Functional Singular Value Decomposition and Multi-Resolution Anomaly Detection

by
Lingsong Zhang

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistics and Operations Research.

Chapel Hill
2007

Approved by:

J. S. Marron, Advisor

Zhengyuan Zhu, Advisor

Haipeng Shen, Advisor

Yufeng Liu, Committee Member

Kevin Jeffay, Committee Member

R. L. Smith, Committee Member

ABSTRACT

LINGSONG ZHANG: Functional Singular Value Decomposition and Multi-Resolution Anomaly Detection.
(Under the direction of J. S. Marron, Zhengyuan Zhu and Haipeng Shen.)

This dissertation has two major parts. The first part discusses the connections and differences between the statistical tool of Principal Component Analysis (PCA) and the related numerical method of Singular Value Decomposition (SVD), and related visualization methods. The second part proposes a Multi-Resolution Anomaly Detection (MRAD) method for time series with long range dependence (LRD).

PCA is a popular method in multivariate analysis and in Functional Data Analysis (FDA). Compared to PCA, SVD is more general, because it not only provides a direct approach to calculate the principal components (PCs), but also simultaneously yields the PCAs for both the row and the column spaces. SVD has been used directly to explore and analyze data sets, and has been shown to be an insightful analysis tool in many fields. However, the connection and differences between PCA and SVD have seldom been explored from a statistical view point. Here we explore the connections and differences between PCA and SVD, and extend the usual SVD method to variations including different centerings based on various types of means. A generalized scree plot is developed to provide a visual aid for selection of different centerings. Several matrix views of the SVD components are introduced to explore different features in data, including SVD surface plots, image plots, rotation movies, and curve movies. These methods visualize both column and row information of a two-way matrix simultaneously, relate the matrix to relevant curves, and show local variations and interactions between columns and rows. Several toy examples are designed to compare the different types of centerings, and three real applications are used to illustrate the matrix views.

In the field of Internet traffic anomaly detection, different types of network anomalies exist at different time scales. This motivates anomaly detection methods that effectively exploit multiscale properties. Because time series of Internet measurements exhibit long range dependence (LRD) and self-similarity (SS), the classical outlier detection methods base on short-range dependent time series may not be suitable for identifying network anomalies. Based on a time series collected at a single scale (the finest scale), we aggregate to form time series of various scales, and propose a MRAD procedure to find anomalies which appear at different time scales. We show that this MRAD method is more conservative than a typical outlier detection method based on a given scale, and has larger power on average than any single scale outlier detection method based on some reasonable assumptions. Asymptotic distribution of the test statistic is developed as well. An MRAD map is developed to show candidate anomalies and the corresponding significance probabilities (p values). This method can be easily extended to be implemented in real time. Simulations and real examples are reported as well, to illustrate the usefulness of the MRAD method.

Keywords: Principal Component Analysis, Functional Data Analysis, Exploratory Data Analysis, Network Intrusion Detection, Outlier detection, Level Shift, Multiscale analysis, Long Range Dependence, Multiple Comparison, p values, Time Series, false discovery rate.

ACKNOWLEDGEMENTS

I am deeply indebted to my advisors, Professor Marron, Professor Zhu and Professor Shen. They led me into the research field of statistical analysis of the network traffic data sets. They helped me in learning the PCA method, the exploration of the relations between PCA and SVD, and comparing the four types of centerings. From them, I improved myself a lot in formulating scientific problems, in developing novel methods to solve those problems, etc.

In the two courses taught by Professor Marron, *Data Statistical Analysis and Modelling of Internet Traffic Data* and *Object Oriented Data Analysis*, I started to know the new field of Functional Data Analysis, Objected Oriented Data Analysis and related statistical methods in network data analysis. After taking these courses, I related my research into the above fields, and made some contributions as this thesis will show. In developing the multiresolution anomaly detection method, Professor Zhu gave me a lot of help in formulating the theoretical properties, and in organizing the results, etc.

Thanks also go to Professor Kevin Jeffay and Professor Don Smith. They showed me the research area of network anomalies detection. Especially from the course, *Network Intrusion Detection* taught by Professor Jeffay, I learned a lot of background knowledge of network intrusions. I am grateful to the Internet Data Study Group at University of North Carolina. From the group meetings, I understood more on network research backgrounds, and gained some research insights. In addition, they, especially Jeff Terrell, helped me to collect Internet data sets for evaluation of the MRAD method.

For the chapter 2, the author gratefully acknowledges Dr. Andrés M. Alonso Fernández of Departamento de Estadística in Universidad Carlos III de Madrid, for providing the Spanish mortality data to us. Thanks also go to the reviewers of *Journal of Computational and Graphical Statistics* for their helpful comments and constructive suggestions related to chapter 2. For the chapter 3, the author thanks Professor Richard Smith, for his discussion and comments on the development of MRAD method and the asymptotic threshold.

I would like to give my special thanks to my girlfriend, Miss. Xuanyao He, for her strong support of my research and my life for the last three years.

