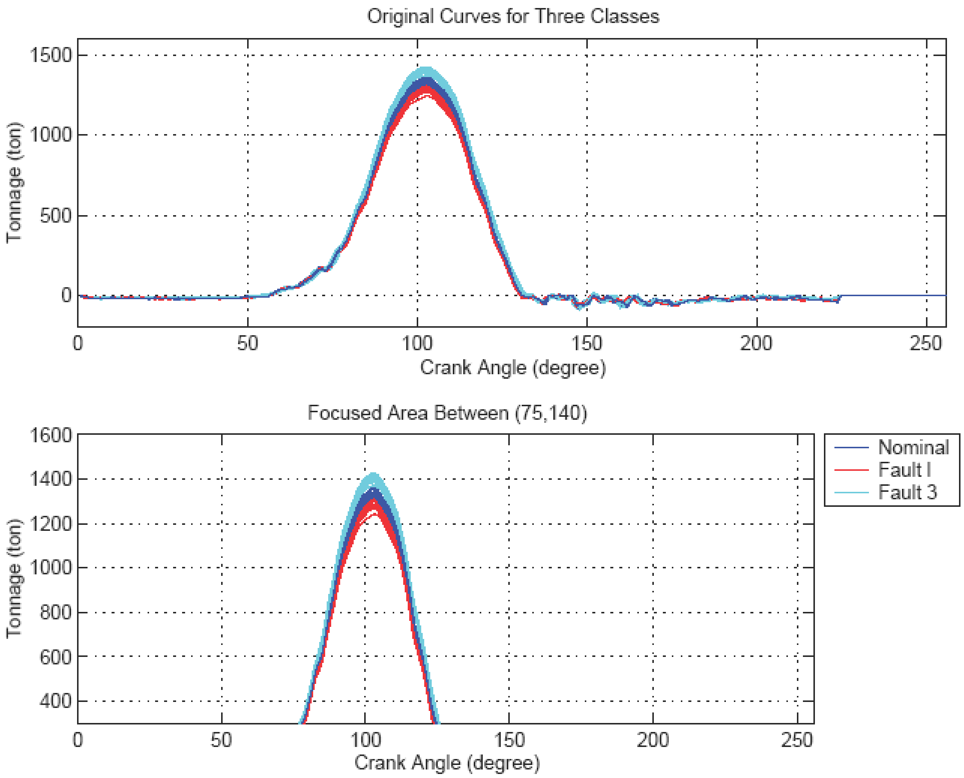


Figure 4.1 Three classes of multiple curves

To capture these between-curve variations, Section 4.2 presents a wavelet-based local random-effect model like the repeated measurement models used in biomedical studies. However, one important property in our model, as discussed above, is its ability to characterize variations in local areas. To elaborate, the following three subfigures in Figure 4.2 are generated from our model based on Haar wavelets using different sizes of supports covering local between-curve variations. In each subfigure, 20 curves of 256 data points at a time domain are generated based on a wavelet random-effect model from the Haar family in which the variance equals four. In Figure 4.2(a) one wavelet coefficient ($c_{4,7}$) at a coarser level in the fourth resolution level is assumed to be a random effect. The support of this coefficient covers from t_{97} to t_{112} . Figure 4.2(b) shows a wider support area (t_{65}, t_{96}) of a coarser level wavelet coefficient ($c_{3,3}$) in the third resolution level. Figure 4.2(c) shows a much wider support area (t_{65}, t_{128}) of a coarser wavelet coefficient ($c_{2,2}$) in the second resolution level. See Example 1 for the linkage of these random-effects to the stamping process data.

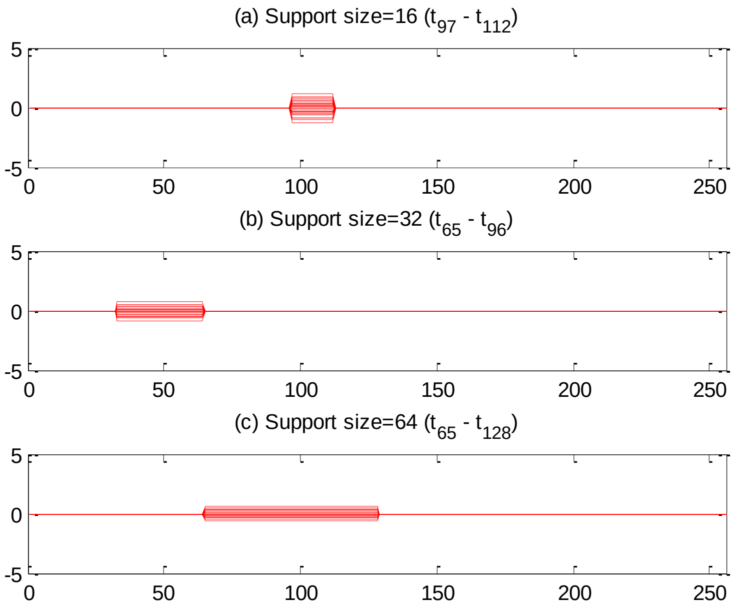


Figure 4.2 Local wavelet random-effect models based on Haar wavelet family

Depending on the support region covered by a few random wavelet coefficients, our model can capture between-curve variations in various sizes of local regions. More important, our model makes no assumption about which wavelet coefficient is random and which is not. We have developed a formal variance-thresholding procedure to identify random wavelet-coefficients. The selected wavelet coefficients serve as reduced-size data for various types of decision analysis. Figure 4.3 illustrates the flowchart of the proposed algorithm for freeway incident detection.

The remainder of this Chapter is composed as follows. Section 4.2 briefly reviews the wavelet background, and Section 4.3 proposes a wavelet-based local random-effect model and a mapping theory between data in the time and wavelet domain. Section 4.4 develops the WMVT procedure and provides guidelines for selecting its regularization parameters. Real-life examples are given in Section 4.5, and the WMVT method is compared with some possible extensions of existing methods in the literature. A anomaly detection procedure is presented in Section 4.6 and the performance of a detection power is compared in Section 4.7. The conclusion and suggestions for possible future works are offered in Section 4.8.

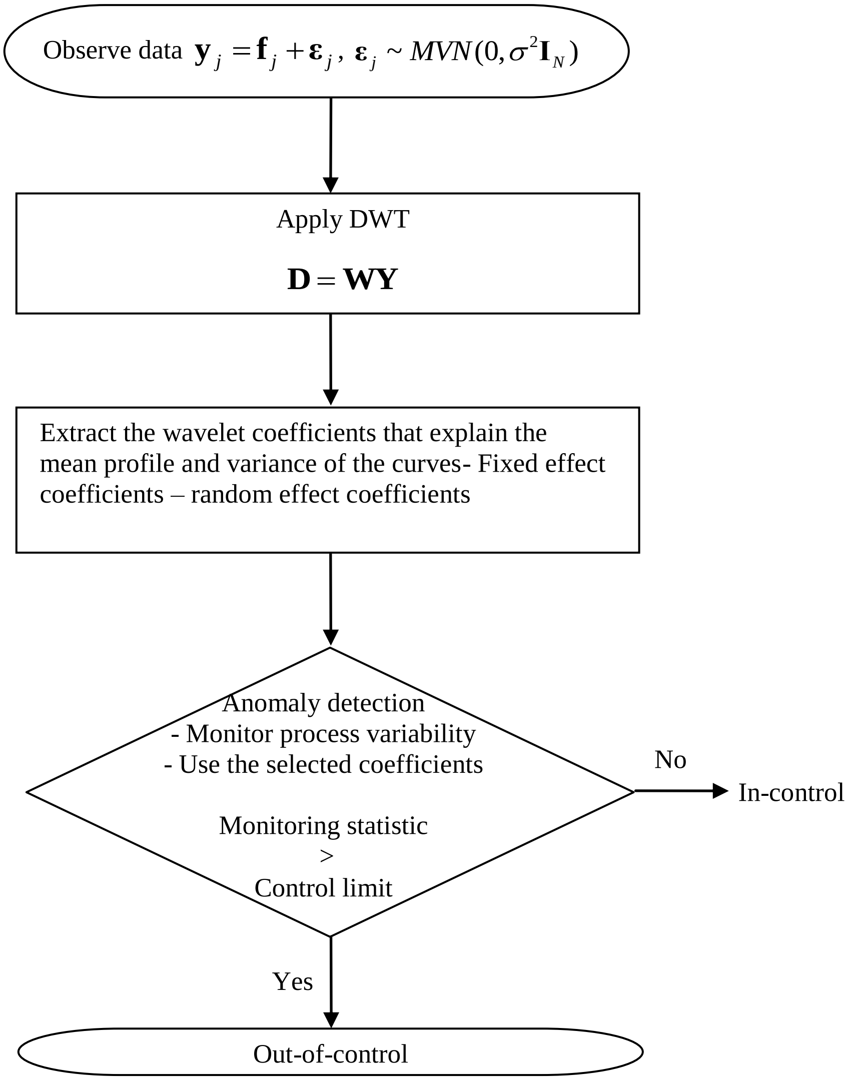

Observe data $\mathbf{y}_j = \mathbf{f}_j + \boldsymbol{\varepsilon}_j,\ \boldsymbol{\varepsilon}_j \sim MVN(0, \sigma^2 \mathbf{I}_N)$
Apply DWT
$\mathbf{D} = \mathbf{WY}$
Extract the wavelet coefficients that explain the mean profile and variance of the curves- Fixed effect coefficients – random effect coefficients
Anomaly detection
- Monitor process variability
- Use the selected coefficients
Monitoring statistic
>
Control limit
No
In-control
Yes
Out-of-control

Figure 4.3 Flowchart of the proposed statistical anomaly detection procedure

4.2 Wavelets

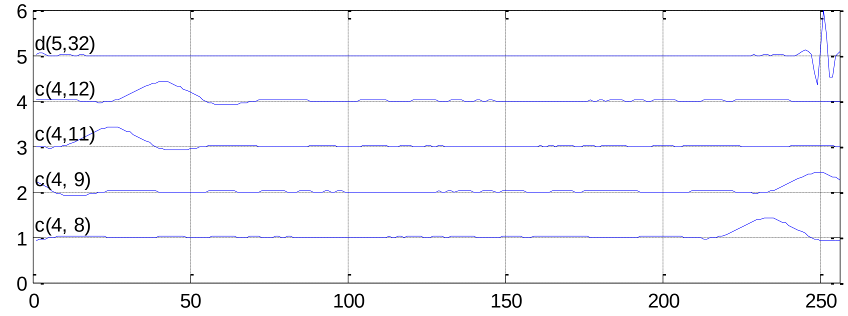
Denoted by $y_i = [y_{i1}, y_{i2}, \cdots, y_{iN}]^T$ a vector of N equally spaced data points from a functional curve, where $N = 2^J$ with some positive integer J and $i = 1, 2, \cdots, M$ for independently replicated curves. The superscript T represents the transpose operator. Let $\mathbf{Y} = [\mathbf{y}_1^T, \mathbf{y}_2^T, \cdots, \mathbf{y}_M^T]^T$. When a DWT $\mathbf{W}$ is applied to the data $\mathbf{Y}$, the vector of wavelet coefficients obtained from this transformation is $\mathbf{D} = \mathbf{YW}$, where $\mathbf{D} = [\mathbf{d}_1^T, \mathbf{d}_2^T, \cdots, \mathbf{d}_M^T]^T$, $d_i = [d_{i1}, d_{i2}, \cdots, d_{iN}]^T$, d_{im} is the wavelet coefficient at the m th wavelet-position for the i th data curve, and $\mathbf{W} = [h_{ij}]$, for $i, j = 1, 2, \cdots, N$ is the orthonormal $N \times N$ wavelet-transform matrix. The original observations $\mathbf{Y}$ can be reconstructed using the inverse DWT, i.e., through $\mathbf{Y} = \mathbf{DW}^T$.

The statistical literature has focused on single-curve data. A popular underlying model with a certain constant variance random error structure, e.g.,

$$\mathbf{y} = \mathbf{f} + \boldsymbol{\varepsilon}, \text{ or } \quad \mathbf{d} = \boldsymbol{\theta} + \boldsymbol{\varepsilon}_d,$$

is assumed for generating the N data points. Then, a few wavelet coefficients can be selected based on some thresholding procedures to estimate the true model (Donoho and Johnstone 1994). In the signal processing literature, a few of the largest coefficients are selected and other coefficients are set as zeros so as to use the inverse DWT to approximate the original data curve (Mallat 1998, Section 9.2). The selected wavelet coefficients in both the statistical and signal processing literature can be used as ``reduced size data" in follow-up decision analysis (e.g., Jeong *et al.* 2006a).

Figure 4.4 yields a better understanding of the relationship between $\mathbf{f}$ in the time domain and its DWT $\boldsymbol{\theta}$ in the wavelet domain. Based on the Symmlet-8 wavelet, Figure 4.4(a) shows that each wavelet coefficient will only affect the original data curve in its support area. Using all these coefficients together with a proper local-random-effect model proposed in Section 4.3, Figure 4.4(b) illustrates that the original data curves and their local-variations can be generated. Note that the data noises were not added into Figure 4.3(b) here. See Figure 4.7 for reconstructed curves with noises.


(a) Wavelet Bases of Five Active Random-effects

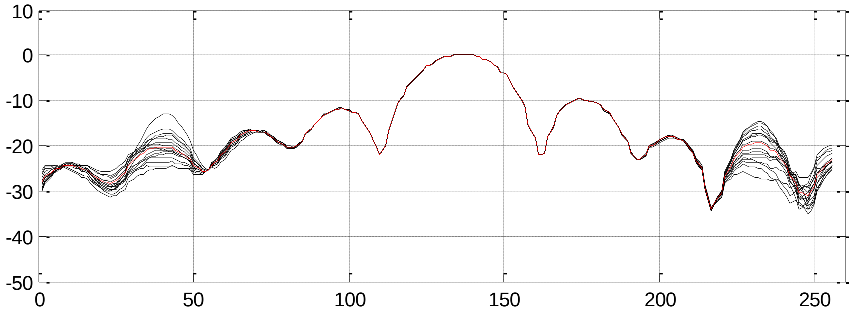

(b) Simulated Antenna Data

Figure 4.4 Support Areas of Active Random Wavelet Coefficients

4.3 Locally Focused Wavelet Random-Effect Model

Denote θ_{ij} the j th true wavelet coefficient for the i th curve and d_{ij} the sample version of θ_{ij}. Most of the classical work in the wavelet literature has focused on single curve. Thus, d_{ij} 's are independent and $N(\theta_{ij}, \sigma^2)$ distributed, where θ_{ij} 's and σ^2 are unknown parameters to be estimated. Our random-effect model follows repeated measurement studies in the biomedical field (e.g., Chen *et al*. 2001). In some support regions of wavelet coefficients, the behavior of data from different replicates are similar and thus we assume that $\theta_{1j} = \ldots = \theta_{Mj} \equiv \theta_{\cdot j}$ for keeping the model simple. In other regions, data curves differ significantly (see Figure 4.4(b)). Then, θ_{ij} 's are modeled as random-effects such as $\theta_{ij} \sim N(\theta_{\cdot j}, \tau_j^2)$, where $\theta_{\cdot j}$ measures the average value of wavelet coefficients in the j th position while τ_j^2 is the wavelet-position-dependent variance. To simplify expressions, we assume that $\theta_{ij} \sim N(\theta_{\cdot j}, \tau_j^2)$ with the convention that $\tau_j^2 = 0$ implies a fixed-effect model of $\theta_{\cdot j}$.

The following theorem presents analytically the mapping theory between the time and wavelet domain data for this random-effect model.

Theorem 4.1 Assume that there is a set of random coefficients, D, in the wavelet domain, i.e., $\tau_j^2 \neq 0$ for $j \in D$; zero, elsewhere. Then, the replicated curves from the wavelet-based random effect model in the time domain will have the following *systematic* variations over the region A, where A is the support area covered by the wavelet coefficients in the set D:

$$y_i(t_j) \sim \begin{cases} N(f_{.j}, \sum_{k \in D} h_{kj}^2 \tau_k^2 + \sigma^2), t_j \in A \\ N(f_{.j}, \sigma^2), \textit{elsewhere}, \end{cases}$$

where $y_i(t_j)$ is the original time domain data for the i th curve at time point t_j and $f_{.j}$ is the mean curve f evaluated at t_j.

Proof of Theorem 4.1

The replicated curves can be reconstructed from the inverse DWT, i.e.,

$$y_i(t_j) = \sum_{k=1}^{N} h_{kj} d_{i,k}, \ (j = 1, 2, \ldots, N)$$

$$= \sum_{k \in D} h_{kj} d_{ik} + \sum_{l \in S/D} h_{lj} d_{il},$$

where S is the set of all wavelet positions. Then, the variability across the curves is given by

$$Var(y_i(t_j)) = \sum_{k \in D} h_{kj}^2 Var(d_{ik}) + \sum_{l \in S/D} h_{lj}^2 Var(d_{il}),$$

$$= \sum_{k \in D} h_{kj}^2 \tau_k^2 + \sigma^2.$$

Example 4.1. Suppose that there is a data curve with $N = 256$ data and the lowest resolution level, L, is 4 in a Haar wavelet family. Assume all mean parameters $\theta_{.j}$'s are equal to zeros and there is only one coarser-level random-effect $c_{4,7}$ in the set D. Its support area is from t_{97} to t_{112} with $h_{7,j} = 1/\sqrt{16},\ j = 97, 98, \cdots, 112$; zeros, otherwise. If we simulated one data curve, according to Theorem 4.1, the support area $(t_{97}, t_{98}, \cdots, t_{112})$ of the random wavelet-coefficient $c_{4,7}$ would have the following systematic changes:

$$Y(t_j) = \begin{cases} \dfrac{1}{\sqrt{16}} c_{4,7} + \varepsilon_j,\ t_j = t_{97}, t_{98}, \cdots, t_{112}, \\ \varepsilon_j,\ elsewhere. \end{cases}$$

Figure 4.5(a) shows 50 simulated curves with Var($c_{4,7}$) = $\tau^2_{4,7} = 2^2$ and Var(ε_j) $= \sigma^2 = 0.1^2$. The area (t_{97}, t_{112}) in the time domain has *systematic* variations contributed from the random-effect with a variance equal to $\dfrac{1}{16}\tau^2_{4,7} + \sigma^2$. Other areas have variations from random noise with constant variance of σ^2. Note that Figure 4.2(a) showed 10 curves using the same random-effect $c_{4,7}$ model with a different variance $\tau^2_{4,7} = 10^2$ and the common variance σ^2 was set at zero.