Contents

ACKNOWLEDGEMENTS **v**

List of Figures **xi**

List of Tables **xiv**

1 Introduction **1**

1.1 PCA, SVD, and visualizations 1

1.2 Multi-Resolution Anomaly Detection 5

1.3 Background on network analysis and intrusion detection 7

1.3.1 Network data analysis 8

1.3.2 Network intrusion detection method 10

2 SVD, PCA and Visualizations **14**

2.1 Introduction 15

2.2 Motivating example 18

2.3 SVD and PCA 23

2.3.1 SVD and its properties 23

2.3.2 Four types of centerings for SVD 24

2.3.3 Geometric interpretations of the means and centerings 32

2.3.4 Model selection and a Generalized scree plot 40

2.4 Generation of the surface plots and the SVD movies 43

2.5 Four types of SVD and toy examples 45

2.5.1 Example 1 46

2.5.2 Example 2 49

2.5.3 Example 3 52

2.5.4 Example 4 55

2.6 Real applications 57

2.6.1 Further analysis of the Internet traffic data set 57

2.6.2 A Chemometrics data set 60

2.6.3 A Demographical data set 63

2.7 Discussion 68

3 Anomaly Detection for Time Series with Long Range Dependence 70

3.1 Introduction 70

3.2 Motivating example and the MRAD map 73

3.3 Two types of aggregation 76

3.3.1 Non-Overlapping Window Aggregation 77

3.3.2 Sliding Window Aggregation 79

3.3.3 MRAD outlier map based on SWA 83

3.4 Theoretical properties for the MRAD method 86

3.4.1 Background on LRD and fractional Gaussian noise 86

3.4.2 Background on extreme value theory of stationary processes 87

3.4.3 The testing problem 88

3.4.4 Simple MRAD procedures . . . 89

3.4.5 Theoretical properties of a 2-scale MRAD procedure . . . 90

3.5 An improved test threshold . . . 95

3.6 Simulations and applications in network intrusion detection . . . 99

3.6.1 Fractional Gaussian noise and local mean level shift . . . 99

3.6.2 Intensity of the mean level shift . . . 104

3.6.3 Comparisons between the MRAD method and the intervention analysis . . . 107

3.6.4 White noise with a local mean level shift and the scale movie . . . 111

3.6.5 Semi-experiments . . . 113

3.6.6 More analysis of the motivating example, zoomed maps and movies . . . 117

3.7 Conclusion and discussion . . . 119

3.8 Theoretical justifications . . . 120

3.8.1 Background on the bivariate Normal distribution . . . 120

3.8.2 Power at one single scale . . . 122

3.8.3 Proof for Proposition 3.4.1 . . . 123

3.8.4 Proof for Proposition 3.4.2 . . . 123

3.8.5 Proof for Proposition 3.4.3 . . . 125

3.8.6 Power for a two-scale MRAD procedure . . . 127

3.8.7 Proof for Theorem 3.4.1 . . . 128

3.8.8 Proof for Theorem 3.4.2 . . . 132

4 Summary and Comments 133

4.1 Functional SVD . . . 133

4.2 The MRAD method . . . 134

4.3 Future work 136

Bibliography **138**

List of Figures

2.1 A time series plot and a mesh plot for the UNC network packet-count data 19

2.2 SSVD surface plots for the UNC network traffic data 21

2.3 One snapshot of the SSVD curve movie for the UNC network data 22

2.4 Approximation and complexity diagram for PCA and SVD 27

2.5 Geometric diagram of different means and centerings 34

2.6 Generalized scree plot for the UNC network data 42

2.7 Two snapshots of the second SSVD component for the network traffic data set . . . 44

2.8 Log-Scale generalized scree plot for the first toy example 47

2.9 The surface plots of the SSVD for the first toy example 48

2.10 The surface plots of the DSVD for the first toy example 48

2.11 Log-scale generalized scree plot for the second toy example 50

2.12 CSVD and SSVD surface plots for the second toy example 51

2.13 Log-scale generalized scree plot for the third toy example 53

2.14 Surface plots of SSVD and RSVD for the third toy example 54

2.15 Generalized scree plot for the fourth toy example 56

2.16 Surface plot of RSVD for the fourth toy example 57

2.17 Snapshots of the SSVD rotation movie for the UNC network data 58

2.18 Scatter plot between the first two singular columns for the UNC network data 59

2.19 Mesh plot for the Chemometrics data 60

2.20 Log-Scale generalized scree plot for the Chemometrics data set 61

2.21 Mesh plot and zoomed mesh plot for CSVD of the Chemometrics data 62

2.22 Zoomed version of the surface plots for the Chemometrics data 63

2.23 Mesh plot for the mortality data 64

2.24 Generalized scree plot for the mortality data 65

2.25 Surface plot of the SSVD for the Spanish mortality data set 66

2.26 Image plot of the SSVD for the Spanish mortality data 67

3.1 Motivating example for MRAD method, the byte-count time series 74

3.2 MRAD outlier map based on NOWA for the byte-count data 75

3.3 MRAD thresholded outlier map based on NOWA for the byte-count data 76

3.4 Non-Overlapping Window Aggregation 77

3.5 Sliding Window Aggregation: to form scale 2 time series 80

3.6 Sliding Window Aggregation: to form scale 3 time series 80

3.7 Sliding Window Aggregation: to form scale 4 time series 81

3.8 The MRAD map based on SWA for the network byte count time series 84

3.9 Comparisons of the simplified Bonferroni asymptotic threshold and the independent asymptotic threshold 95

3.10 The improved test threshold vs. Hurst parameter 97

3.11 The improved threshold vs. the number of scales 98

3.12 MRAD outlier map for one simulated trace 103

3.13 FDR and FNR boxplots under different intensities 105

3.14 The box plot for the TDRs under different intensities 106

3.15 True discovery rate comparisons between the MRAD method and other methods . . 109

3.16 FDR and FNR comparisons between the MRAD method and other methods 110

3.17 Time series plot for the white noise series plus mean level shift 111

3.18 Snapshots of the MRAD outlier movie for the contaminated white noise series 112

3.19 MRAD outlier map for packet series of the semi-experiment example 115

3.20 MRAD outlier map for flow series of the semi-experiment example 116

3.21 Snapshots of the zoomed MRAD outlier movie for the motivating example 118

List of Tables

3.1 Improved test thresholds for the MRAD method 96

3.2 The classical FDR definition table. 101

3.3 Evaluation of MRAD based on simulation 102

3.4 More simulation evaluations 103

Chapter 1

Introduction

Two major parts are included in this dissertation. The first part, Chapter 2, discusses the connections and differences between Singular Value Decomposition (SVD) and Principal Component Analysis (PCA), especially from a Functional Data Analysis (FDA) view point. Several new matrix views for the SVD components are proposed to find underlying features of data sets. The second part, Chapter 3, proposes a Multi-Resolution Anomaly Detection (MRAD) method for time series with long range dependence (LRD), along with its theoretical and empirical properties. In the next two sections, a brief summary of the above two parts is given. In addition, Section 1.3 provides some background on network data analysis and network intrusion detection.