Example 4.2. Consider the same setup as in Example 4.1. Assume the following three wavelet random coefficients: $c_{4,7}, d_{5,28}$, and $d_{7,6}$. Note that $c_{4,7}$ is at a coarser level with a support covering 16 data locations, $d_{5,28}$ is at a finer level with a support covering eight data locations, and $d_{7,6}$ is at the finest level with a support of two data locations.

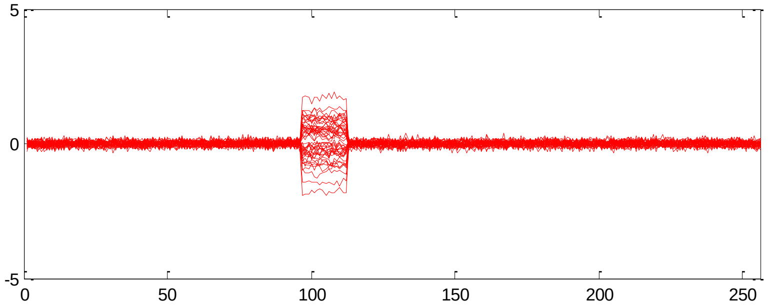
Theorem 4.1 leads to systematic changes in the following support areas:

$Y(t_j) = \frac{1}{\sqrt{2}} d_{7,6} + \varepsilon_j, t_j = t_{11}$;

$Y(t_j) = -\frac{1}{\sqrt{2}} d_{7,6} + \varepsilon_j, t_j = t_{12}$; $Y(t_j) = \frac{1}{\sqrt{8}} d_{5,28} + \varepsilon_j, t_j \in (t_{217}, t_{220})$;

$Y(t_j) = -\frac{1}{\sqrt{8}} d_{5,28} + \varepsilon_j, t_j \in (t_{221}, t_{224})$; $Y(t_j) = \frac{1}{4} c_{4,7} + \varepsilon_j, t_j \in (t_{97}, t_{112})$; $Y(t_j) = \varepsilon_j, t_j$ is elsewhere.

Figure 4.5(b) shows 50 simulated curves with $\tau_{4,7}^2 = 2^2, \tau_{5,28}^2 = 1^2, \tau_{7,6}^2 = 1.5^2$ and $\sigma^2 = 0.1^2$. Along the time line as shown in Figure 4.5(b), the level of systematic variations over the area $(t_{11}, t_{12}), (t_{97}, t_{112}),$ and (t_{217}, t_{224}) is $\frac{1}{2}\tau_{7,6}^2 + \sigma^2$, $\frac{1}{16}\tau_{4,7}^2 + \sigma^2$, and $\frac{1}{8}\tau_{5,28}^2 + \sigma^2$, respectively.


(a) Simulated curves with one wavelet coefficient random coefficient model

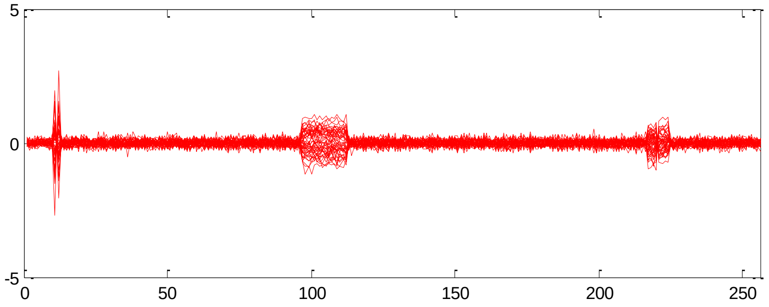

(b) Simulated curves with a wavelet coefficient random coefficients model: $c_{4,7}$, $d_{7,28}$, $d_{8,6}$ are random

Figure 4.5 Simulated curves with the wavelet random coefficients models

Next, the above illustrated examples are linked to real-life data curves.

Example 4.3. (Local Variations Around Center − Tonnage Data): Focusing only on the center portion of the tonnage signals, Figure 4.6(a) shows the original data curves in the normal-condition stamping process. Figure 4.6(b) shows 24 replicated curves from a random-effect model with a variance $\tau_{2,2}^2 = 100^2$ for the coefficient $c_{2,2}$ as studied in Figure 4.2(a). The simulated data from the random-effect model as shown in Figure 4.6(b) captures the local variations in the real data.

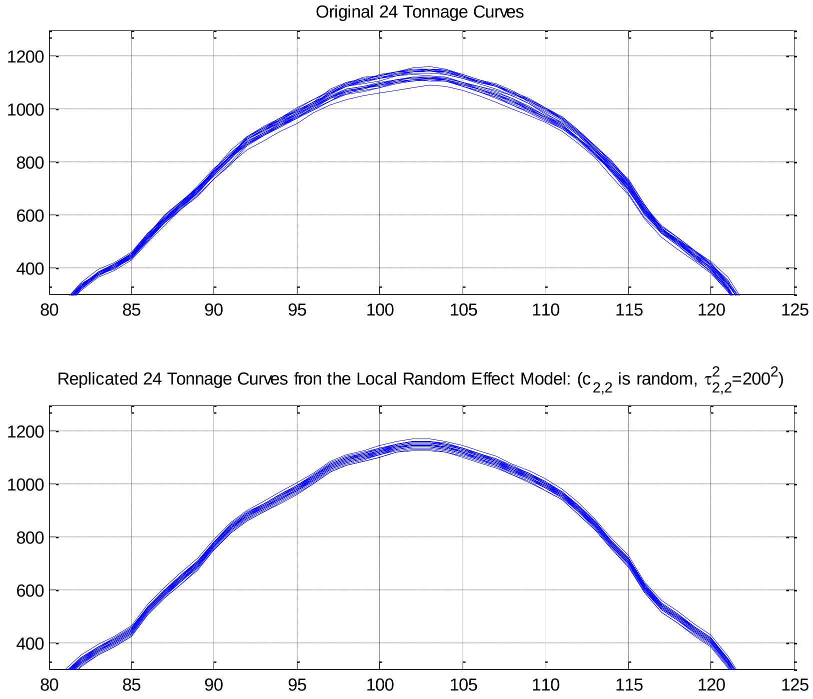


Figure 4.6 Replicated tonnage curves from the random effect model

Example 4.4 (Local Variations at Side-Regions − Antenna Data): Suppose that only the following five wavelet coefficients $c_{4,8}, c_{4,9}, c_{4,11}, c_{4,12}$, and $d_{5,32}$ are random. See Figure 4.4(a) for the support areas of these wavelet bases in the case of the Symmlet-8 wavelet family. Note that all the support areas from these random effects are only on two sides of the antenna data. Figure 4.4(b) shows simulated curves with σ^2 set as zero to display the impact of these random effects. See Figure 4.7 for the simulated curves with estimated σ^2 from the antenna data. Notice the similarity of these figures compared with the original curves presented in Figure 4.7, especially with the variations along two sides.

These examples show that besides the typical mean modeling with thresholded wavelet methods (e.g., Jeong *et al.* 2006a), it is important to decide which wavelet coefficients should be random and estimate the variances of random effects. The next section proposes a thresholding method to capture simultaneously both the mean pattern and the local variation of multiple curves.

4.4 Wavelet-Based Mean and Variance Thresholding Procedure

The local random-effect model proposed in Section 4.3 can be summarized as follows:

$$\mathbf{D} = \mathbf{\Theta} + \mathbf{Z}, \tag{4.1}$$

where $\mathbf{D} = [d_{ij}]$ is a $M \times N$ vector of all DWT transformed wavelet coefficients, $\mathbf{\Theta} = [\boldsymbol{\theta}_1{}^T, \ldots, \boldsymbol{\theta}_M{}^T]^T$, $\theta_i = [\theta_{i1}, \theta_{i2}, \ldots, \theta_{iN}]^T$, $\mathbf{Z} = [\mathbf{z}_1{}^T, \ldots, \mathbf{z}_M{}^T]^T$, and $\mathbf{z}_i$ is a column of $1 \times N$ random errors from the normal distribution $N(0, \sigma^2 + \tau_j^2)$. Note that we do not know which wavelet coefficients are random effects. It will be decided based on the procedure we propose below.

Let us start with the situation that all coefficients are random. Estimation of the mean and variance parameters can be achieved by minimizing the following negative log-likelihood of $\mathbf{D}$

$$M \sum_{j=1}^{N} \ln(\sigma^2 + \tau_j^2) + \sum_{i=1}^{M} \sum_{j=1}^{N} (d_{ij} - \theta_{.j})^2 \sigma^2 + \tau_j^2. \tag{4.2}$$

To encourage sparsity among $\theta_{.j}$'s and $\tau_{.j}$'s to keep the number of coefficients small for the purpose of data reduction, we impose two penalties at the end of the log-likelihood function:

$$M\sum_{j=1}^{N}\ln(\sigma^2+\tau_j^2)+\sum_{i=1}^{M}\sum_{j=1}^{N}(d_{ij}-\theta_{.j})^2\sigma^2+\tau_j^2+\lambda_1\sum_{j=1}^{N}|\theta_{.j}|+\lambda_2\sum_{j=1}^{N}\tau_j^2. \qquad (4.3)$$

Minimization of Eq. (4.3) follows the spirit of soft-thresholding and ridge regression. See Remark [1] below for details.

The first penalty term with a regularization parameter λ_1 encourages sparsity among mean parameters $\theta_{.j}$'s. The second term with λ_2 encourages some of the τ_j's to be zero, which implies that the j th position wavelet coefficient is a fixed effect. See remarks after the parameter estimation algorithm for more insights about the thresholding effects. Tuning parameters λ_1 and λ_2 control the tradeoff between modeling accuracy (in terms of maximizing the likelihood function) and sparsity. By sharing information across all multiple curves, the proposed approach achieves both mean and variance thresholding.

Algorithm for Parameter Estimation: Given λ_1 and λ_2,

(1) Initialize an estimate of σ^2:

Based on our experiments, wavelet coefficients at the finest level are less likely to be random effects. Thus, an initial estimate of σ^2 can be obtained from the following pooled variance idea. For each curve, obtain an estimate of σ^2 based on Donoho and

Johnston's (1994) robust estimate. Then, the common variance σ^2 for M curves can be estimated by averaging these robust estimates: $\hat{\sigma} = M^{-1}\sum_{i=1}^{M} 0.6745^{-1} median(|d_{im}|: N/2+1 \le m \le N)$, where the index m indicates wavelet coefficients at the finest level.

(2) Initialize an estimate of τ_j's:

(i) If the sample variance of $d_{\cdot j}$ is larger than the current estimate of σ^2, estimate τ_j^2 by the difference between the two. That is, this position of wavelet coefficients has a random effect.

(ii) Otherwise, estimate τ_j^2 by zero.

(3) Update $\theta_{\cdot j}$'s by minimizing (4.3) with respect to $\theta_{\cdot j}$'s:

By minimizing the penalized log-likelihood function with respect to $\theta_{\cdot j}$'s, we obtain the following closed form solution for the estimate of θ_j's (see Appendix for its detailed derivation).

$$\theta_{\cdot j} = \left(|\bar{d}_{\cdot j}| - \lambda_1(\sigma^2 + \tau_j^2)/(2M)\right)_+ sign(\bar{d}_{\cdot j}), \tag{4.4}$$

where $(x)_+ = \max(x, 0)$ and $\bar{d}_{\cdot j} = (d_{1j} + \ldots + d_{Mj})/M$.

(4) Update τ_j^2 by minimizing (4.4) with respect to τ_j's:

Similarly, by minimizing the penalized log-likelihood function with respect to τ_j's and by defining $s_j^2 = \sum_{i=1}^{M}(d_{ij} - \theta_{\cdot j})^2/M$, we can also obtain a closed form solution for the estimate of τ_j^2 as follows:

$$\tau_j^2 = \left(\frac{-1+\sqrt{1+4s_j^2\lambda_2/M}}{2\lambda_2/M} - \sigma^2 \right)_+ . \tag{4.5}$$

(5) Update σ^2 by minimizing (4.3) with respect to σ^2:

We can solve the following equation to obtain the updated estimate of σ^2:

$$\sum_{j=1}^{N} \frac{\sigma^2+\tau_j^2-s_j^2}{(\sigma^2+\tau_j^2)^2} = 0. \tag{4.6}$$

(6) Repeat Steps (3)-(5) until convergence.

Derivation of Parameter Estimates for $\theta_{.j}, \tau_j^2$, and σ^2:

The penalized log-likelihood function is given by

$$h(\theta_{.j}, \tau_j^2, \sigma^2) = M\sum_{j=1}^{N} \ln(\sigma^2+\tau_j^2) + \sum_{i=1}^{M}\sum_{j=1}^{N} (d_{ij}-\theta_{.j})^2 \sigma^2 + \tau_j^2 + \lambda_1 \sum_{j=1}^{N} |\theta_{.j}| + \lambda_2 \sum_{j=1}^{N} \tau_j^2 .$$

Taking the partial derivative of h with respect to $\theta_{.j}$, we obtain

$$\frac{\partial h}{\partial \theta_{.j}} = -2\sum_{i=1}^{M} \frac{d_{ij}-\theta_{.j}}{\sigma^2+\tau_j^2} + \lambda_1 sign(\theta_{.j})$$

$$= \frac{-2M}{\sigma^2+\tau_j^2}(\bar{d}_{.j}-\theta_{.j}) + \lambda_1\, sign(\theta_{.j}) = 0,$$

where $\bar{d}_{.j} = \frac{1}{M}\sum_{i=1}^{M} d_{ij}$. Therefore,

$$\hat{\theta}_{\cdot j} = \left(|\,\bar{d}_{\cdot j}\,| - \lambda_1(\sigma^2+\tau_j^2)/2M\right)_+ \mathrm{s}ign(\bar{d}_{\cdot j}),\ \mathrm{w}here\ \ (y)_+ = \max(y,0).$$

In a similar way, by taking the partial derivative of h with respect to τ_j^2, we obtain

$$\frac{\partial h}{\partial \tau_j^2} = \frac{M}{\sigma^2+\tau_j^2} - \sum_{i=1}^{M}\frac{(d_{ij}-\theta_{\cdot j})^2}{(\sigma^2+\tau_j^2)^2} + \lambda_2$$

$$= \frac{M}{(\sigma^2+\tau_j^2)^2}(\sigma^2+\tau_j^2 - s_j^2 + \lambda_2(\sigma^2+\tau_j^2)^2/M) = 0,$$

where $s_j^2 = \sum_{i=1}^{M}(d_{ij}-\theta_{\cdot j})^2/M$. Letting $a = \sigma^2+\tau_j^2$, we obtain $\lambda_2 a^2/M + a - s_j^2 = 0$, and

$$a = \sigma^2+\tau_j^2 = -1+\sqrt{1+4s_j^2\lambda_2/M}\,2\lambda_2/M$$

because $a > 0$. Therefore,

$$\hat{\tau}_j^2 = \left(-1+\sqrt{1+4s_j^2\lambda_2/M}\,2\lambda_2/M - \sigma^2\right)_+.$$

Finally, we can obtain the estimate of σ^2 by solving the following equation:

$$\frac{\partial h}{\partial \sigma^2} = M\sum_{j=1}^{N}\frac{1}{\sigma^2+\tau_j^2} - \sum_{i=1}^{M}\sum_{j=1}^{N}(d_{ij}-\theta_{\cdot j})^2(\sigma^2+\tau_j^2)^2 = M\sum_{j=1}^{N}\frac{\sigma^2+\tau_j^2-s_j^2}{(\sigma^2+\tau_j^2)^2} = 0,$$

which is equivalent to Eq. (4.6).

Remarks:

[1] The algorithm presented above reveals some operating characteristics of the proposed approach. Step (3) is similar to soft thresholding. Although a hard-thresholding procedure (set smaller $\hat{\theta}_{ij}$ to zero if it is less than the threshold) will retain fewer coefficients and thus achieve better data reduction, soft thresholding has various advantages, such as continuity of the shrinkage rule (Bruce and Gao 1996). Hard thresholding also leads to a larger variance of estimates and is also sensitive to small changes in the data. Interestingly, the minimization of Eq. (4.3) leads to the use of *varying* threshold values for means at different wavelet positions when different *variability* at different positions is considered. Thus, it is expected that our estimate should outperform soft thresholding with the *fixed* threshold value developed under the constant variance model used in most of the wavelet thresholding literature (e.g., Donoho and Johnstone 1994, Jung *et al*. 2006).

[2] Step (4) discloses the mechanism behind variance thresholding. By going through some algebric simplification, one can see that (6) implies that $\tau_j^2 = 0$ if

$$s_j^2 < \sigma^2 + \lambda_2 \sigma^4 / M. \tag{4.7}$$

Therefore, the positions whose coefficients display limited variation will be set as fixed effects by shrinking τ_j^2's to zero. Then, these zero coefficients will not be used as reduced-size data in later decision analysis.

[3] Step (5) updates the estimate of σ^2. Although it is an easy one-dimensional optimization problem, it is the most time consuming step of the algorithm because of the lack of a closed form solution.

[4] With the estimates of model parameters, M multiple curves can be reconstructed in the following way. First, obtain the estimate of θ_{ij}'s as follows: (1) for a random-effect position, obtain $\hat{\theta}_{ij}$ from simulated normal random variates with mean $\hat{\theta}_{.j}$ and variance $\hat{\sigma}^2 + \hat{\tau}_j^2$; (2) for a fixed-effect position, simulate $\hat{\theta}_{ij}$ from normal distribution with mean $\hat{\theta}_{.j}$ and variance $\hat{\sigma}^2$. Then, apply the inverse DWT with these estimates to reconstruct multiple curves.

Guideline for the Selection of Tuning Parameters: The effectiveness of the proposed WMVT procedure depends on tuning parameters λ_1 and λ_2. We apply the leaving-one-out cross validation technique (e.g., Stone 1974) to our problem. Let $\theta_{.j}^{[k]}$'s, $\tau_j^{2[k]}$'s, and $\sigma^{2[k]}$ be the estimates obtained by minimizing the penalized log-likelihood function in Eq. (4.4) based on all data curves except the kth. The measure of the quality of these estimates is based on the log-likelihood for data d_k (see below). Then, the cross-validation estimate of λ_1 and λ_2 is defined to be the minimizer of the following log-likelihood function for all M curves being left out one at a time in the cross-validation process:

$$V_0^*(\lambda_1,\lambda_2)=\sum_{k=1}^{M}[\sum_{j=1}^{N}\ln(\sigma^{2[k]}+\tau_j^{2[k]})+\sum_{j=1}^{N}(d_{kj}-\theta_{\cdot j}^{[k]})^2\sigma^{2[k]}+\tau_j^{2[k]}\ ].$$

4.5 Real-Life Examples

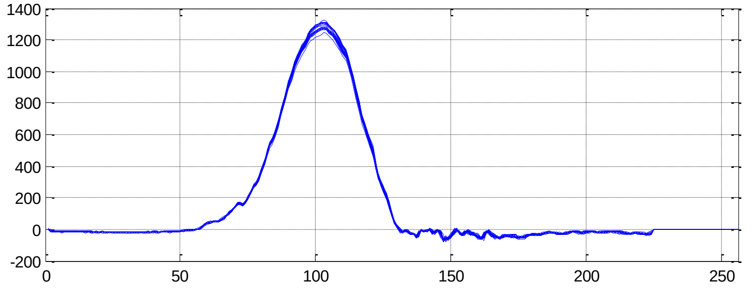
The evaluation of their performance uses the following criteria commonly seen in the wavelet thresholding and signal compression literature. Note that there are a total of $M \times N$ data points from M curves with N wavelet positions.