1.1 PCA, SVD, and visualizations

Traditional univariate statistical methods treat *numbers* as the objects of interest. Multivariate analysis expands the objects to *vectors*. Functional Data Analysis (FDA) deals with curves as data (e.g., Marron et al. (2004); Ramsay and Silverman (2002, 2005)). A new statistical framework, Object Oriented Data Analysis (OODA) allows analysis of more complicated objects, such as images (Locantore et al., 1999), trees (Wang and Marron, 2006) and so on.

Principal Component Analysis (PCA) is known to be a very useful functional data analysis method (for an accessible introduction, see Jolliffe (2002); Ramsay and Silverman (2002, 2005)). Compared to PCA, Singular Value Decomposition (SVD) is more general, because SVD not only can be used to calculate PCA, but also simultaneously provides PCAs for both the case where the data vectors are the rows of the data matrix and the case where they are the columns. In addition, the SVD method has also been applied directly to analyze data sets in many fields, for example, for Gene Microarray data (Kluger et al., 2003), for image data analysis (Muller et al., 2004), etc.

The major contrast between PCA and SVD is that the usual PCA method is a factorization of the sample covariance matrix, while the SVD method is a factorization of the data matrix. Although PCA and SVD are closely related, there is very little literature that explores the connections and differences between them (but see Okamoto (1972) for an exception) from a statistical viewpoint. In this dissertation, we explore the relations between SVD and PCA from a FDA view point. This motivates extending the usual SVD to include four types of centerings (ref. Section 2.3.2). Several matrix views of the SVD components are proposed to provide new insights of the data set (ref. Section 2.2, 2.5, 2.6).

Let X be a data matrix, where its *column mean* (ref. section 2.3.2 for details) is 0. The usual PCA can be viewed as an eigen-analysis of the sample (i.e. empirical) covariance matrix, which is proportional to X^TX in this setting. Rows of X can be viewed as observed data vectors in the usual statistical literature (curves in FDA, or objects from the OODA viewpoint). Correspondingly, columns of X are covariates (or feature vectors in FDA or OODA). For the above column centered data set X, the corresponding singular value decomposition, $X = USV^T$ (ref. 2.3.1 for details), is called *column SVD* (CSVD). The resulting *singular rows*, column vectors of V, are the same as the PC directions of the usual PCA in the column space (i.e., covariate space). Thus it follows that

CSVD can be used to calculate the usual PCs.

Similarly we can define *row SVD* (RSVD) as the singular value decomposition of X after removing the *row mean.* The corresponding *singular columns* (the column vectors of U) are the PC directions of the PCA in the row space (i.e., the observation space). We could also define the *double SVD* (DSVD) as the singular value decomposition of X after removing the *double mean*, which makes both the column mean and row mean of X zero. In summary, for any data matrix X, we can provide four types of factorization based on different centerings: simple SVD (i.e., without any centering), column SVD, row SVD and double SVD. Details of the four types of SVD, and how they compare, are discussed in Section 2.3.2.

There are several natural questions for the four types of centerings. Which centering should be used in which application? In order to select the so-called optimal centering, we provided several criteria for model selection (i.e., to select the optimal decomposition of the data matrix among the four types). The optimal model should contain few components, have small residual and good interpretation. These three criteria, model complexity, approximation performance and interpretability, should be considered simultaneously.

Suppose the data matrix is viewed as the sum of an approximation matrix and the residual matrix, while the rank of the approximation matrix is the same. The row SVD and the column SVD provide simpler models (i.e., fewer components) than the simple SVD and the double SVD (this is called *model complexity* in Section 2.3.4). In addition, the simple SVD provides the best rank k approximation among the four types, thus, it is the optimal model in terms of smallest residual (this criterion is called *approximation performance* in Section 2.3.4). There is no general answer as to which is the most interpretable. In order to balance the goals of finding the simplest, with smallest residual, and the most interpretable model, it is useful to apply all four types of factorization to a data matrix together, and then compare them based on the context. We might need to consider all these criteria simultaneously, but with different weights on the criteria for different purposes. From a statistical view point, different factorization methods can provide different types of insights. There will not be a unique best interpretable decomposition. Several examples are given in Section 2.5 to illustrate that each of the four types of decomposition can be the optimal one.

In order to select the best model among the four types of decomposition, we generalize the usual *scree* plot (Cattell, 1966) (section 6.1.3 in Jolliffe (2002) provides a good introduction) to include all four types of centerings. The generalized scree plot visualizes the model complexity and model approximation in the same graph, and shows the tradeoff between them. It provides a good starting point for model selection. But for interpretability, it is recommended to explore the data using all four types of decomposition. Visualization methods, including our proposed matrix views of SVD (ref. Section 2.2, 2.5, 2.6), will provide some aids for selecting the most interpretable one.

The new proposed set of visualization methods, the matrix views of SVD, includes the SVD surface plots, the image plots, the curve movie, and the rotation movie, can reveal additional insights of the PCs or the SVD components. These methods simultaneously show the row and column curves, and reveal the interactions between them. See Section 2.2 and Section 2.4 for details. We will use an Internet traffic data set (ref. Section 2.2, 2.6.1) to show the usefulness of the surface plots, curve movies and rotation movies. Zoomed versions of the plots and movies are motivated by a chemometrics data set (ref Section 2.6.2). A Spanish mortality data set (ref section 2.6.3) is studied as well to show that the image plots highlight the *cohort* of ages and years (i.e., interactions between columns and rows). See Chapter 2 for details on the connections between SVD and PCA, and on the new visualization methods.

The above content (Zhang et al., 2006) won the 2005 student paper award of the American

Statistical Association Sections of Statistical Computing and Statistical Graphics, and has been accepted by *Journal of Computational and Graphical Statistics* for publication.

1.2 Multi-Resolution Anomaly Detection

Internet intrusions, including Distributed Denial of Service (DDoS) attacks, worms etc., are major threats for the data network. These intrusions can consume all the network resources or bandwidths in a very short period of time, such that legitimate network users are no longer capable of using some part of the network. Detection of network anomalies has been one major approach for identifying Internet intrusions (Section 1.3 provides review of some Intrusion detection methods). Anomaly detection in the networking field is similar to outlier detection in Statistics.

Internet traffic, collected over time at a single location, forms time series of features, which have self-similarity (SS) and long range dependence (LRD) (Leland et al., 1994; Willinger et al., 1997). These features make classical outlier detection methods based on short-range dependent time series unsuitable in this case, because time series with LRD are more bursty, such that those methods will generate more false alarms (i.e., larger type I error). Furthermore, different types of network anomalies (e.g. flash crowd, DDoS attacks etc.) exist in different time scales (Barford et al., 2002). Thus, a good Internet anomaly detection method should incorporate multiscale properties. Various network anomalies might exhibit abnormal statistical behaviors with different network measurements (Terrell et al., 2005). Note that, in this thesis, we will focus on the time series of one single network measurement.

In this dissertation, we propose a Multi-Resolution Anomaly Detection (MRAD) method to detect outliers in a LRD time series. Let $\{Y_1(i)\}$ be the observed time series of one network measurement, assumed to be generated from a standard fractional Gaussian noise (fGn) with Hurst parameter H plus a mean level shift with unknown duration K. The problem we intend to test is whether the ith observation $Y_1(i)$ has a level shift or not. Details of this problem are discussed in section 3.4.3. The proposed MRAD method can be described as follows.

1. Aggregate the time series $\{Y_1(i)\}$ (the observed time series, or the time series at scale 1) to obtain time series $\{Y_L(i)\}$ at time scale L (details ref. Section 3.4.4), such that at one particular time location, the aggregate observations have the same marginal distribution under the null hypothesis (i.e., the observed time series is only a realization of a fGn with no outliers).

2. The rejection region of the MRAD method is

$$\max_L |Y_L(i)| > C_\alpha^M,$$

   where the threshold C_α^M is a function of the significance level α and the Hurst parameter H of the fGn.

For each time location i, and each scale L, the rejection region of a naive testing method has the form "$|Y_L(i)| > C$". If the naive method and the MRAD are set to have the same probability of type I error (i.e., the same level of significance), the thresholds of the naive method of any single scale are the same, which is denoted as C_α for level of significance α. See Section 3.4.5 for details. We will prove in Chapter 3 that the 2-scale MRAD method provides a more conservative threshold than the naive method using one single scale, i.e., $C_\alpha^M \geq C_\alpha$ (ref. Proposition 3.4.1 in Section 3.4.5). We have proven an important theorem, that the MRAD method has larger power on average than the power of the naive method (ref. Theorem 3.4.1 in section 3.4.5). Thus the MRAD method is more conservative than the outlier detection method at a given scale, but has larger power than the outlier detection method at the given scale under certain conditions (ref. Section 3.4.5, 3.8). In addition, we explored the asymptotic distribution of $\max_L\{Y_L(i)\}$, and developed an asymptotic test threshold for the MRAD method (ref Section 3.4.5).

In Section 3.4.4, an MRAD outlier map is developed to visualize the possibility of an observation as an outlier (significance probability, or p values) in color at different scales. The MRAD outlier map highlights anomalies at different locations and different scales, using different colors to represent corresponding significance probabilities. Details of the map are discussed in section 3.4.4. Toy examples of some special designed time series with local mean level shifts are designed to illustrate the usefulness of the MRAD method (ref. section 3.6). One real network data set is studied in section 3.2 and Section 3.6.6, where network anomalies exist in different scales, and they are obvious now in the MRAD map.

The rest of the dissertation is arranged as following. The next section, Section 1.3 describes the background of network traffic analysis and network anomalies detection. Chapter 2 shows the connections and differences between SVD and PCA, four types of SVD decomposition, and the proposed matrix view of the SVD components. Chapter 3 proposes a multiresolution outlier detection method for a time series with long range dependence, compares the method with other existing methods, and discusses some properties of the proposed method. Further comments are summarized in Chapter 4.

1.3 Background on network analysis and intrusion detection

The Internet transfers files (webpages, data files or media files, etc.) by dividing them into small chunks called *packets*. Network protocols are designed to setup a connection between two *hosts* (i.e. ending computers on the edge of the network), and transfer files between them. For example, the Transmission Control Protocol (TCP) is one such network protocol. Classical textbooks about computer networks (e.g. Stevens (1994) and Kurose and Ross (2001)) provide good introduction.

Analysis of network traffic data is a wide research area. Network features, including packet counts, byte counts and flow counts, collected over time at a single location form time series. There are many statistical methods to understand the features of time series. For example, queueing theory (e.g. Rolls et al. (2005)), multivariate analysis methods (e.g. Lakhina et al. (2004a,b,c) etc.), time series analysis etc.

The current Internet is threatened by all kinds of malicious usages, for example the Distributed Denial of Service (DDoS) attacks (Specht and Lee, 2004), worms (Seely, 1989; Spafford, 1989) etc. Internet intrusions consume a lot of network resources or network bandwidth, thus detection and defense against Internet attacks has become a new important field in network security research. There are many statistical methods which help to identify Internet intrusions. One usual way to deal with network intrusions is to treat common usages of network as the normal traffic, and the malicious usages as network anomalies.