(1) K_1: number of non-zero mean wavelet coefficients;

(2) K_2: number of positions with wavelet random-effects;

Example 4.5 (Tonnage Signals): Tonnage signals were used for the monitoring and diagnosis of a stamping process (Jin and Shi 1999 and 2001). Tonnage signals contain process information relating to the deformation stage. Figure 4.7(a) shows 24 sets of tonnage signals under normal working conditions (class 1), and the data size of each curve is 256. Figure 4.7(b) shows only the center area and indicates that all tonnage signals have similar characteristics, but the center area has a larger local between-curve variation. The local-variations are contributions of the randomness of the distribution of lubricants and material uniformity (Zhou *et al*. 2006).

(a) Original Tonnage Curves



(B) Focused Area Between (75,125)

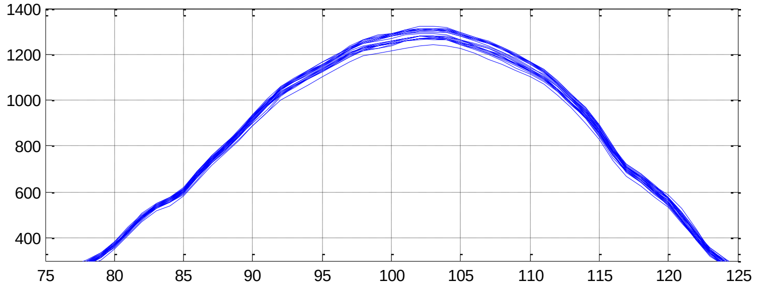


Figure 4.7 Tonnage curves of class 1

Figure 4.8(a) shows the reconstructed tonnage curves from the WMVT procedure with $\lambda_1 = 400$ and $\lambda_2 = 4000$. The WMVT procedure uses 22 non-zero wavelet coefficients for mean modeling and only three wavelet random-effects for variance modeling.

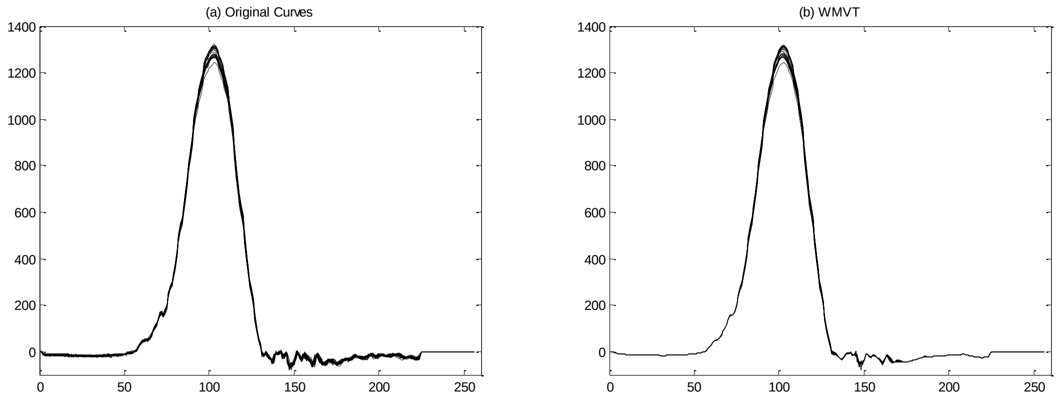


Figure 4.8 Reconstructed multiple curves for tonnage signals

Example 4.6 (Continued for Antenna Signals): The popularity of wireless communications has increased the need for high quality, technically sophisticated antennae. We collected data sets to develop procedures to monitor antenna manufacturing quality and detect process problems. Equipment used in such testing receives antenna signals at different degrees of elevation and azimuth (Jeong *et al*. 2006b, Jeong *et al*. 2006c). This study focuses on the zero-azimuth cut data curves generated from 20 antenna data sets under normal conditions. The antenna quality is evaluated according to various regulations regarding the signal patterns.

Figure 4.9 shows the reconstructed multiple curves based on different procedures. In particular, Figure 4.9(a) shows the reconstructed curves from the WMVT procedure with $\lambda_1 = 150$ and $\lambda_2 = 600$. The WMVT procedure uses a total of 87 coefficients ($K_1 + K_2 = 86$ and one from σ^2) to model the M curves.

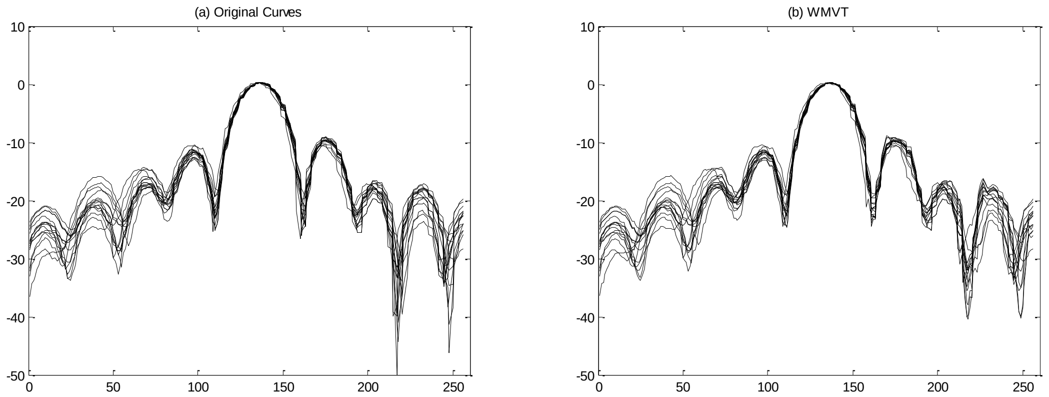


Figure 4.9 Reconstructed Multiple Curves for Antenna Signals

4.6 Profile Monitoring via Mixed-Effects Model in the Wavelet Domain

This section shows how to extend SPC procedures to selected wavelet coefficients for monitoring possible systematic changes of curves at certain local regions. For example, suppose that the antenna assembly process has a process change at time *i* and thus, the antenna or tonnage curves (see Figures 4.1 and 4.4, respectively) collected after the time *i* have larger systematic "local variations" compared with the original curves. Figure 4.10 shows an example of certain local changes around the center. This figure presents the multiple curves, which are coming from the normal condition (blue line) and anomaly condition (red line). The center areas exhibit systematic local variations even under the normal contition. For monitoring process variations in this case with low false alarm rates, a SPC model for monitoring local variances should be considered. However, there are fewer publications in variance monitoring, especially in local variance monitoring. Therefore, unlike the traditional time domain based SPC procedures in profile monitoring

(Reynolds and Cho 2006, Huwang *et al*. 2007, Zou *et al*. 2007), this section shows that by monitoring a few selected wavelet coefficients that captures those systematic local variations, the probability of detecting the changes of process variability at certain local areas can be much improved.

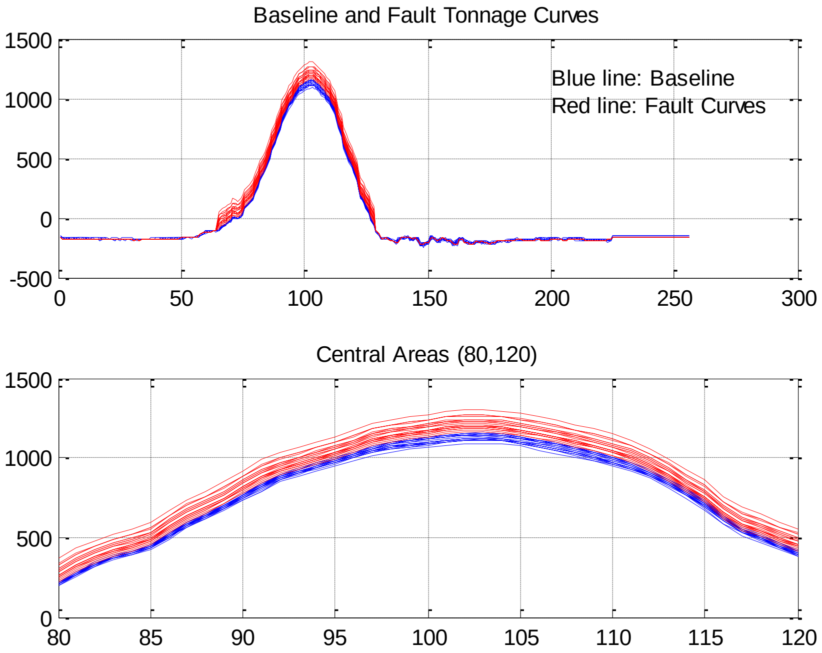


Figure 4.10 An example of the change of the size of local variations around center area in tonnage curves

4.6.1 Only process variance is changed

Let $\mathbf{d}_i = (d_{i,1}, d_{i,2}, \ldots, d_{i,N})$ be the wavelet coefficients of the given *i*-th observation $\mathbf{y}_i$. When the process is in control, the $d_{i,j}$'s are independent and $N(\theta_{i,j}, \sigma^2 + \tau_j^2)$ distributed where $\theta_{i,j}$'s, σ^2 and τ_j^2 were estimated using the WMVT procedure. After we identify the fixed effect and random effect wavelet coefficients through the WMVT procedure, we have rearranged the positions of wavelet variables so that the first p_1 variables are random effect coefficients, next p_2 variables are fixed effect coefficients, and others are shrunken coefficients. Thus, the covariance matrix of wavelet coefficients $\mathbf{\Sigma}_i$ can be expressed as

$$\mathbf{\Sigma}_i = \begin{bmatrix} \mathbf{\Sigma}_i^{r(p_1 \times p_1)} & \mathbf{0}^{p_1 \times p_2} & \mathbf{0}^{p_1 \times (N-p_1-p_2)} \\ \mathbf{0}^{p_2 \times p_1} & \mathbf{\Sigma}_i^{f(p_2 \times p_2)} & \mathbf{0}^{p_2 \times (N-p_1-p_2)} \\ & \mathbf{0}^{(N-p_1-p_2) \times N} & \end{bmatrix}_{N \times N}$$

where $\Sigma_i^r = (\sigma^2 + \tau_j^2)\mathbf{I}_{p_1}$ and $\Sigma_i^f = \sigma^2 \mathbf{I}_{p_2}$ are the covariance matrix of random and fixed effect variables, respectively and $\mathbf{I}_{p_i}$ is $p_i \times p_i$ identity matrix. Letting $\tilde{\mathbf{d}}_i^{rf} = (d_{i,1}, \ldots, d_{i,p_1}, d_{i,p_1+1}, \ldots, d_{i,n_1})$ be of the vector of only random and fixed effect wavelet coefficients and $n_1 = p_1 + p_2$. Then, the mean and covariance matrix of $\tilde{\mathbf{d}}_i^{rf}$ are given by

$$\boldsymbol{\mu}_i^{rf} = [\boldsymbol{\mu}_i^r \vdots \boldsymbol{\mu}_i^f]'; \ \mathbf{\Sigma}_i^{rf} = \begin{bmatrix} \mathbf{\Sigma}_i^{r(p_1 \times p_1)} & \mathbf{0}^{p_1 \times p_2} \\ \mathbf{0}^{p_2 \times p_1} & \mathbf{\Sigma}_i^{f(p_2 \times p_2)} \end{bmatrix}_{n_1 \times n_1}.$$

where $\boldsymbol{\mu}_i^r$ and $\boldsymbol{\mu}_i^f$ are the mean vectors of random and fixed effect variables, respectively.

Based on Theorem 4.2, under the assumption that only the process variance is changed, the hypothesis-testing formulation for a process-monitoring procedure in the time domain is given as follows:

$$\text{H}_0\text{:}\ y_i(t_j) \sim \begin{cases} N(f_{.j}, \sum_{k \in D} h_{kj}^2 \tau_k^2 + \sigma^2), t_j \in A \\ N(f_{.j}, \sigma^2), \textit{elsewhere}, \end{cases} \quad \text{versus H}_1\text{:}$$

$$y_i(t_j) \sim \begin{cases} N(f_{.j}, \rho_k + \sigma^2), t_j \in A \\ N(f_{.j}, \sigma^2), \textit{elsewhere}, \end{cases}$$

where $\rho_k = \sqrt{\frac{1}{\sigma^2}\sum_{k\in D} h_{kj}^2 \tau_{k'}^2}$, $\tau_{k'}^2 = \tau_k^2 + \eta_k$, and η_k is the changed level of process variance of the *k*-th random effect variable.

The process change under the above assumption can be detected by monitoring only random effect variables, $\tilde{\mathbf{d}}_i^r = (d_{i,1},\ldots,d_{i,p_1})$ in the wavelet domain. For the given *i*-th observation $\mathbf{y}_i$, after transforming it into the wavelet domain and selecting only random effect variables, we consider that $\boldsymbol{\mu}_i^r = \boldsymbol{\mu}_0^r$ and $\boldsymbol{\Sigma}_i^r = \boldsymbol{\Sigma}_0^r$ at time *i* where both $\boldsymbol{\mu}_0^r$ and $\boldsymbol{\Sigma}_0^r$ are known from baseline profiles. In addition, the standardized version of $\tilde{\mathbf{d}}_i^r$, $\tilde{\mathbf{u}}_i = \boldsymbol{\Sigma}_0^{r-1/2}(\tilde{\mathbf{d}}_i^r - \boldsymbol{\mu}_0^r)$, follows the normal distribution with mean $\boldsymbol{\mu}_i^s = \boldsymbol{\Sigma}_0^{r-1/2}(\boldsymbol{\mu}_i^r - \boldsymbol{\mu}_0^r)$ and covariance $\boldsymbol{\Sigma}_i^s = \boldsymbol{\Sigma}_0^{r-1/2}\boldsymbol{\Sigma}_i^r\boldsymbol{\Sigma}_0^{r-1/2}$. Thus, when the process is in control, $\tilde{\mathbf{u}}_i$ is distributed as $N(\mathbf{0},\mathbf{I}_{p_1})$. The proposed SPC model will be constructed based on the standardized coefficients $\tilde{\mathbf{u}}_i$.

In case of Phase II process monitoring with an individual observation, an unbiased estimator for $\boldsymbol{\Sigma}_i^s$ is given as $\mathbf{A}_i = \tilde{\mathbf{u}}_i\tilde{\mathbf{u}}_i^{'}$ when process mean does not change ($\boldsymbol{\mu}_i^s = 0$). One way to combine as much as possible information contained in $\mathbf{A}_i$ is to utilize the exponentially weighted moving average (EWMA) chart (Macgregor and Harris 1993). Therefore, we can define EWMA of $\mathbf{A}_i$ at the *i*-th observation (profile) as follows;

$$\mathbf{Z}_i = \delta\mathbf{A}_i + (1-\delta)\mathbf{Z}_{i-1},\ 1 \le i \le M \tag{4.8}$$

where $0 < \delta < 1$ is a smoothing constant, and $\mathbf{Z}_0 = \tilde{\mathbf{u}}_1\tilde{\mathbf{u}}_1^{'}$, which is an initial estimate of the covariance (Macgregor and Harris 1993). After some computational manipulations, Eq. (4.8) can be expressed alternatively as

$$\mathbf{Z}_i = \sum_{k=1}^{i} \delta(1-\delta)^{i-k}\mathbf{A}_k$$

where $\sum_{k=1}^{i} \delta(1-\delta)^{i-k} = 1$.

When the process mean does not change, $\mathrm{E}(\mathbf{Z}_i) = \sum_{k=1}^{i} \delta(1-\delta)^{i-k}\,\mathrm{E}(\mathbf{A}_i) = \mathbf{\Sigma}_i^s$ and then $\mathbf{Z}_i$ can be used to estimate $\mathbf{\Sigma}_i^s$. Since the trace, which is the sum of the diagonal of the covariance matrix, measures the overall variability in a covariance matrix, we propose the following monitoring statistic

$$T_{W_1}^2 = tr(\mathbf{Z}_i) = \sum_{k=1}^{i} \delta(1-\delta)^{i-k} \left(\sum_{j=1}^{p_1} u_{k,j}^2 \right) \tag{4.9}$$

This test statistic is a modified version of Huwang's process variability monitoring for an individual observation (Huwang *et al*. 2007). Larger $T_{W_1}^2$ values indicate that a process variance is increased because it is assumed that process mean does not change. When the process is in control, $\sum_{j=1}^{p_1} u_{k,j}^2$ follows a χ^2 distribution with the degree of freedom of p_1, thus, the mean and variance of $T_{W_1}^2$ are given as follows, respectively;

$$E(T_{W_1}^2) = \sum_{k=1}^{i} \delta(1-\delta)^{i-k} E\left(\sum_{j=1_1}^{p_1} u_{k,j}^2 \right) = p_1$$

$$Var(T_{W_1}^2) = \sum_{k=1}^{i} \left(\delta(1-\delta)^{i-k}\right)^2 Var\left(\sum_{j=1}^{p_1} u_{k,j}^2 \right) = \sum_{k=1}^{i} \left(\delta(1-\delta)^{i-k}\right)^2 2p_1$$

Thus, by using large-sample normal approximation theory, the control limit of $T_{W_1}^2$ is given by

$$CL_{T_{W_1}^2} = p_1 \pm \Phi^{-1}(1-\alpha)\sqrt{\sum_{k=1}^{i} \left(\delta(1-\delta)^{i-k}\right)^2 2p_1}$$

where α is the significance level and Φ is the standard normal distribution function.