The next two subsections here give a short brief introduction of network analysis and network intrusion detection. Section 1.3.1 reviews a small portion of network traffic data analysis, and Section 1.3.2 discusses some popular network intrusion detection methods.

1.3.1 Network data analysis

This subsection shows a small aspect of network data analysis. As mentioned earlier, network features collected over time at a single location form a time series. Some common network features include packet counts, byte counts etc. The network data in Figure 2.1(a) of Section 2.2 is an example of packet count time series.

There is a large literature in the network traffic data analysis field, using many kinds of statistical methods. The multiscale property of network traffic data is one of the fundamental properties. The time series of network features, e.g. packet counts over equal time intervals, shows self-similarity. Thus the time series has long range dependence (Leland et al., 1994). Paxson and Floyd (1995) showed that the usual Poisson modeling designed for telephony networks failed in the Internet context. The duration of network connections are approximately modeled with heavy tailed distributions rather than the classical exponential. Willinger et al. (1996) provided a good review of self-similarity of network analysis. Cao et al. (2002) showed that, in some ideal contexts, the network packet count time series converges to a Poisson process, when the number of connections goes to infinity, and the time intervals go to zero.

The following analyses are for time series collected at the link with relative lower speed, or with smaller network load. In this case the time series has long range dependence and self-similarity (Hannig et al., 2001; Embrechts and Maejima, 2002) and the Poisson modeling is not appropriate here. Because of the self-similarity of the time series, the usual ARIMA models might not be appropriate here. The Hurst parameter H of a time series with long range dependence, which is related to the self-similarity parameter, is estimated from the network traffic time series. Fractional Gaussian noise, fractional Brownian motion, and the Lévy process have been applied as approximations of the underlying flow of network traffic. In addition, some signal processing methods, e.g. Wavelet analysis and Fourier analysis, can provide additional understanding of the network traffic data.

Connection type of data are another interesting type of network data, which can be collected by a larger network service provider, e.g. AT&T, Sprint, or even larger backbone network providers. This type of data measures the connections, which are from one host in the network to another host in the network. Network measurements can also include the packet counts, byte counts and flow counts between these two hosts, which provide a global view of the network usage. The data also form many different time series based on different origin-destination pairs. In other words, the data, from the same origin and the same destination, are aggregated together and form a multiple time series. Then some multivariate analysis and multiple time series analysis methods can be applied here to find interesting features (Lakhina et al., 2004a,b,c).

1.3.2 Network intrusion detection method

Since the 1980's, the Internet has become more and more popular. Due to network system weaknesses and some malicious users of the network, network intrusions have become one of key problems for network engineers, network designers and network researchers. The first generation of system intrusions usually was based on gaining the power user's privileges or obtaining the access rights of the administrators. Malicious users use some non-secure commands or scripts to access some "confidential" information ("confidential" here means those users are not authorized to access the materials). Nowadays, most attacks are Distributed Denial of Service (DDoS) attacks. The attackers use bugs or weaknesses of the current operating systems, or some properties of the Internet protocols, and setup a lot of "zombie" armies (the computers act as anonymous attack agents, which are controlled by the hackers through some self-executed scripts or programs). They use the zombie armies as agents to attack some specific hosts, by jamming the network bandwidth, or exploiting the resources of the system, such that other normal users can not access the hosts or the network, or can not use any resources in the hosts (Specht and Lee, 2004). The rate of spread of the DDoS attacks is really fast, so the detection and defense methods should be close to real time.

There are many methods to detect network intrusions. The following materials are based on the network intrusion course (Jeffay, 2005), which was taught by Prof. Kevin Jeffay in the department of

Computer Science of UNC. Common intrusion detection methods include signature-based detection methods, signal processing methods, multivariate analysis methods, data mining methods, machine learning methods etc.

Network intrusion attacks might have some specific patterns in the TCP/UDP headers. It is natural to scan all of the packets going through some specific network connection, and to check whether the through traffic matches the signature of attacks (i.e. the specific pattern in the header or the payload). Some shareware or commercial detection software include Snort (Roesch, 1999) or Bro (Paxson, 1999). Based on the signatures, the packets can be classified as belonging to some known attacks or not (Sommer and Paxson, 2003). The advantage of Signature-based detection methods is that detection of a specific attack has small false alarm rate, if the patterns of the attack are stored in the detection database. One drawback of these signature-based methods is that the pattern should be exactly matched. If the attacker modifies the patterns a little bit, the packets he/she sends out might not be detectable unless the new patterns are updated in the detection box. In addition, new types of attacks might not be detectable because the patterns might not be recorded in the detection database. We need to extend the detection database every day or in a short period of time.

For the time series of network features, the malicious users, including network intrusion attackers, might yield traffic series which are "different" from the normal traffic generated by the common users. One way to detect intrusions is to treat the normal traffic as the background or major part of the time series, and the intrusions as the signal or outliers (or anomalies) in it. Thus, many methods of outlier detection or signal detection, can be used to detect anomalies.

For example, Signal processing based methods can be used to detect network anomalies. The wavelet method (Barford et al., 2002) can separate the time series of packet counts or byte counts into high, medium and low frequency parts, where at each part some outlier detection methods can be applied, and find the anomalies there. The high frequency anomalies are more related to the short-term DDoS attacks, while the low and medium parts are more related to long term network anomalies (e.g. flash flood), etc. Using the periodicity of TCP round trip time and spectral analysis is another approach to detection (Cheng et al., 2002).