4.6.2 Both process mean and variance are changed

Under the assumption that both process mean and variance may change during the monitoring period, we can detect process change by monitoring both random and fixed effect variables. When the process is in-control, we can assume that $\boldsymbol{\mu}_i^{rf} = \boldsymbol{\mu}_0^{rf}$ and $\boldsymbol{\Sigma}_i^{rf} = \boldsymbol{\Sigma}_0^{rf}$ where both $\boldsymbol{\mu}_0^{rf}$ and $\boldsymbol{\Sigma}_0^{rf}$ are known from baseline profiles. The standardized version of $\tilde{\mathbf{d}}_i^{rf}$, $\tilde{\mathbf{v}}_i = \boldsymbol{\Sigma}_0^{rf\,-1/2}(\tilde{\mathbf{d}}_i^{rf} - \boldsymbol{\mu}_i^{rf})$, follows the normal distribution with mean $\boldsymbol{\mu}_i^{v} = \boldsymbol{\Sigma}_0^{rf\,-1/2}(\boldsymbol{\mu}_i^{rf} - \boldsymbol{\mu}_0^{rf})$ and covariance $\boldsymbol{\Sigma}_i^{v} = \boldsymbol{\Sigma}_0^{rf\,-1/2}\boldsymbol{\Sigma}_i^{rf}\boldsymbol{\Sigma}_0^{rf\,-1/2}$. Thus, when the process is in control, $\tilde{\mathbf{v}}_i$ is normally distributed as $N(\mathbf{0}, \mathbf{I}_{m_1})$.

When the process mean changes during the monitoring period, $\mathbf{Z}_i$ is modified by

$$\mathbf{C}_i = \sum_{k=1}^{i} \delta(1-\delta)^{i-k}(\tilde{\mathbf{v}}_k - \boldsymbol{\gamma}_k)(\tilde{\mathbf{v}}_k - \boldsymbol{\gamma}_k)' \,,\quad 1 \le i \le M \tag{4.10}$$

where $\boldsymbol{\gamma}_k$ is the estimate of the process mean. The optimal estimate $\boldsymbol{\gamma}_i$ for process mean at time *i*, is $\varphi\tilde{\mathbf{v}}_i + (1-\varphi)\tilde{\mathbf{v}}_{i-1}$ with smoothing weight $0 < \varphi < 1$ (Macgregor and Harris 1993). Because $\mathrm{E}(\mathbf{C}_i) \rightarrow \frac{2(1-\varphi)}{(2-\varphi)}\boldsymbol{\Sigma}_i^{v}$ as $i \rightarrow \infty$, $\frac{(2-\varphi)}{2(1-\varphi)}\mathbf{C}_i$ can be used as the estimator of $\boldsymbol{\Sigma}_i^{v}$.

Derivation of estimator $\boldsymbol{\Sigma}_i^{v}$:

By definition of $\mathrm{EWMA}(\tilde{\mathbf{v}}_i)$ with smoothing weight $0 < \alpha < 1$, $\mathbf{r}_i = \sum_{k=1}^{i} \alpha(1-\alpha)\tilde{\mathbf{v}}_k$.

Thus, $\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_i = \tilde{\mathbf{v}}_i - \sum_{k=1}^{i} \alpha(1-\alpha)\tilde{\mathbf{v}}_k$ and

$$\begin{aligned} \mathrm{E}\left(\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_i\right) &= \mathrm{E}\left(\tilde{\mathbf{v}}_i - \sum_{k=1}^{i} \alpha(1-\alpha)\tilde{\mathbf{v}}_k\right) = \mathrm{E}(\tilde{\mathbf{v}}_i) - \sum_{k=1}^{i} \alpha(1-\alpha)\mathrm{E}(\tilde{\mathbf{v}}_k) \\ &= \boldsymbol{\mu}_i - [1-(1-\alpha)^i]\boldsymbol{\mu}_i = (1-\alpha)^i \boldsymbol{\mu}_i \end{aligned}$$

In addition,

$$\begin{aligned} \mathrm{E}[(\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_i)(\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_i)'] &= \mathrm{Cov}(\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_i) + \mathrm{E}(\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_i)\mathrm{E}(\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_i)' \\ &= \mathrm{Cov}[\tilde{\mathbf{v}}_i - \sum_{k=1}^{i} \alpha(1-\alpha)^{i-k}\,\tilde{\mathbf{v}}_k] + (1-\alpha)^{2i}\boldsymbol{\mu}_i^v \\ &= \mathrm{Cov}[\tilde{\mathbf{v}}_i] - \mathrm{Cov}[\sum_{t=1}^{n} \alpha(1-\alpha)^{i-k}\,\tilde{\mathbf{v}}_k] + (1-\alpha)^{2i}\boldsymbol{\mu}_i^v \\ &= \left[1 + \frac{\alpha}{2-\alpha}\left((1-\alpha)^{2i} - 1\right)\right]\boldsymbol{\Sigma}_0^{rf} + (1-\alpha)^{2i}\boldsymbol{\mu}_i^v \\ &= \frac{1}{2-\alpha}\left[2(1-\alpha) + \alpha(1-\alpha)^{2i}\right]\boldsymbol{\Sigma}_0^{rf} + (1-\alpha)^{2i}\boldsymbol{\mu}_i^v \end{aligned}$$

Based on these results, we can obtain

$$\begin{aligned} \mathrm{E}(\mathbf{C}_i) &= \sum_{k=1}^{i} \delta(1-\delta)^{i-k} E((\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_k)(\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_k)') \\ &= \sum_{k=1}^{i} \delta(1-\delta)^{i-k}\left[\frac{1}{2-\alpha}\left(2(1-\alpha) + \alpha(1-\alpha)^{2i}\right)\right]\boldsymbol{\Sigma}_i^v + (1-\alpha)^{2i}\boldsymbol{\mu}_i^v \\ &= \frac{2(1-\alpha)}{2-\alpha}\boldsymbol{\Sigma}_i^v + \sum_{k=1}^{i} \delta(1-\delta)^{i-k}\,\frac{\alpha}{2-\alpha}(1-\alpha)^{2k}\,\boldsymbol{\Sigma}_i^v + (1-\alpha)^{2i}\boldsymbol{\mu}_i^v \\ &= \frac{2(1-\alpha)}{2-\alpha}\boldsymbol{\Sigma}_i^v \quad \text{as} \quad i \to \infty \end{aligned}$$

Because $\frac{(2-\varphi)}{2(1-\varphi)}$ is a constant, we propose the following monitoring statistic

$$T_{W_2}^2 = \mathrm{tr}(\mathbf{C}_i) = \sum_{k=1}^{i} \delta(1-\delta)^{k-i}\, tr((\tilde{\mathbf{v}}_k - \boldsymbol{\gamma}_k)(\tilde{\mathbf{v}}_k - \boldsymbol{\gamma}_k)^{'})$$

By using matrix forms, the above statistic can be simplified as follows.

$$T_{W_2}^2 = \mathrm{tr}((\tilde{\mathbf{v}}_i - \mathbf{H}_i)'\boldsymbol{\Delta}(\tilde{\mathbf{v}}_i - \mathbf{H}_i)$$

where

$$\begin{aligned}
&\tilde{\mathbf{V}}_i = [\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \ldots, \tilde{\mathbf{v}}_i]' \\
&\mathbf{H}_i = [\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_i]' \\
&\boldsymbol{\Delta} = diag((1-\delta)^{i-1}, \delta(1-\delta)^{i-2}, \ldots, \delta(1-\delta), \delta)
\end{aligned}$$

In addition,

$$(\tilde{\mathbf{V}}_i - \tilde{\mathbf{H}}_i) = \begin{pmatrix} (\tilde{\mathbf{v}}_1 - \boldsymbol{\gamma}_1)' \\ (\tilde{\mathbf{v}}_2 - \boldsymbol{\gamma}_2)' \\ \vdots \\ (\tilde{\mathbf{v}}_i - \boldsymbol{\gamma}_i)' \end{pmatrix} = \begin{pmatrix} (1-\varphi)\tilde{\mathbf{v}}_1 \\ (1-\varphi)\tilde{\mathbf{v}}_2 - \varphi(1-\varphi)\tilde{\mathbf{v}}_1 \\ \vdots \\ (1-\varphi)\tilde{\mathbf{v}}_i - \varphi(1-\varphi)\tilde{\mathbf{v}}_{i-1} - \cdots - \varphi(1-\varphi)^{i-1}\tilde{\mathbf{v}}_1 \end{pmatrix} = \mathbf{B}\tilde{\mathbf{V}}_i$$

where $\mathbf{B} = \begin{pmatrix} 1-\varphi & 0 & \cdots & 0 \\ -\varphi(1-\varphi) & 1\text{-}\varphi & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -\varphi(1-\varphi)^{i-1} & \cdots & -\varphi(1-\varphi) & (1-\varphi) \end{pmatrix}$.

Thus,

$$
\begin{aligned}
T_{W_2}^2 &= \mathrm{tr}((\tilde{\mathbf{V}}_i - \mathbf{H}_i)'\boldsymbol{\Delta}(\tilde{\mathbf{V}}_i - \mathbf{H}_i)) \\
&= \mathrm{tr}(\tilde{\mathbf{V}}_i\mathbf{B}'\boldsymbol{\Delta}\mathbf{B}\tilde{\mathbf{V}}_i) = \mathrm{tr}(\boldsymbol{\Theta}\tilde{\mathbf{V}}_i\tilde{\mathbf{V}}_i^{'}) \\
&= \sum_{k=1}^{i}\sum_{j=1}^{i}\pi_{kj}(\sum_{l=1}^{m_1}\tilde{v}_{k,l}\tilde{v}_{j,l})
\end{aligned}
$$

where $\mathbf{\Pi}_{i\times i} = (\pi_{kj})_{i\times i} = \mathbf{B}'\boldsymbol{\Delta}\mathbf{B}$, $1 \le i, j \le M$. The matrix $\mathbf{\Pi}_{i\times i}$ indicates that recent profiles are heavily weighted such as $\pi_{ii} = \delta(1-\varphi)^2$, $\pi_{(i-1)(i-1)} = \delta(1-\delta)(1-\varphi)^2 + \delta[-\varphi(1-\varphi)]^2$, and $\pi_{(i-2)(i-2)} = \delta(1-\delta)^2(1-\varphi)^2 + \delta[-\varphi(1-\varphi)^2]^2 + \delta(1-\delta)[-\varphi(1-\varphi)]^2$, and so on.

Thus, when the process is in control, the mean and variance of $T_{W_2}^2$ are given as follows, respectively

$$\mathrm{E}(T_{W_2}^2) = m_1\sum_{k=1}^{i}\pi_{kk}$$

$$\mathrm{V}(T_{W_2}^2) = 2m_1\sum_{k=1}^{i}\sum_{j=1}^{i}\pi_{kj}^2$$

By using large-sample normal approximation theory, the control limit of $T_{W_2}^2$ is given by

$$CL_{T_{W_2}^2} = m_1\sum_{k=1}^{i}\pi_{kk} \pm \Phi^{-1}(1-\alpha)\sqrt{2m_1\sum_{k=1}^{i}\sum_{j=1}^{i}\pi_{kj}^2}\ .$$

Derivation of mean and variance of $T_{W_2}^2$:

$$
\begin{aligned}
\mathrm{E}(T_{W_2}^2) &= \mathrm{E}(\sum_{k=1}^{i}\sum_{j=1}^{i}\pi_{kj}(\sum_{l=1}^{m_1}\tilde{v}_{kl}\tilde{v}_{jl})) \\
&= \sum_{k=1}^{i}\pi_{kk}\,\mathrm{E}(\sum_{l=1}^{m_1}\tilde{v}_{kl}^2) + \sum_{k=1}^{i}\sum_{k\neq j}^{\mathrm{i}}\pi_{kj}\,\mathrm{E}(\sum_{l=1}^{m_1}\tilde{v}_{kl}\tilde{v}_{jl}) \\
&= \sum_{k=1}^{i}\pi_{kk}\,\mathrm{E}(\sum_{l=1}^{m_1}\tilde{v}_{kl}^2) + \sum_{k=1}^{i}\sum_{k\neq j}^{\mathrm{i}}\pi_{kj}\,\mathrm{E}(\tilde{v}_{11}\tilde{v}_{21} + \tilde{v}_{11}\tilde{v}_{22} + \cdots + \tilde{v}_{im_1}\tilde{v}_{i-1m_1}) \\
&= m_1\sum_{k=1}^{i}\pi_{kk}
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Var}(T_{W_2}^2) &= \mathrm{Var}(\sum_{k=1}^{i}\sum_{j=1}^{i}\pi_{kj}(\sum_{l=1}^{m_1}\tilde{v}_{kl}\tilde{v}_{jl})) \\
&= \mathrm{Var}\left[\sum_{k=1}^{i}\pi_{kk}\sum_{l=1}^{m_1}\tilde{v}_{kl}^2 + 2\sum_{k=1}^{i}\sum_{j<k}^{\mathrm{i}}\pi_{kj}\sum_{l=1}^{m_1}\tilde{v}_{kl}\tilde{v}_{jl}\right] \\
&= \sum_{k=1}^{i}\pi_{kk}^2\,\mathrm{Var}(\sum_{l=1}^{m_1}\tilde{v}_{kl}^2) + 4\sum_{k=1}^{i}\sum_{j<k}^{\mathrm{i}}\pi_{kj}^2\,\mathrm{Var}(\sum_{l=1}^{m_1}\tilde{v}_{kl}\tilde{v}_{jl}) \\
&= \sum_{k=1}^{i}\pi_{kk}^2\,\mathrm{Var}(\sum_{l=1}^{m_1}\tilde{v}_{kl}^2) + 4\sum_{k=1}^{i}\sum_{j<k}^{i}\pi_{kj}^2\left[\mathrm{Var}(\tilde{v}_{11}) + \mathrm{Var}(\tilde{v}_{12}) + \cdots + \mathrm{Var}((\tilde{v}_{im_1})\mathrm{Var}(\tilde{v}_{i-1m_1})\right] \\
&= 2m_1\sum_{k=1}^{i}\pi_{kk}^2 + 4m_1\sum_{k=1}^{i}\sum_{j<k}^{i}\pi_{kj}^2 \\
&= 2m_1\left(\sum_{k=1}^{i}\pi_{kk}^2 + 2\sum_{k=1}^{i}\sum_{j<k}^{i}\pi_{kj}^2\right) \\
&= 2m_1\sum_{k=1}^{i}\sum_{j=1}^{i}\pi_{kj}^2
\end{aligned}
$$

4.6.3 Comparison of average run length

This subsection presents simulation results using tonnage stamping signals for comparing average run length (ARL_1) values for the following four test-statistics: all wavelet coefficients-based statistic T_A^2, *VisuShrink*-based statistic T_{Js}^2, *VET*-based statistic

T^2_{Vet}, *VertiShrink*-based statistic T^2_{Vs}, and the proposed statistic $T^2_{W_1}$ and $T^2_{W_2}$. Control limits are set to make in-control ARL$_0$=200 (Kang and Albin 2000).

On the other hands, based on the engineering knowledge of stamping processes, we know that different potenital process faluires may occur in different areas of the tonnage signlas (Jin and Shi 1999). For example, the problems of loose tie rods and worn bearings ususally occurs at the central areas of a signal, showing different features of the peak tonnage. A excessive dynamic interaction between the press and the nitrogen cushion system are frequently happened at transition jump edges (see Jin and Shi 1999 for detailed process faulures in a stamping process). We will utilize those engineering knowledge for our simulation stuides.

4.6.3.1 Only process variance is changed

In order to evaluate the effectivenss of the proposed monitoring method for functional data, we use 24 tonnage signals with *N*=256 as baseline, as shown in Figure 4.1. In our studies, we used Haar wavelets and set the lowest resolution level (*L*) as 4. Random noises from normal distribution $N(0,\sigma^2)$ with $\sigma^2=1$ are added to generate 1,000 replications for each study. We compare ARL$_1$ values of the proposed statistic $T^2_{W_1}$ with those of the existing methods such as T^2_A, T^2_{Js}, T^2_{Vet}, and T^2_{Vs}. The simulated curve at time *i* is generated under the shift level of $\kappa\ (>1)$ as follows:

$$y_i(t_j)=\begin{cases} f_0(t_j)+\gamma_{t_j}+\varepsilon(t_j), & t_j\in A \\ f_0(t_j)+\varepsilon(t_j), & elsewhere, \end{cases}$$

where variance shift $\gamma_{t_j} \sim N(0, \kappa\sigma_\gamma^2)$ where $\kappa\ (>1)$ is the level of shift ($\kappa = 1$ means no variance change), $\sigma_\gamma^2 = \sum_{k \in D} h_{kj}^2 \tau_k^2$ and A is the shift areas [91, 110] in the time domain. In reality, the detection of loose tie rod or worn bearing can be identified based on the information of the signal in those areas (Jin and Shi 1999).

Table 4.1 gives the ARL_1 values for the five methods over different level of variance changes. As expected, T_A^2 chart does not work well for high-dimensional functional data (with N=256), and it has large ARL_1 values than other charts. Because T_W^2 chart monitors only random wavelet coefficients, it performs better than other charts ($T_{Js}^2, T_{Vs}^2, T_{Vet}^2$) over the entire level of the variance shifts. Based on the proposed WMVT, in this experiment, three wavele coeffeicnts, ($c_{4,6}, c_{4,7}, c_{4,8}$), are selected as random effects variables. The support areas in the time domain of these coefficients cover from from t_{81} to t_{128}. In particular, the proposed T_W^2 performs better in detecting smaller shifts compared to other procedures. Based on the simulation studies, the proposed chart is generally effective for detecting process changes than the methods extended from ideas given in the literature.

Table 4.1 Comparison of ARLs when only process variance is changed

Level of variance shift (κ)	T_A^2	T_{Js}^2	T_{Vs}^2	T_{Vet}^2	$T_{W_1}^2$
1	202.22	201.38	201.85	199.95	200.37
1.5	190.85	186.25	182.37	180.78	155.58
2	180.74	160.82	156.12	140.24	89.56
2.5	167.12	141.29	135.31	122.36	43.71
3	124.06	90.28	60.38	54.10	23.14
3.5	74.06	56.63	50.94	28.86	13.73
4	30.778	21.276	20.23	11.946	6.773
4.5	20.478	15.248	14.746	8.691	4.676
5	15.353	11.137	10.544	6.449	3.603

4.6.3.2 Both process mean and variance are changed

In this case, both process mean and variance are shifted at different local segments. We also compare ARL$_1$ performance of the proposed statistic $T_{W_2}^2$ with those of other charts. The monitoring curves are generated with a corresponding shift as follows:

$$y_i(t_j) = \begin{cases} f_0(t_j) + \gamma_{t_j} + \varepsilon(t_j), & t_j \in A_j \\ f_0(t_i) + \xi_{t_i} + \varepsilon(t_i), & t_i \in A_i \\ f_0(t_j) + \varepsilon(t_j), & elsewhere, \end{cases}$$

where mean shift $\xi_{t_j} \sim N(0, \sigma_\xi^2)$ and A_i, A_j are the changed areas in the time domain. In practice, loose tie rods or worn bearings are usually occurred at central peak areas and excessive dynamic interaction between the press and the nitrogen cushion system are shown at transient rising edge areas (Jin and Shi 1999).