As mentioned in Section 1.3.1, multivariate analysis methods, such as Principal Component Analysis (PCA), can be used to detect network anomalies for multivariate time series. PCA decomposes the data set into several components, where the first several contain the normal traffic and some obvious anomalies which affect the variance, and the last several show those anomalies which exist in a short period of time or in only a few connections. Lakhina et al. (2004a,b,c) explored the connection types of data, and classified the traces (same feature traces by different origin-destination pairs) into normal and anomaly traffic. Terrell et al. (2005) explored multivariate time series of different features which collected at the main Internet link between the UNC campus and outside, and concluded that multiple features help to reveal network anomalies.

The above statistical detection methods are based on off-line data, where the data will not be updated while the data are processed and analyzed. The usual online detection boxes are based on signature-based detection methods. As mentioned earlier, signature-based detection methods have strong power to detect well-known intrusions, but are weak on finding variations of intrusions and new versions of attacks. And statistical anomaly detections are strong at detecting those attacks which have different statistical features from normal usages. Auditing the offline data and automatically updating the internet detection system box lead to the detection methods based on data mining (Lee et al., 1999; Stolfo et al., 2001; Lee and Fan, 2001).

There are some other statistical analysis methods to detect network anomalies, such as machine learning methods, statistical classification methods, etc. For example, the robust support vector machine method in Hu et al. (2003), classifying user behaviors in Lane and Brodley (1997), the sequential hypothesis testing method of Jung et al. (2004), etc.

Chapter 2

SVD, PCA and Visualizations

Singular Value Decomposition (SVD) of a data matrix is a useful tool in Functional Data Analysis (FDA). Compared to Principal Component Analysis (PCA), SVD is more general, because SVD simultaneously provides the PCAs in both the row and the column spaces. In Section 2.3.2, we compare SVD and PCA from an FDA view point, and extend the usual SVD to potentially useful variations by considering different centerings. A generalized scree plot is proposed in Section 2.3.4 as a visual aid for model selection. Several matrix views of the SVD components are introduced to explore different features in data, including SVD surface plots, rotation movies, curve movies and image plots. These methods visualize both column and row information of a two-way matrix simultaneously, relate the matrix to relevant curves, and show local variations and interactions between columns and rows. Several toy examples (Section 2.5) are designed to compare as well as reveal the different variations of SVD, and real data examples (Section 2.2 and 2.6) are used to illustrate the usefulness of the visualization methods.

2.1 Introduction

Functional Data Analysis (FDA) is the study of curves as data (See Ramsay and Silverman (2002, 2005) for a good summary of FDA). A new statistical framework, Object Oriented Data Analysis (OODA) allows analysis of more complicated objects, such as images (Locantore et al., 1999), trees (Wang and Marron, 2006) and so on. Methods related to Principal Component Analysis (PCA) have provided many insights. Related to the PCA method, Singular Value Decomposition (SVD) can be thought of as more general, in the sense that SVD not only provides a direct approach to calculate the principal components (PCs), but also derives the PCAs in the row and the column spaces simultaneously. In this Chapter, we view a set of curves as a two-way data matrix, explore the connections and differences between SVD and PCA from a FDA view point, and propose several visualization methods for the SVD components.

Let X be a data matrix, where the rows are the observations of an experiment (or feature vectors of different objects in FDA or OODA), and the columns are the covariate vectors. SVD provides a useful factorization of the data matrix X, while PCA provides a nearly parallel factoring, via eigen-analysis of the sample covariance matrix, i.e. $X^T X$, when X is column centered at 0 (meaning the mean of each column is 0, i.e., the feature vectors (rows) have mean 0). The eigenvalues for $X^T X$ are then the squares of the singular values of X, and the eigenvectors for $X^T X$ are the singular rows of X. In this Chapter, we extend the usual (column centered) PCA method into a general SVD framework, and consider four types of SVDs based on different centerings: Simple SVD (SSVD), Column SVD (CSVD), Row SVD (RSVD) and Double SVD (DSVD) (ref. Section 2.3 for details). Several criteria are discussed in this Chapter for model selection, i.e. selecting the appropriate type of SVD, including model complexity, approximation performance, and interpretability etc. We introduce a generalized scree plot, which provides a simple way to understand the tradeoff between model complexity and approximation performance, and provides a visual aid for model selection in terms of these two criteria. See Section 2.3.4 for details. Several toy examples in Section 2.5 are designed to illustrate the generalized scree plot and the differences between the four types of centerings. These simulated toy examples show that each of the centerings can be the best choice under certain contexts. See Section 2.5 for details.

Visualization methods can be very helpful in finding underlying features of a data set. In the context of PCA or SVD, common visualization methods include the biplot (Gabriel, 1971), scatter plots between singular columns or singular rows (Section 5.1 in Jolliffe (2002) provides a good introduction, or the analysis of call center data sets in Shen and Huang (2005) is a good example), etc. The biplot shows the relations between the rows and columns, and the scatter plot can be used to show possible clusters in the rows or columns. Each scatter plot used to visual singular columns or singular rows is a two dimensional projection of (the point clouds formed from) the data set in the column spaces or the row spaces respectively. However, for FDA data sets, these plots fail to show the functional curves. There are other forms of low-dimensional projection of the data set, for example three dimensional extensions of the biplot (Gower and Hand, 1996) and the (three dimensional) scatter plots.

In the FDA field, besides the above plots, it is common to plot singular columns or singular rows as curves (Ramsay and Silverman, 2005). This method is also used in the context of high dimensional data sets (ref. Section 5.6 in Jolliffe (2002)) . Marron et al. (2004) provided a visualization method for functional data (using functional PCA, which is CSVD in the notation of this Chapter), which shows the functional objects (curves), projections on the PCs and the residuals. These methods can also be applied in the SVD framework, but to understand all the structure in the data, they need to be applied twice, once for the rows and once for the columns. When the data set is a time series of curves, the method in Marron et al. (2004) uses different colors to show the time ordering. However, if the time series structure is complicated, color coding curves might not be enough to reveal the time effect, because of the overplotting problems.