Table 4.2 gives the ARL$_1$ values for six methods. This table indicates that $T_{W_2}^2$ chart shows better performance than other charts. In addition, regardless of the level of mean shift, $T_{W_1}^2$ chart has similar ARL$_1$ results with same variance shift because random effect variables do not cover mean shift areas.

Table 4.2 Comparison of ARLs under both mean shift [65, 70] and variance shift [91, 110]

Variance level (κ)	Mean shift (σ_ξ^2)	T_A^2	T_{Js}^2	T_{Vs}^2	T_{Vet}^2	$T_{W_1}^2$	$T_{W_2}^2$
1	2	191.24	190.32	194.31	180.23	197.74	130.60
	4	184.42	165.48	169.57	137.16	199.30	72.90
	6	166.70	154.38	148.47	96.48	198.58	46.50
	8	136.20	128.98	125.46	72.96	199.42	36.30
	10	121.24	115.16	111.65	56.30	200.88	23.12
2	2	185.83	175.15	168.42	127.59	121.06	63.47
	4	165.90	146.21	137.86	88.75	122.24	41.20
	6	135.99	108.76	105.61	66.70	119.30	32.18
	8	130.46	93.27	90.89	52.29	128.76	23.70
	10	108.15	83.77	76.27	44.33	121.94	21.31
3	2	182.60	160.32	157.73	122.71	87.60	61.21
	4	164.90	148.38	139.72	96.77	82.81	41.89
	6	136.64	118.32	105.67	66.20	85.20	32.59
	8	119.77	93.35	91.48	53.49	87.40	25.16
	10	105.06	82.40	78.51	44.70	82.35	19.62
4	2	173.10	138.16	147.17	101.83	58.69	44.97
	4	140.20	122.30	116.35	73.22	60.37	34.26
	6	131.55	98.40	91.63	58.84	57.86	27.03
	8	109.01	83.40	80.84	46.81	59.20	22.33
	10	96.90	71.63	64.55	37.01	58.24	18.23

4.7 Concluding Remarks

In this research, we proposed a new SPC procedure for functional data to consider systematic variations of curves at certain local regions. For this, we presented a wavelet-based local-random-effect model to characterize between-curve local variations. The penalized likelihood-based wavelet mean and variance thresholding method (WMVT) is easy to understand and implement. Closed-form expressions are provided for the estimates of the mean and variance thresholding parameters. Based on real-life data analyses, we found that the WMVT model adequately describes local variations and uses fewer model parameters, and the proposed SPC model can detect the changes of process local variations efficiently by using fewer coefficients in the wavelet domain.

Although the penalized likelihood method limits the number of coefficients in the model, this research did not delve deeper into data reduction because it it was considered beyond its scope. Although procedures like those used by Jeong *et al.* (2006a) could be used to formulate data reduction metrics, these procedures may not be sufficient in these circumstances. This is because the penalized likelihood procedure usually uses cross-validations to decide tuning parameters. Consequently, the problem is more complicated than the single-curve studies of data reduction contained in Jeong *et al.*(2006a), and further work is needed in this direction.

CHAPTER 5

Kernel-Based Regression with Lagged Dependent Variables

5.1 Introduction

Regression is the statistical methodology for predicting values of one or more dependent variables based on a collection of independent variables. The traditional regression models such as the ordinary least squares (OLS) regression, assumes that there is no autocorrelation in the residuals of observations (Keele and Kelly 2005). That is, the residual at any observation is not correlated with any other residual. However, many functional data and time series data usually violate this assumption. That is why a traditional model often fails to describe the data where there are autocorrelations between dependent variables. If the dependent variable is autocorrelated, OLS estimators will be biased, inconsistent, and inefficient regardless of the properties of the error term (Anselin 1998).

In regression analysis, one of the effective ways of reflecting autocorrelations for better accurate prediction is to include lagged dependent variables (LDVs) as a part of independent variables. By introducing LDVs, regression model reflects that a dependent variable changes incrementally across time and provides better fits for data with autocorrelations. Recently, there has been growing interest in considering the model with lagged dependent variables. For instance, Beck and Kats (1995) utilized LDVs as a means of capturing the dynamics of politics. They proposed to consider LDVs in the OLS-panel-corrected standard errors (PCSE) model to capture dynamic tendencies. Thies and Porche (2007) utilized LDVs to assess the dynamic aspects of agricultural producer support. Garin and Montero (2007) also developed tourism demand function using ordinary regression model with LDVs considering the correlation between the numbers of tourist on each year.

Also, the kernel-based regression method has been recently used to explore the nonlinearity of data in an easy way through the use of various kernel functions (Muller *et al*. 2001, Ruiz and Lopez 2001). Popular kernel-based regression methods include the support vector machines for regression (SVR) and kernel ridge regression (KRR) (Saunders *et al*. 1998). However, existing kernel-based regression method has the drawback where it assumes that there is no autocorrelation in the residuals of observations. To avoid such a problem, this paper proposes a kernel-based regression model with lagged dependent variables (LDVs) to consider both autocorrelations of response variables and nonlinearity of data. To the best of our knowledge, little prior works deal with LDVs taking as input in kernel-based regression methods.

Among several kernel-based approaches, kernel ridge regression (KRR), a kernel version of the ridge regression, makes it possible to perform a sparse non-linear regression by constructing a linear regression function in a high dimensional feature space (Saunders *et al*. 1998). Therefore, in the paper, we propose a RR-based LDVs model, which combines RR with LDV and the KRR-based LDVs model that explores the nonlinearity of data using various kernel functions. In other words, this study explores the nonlinearity between a response variable, independent variables and even previous response variables. The nonlinearity of model is a critical limitation of existing LDV models. We will present the procedure of how to kernelize the LDV model when previous response variables are considered. The proposed model can explore the nonlinear relationship between the response and both independent variables and past response variables using various kernel functions. In this case, however, it will be difficult to apply existing kernel trick directly because of LDVs. We derive a kernel ridge estimator with LDVs using a new mapping idea so that the nonlinear mapping does not have to be computed explicitly depending on kernel types. The experimental results show that the proposed algorithms are promising alternatives for high dimensional dataset with autocorrelations of dependent variables.

The remainder of this paper is organized as follows. Section 5.2 briefly reviews relevant literatures. Our proposed RR-based LDVs and KRR-based LDVs models are presented in Section 5.3. The performances of the proposed approaches are compared with conventional methods in Section 5.4. Finally, conclusions and recommendations for future study are presented in Section 5.5.

5.2 Ordinary Regression Model with Lagged Dependent Variables

Following Tanizaki (2000), we formulate a multivariate regression model with LDVs as the basis of our study. Consider a multivariate regression model with LDVs in a matrix form that takes the form

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Y}\boldsymbol{\alpha} + \boldsymbol{\varepsilon},$$

where

$$\mathbf{y} = \begin{pmatrix} y_{p+1} \\ y_{p+2} \\ \vdots \\ y_T \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} x_{p+11} & x_{p+12} & \cdots x_{p+1k} \\ x_{p+21} & x_{p+22} & \cdots x_{p+2k} \\ \vdots & \vdots & \vdots \quad \vdots \\ x_{T1} & x_{T2} & \cdots x_{Tk} \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix}, \quad \mathbf{Y} = \begin{pmatrix} y_p & y_{p-1} & \cdots y_1 \\ y_{p+1} & y_p & \cdots y_2 \\ \vdots & \vdots & \vdots \quad \vdots \\ y_{T-1} & y_{T-2} & \cdots y_{T-p} \end{pmatrix}, \quad \boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{pmatrix},$$

$\boldsymbol{\varepsilon} \sim N(0,\sigma^2)$, $\mathbf{X}$ is predictor variables related to a response variables $\mathbf{y}$ and $\mathbf{Y}$ is dependant variables in which the model has autocorrelation among response variables with a specific lag *p*.

Regression coefficients $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ can be estimated by least square method as

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{OLSE} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{y} - \mathbf{Y}\hat{\boldsymbol{\alpha}}_{OLSE}) \\ \hat{\boldsymbol{\alpha}}_{OLSE} &= (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{OLSE}) \end{aligned} \tag{5.1}$$

$\hat{\boldsymbol{\beta}}_{OLSE}$ in Eq. (5.1) can be rewritten as

$$\hat{\boldsymbol{\beta}}_{OLSE} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\left[\mathbf{y} - \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{OLSE})\right]$$

$$= \left[(\mathbf{X}^T\mathbf{X}) - \mathbf{X}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{X}\right]^{-1}\mathbf{X}^T(\mathbf{y} - \mathbf{Y}\hat{\boldsymbol{\alpha}}^*)$$

where, $\hat{\boldsymbol{\alpha}}^* = \left(\mathbf{Y}^T\mathbf{Y}\right)^{-1}\mathbf{Y}^T\mathbf{y}$, which represents ordinary least square estimates of coefficients in the AR (*p*) model. Therefore, least square estimates of $(\hat{\boldsymbol{\beta}}_{OLSE}, \hat{\boldsymbol{\alpha}}_{OLSE})$ can be estimated as follows

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{OLSE} &= \left[(\mathbf{X}^T\mathbf{X}) - \mathbf{X}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{X}\right]^{-1}\mathbf{X}^T(\mathbf{y} - \mathbf{Y}\hat{\boldsymbol{\alpha}}^*)\\ \hat{\boldsymbol{\alpha}}_{OLSE} &= (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{OLSE})\end{aligned}$$

where, $\hat{\boldsymbol{\alpha}}^* = \left(\mathbf{Y}^T\mathbf{Y}\right)^{-1}\mathbf{Y}^T\mathbf{y}$

In practice, it is common to include an intercept term in the model. To obtain least square estimates of $(\hat{\boldsymbol{\beta}}_{OLSE}, \hat{\boldsymbol{\alpha}}_{OLSE}, \hat{b}_{OLSE})$ in the model with the intercept term *b*, the following optimization problem should be solved.

$$\min_{\boldsymbol{\beta},\boldsymbol{\alpha},b} \left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Y}\boldsymbol{\alpha} - \mathbf{1}b\right)^T \left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Y}\boldsymbol{\alpha} - \mathbf{1}b\right)$$

As a result, the estimates can be derived as follows.

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{OLSE} &= \left[(\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}) - \tilde{\mathbf{X}}^T\tilde{\mathbf{Y}}(\tilde{\mathbf{Y}}^T\tilde{\mathbf{Y}})^{-1}\tilde{\mathbf{Y}}^T\tilde{\mathbf{X}}\right]^{-1}\tilde{\mathbf{X}}^T(\tilde{\mathbf{y}} - \tilde{\mathbf{Y}}\hat{\boldsymbol{\alpha}}^*)\\ \hat{\boldsymbol{\alpha}}_{OLSE} &= (\tilde{\mathbf{Y}}^T\tilde{\mathbf{Y}})^{-1}\tilde{\mathbf{Y}}^T(\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}_{OLSE})\\ \hat{b}_{OLSE} &= \frac{1}{l}\mathbf{1}^T(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{OLSE} - \mathbf{Y}\hat{\boldsymbol{\alpha}}_{OLSE})\end{aligned}$$

where $\mathbf{1}$ is the vector with all 1's and $\tilde{\mathbf{X}} = (\mathbf{X} - \bar{\mathbf{X}}\mathbf{1})$ and $\tilde{\mathbf{Y}} = (\mathbf{Y} - \bar{\mathbf{Y}}\mathbf{1})$ which represents the centered matrices of $\mathbf{X}$ and $\mathbf{Y}$, respectively. Also, $\bar{\mathbf{X}}$ and $\bar{\mathbf{Y}}$ are the column mean of $\mathbf{X}$ and $\mathbf{Y}$, respectively. In addition, $\tilde{\mathbf{y}}$ means the centered vector of $\mathbf{y}$ obtained by $\tilde{\mathbf{y}} = (\mathbf{y} - \bar{\mathbf{y}}\mathbf{1})$. Also, l is the number of observations.

5.3 Proposed Methodology

In this section, we present our proposed ridge regression with LDVs model and its kernelized version.

5.3.1 Ridge regression with lagged dependent variables

Ridge regression is a popular regularization method for ill-posed problems (Hastie 2001). The regression coefficients in RR with LDVs minimizes the following penalized residual sum of squares (RSS)

$$RSS(\boldsymbol{\beta},\boldsymbol{\alpha}) = \left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Y}\boldsymbol{\alpha}\right)^T \left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Y}\boldsymbol{\alpha}\right) + \lambda\boldsymbol{\beta}^T\boldsymbol{\beta} + \gamma\boldsymbol{\alpha}^T\boldsymbol{\alpha}$$

where, $\lambda \geq 0$ and $\gamma \geq 0$ are predetermined constants controlling the amount of shrinkage. Therefore, estimates of regression coefficients can be obtained by solving the following equations

$$\frac{\partial RSS(\boldsymbol{\beta},\boldsymbol{\alpha})}{\partial\boldsymbol{\beta}} = -2\mathbf{X}^T\left(\mathbf{y} - \mathbf{Y}\boldsymbol{\alpha}\right) + 2\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} + 2\lambda\boldsymbol{\beta} = 0$$

$$\frac{\partial RSS(\boldsymbol{\beta},\boldsymbol{\alpha})}{\partial\boldsymbol{\alpha}} = -2\mathbf{Y}^T\left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\right) + 2\mathbf{Y}^T\mathbf{Y}\boldsymbol{\alpha} + 2\gamma\boldsymbol{\alpha} = 0$$

Hence,

$$\hat{\boldsymbol{\beta}}_{rr} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T(\mathbf{y} - \mathbf{Y}\hat{\boldsymbol{\alpha}}_{rr}) \tag{5.2}$$

$$\hat{\boldsymbol{\alpha}}_{rr} = (\mathbf{Y}^T\mathbf{Y} + \gamma\mathbf{I})^{-1}\mathbf{Y}^T(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{rr}) \quad (5.3)$$

To proceed to the next step, Eq. (5.3) should be modified as an alternative expression. The different expression of Eq. (5.3) can be derived using identity properties as follows;

$$\hat{\boldsymbol{\alpha}}_{rr} = (\mathbf{Y}^T\mathbf{Y} + \gamma\mathbf{I})^{-1}\mathbf{Y}^T(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{rr}) = \mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \gamma\mathbf{I})^{-1}(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{rr}) \quad (5.4)$$

Derivation of Eq. (5.4)

For any matrix U and V with $\left(\mathbf{I} + \mathbf{UV}\right)$and $\left(\mathbf{I} + \mathbf{VU}\right)$ are nonsingular, the following simple identity property holds (Henderson and Searle 1981)

$$(\mathbf{UV} + \mathbf{I})^{-1}\mathbf{U} = \mathbf{U}(\mathbf{VU} + \mathbf{I})^{-1} \quad (5.5)$$

Letting $\mathbf{U} = \frac{1}{\gamma}\mathbf{Y}^\mathbf{T}$, $\mathbf{V} = \mathbf{Y}$ in Eq. (5.5), we have

$$(\mathbf{UV} + \mathbf{I})^{-1}\mathbf{U} = (\frac{1}{\gamma}\mathbf{Y}^\mathbf{T}\mathbf{Y} + \mathbf{I})^{-1}\frac{1}{\gamma}\mathbf{Y}^\mathbf{T}$$
$$\mathbf{U}(\mathbf{VU} + \mathbf{I})^{-1} = \frac{1}{\gamma}\mathbf{Y}^\mathbf{T}(\mathbf{Y}\frac{1}{\gamma}\mathbf{Y}^\mathbf{T} + \mathbf{I})^{-1}$$

Therefore,

$$(\mathbf{Y}^\mathbf{T}\mathbf{Y} + \gamma\mathbf{I})^{-1}\mathbf{Y}^\mathbf{T} = \mathbf{Y}^\mathbf{T}(\mathbf{Y}\mathbf{Y}^\mathbf{T} + \gamma\mathbf{I})^{-1}$$

Thus, the estimates of ridge coefficients $\hat{\boldsymbol{\alpha}}_{rr}$ can be rewritten as follows.

$$\hat{\boldsymbol{\alpha}}_{rr} = (\mathbf{Y}^\mathbf{T}\mathbf{Y} + \gamma\mathbf{I})^{-1}\mathbf{Y}^\mathbf{T} = \mathbf{Y}^\mathbf{T}(\mathbf{Y}\mathbf{Y}^\mathbf{T} + \gamma\mathbf{I})^{-1}$$

The ridge coefficient $\boldsymbol{\beta}_{rr}$ can be represented using Eq. (5.2) and Eq. (5.4)

$$\hat{\boldsymbol{\beta}}_{rr} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T(\mathbf{y} - \mathbf{Y}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \gamma\mathbf{I})^{-1}(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{rr}))$$

$$(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})\hat{\boldsymbol{\beta}}_{rr} = \mathbf{X}^T\mathbf{y} - \mathbf{X}^T\mathbf{Y}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \gamma\mathbf{I})^{-1}(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{rr})$$

Namely,

$$\hat{\boldsymbol{\beta}}_{rr} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I} - \mathbf{X}^T\mathbf{Y}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \gamma\mathbf{I})^{-1}\mathbf{X})^{-1}\left(\mathbf{X}^T - \mathbf{X}^T\mathbf{Y}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \gamma\mathbf{I})^{-1}\right)\mathbf{y} \qquad (5.6)$$

From Eq. (5.4) and Eq. (5.6), the coefficient estimators for ridge regression with LDVs are given as follows.

$$\hat{\boldsymbol{\beta}}_{rr} = (\mathbf{X}^T(\mathbf{X} - \mathbf{Y}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \gamma\mathbf{I})^{-1}\mathbf{X}) + \lambda\mathbf{I})^{-1}\mathbf{X}^T\left(\mathbf{I} - \mathbf{Y}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \gamma\mathbf{I})^{-1}\right)\mathbf{y}$$

$$\hat{\boldsymbol{\alpha}}_{rr} = \mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \gamma\mathbf{I})^{-1}(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{rr}).$$

The coefficient estimators for ridge regression model including intercept term *b* can be obtained by minimizing the following RSS.

$$RSS(\boldsymbol{\beta}, \boldsymbol{\alpha}, b) = \left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Y}\boldsymbol{\alpha} - \mathbf{1}b\right)^T\left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Y}\boldsymbol{\alpha} - \mathbf{1}b\right) + \lambda\boldsymbol{\beta}^T\boldsymbol{\beta} + \gamma\boldsymbol{\alpha}^T\boldsymbol{\alpha}$$

The obtained coefficient estimators with $\mathbf{X}$ and $\mathbf{Y}$ replaced by the centered input $\tilde{\mathbf{X}} = (\mathbf{X} - \bar{\mathbf{X}}\mathbf{1})$, $\tilde{\mathbf{Y}} = (\mathbf{Y} - \bar{\mathbf{Y}}\mathbf{1})$ and $\tilde{\mathbf{y}}$ replaced by the centered ouptut $\tilde{\mathbf{y}} = (\mathbf{y} - \bar{\mathbf{y}}\mathbf{1})$

are as follows.

$$
\begin{aligned}
\hat{\boldsymbol{\beta}}_{rr} &= (\tilde{\mathbf{X}}^T(\tilde{\mathbf{X}} - \tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^T(\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^T + \gamma\mathbf{I})^{-1}\tilde{\mathbf{X}}) + \lambda\mathbf{I})^{-1}\tilde{\mathbf{X}}^T\left(\mathbf{I} - \tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^T(\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^T + \lambda\mathbf{I})^{-1}\right)\tilde{\mathbf{y}} \\
\hat{\boldsymbol{\alpha}}_{rr} &= \tilde{\mathbf{Y}}^T(\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^T + \gamma\mathbf{I})^{-1}(\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}_{rr}) \\
\hat{b}_{rr} &= \frac{1}{l}\mathbf{1}^T(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{rr} - \mathbf{Y}\hat{\boldsymbol{\alpha}}_{rr})
\end{aligned}
$$

5.3.2 Kernel ridge regression model with lagged dependent variables

Kernel-based method is to map nonlinear data matrix in original space into linear ones in high dimensional feature space (Muller *et al*. 2001, Ruiz and Lopez 2001). In the regression model with lagged dependent variables, the original data matrix X and Y are transformed to $\Phi(\mathbf{X})$ and $\Phi(\mathbf{Y})$ in the feature space as follows:

$$\mathbf{y} = \Phi(\mathbf{X})\boldsymbol{\beta} + \Phi(\mathbf{Y})\boldsymbol{\alpha} + \boldsymbol{\varepsilon}$$

where, $\Phi(\cdot)$ represents a mapping function to transform the original input into a high dimensional feature space, in which linear regression is equivalent to nonlinear regression in the input space. Note that the nonlinear mapping $\Phi(\cdot)$ does not have to be computed explicitly in a kernel method. Instead, kernel function K replaces the dot products $\langle \Phi(u_1), \Phi(u_2) \rangle$, which is called the "kernel trick". That is,

$$\mathrm{K}_{ij} = <\Phi(\mathbf{X}_i), \Phi(\mathbf{X}_j)> = K(\mathbf{X}_i, \mathbf{X}_j)$$

However, it is hard to apply kernel trick to Eq. (5.6) directly. The following Lemma makes it possible to apply a nonlinear mapping with kernel trick to ridge regression with LDVs. Using these results, we can derive simple formula to obtain the kernel ridge estimators with dependent variables.

Lemma 5.1.

Let $\mathbf{K}_1 = \Phi(\mathbf{X})\Phi(\mathbf{X})^T$ and $\mathbf{K}_2 = \Phi(\mathbf{Y})\Phi(\mathbf{Y})^T$ be the kernel matrices of independent variables and dependent variables, respectively. Then, the regression coefficients for KRR model with LDVs can be expressed as follows.

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{krr} &= \Phi(\mathbf{X})^T((\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{K}_1+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{y}\\ \hat{\boldsymbol{\alpha}}_{krr} &= \Phi(\mathbf{Y})^T(\mathbf{K}_2+\gamma\mathbf{I})^{-1}(\mathbf{y}-\Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr})\end{aligned}$$

Proof of Lemma 5.1

Letting $\mathbf{A}$ be $\mathbf{Y}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T+\gamma\mathbf{I})^{-1}$, $\hat{\boldsymbol{\beta}}_{rr}$ in Eq. (5.6) can be represented as follows.

$$\hat{\boldsymbol{\beta}}_{rr} = \left(\mathbf{X}^\mathbf{T}(\mathbf{X}-\mathbf{A}\mathbf{X})+\lambda\mathbf{I}\right)^{-1}\mathbf{X}^\mathbf{T}(\mathbf{I}-\mathbf{A})\mathbf{y}$$

Letting $\mathbf{U} = \frac{1}{\lambda}\mathbf{X}^T$, $\mathbf{V} = (\mathbf{X}-\mathbf{A}\mathbf{X})$ in Eq. (5.5), we have

$$\begin{aligned}(\mathbf{U}\mathbf{V}+\mathbf{I})^{-1}\mathbf{U} &= (\frac{1}{\lambda}\mathbf{X}^\mathbf{T}(\mathbf{X}-\mathbf{A}\mathbf{X})+\mathbf{I})^{-1}\frac{1}{\lambda}\mathbf{X}^\mathbf{T}\\ \mathbf{U}(\mathbf{V}\mathbf{U}+\mathbf{I})^{-1} &= \frac{1}{\lambda}\mathbf{X}^\mathbf{T}((\mathbf{X}-\mathbf{A}\mathbf{X})\frac{1}{\lambda}\mathbf{X}^\mathbf{T}+\mathbf{I})^{-1}\end{aligned}$$

Namely,

$$(\mathbf{X}^\mathbf{T}(\mathbf{X}-\mathbf{A}\mathbf{X})+\lambda\mathbf{I})^{-1}\mathbf{X}^\mathbf{T} = \mathbf{X}^\mathbf{T}((\mathbf{X}-\mathbf{A}\mathbf{X})\mathbf{X}^\mathbf{T}+\lambda\mathbf{I})^{-1}$$

Therefore, the ridge coefficient $\hat{\boldsymbol{\beta}}_{rr}$ can be rewritten in an alternative expression as follows

$$\hat{\boldsymbol{\beta}}_{rr} = (\mathbf{X}^T(\mathbf{X}-\mathbf{A}\mathbf{X})+\lambda\mathbf{I})^{-1}\mathbf{X}^T(\mathbf{I}-\mathbf{A})\mathbf{y} = \mathbf{X}^T((\mathbf{I}-\mathbf{A})\mathbf{X}\mathbf{X}^T+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{A})\mathbf{y}$$

Based on Eq. (5.5) and the above equation of the ridge coefficient $\hat{\boldsymbol{\beta}}_{rr}$, we have

$$\hat{\boldsymbol{\beta}}_{rr} = \mathbf{X}^T((\mathbf{I}-\mathbf{A})\mathbf{X}\mathbf{X}^T+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{A})\mathbf{y} \tag{5.7}$$

$$\hat{\boldsymbol{\alpha}}_{rr} = \mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T+\gamma\mathbf{I})^{-1}(\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}}_{rr}) \tag{5.8}$$

where $\mathbf{A} = \mathbf{Y}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T+\gamma\mathbf{I})^{-1}$.

The original data matrices $\mathbf{X}$ and $\mathbf{Y}$ are nonlinearly transformed to $\Phi(\mathbf{X})$ and $\Phi(\mathbf{Y})$ in the feature space, respectively. Similarly, Eqs. (5.7)-(5.8) can be expressed using $\Phi(\mathbf{X})$ and $\Phi(\mathbf{Y})$ in the feature space as follows

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{krr} &= \Phi(\mathbf{X})^T((\mathbf{I}-\mathbf{A})\Phi(\mathbf{X})\Phi(\mathbf{X})^T+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{A})\mathbf{y}\\ \hat{\boldsymbol{\alpha}}_{krr} &= \Phi(\mathbf{Y})^T(\Phi(\mathbf{Y})\Phi(\mathbf{Y})^T+\gamma\mathbf{I})^{-1}(\mathbf{y}-\Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr})\end{aligned}$$

where $\mathbf{A} = \Phi(\mathbf{Y})\Phi(\mathbf{Y})^T(\Phi(\mathbf{Y})\Phi(\mathbf{Y})^T+\gamma\mathbf{I})^{-1}$

Letting $\mathbf{K}_1$ be $\Phi(\mathbf{X})\Phi(\mathbf{X})^T$ and $\mathbf{K}_2$ be $\Phi(\mathbf{Y})\Phi(\mathbf{Y})^T$, the kernel ridge regression with LDVs estimator can be represented using $\mathbf{K}_1$ and $\mathbf{K}_2$ as

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{krr} &= \Phi(\mathbf{X})^T((\mathbf{I}-\mathbf{A})\mathbf{K}_1+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{A})\mathbf{y}\\ \hat{\boldsymbol{\alpha}}_{krr} &= \Phi(\mathbf{Y})^T(\mathbf{K}_2+\gamma\mathbf{I})^{-1}(\mathbf{y}-\Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr})\end{aligned}$$

where $\mathbf{A} = \mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1}$

That is,

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{krr} &= \Phi\left(\mathbf{X}\right)^T((\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{K}_1+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{y}\\ \hat{\boldsymbol{\alpha}}_{krr} &= \Phi(\mathbf{Y})^T(\mathbf{K}_2+\gamma\mathbf{I})^{-1}(y-\Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr}).\end{aligned}$$

The explicit expressions of $\hat{\boldsymbol{\beta}}_{krr}$ and $\hat{\boldsymbol{\alpha}}_{krr}$ are not available because explicit forms of $\Phi(\mathbf{X})$ and $\Phi(\mathbf{Y})$ are unknown in a kernel-based method. Instead, the prediction of the observation can be obtained through the kernel trick as follows:

$$\begin{aligned}\hat{y} &= \Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr}+\Phi(\mathbf{Y})\hat{\boldsymbol{\alpha}}_{krr}\\ &= \Phi(\mathbf{X})\Phi(\mathbf{X})^T((\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{K}_1+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{y}\\ &\quad +\Phi(\mathbf{Y})\Phi(\mathbf{Y})^T(\mathbf{K}_2+\gamma\mathbf{I})^{-1}(\mathbf{y}-\Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr})\\ &= \mathbf{K}_1((\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{K}_1+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{y}+\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1}(\mathbf{y}-\Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr})\end{aligned}$$

Similarly, the point prediction estimate of a new observation can be obtained by

$$\begin{aligned}\hat{y}_{new} &= \Phi(\mathbf{X}_{new})\hat{\boldsymbol{\beta}}_{krr}+\Phi(\mathbf{Y}_{new})\hat{\boldsymbol{\alpha}}_{krr}\\ &= \Phi(\mathbf{X}_{new})\Phi(\mathbf{X})^T((\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{K}_1+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{y}\\ &\quad +\Phi(\mathbf{Y}_{new})\Phi(\mathbf{Y})^T(\mathbf{K}_2+\gamma\mathbf{I})^{-1}(y-\Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr})\\ &= \mathbf{K}_{new1}((\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{K}_1+\lambda\mathbf{I})^{-1}(\mathbf{I}-\mathbf{K}_2(\mathbf{K}_2+\gamma\mathbf{I})^{-1})\mathbf{y}+\mathbf{K}_{new2}(\mathbf{K}_2+\gamma\mathbf{I})^{-1}(y-\Phi(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr})\end{aligned}$$

where, $\mathbf{K}_{new1}=\Phi(\mathbf{X}_{new})\Phi(\mathbf{X})^T$, $\mathbf{K}_{new2}=\Phi(\mathbf{Y}_{new})\Phi(\mathbf{Y})^T$.

5.3.3 Centering of mapped data points for KRR-LDVs models

In KRR, the intercept term b is helpful for achieving better prediction accuracy but, often overlooked. To consider the intercept term b in KRR-LDVs model, the coefficient estimators can be obtained by minimizing the following RSS,

$$RSS(\boldsymbol{\beta},\boldsymbol{\alpha},b) = \left(\mathbf{y} - \Phi\left(\mathbf{X}\right)\boldsymbol{\beta} - \Phi\left(\mathbf{Y}\right)\boldsymbol{\alpha} - \mathbf{1}b\right)^T \left(\mathbf{y} - \Phi\left(\mathbf{X}\right)\boldsymbol{\beta} - \Phi\left(\mathbf{Y}\right)\boldsymbol{\alpha} - \mathbf{1}b\right) + \lambda\boldsymbol{\beta}^T\boldsymbol{\beta} + \gamma\boldsymbol{\alpha}^T\boldsymbol{\alpha}$$

.

We can show that the coefficient estimators for kernel based regression methods can be calculated using the centered nonlinear mapping of both $\mathbf{X}$ and $\mathbf{Y}$ as follows.

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{krr} &= (\tilde{\Phi}(\mathbf{X})^T(\tilde{\Phi}(\mathbf{X}) - \tilde{\Phi}(\mathbf{Y})\tilde{\Phi}(\mathbf{Y})^T(\tilde{\Phi}(\mathbf{Y})\tilde{\Phi}(\mathbf{Y})^T + \gamma\mathbf{I})^{-1}\tilde{\Phi}(\mathbf{X})) + \lambda\mathbf{I})^{-1}\tilde{\Phi}(\mathbf{X})^T \\ &\quad \times\left(\mathbf{I} - \tilde{\Phi}(\mathbf{Y})\tilde{\Phi}(\mathbf{Y})^T(\tilde{\Phi}(\mathbf{Y})\tilde{\Phi}(\mathbf{Y})^T + \lambda\mathbf{I})^{-1}\right)\tilde{\mathbf{y}} \\ \hat{\boldsymbol{\alpha}}_{krr} &= \tilde{\Phi}(\mathbf{Y})^T(\tilde{\Phi}(\mathbf{Y})\tilde{\Phi}(\mathbf{Y})^T + \gamma\mathbf{I})^{-1}(\tilde{\mathbf{y}} - \tilde{\Phi}(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr})\end{aligned}$$

where

$\tilde{\Phi}(\mathbf{X}) = \Phi(\mathbf{X}) - \frac{1}{l}\sum_{m=1}^{l}\Phi(\mathbf{X}_m)$ and $\tilde{\Phi}(\mathbf{Y}) = \Phi(\mathbf{Y}) - \frac{1}{l}\sum_{n=1}^{l}\Phi(\mathbf{Y}_n)$ are centered mapping of $\mathbf{X}$ and $\mathbf{Y}$ in the feature space, respectively. The centering of the mapped data points lead to the coefficients of the KRR model, which does not consider implicit intercept term *b* in the feature space. Instead, the following term *b* is added to the model explicitly.

$$\hat{b} = \frac{1}{l}\mathbf{1}^T\left(\mathbf{y} - \Phi\left(\mathbf{X}\right)\hat{\boldsymbol{\beta}}_{krr} - \Phi\left(\mathbf{Y}\right)\hat{\boldsymbol{\alpha}}_{krr}\right)$$

However, the centered nonlinear mapping $\tilde{\Phi}(\mathbf{X})$ does not have to be computed explicitly. The centering of the individual mapped data points may be achieved by using the kernel matrix $\mathbf{K} = \Phi(\mathbf{X})\Phi(\mathbf{X})^T$ (Scholkopf *et al*. 1998, Shawe and Cristianini 2004). It is noted that the centered nonlinear mapping has an advantage that we do not have to worry whether the selected kernel includes implicit intercept terms or not. For example, the polynomial kernel and RBF kernel include intercept terms implicitly (Abe 2005). However, even thought the selected kernel is poly kernel or RBF kernel, the centered nonlinear mappings of both $\mathbf{X}$ and $\mathbf{Y}$ make it possible not to consider implicit intercept terms *b.* The main advantage is that we can utilize the obtained coefficient estimators with the centered nonlinear mapping without any changes to consider the intercept term regardless of the kernel types.

The kernel matrix in the transformed space can be derived as

$$\begin{aligned}\tilde{\mathbf{K}}_{ij} = \tilde{\Phi}(\mathbf{X}_i)\tilde{\Phi}(\mathbf{X}_j)^T &= \left(\Phi(\mathbf{X}_i) - \frac{1}{l}\sum_{m=1}^{l}\Phi(\mathbf{X}_m)\right)\left(\Phi(\mathbf{X}_j) - \frac{1}{l}\sum_{n=1}^{l}\Phi(\mathbf{X}_n)\right)^T \\ &= \Phi(\mathbf{X}_i)\Phi(\mathbf{X}_j)^T - \frac{1}{l}\sum_{m=1}^{l}\Phi(\mathbf{X}_m)\Phi(\mathbf{X}_j)^T - \frac{1}{l}\sum_{n=1}^{l}\Phi(\mathbf{X}_i)\Phi(\mathbf{X}_n)^T + \frac{1}{l^2}\sum_{m,n=1}^{l}\Phi(\mathbf{X}_m)\Phi(\mathbf{X}_n)^T \\ &= \mathbf{K}_{ij} - \frac{1}{l}\sum_{m=1}^{l}\mathbf{1}_{im}\mathbf{K}_{mj} - \frac{1}{l}\sum_{n=1}^{l}\mathbf{K}_{in}\mathbf{1}_{nj} + \frac{1}{l^2}\sum_{m,n=1}^{l}\mathbf{1}_{im}\mathbf{K}_{mn}\mathbf{1}_{nj}\end{aligned}$$

The centered kernel matrix can be represented in matrix form as

$$\tilde{\mathbf{K}} = \mathbf{K} - \frac{1}{l}\mathbf{1}_l\mathbf{1}_l^T\mathbf{K} - \frac{1}{l}\mathbf{K}\mathbf{1}_l\mathbf{1}_l^T + \frac{1}{l^2}\mathbf{1}_l\mathbf{1}_l^T\mathbf{K}\mathbf{1}_l\mathbf{1}_l^T$$

where, $\mathbf{1}_l$ represent the vectors of one of the length l, and $\mathbf{I}$ is l dimensional identity matrix. Therefore, the centralization of the data leads to the modification of $\mathbf{K}_1$ and $\mathbf{K}_2$ matrices as follows.

$$\begin{aligned}\tilde{\mathbf{K}}_1 &\leftarrow \left(\mathbf{I} - \frac{1}{l}\mathbf{1}_l\mathbf{1}_l^T\right)\mathbf{K}_1\left(\mathbf{I} - \frac{1}{l}\mathbf{1}_l\mathbf{1}_l^T\right) \\ \tilde{\mathbf{K}}_2 &\leftarrow \left(\mathbf{I} - \frac{1}{l}\mathbf{1}_l\mathbf{1}_l^T\right)\mathbf{K}_2\left(\mathbf{I} - \frac{1}{l}\mathbf{1}_l\mathbf{1}_l^T\right)\end{aligned}$$

Therefore, the point prediction estimate of a new observation is replaced as

$$\begin{aligned}\hat{y} &= \tilde{\Phi}(\mathbf{X}_{new})\hat{\boldsymbol{\beta}}_{krr} + \tilde{\Phi}(\mathbf{Y}_{new})\hat{\boldsymbol{\alpha}}_{krr} + \hat{b}_{krr} \\ &= \tilde{\mathbf{K}}_{new1}((\mathbf{I} - \tilde{\mathbf{K}}_2(\tilde{\mathbf{K}}_2 + \gamma\mathbf{I})^{-1})\tilde{\mathbf{K}}_1 + \lambda\mathbf{I})^{-1}(\mathbf{I} - \tilde{\mathbf{K}}_2(\tilde{\mathbf{K}}_2 + \gamma\mathbf{I})^{-1})\tilde{\mathbf{y}} \\ &\quad + \tilde{\mathbf{K}}_{new2}(\tilde{\mathbf{K}}_2 + \gamma\mathbf{I})^{-1}(\tilde{\mathbf{y}} - \tilde{\Phi}(\mathbf{X})\hat{\boldsymbol{\beta}}_{krr}) + \hat{b}_{krr}\end{aligned}$$

where, $\tilde{\mathbf{K}}_{new1} = \tilde{\Phi}(\mathbf{X}_{new})\tilde{\Phi}(\mathbf{X})^T$, $\tilde{\mathbf{K}}_{new2} = \tilde{\Phi}(\mathbf{Y}_{new})\tilde{\Phi}(\mathbf{Y})^T$

$$\hat{b}_{krr} = \frac{1}{l}\mathbf{1}^T\left(\mathbf{y} - \Phi\left(\mathbf{X}\right)\hat{\boldsymbol{\beta}}_{krr} - \Phi\left(\mathbf{Y}\right)\hat{\boldsymbol{\alpha}}_{krr}\right).$$

5.4 Experimental Results

Computational experiments were conducted using three data sets in terms of prediction accuracy. The performance of our proposed RR and KRR models with LDVs has been compared with OLS, OLS with LDVs, RR, and KRR. In the experimental design, shrinkage parameters λ and γ for RR and KRR are optimized by a grid search. λ and γ are varied such that λ and $\gamma = 2^{-15}, 2^{-14}, \cdots, 2^{14}$ and 2^{15}, respectively. Also, Gaussian RBF kernel is used for kernel ridge regression. Gaussian RBF kernel is presented as follows.

$$K(u_1, u_2) = \exp\left(-\left\|u_1 - u_2\right\| / 2\tau^2\right)$$

where, τ is the width parameter that controls the amplitude of the RBF. τ is varied such that $\tau = 2^{-15}, 2^{-14}, \cdots, 2^{14}$ and 2^{15}.