Here we propose several matrix views of the SVD components. These visualizations may reveal new underlying features of the data set, which are not likely available for earlier methods. The following describes our proposed visualization methods.

- The major visualization tool is *a set of surface plots*. The surface plots for the SVD components show the functional curves of rows and columns simultaneously. See Section 2.2 for details.

- *The SVD rotation movie* shows the surface plots from different view angles, which makes it easier to find underlying information. In fact, the visualization method in Marron et al. (2004) (for example, the first column in Figure 3 of their paper) can be viewed as the surface plots from a special angle.

- Another special view angle, the overlook view of the surface plots, becomes *the image plot*, which can show the relative variation of the surface, and highlight the interaction between the columns and the rows.

- Motivated by the definition of the SVD components (ref. Section 2.3), we introduce *the SVD curve movie*, which can show the time varying features of the functions.

One motivating example is an Internet traffic data set, which is discussed in Section 2.2 to illustrate the usefulness of the surface plots, the rotation movie and the curve movie. Two other real applications are reported in Section 2.6 as well. One chemometrics data set is used to show that a zoomed version of the surface plots and the SVD movies can highlight local behaviors. A Spanish mortality data set is analyzed to illustrate that the image plot highlights the cohort effect (i.e., the interaction between age groups and years).

The remaining part of this Chapter is organized as follows. The motivating example in network traffic analysis is in Section 2.2. Section 2.3 gives a brief introduction of SVD, and compares it with PCA. Section 2.4 describes the generation of the plots and the movies in detail. Section 2.5 shows several toy examples to illustrate the four types of centerings. Two more real applications in chemometrics and demography are reported in Section 2.6.

2.2 Motivating example

Internet traffic, measured over time at a single location, forms a very noisy time series. Figure 2.1(a) shows an example of network traffic data collected at the main Internet link of the UNC campus network, as packet counts per half hour over a period of 7 weeks. This period covers part of two sessions of UNC summer school in 2003. The approximately 49 tall thin spikes represent peak daytime network usage, which suggests that there is a strong daily effect. The tallest spikes are grouped into clusters of 5 corresponding to weekdays, with in-between gaps corresponding to weekends.

Because of the expected similarity of the daily shapes, and as a device for studying potential contrasts between these shapes (e.g. differences between weekdays and weekends), we analyze the time series in an unusual way. We rearrange the data as a 49×48 matrix, so that each row represents one day, and each column represents one half-hour interval within a day. Thus each row of the resized matrix is the network daily usage profile of each day, and each column of it is the time series across days for each given time in a day. This treatment is similar to the singular-spectrum analysis method (Golyandina et al., 2001). A mesh plot showing the structure of the data matrix

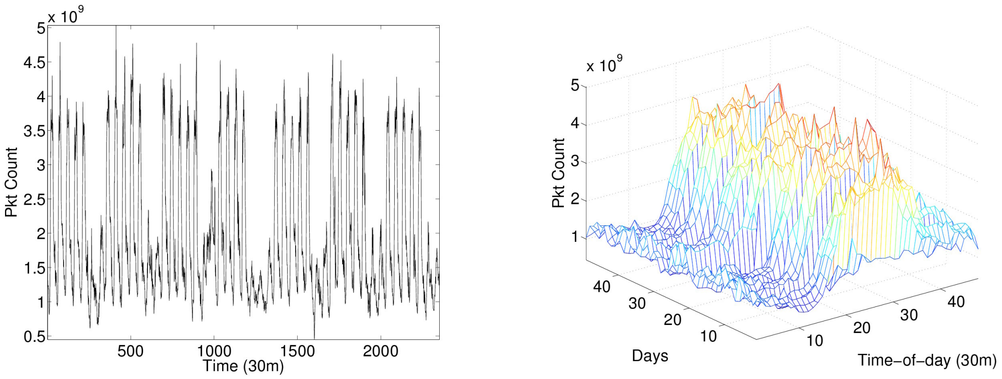

Figure 2.1: *(a) Original time series plot of packet counts per half-an-hour over 49 days. The 49 spikes correspond to peak daytime usage, i.e., there is a strong daily effect. Tallest spikes are grouped into clusters of 5 corresponding to weekdays, with in-between gaps corresponding to weekends. (b) Mesh plot of the resized matrix (ref. section 2.2 for details) for the original network traffic data. It shows a clear daily shapes, and contrasts between weekdays and weekends.*

.

is in Figure 2.1(b). This shows that the data matrix is noisy, but we can still see a weekly pattern and also clear daily shapes from it.

One way to analyze the data set is to treat the daily shapes, the *rows* of the data matrix, as functions (curves), and to use some functional data analysis methods to understand the characteristics of the data. PCA has proven to be very useful for this purpose. But observe that the *columns* of the data matrix are also curves (i.e. time series) of interest as well. In particular, these are the counts over days, for each half hour interval. A natural eigen-analysis for simultaneous PCA of rows and columns is contained in the SVD of the data matrix. Similar to the PCA method, SVD provides a useful first tool for exploratory data analysis. SVD decomposes the data matrix into a sum of rank one matrices (which are the SVD components). Each component provides insights into features of the data matrix.

Here we provide a new visualization method to find data characteristics using SVD. This is a set of surface plots of the SVD components, which help to examine the data matrix in both directions (i.e., both rows as data, and columns as data). The set of surface plots include the following components:

- the surface plot of the original data matrix,
- the surface plot of the mean matrix, if applicable,
- the surface plot of the first several SVD components,
- the surface plot of the reconstruction matrix, which is the summation of the mean matrix and the first several SVD components,
- the surface plot of the residual matrix.