The root mean square error (RMSE) is used to evaluate the prediction ability of each method. The data set is divided into three subsets: training, validation and testing data sets. After parameter tuning based on the validation data set, the RMSE is then calculated for testing data as follows.

$$\mathrm{RMSE}=\sqrt{\frac{\sum_{i=1}^{n_t}\left(y_{t,i}-\hat{y}_{t,i}\right)^2}{n_t}}$$

where, n_t is the number of testing samples, and $y_{t,i}$ and $\hat{y}_{t,i}$ are the actual and the predicted values, respectively. Since the time-order of observation should be maintained for considering the lagged variable, the cross-validation for all data sets cannot be used.

5.4.1 The Mackey-Glass time series prediction

The first data set used for the computational experiments is the Mackey-Glass Time Series (Mackey and Glass 1977). The data is generated by the following differential equation.

$$\frac{dx(t)}{dt}=-0.1x(t)+\frac{0.2x(t-\Delta)}{1+x(t-\Delta)^{10}}$$

where, Δ =17. We considered 600 training observations and next 200 for optimal parameter selection and last 200 for testing. This data set includes only one independent variable and one lagged dependent variable for methods considering LDVs. The RMSE values of each method are summarized in Table 5.1 with respect to the data set. From Table 5.1, we observe that the methods considering LDVs yield better results than the conventional regression methods without LDVs. Note that OLS with LDVs and RR with LDVs perform similarly in terms of prediction accuracy. That’s because we considered only one independent variable and one lagged dependent variable in this data set. For the reason, the shrinkage effect of RR is insignificant.

Table 5.1 Summary of the computational results for the simulated data set

Methods	Validation Error (RMSE)	Testing Error (RMSE)
OLS	0.234	0.243
RR	0.234	0.243
KRR	0.220	0.228
OLS_LDVs	0.038	0.038
RR_LDVs	0.038	0.038
KRR_LDVs	0.037	0.037

5.4.2 The internal bond strength prediction

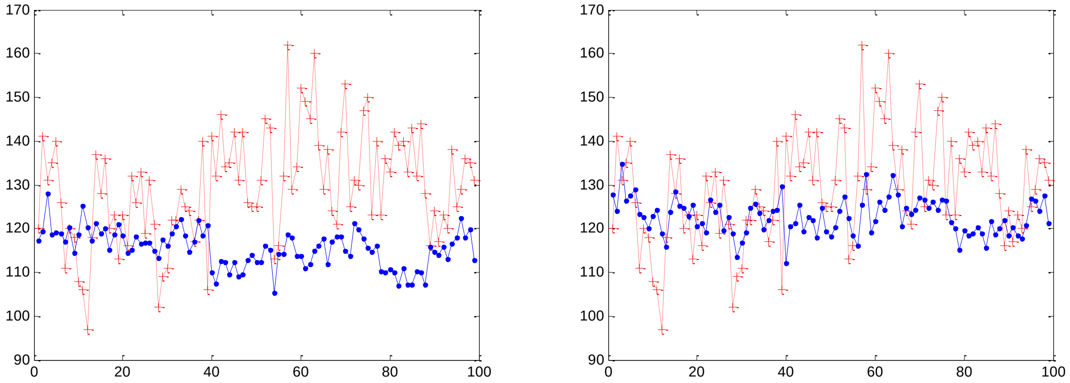
The second data set is Internal Bond (IB) strength which is real life data set. Accurate prediction of internal bond (IB) strength in a medium density fiberboard (MDF), a key product produced by the wood composites industry, is significant and challenging (Andre *et al*. 2008). The IB strength is an indicator of the cohesion of the panel in the direction perpendicular to the plane of the panel. A special measuring device is utilized that pulls the cross section apart and stresses the specimen until failure. The IB strength has known to be affected by fibers moisture contents, resin percentage, and press movement time. In addition, because IB strengths were collected by time-order records, it is possible to have autocorrelations between them. In this study, 495 time-order IB strength were obtained with 164 process variables. The dimension of each observation is 164 and the total number of observations is 495. Because original dimensionality is too high, we extracted

14 process variables by using variable selection technique. (see Andre *et al*. (2008) for detailed description of the data sets and variable selection procedure).

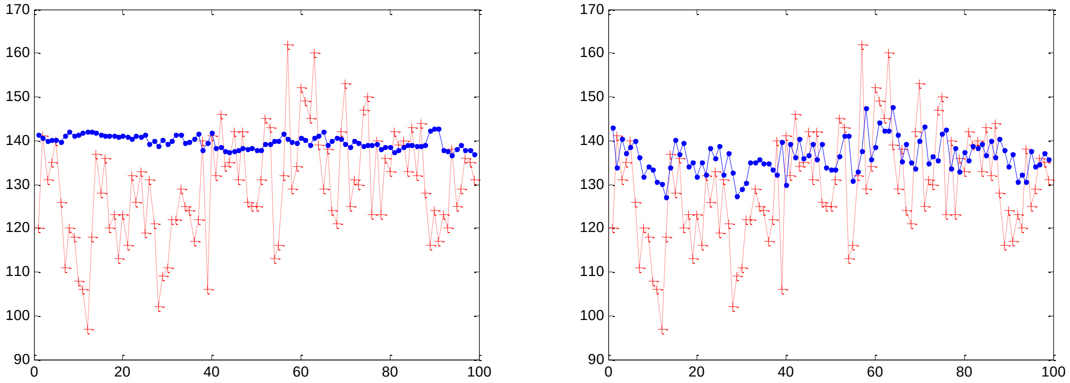
In this experiment, we use the LDVs model with autocorrelation among response variables with a specific lag 1. Table 5.2 shows computational results for each regression model. Also the fitted models are presented in Figure 5.1. This result demonstrates that the proposed RR and KRR models with LDVs produce smaller prediction errors than conventional regression methods even in this real-world application where the observations are often subject to noise or outliers. In addition, Table 5.2 shows that the RMSE for OLS with LDVs is higher than that for the other methods considering LDVs. The reason for this is believed to be due to the multicollinearity of independent variables. Note that the KRR model with LDVs yields the best performance. That's because the KRR model with LDVs may consider the nonlinearity of data and reflect autocorrelations of response variables simultaneously.

Table 5.2 Summary of the computational results for IB strength data set

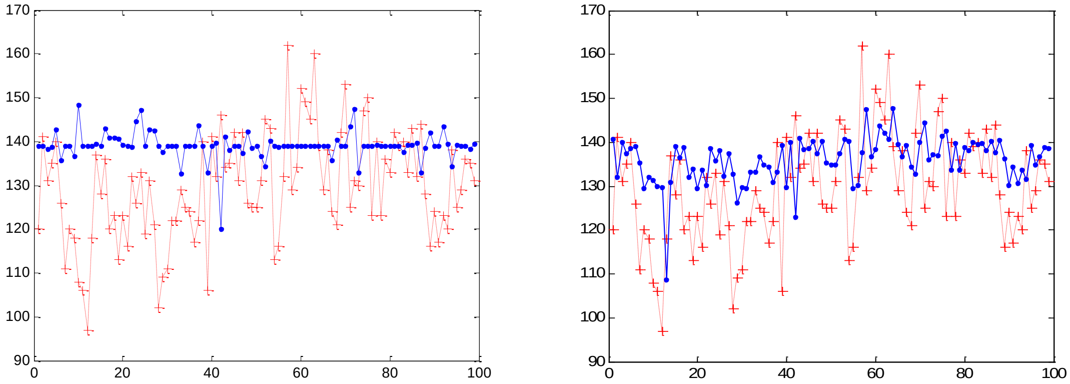
Methods	Validation Error (RMSE)	Testing Error (RMSE)
OLS	22.074	19.784
RR	15.066	16.349
KRR	14.058	16.267
OLS_LDVs	17.283	14.168
RR_LDVs	13.327	13.423
KRR_LDVs	12.666	13.260



(a) OLS (b) OLS with LDVs



(c) RR (d) RR with LDVs



(e) KRR (f) KRR with LDVs

Figure 5.1 Output from six different models for IB strength testing data set: The dashed line (the actual value) and the solid line (the predicted value)

5.4.3 The tourism demand prediction

The third data set obtained from the Korea National Tourism Corporation (KNTC) Annual Statistical Report represents the demand for tourism to Korea by the major tourism-generating country, USA, which was originally used in Song *et al*. (2009). Since the success of many tourism businesses depends on the state of the tourism demand, the accurate forecasts of expected future demand are essential for efficient planning for all tourism-related businesses (Song *et al*. 2009). Note that people may be likely to tell others about their favorable experiences related to the destination. Hence, the lagged dependent variable for the tourism demand may be considered as one of the independent variables (Garin-Munoz and Montero-Martin 2007). The total number of tourist arrivals by the country ranging from 1962 to 1994 is used as the dependent variable. The data set consists of 33 samples and was divided into the three sets. The data from 1962 to 1980 are used for training data and validation (from 1981 to 1987), respectively and testing samples (from 1988 to 1994) of 7 data points are used for the experiments. Also, the gross domestic product (GDP), the trade volume (TV) and the relative consumer price index (RCPI) are used as the independent variables (Song *et al*. 2009). The GDP of the tourism generating country is included to consider the travelers' income. The TV variable measured by the sum of total imports and exports between Korea and USA is also involved in the model. That's because TV may reflect the influence of business travelers on tourism demand.

In addition, RCPI is measured by the relative consumer price index (CPI) of Korea to that of USA considering the corresponding exchange rate (EX) as follows.

$$RCPI_{USA} = \frac{CPI_{Korea} / EX_{Korea/USA}}{CPI_{USA}}$$

RCPI as independent variable explains the effects of both relative inflation and the exchange rate on the demand for tourism to Korea. For this study, the values of all variables were transformed by natural logarithm before analysis.

Validation and testing errors in terms of RMSE for the tourism demand data to Korea by USA are summarized in Table 5.3. As same to other cases, the best performance in terms of testing error was obtained with the KRR model with LDVs. In addition, the models including the LDVs are superior to those excluding the LDVs. The results show that the word-of-mouth effect is an important factor for the decision of the travel destination.

Table 5.3 Computational results for the tourism demand to Korea by USA

Methods	Validation Error (RMSE)	Testing Error (RMSE)
OLS	0.228	0.475
RR	0.075	0.115
KRR	0.064	0.092
OLS_LDVs	0.050	0.160
RR_LDVs	0.032	0.099
KRR_LDVs	0.036	0.057

In summary, the experimental results suggest that the proposed approaches are promising alternatives to existing algorithms when there are autocorrelations between dependent variables.

5.5 Concluding Remarks

This paper proposed RR-based and KRR-based LDVs models, and compared them with existing regression methods in terms of RMSE using one simulated and two real-life data sets. Experimental results show that the proposed RR-based KRR-based models perform consistently better than conventional regression methods without LDVs regardless of the data set and more importantly, the proposed RR and KRR approach yields consistently better results than OLS with LDVs regardless of the data set. This is an encouraging result since the performance of proposed approach increase markedly by considering LDVs for autocorrelated dependent variables, and the performance of the proposed approaches do not deteriorate even for high dimensional data, and therefore, may be considered as a useful alternative when the dependent variables are autocorrelated and data are high-dimensional.

The proposed ideas can be expended to the advanced regression model such as relevance vector machine (RVM) for regression, which uses Bayesian theory to obtain sparse solutions.

CHAPTER 6

Concluding Remarks and Future Researches

6.1 Concluding Remarks

In this disseration, we have proposed and subsequently implemented several methodolgies for spatial and time series data mining. In Chapter 2, we proposed a methodology for detecting the presence of spatial autocorrelations and classifying spatial patterns using binary spatial data. As a specific application, the identification of spatial defect patterns on wafer maps occurring during the manufacturing of semiconductors was investigated. We also derive the generalized join-count (JC)-based statistic and then its optimal weights. The spatial randomness test was developed for the detection of spatial autocorrelation. By combining the spatial correlogram, which transforms binary spatial data into time sequence data, with the dynamic time warping algorithm, we have been able to construct a classification model for detecting spatial defect patterns on wafer maps. The experimental results show that the proposed algorithm is superior to existing methods, such as neural networks and nearest neighbor with Euclidean distance.

In Chapter 3, we proposed a novel distance measure, called weighed dynamic time warping (WDTW), for time series classification and clustering. We also explore a number of mathematical properties of the WDTW. Unlike standard DTW, WDTW does account for the relative importance of the phase distance between time series points. This property can lead to an accurate classification, especially in applications where the phase difference between two time series points plays a key role in the discrimination of classes. The rationale underlying the performance advantage of WDTW was investigated by illustrating a number of practical examples in which WDTW is clearly more effective than standard DTW. We also extended a weight concept to a variant of DTW and then proposed a weight-based derivative DTW (WDDTW). The extensive experimental results show that the proposed weighed-based DTW with optimal weights has a great potential for improving the accuracy of time series classification and clustering

In Chapter 4, we proposed a new statistical process control procedure for functional data to be used for considering these systematic variations of curves at certain local regions. To this end, we present a wavelet-based local-random-effect model to capture local variations of curves. To deal with the large number of parameters in both the mean and variance models, we developed an integrated mean and variance thresholding procedure (WMVT) to keep the model simple and also to fit the data curves well. Based on the WMVT procedure, we then developed process monitoring procedures for detecting process changes using the selected wavelet coefficients within the framework of the wavelet-based mixed effects model. The experimental results show that the proposed procedure performs better than several techniques extended from methods based on single curve-based data reduction.

Finally, in Chapter 5, we proposed a kernel-based regression model with lagged dependent variables (LDVs) that takes both the autocorrelations of the response variables and the nonlinearity of data into consideration. In addition, we derived the kernel ridge estimators with LDVs using a new mapping concept so that the nonlinear mapping does not have to be computed explicitly depending on kernel types. The experimental results show that the proposed kernel ridge regression-based models perform consistently better than conventional regression methods without LDVs regardless of the data set and, more importantly, that the proposed approach yields consistently better results than ordinary least squares model with LDVs regardless of the data set.

6.2 Future Researches

Future studies are needed that focus on improving and applying the approaches proposed here to other application domains. Unlike binary wafer maps, which are composed of a binary matrix independent of types of failure patterns, the wafer bin map (WBM) presents specific failure patterns in order to provide more details that can be used to track the process problems in the semiconductor manufacturing process. Even though WBM contains more useful information than the binary wafer map, little research has been done on the application of WBM for the analysis of spatial defect patterns on wafers. A meaningful line of research would be to extend spatial correlogram-based classification approaches into WBMs with the aim of identifying defect patterns on wafer maps.

In this dissertation, we have focused on the effectiveness of the proposed weighted DTW. As a means to improve efficiency, the weighted-based DTW algorithm could be combined with certain of the pruning techniques, such as LB_Keogh and warping-window-DTW, to reduce the computational time for much longer time series datasets.

Finally, the concept of lagged dependent variables can be expended to the advanced regression model, such as the relevance vector machine (RVM) for regression, which utilizes Bayesian theory to obtain sparse solutions.

REFERENCES

1. Abe, S. (2005) *Support vector machines for pattern classification*. Springer, London, UK
2. Agresti, A. (1990), *Categorical data analysis*, Wiley.
3. Alwan, L. and Roberts, H. (1998), "Time-series modeling for statistical process control," *Journal of Business and Economic Statistics*, 6, 87-97.
4. Andre, N., Cho, H.W., Baek, S.H., Jeong, M.K., and Young, T.M. (2008), "Prediction of internal bond strength in a medium density fiberboard process using multivariate statistical methods and variable selection," *Wood Science Technology*, 42, 521-534.
5. Anselin, L. (1988), *Spatial econometrics: methods and models*, Kluwer Academic Publisher, Netherlands.
6. Bailey, T. C. and Gatrell, A. C. (1995), *Interactive spatial data analysis*, Prentice Hall.
7. Beck, N. and Katz, J.N. (1995), "What to do (and not to do) with time–series cross-section data," *American Political Science Review*, 89, 634-647.
8. Bruce, A. G., and Gao, H.-Y. (1996), "Understanding waveshrink: Variance and bias estimation," *Biometrika*, 83, 727-745.
9. Castro, B. F., Guillas, S., and Manteiga, W. G. (2005), "Functional samples and bootstrap for predicting sulfur dioxide levels," *Technometrics*, 47, 212-222.
10. Chen, D., Lu, J.-C., X. Huo, and Ming, Y. (2001), "Robust estimation with estimating equations for nonlinear random coefficients model," *Journal of Statistical Planning and Inference*, 37, 275-292.
11. Chen, F. L. and Liu, S. F. (2000), "A neural-network approach to recognize defect spatial pattern in semiconductor fabrication," *IEEE Transactions on Semiconductor Manufacturing*, 13, 366-373.
12. Cliff, A. D. and Ord, J. K. (1981), *Spatial processes: Models and applications*, Pion.
13. Cunningham, S. P. and McKinnon, S. (1998), "Statistical methods for visual defect metrology," *IEEE Transactions on Semiconductor Manufacturing*, 11, 48-53.
14. DeNicolao, G., Pasquinetti, E., Miraglia, G., and Piccinini, F. (2003), "Unsupervised spatial pattern classification of electrical failures in semiconductor manufacturing," *Artificial Neural Networks Pattern Recognition Workshop*, 125-131, 2003.
15. Dietrich, C. D., Palm, G., Riede, K., and Schwenker, F. (2004), "Classification of bioacoustic time series based on the combination of global and local decision," *Pattern Recognition*, 37, 2293-2305.
16. Donoho, D. L. and Johnstone, I. M. (1994), "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, 81, 425-455.
17. Eads, D., Hill, D., Davis, S., Perkins, S., Ma, J., Porter, R., and Theiler, J. (2002), "Genetic algorithms and support vector machines for time series classification," *Proceeding SPIE 4787*, 74-85.

18. Fang, Y., Park, J. I., Jeong, Y. S., Jeontg, M. K., Baek, S. H., and Cho, H. W. (in press), "Enhanced predictions of wood properties using hybrid models of PCR and PCS with high-dimensional NIR spectral data," *Annals of Operations Research*.
19. Fenner, J. S., Jeong, M. K., and Lu, J. C. (2005), "Optimal automatic control of multistage production processes," *IEEE Transactions on Semiconductor Manufacturing*, 18, 94-103.
20. Ganesan, R., Das, T. K., Sikder, A. K. and Kumar, A. (2003), "Wavelet based identification of delamination emission signal," *IEEE Transactions on Semiconductor Manufacturing*, 16, 677-685.
21. Garin-Munoz, T. and Montero-Martin, L.F. (2007), "Tourism in the Balearic Islands: A dynamic model for international demand using panel data," *Tourism Management*, 28, 1224-1235.
22. Guler, I. and Ubeyli, E. D. (2005), "Adaptive neuro-fuzzy inference system for classification of EEG signals using wavelet coefficient," *Journal of Neuroscience Methods*, 148, 113-121.
23. Gullo, F., Ponti, G., Tagarelli, A., and Greco, S. (2009), "A time series representation model for accurate and fast similarity detection," *Pattern Recognition*, 42, 2998-3014.
24. Hansen, C. K. and Thyregod, P. (1998), "Use of wafer maps in integrated circuit manufacturing," *Microelectronics Reliability*, 38, 1155-1164.
25. Hansen, M. H., Nair, V. N., and Friedman, D. J. (1997), "Monitoring wafer map data from integrated circuit fabrication processes for spatially clustered defects," *Technometrics*, 39, 241-253.
26. Hastie, T., Tibshirani, R., and Friedman, J. (2001), *The elements of statistical learning*, Springer, New York, NY.
27. Hsieh, H. W. and Chen, F. L. (2004), "Recognition of defect spatial patterns in semiconductor fabrication," *International Journal of Production Research*, 42, 4153-4172.
28. Hsu, S. C. and Chen, F. L. (2007), "Hybrid data mining approach for pattern extraction from wafer bin map to improve yield in semiconductor manufacturing," *International Journal of Production Economics*, 107, 88-103.
29. Husken, M. and Stagge, P. (2003), "Recurrent neural networks for time series classification," *Neurocomputing*, 50, 223-235.
30. Huang, C. J. (2007), "Clustered defect detection of high quality chips using self-supervised multilayer perceptron," *Expert System with Applications*, 33, 996-1003.
31. Huwang, L., Yeh, A.B., and Wu, C.W. (2007), "Monitoring multivariate process variability for individual observations," *Journal of Quality Technology*, 39, 258-278.
32. Itakura, F (1975), "Minimum prediction residual principle applied to speech recognition," *Proceedings of IEEE Transactions Acoustics, Speech, and Signal*, 52-72.
33. Jalba, A. C., Wilkinson, M., Roerdink, J., Bayer, M. M. and Juggins, S. (2005), "Automatic diatom identification using contour analysis by morphological curvature scale spaces," *Machine Vision and Applications*, 16, 217-228.

34. Jeong, M. K., Lu, J. C., Huo, X., Vidakovic, B. and Chen, D. (2006a), "Wavelet-based data reduction techniques for fault detection," *Technometrics*, 48, 26-40.
35. Jeong, M. K., Lu, J. C., and N. Wang (2006b), "Wavelet-based SPC procedure for complicated functional data," *International Journal of Production Research*, 44, 1-16.
36. Jeong, M. K., Lu, J. C., and N. Zhou, and Ghosh, S. K. (2006c), "Data-reduction method for spatial data using a structured wavelet model," *International Journal of Production Research*, 45, 2295-2311.
37. Jeong, Y. S., Kim, S. J., and Jeong, M. K. (2008), "Automatic identification of defect patterns in semiconductor wafer maps using spatial correlogram and dynamic time warping," *IEEE Transactions on Semiconductor manufacturing*, 21, 625-637.
38. Jeong, Y. S., Jeong, M. K., and O. "Omitaomu (in press), "Weighted dynamic time warping using time series classification," *Pattern Recognition*.
39. Jeong, Y. S. and Jeong, M. K. (2009), "Classification of spatial defect patterns using weighed dynamic time warping," Technical report, Department of Industrial and Systems Engineering, Rutgers University.
40. Jeong, Y. S., Castro-Neto, M., Jeong. M. K., and Han, L. (2011), "A wavelet-based freeway incident detection algorithm with adapting threshold parameters," *Transportation Research Part C*, 19, 1-19.
41. Jin, J. and Shi, J. (1999), "Feature-preserving data compression of stamping tonnage information using wavelets," *Technometrics*, 41, 327-339.
42. Jin, J. and Shi, J. (2001), "Automatic feature extraction of waveform signals for in process diagnostic performance improvement," *Journal of Intelligent Manufacturing*, 12, 257-268.
43. Jun, C. H., Hong, Y., Kim, S. Y., Park, K. S., and Park, H. (1999), "A simulation based semiconductor chip yield model incorporating a new defect cluster index," *Microelectronics Reliability*, 39, 451-456.
44. Jung, U., Jeong, M. K., and Lu, J. C. (2006), "A vertical energy thresholding procedure for data reduction with multiple complex curves," *IEEE Transactions on Systems, Man, Cybernetics, Part B*, 36, 1128-1138.
45. Kang, L. and Albin, S. L. (2000), "On-line monitoring when the process yields a linear profile," *Journal of Quality Technology*, 32, 418-426.
46. Keele, L. and Kelly, N. (2005), "Dynamic models for dynamic theories: The ins and outs of lagged dependent variables," *Political Analysis*, 14, 186-205.
47. Keogh, E. and Pazzani, M. (2001), "Derivative dynamic time warping," *SIAM International Conference on Data Mining*, Chicago.
48. Keogh, E., Xi, X., Wei, L., and Ratanamahatana, C.A. (2006), "*The UCR Time Series Data Mining Archive*," Available at: http://www.cs.ucr.edu/~eamonn/time_series_data.
49. Keogh, E. and Ratanamahatana, C. A. (2005), "Exact indexing of dynamic time warping," *Knowledge and Information Systems*, 3, 358-386.
50. Keogh, E and Lin, J. (2005), "Clustering of time series subsequences is meaningless: Implications for previous and future research," *Knowledge and Information Systems*, 8, 154-177.

51. Lada, E. K., Lu, J. C. and Willson, J. R. (2002), “A wavelet- based procedure for process fault detection,” *IEEE Transactions on Semiconductor Manufacturing*, 15, 79-90.
52. Lee, D. J., Schoenberger, R., Shiozawa, D., Xu, X., and Zhan, P. (2004), “Contour matching for a fish recognition and migration monitoring system,” *SPIE Optics East, Two and Three-Dimensional Vision Systems for Inspection, Control, and Metrology II,* 5606-05, 37-48, Philadelphia, PA.
53. Lemire, D. (2009), “Faster retrieval with a two-pass dynamic-time-warping lower bound,” *Pattern Recognition*, 42, 2169-2180.
54. Liu, S. F., Chen, F. L., and Lu, W. B. (2002), “Wafer bin map recognition using a neural network approach,” *International Journal of Production Research*, 40, 2207-2223.
55. Lu, Y., Ouyang, Y., Sheng, H., and Xiong, Z. (2008), “An incremental algorithm for clustering search results,” *IEEE International Conference on Signal Image Technology and Internet Based Systems*.
56. Montgomery, D. C. (2005), *Introduction to statistical quality control*, Wiley.
57. Mallat, S. G. (1998), *A wavelet tour of signal processing*, Academic Press, San Diago.
58. Macgregor, J.F. and Harris, T.J. (1993), “The exponentially weighted moving variance,” *Journal of Quality Technology*, 25, 106-118.
59. Mackey, M. C. and Glass, L. (1977), “Oscillation and chaos in physiological control systems,” *Science*, 197, 287-289.
60. Morris, J. S., Vannucci, M., Brown, P. J., Carroll, R. J. (2003), “Wavelet-based nonparametric modeling of hierarchical functions in colon carcinogenesis,” *Journal of the American Statistical Association*, 98, 573-597.
61. Morse, M. D. and Patel, J. M. (2006), “An efficient and accurate method for evaluating time series similarity,” *Proceedings of the ACM SIGMOD International on Information and Knowledge Management*, 14-23.
62. Muller, K. R., Mika, S., Ratsch, G., Tsuda, K., Scholkopf, B. (2001), “An introduction to kernel-based learning algorithms,” *IEEE Transactions on Neural Networks*, 12, 2, 181-201.
63. Nieeattrakul, V and Ratanamahatana, C. (2007), “On clustering multimedia time series data using K-means and dynamic time warping,” *IEEE International Conference on Multimedia and Ubiquitous Engineering*.
64. Palma, F. D., Nicolao, G. D., Miraglia, Pasquinetti, G., E., and Piccinini, F. (2005), “Unsupervised spatial pattern classification of electrical-wafer-sorting maps in semiconductor manufacturing,” *Pattern Recognition Letter*, 26, 1857-1865.
65. Pierre, L. and Louis, L. (1998), *Numerical Ecology*, Elsevier.
66. Ramirez, J. and Taam, W. (2000), “An autologistic model for integrated circuit manufacturing,” *Journal of Quality Technology*, 32(3), 254-262.
67. Ratanamahatana, C. A. and Keogh, E. (2004a), “Making time-series classification more accurate using learned constraints,” in *Proc. of the 4th SLAM Int. Conf. on data mining*, April.

68. Ratanamahatana, C. A. and Keogh, E. (2004b), "Everything you know about dynamic time warping is wrong," in *Proc. 10th ACM SIGKDD Int. Conf. on knowledge discovery and data mining*, August.
69. Rath, T. M. and Manmatha, R. (2003), "Word image matching using dynamic time warping," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.
70. Reynolds, M and Cho, G.Y. (2006), "Multivariate control charts for monitoring the mean vector and covariance matrix," *Journal of Quality Technology*, 38, 230-253.
71. Ruiz, A. and Lopez-de-Teruel, P. E. (2001), "Nonlinear kernel-based statistical pattern analysis," *IEEE Transactions on Neural Networks*, 12, 16-32.
72. Sakoe, H. and Chiba, S. (1978), "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Process*, 43-49.
73. Saunders, C., Gammerman, C. A., and Vovk, V. (1998), "Ridge regression learning algorithm in dual variables," *Proceeding of the 15th International Conference on Machine Learning*, ICML.
74. Scholkopf, B., Smola, A., and Müller, K. R. (1998), "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, 10, 1299-1319.
75. Shawe-Taylor, J and Cristianini, N. (2004), *Kernel methods for pattern analysis*, Cambridge University Press, Cambridge, UK.
76. Song, H., Witt, S. F., and Li, G. (2009), *The advanced econometrics of tourism demand. Routledge*, New York, NY.
77. Stone, M. (1974), "Cross-validatory choice and assessment of statistical predictions," *Journal of the Royal Statistical Society, Series B*, 36, 111-147.
78. Taam, W. and Hamada, M. (1993), "Detecting spatial effects from factorial experiment: an application from IC manufacturing," *Technometrics*, 35, 149-160.
79. Tanizaki, H. (2000), "Bias correction of OLSE in the regression model with lagged dependent variables," *Computational Statistics and Data Analysis*, 34, 495-511.
80. Temiyasathit, C., Kim, S. B., and Park, S.-K. (2009), "Spatial prediction of ozone concentration profiles," *Computational Statistics and Data Analysis*, 53, 3892-3906.
81. Thies, C. G. and Porche, S. (2007), "The political economy of agricultural protection," *Journal of Politics*, 69, 116-127.
82. Tong, L. I., Wang, C. H., and Huang, C. L. (2005), "Monitoring defects in IC fabrication using Hotelling T^2 control chart," *IEEE Transactions on Semiconductor Manufacturing*, 18, 140-147.
83. Ubeyli, E. D. (2008), "Wavelet/mixture of experts network structure of ECG signals classification," *Expert Systems with Applications*, 34, 1954-1962.
84. Vlachos, M., Kollios, G., and Gunopulos, D. (2002), "Discovering similar multidimensional trajectories," *Proceeding of the International Conference Data Engineering*.
85. Xi, X., Keogh, E., Wei, L., and Ratanamahatana, C. A. (2006), "Fast time series classification using numerosity reduction," *Proceedings of the 23rd International Conference on Machine Learning*, Pittsburgh, PA.

86. Xie, Y. and Wiltgen, B. (2010), “Adaptive feature based dynamic time warping,” *International Journal of Computer Science and Network Security*, 10, 264-273.
87. Yu, F., Dong, K., Chen, F., Jiang, Y. and Zeng, W. (2007), “Clustering time series with granular dynamic time warping method,” *IEEE International Conference on Granular Computing*.
88. Xi, X., Keogh, E., Wei, L., and Ratanamahatana, C. A. (2006), “Fast time series classification using numerosity reduction,” *Proceedings of the 23rd International Conference on Machine Learning*, Pittsburgh, PA.
89. Xie, Y. and Wiltgen, B. (2010), “Adaptive feature based dynamic time warping,” *International Journal of Computer Science and Network Security*, 10, 264-273.
90. Zhao, F. and Park, N. (2004), “Using geographically weighted regression models to estimate annual average daily traffic,” *Journal of the Transportation Research Board*, 1897, 99-107.
91. Zhao, W., Serpedin, E., and Dougherty, E.R. (2010), “Spectral preprocessing for clustering time-series gene expressions,” *EURASIP Journal on Bioinformatics and Systems Biology*, 1-10.
92. Zhou, S., Sun, B., and Shi, J. (2006), “An SPC monitoring system for cycle-based waveform signals using Haar transform,” *IEEE Transactions on Automation Science and Engineering*, 3, 60-72.
93. Zou, C., Tsung, F., and Wang, Z. (2007) "Monitoring general linear profiles using multivariate EWMA schemes," *Technometrics*, 49, 395-408.

CURRICULUM VITAE

YOUNG-SEON JEONG

2011 Ph.D., Industrial and Systems Engineering, Rutgers, The State University of New Jersey, New Brunswick, NJ, USA.

2001 M.S., Industrial and Information Engineering, Korea University, Seoul, South Korea.

1997 B.S., Industrial Engineering, Chonnam National University, Gwangju, South Korea.

Publications

2011 Y. S. Jeong, M. K. Jeong, and O. A. Omitaomu, in press, "Weighted dynamic time warping for time series classification," *Pattern Recognition*.

2011 Zheng Du, Y. S. Jeong, M. K. Jeong, and S. G. Kong, in press, "Multidimensional local spatial autocorrelation measure for integrating spatial and spectral Information in hyperspectral image band selection," *Applied Intelligence*.

2011 Y. S. Jeong, M. Castro-Neto, M. K. Jeong, and Lee D. Han, 2011, "Wavelet based incident detection algorithm with adapting threshold parameters," *Transportation Research Part C*, 19, 1, 1-19.

2011 Y. Fang, J. I. Park, Y. S. Jeong, M. K. Jeong, S. H. Baek, H. W. Cho, in press, “Enhanced predictions of wood properties using hybrid models of PCR and PLS with high-dimensional NIR spectral data,” *Annals of Operation Research*.

2010 S. H. Choi, Y. S. Jeong, and M. K. Jeong, 2010, "A hybrid recommendation method reduced data for large-scale application," *IEEE Transactions on Systems, Man, and Cybernetics-Part C*, 40, 5, 557-566.

2010 Y. D. Ko, Y. S. Jeong, M. K. Jeong, A. Garcia-Diaz, and B. W. Kim, 2010, "Functional kernel based modeling of wavelet compressed optical emission spectral data: Prediction of plasma etch process," *IEEE Sensors Journal*, 10, 3, 746-754.

2009 J. S. Fenner, Y. S. Jeong, M. K. Jeong, and J.-C. Lu, 2009, "A Bayesian parallel site methodology with an application to uniformity modeling in semiconductor manufacturing," *IIE Transactions on Quality and Reliability*, 41, 9, 754-763.

2009 M. Castro-Neto, Y. S. Jeong, M. K. Jeong, and Lee D. Han, 2009, " Annual average daily traffic (AADT) prediction using support vector regression with data-dependent parameters," *Expert Systems with Applications*, 36, 2979-2986.

2009 M. Castro-Neto, Y. S. Jeong, M. K. Jeong, and Lee D. Han, 2009, "Online-SVR for short term traffic flow prediction under typical and atypical traffic conditions," *Expert Systems with Applications*, 36, 6164-6173.

2008 Y. S. Jeong, S. J. Kim, and M. K. Jeong, 2008, "Automatic identification of defect patterns in semiconductor wafer maps using spatial correlogram and dynamic time warping," *IEEE Transactions on Semiconductor Manufacturing*, 21, 4, 625-637