Long (1983) used a similar method to illustrate matrix approximation of SVD from a mathematical education view point. Interpretation of the surface plots is aided by a movie which shows the plots from different angles and also by a movie which highlights the rows and columns. The movies can be used to demonstrate the time-varying features of the components, and highlight some special features or outliers. These visualization methods can be used alone or with other visualization methods, to find interesting structures in the data.

For this network traffic data, the set of SVD surface plots is in Figure 2.2. Note that, here we use the SSVD model with three SVD components, so the set of the surface plots does not include the mean matrix.

- The top left is the mesh plot for the original data matrix (the same as in Figure 2.1(b)).
- The top right is the first SVD component, which turns out to be a smoothed version of the original data matrix, showing a clear weekly pattern.

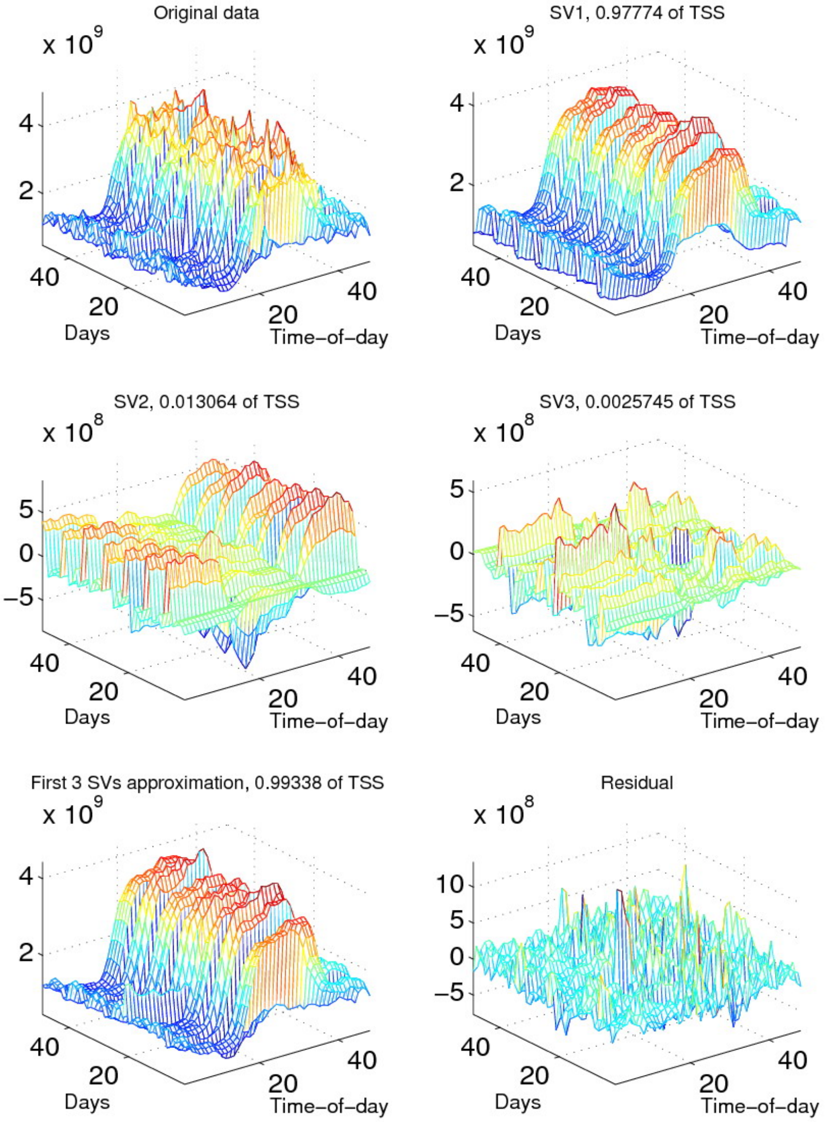

Figure 2.2: *The surface plots of SSVD for the 49 days network traffic data. SV1-SV3 are a decomposition of the data matrix. These are combined to give the model in the lower left, with corresponding residual shown in the lower right.*

- The first SVD component also indicates that weekdays and weekends might not share the same daily shape.

- The middle left is the second component, which has a clear shape for weekends, and is relatively flat for weekdays. This indicates the existence of a weekday-weekend effect, and suggests analyzing weekend and weekday data separately as a good option.

- The middle right is the third component, which shows that there are very large bumps in some days. Those bumps might indicate that the corresponding dates have some special features, as discussed below.

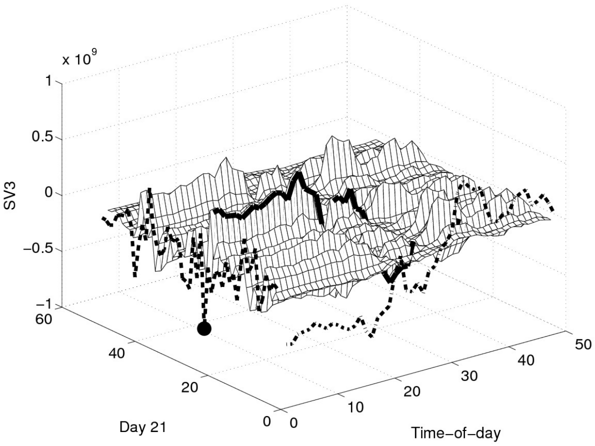

Figure 2.3: *One carefully chosen snapshot of the SVD curve movie of the third SVD component for the network data. The movie highlights some outlying days by showing high spikes on the surface. For instance, this snapshot shows the 21st row, which suggests that June 29 is a special day, as explained in the text.*

The SVD movies for this data highlight those features. All movies for the network traffic data set can be easily viewed at the website of Zhang (2006b). Figure 2.3 shows one carefully chosen snapshot of the SVD curve movie for the third component. It highlights Sunday, June 29 (Figure 2.3) as a possible outlier, by showing a large bump in the movie. This day had a network workload that was between the weekday and weekend data. A check of the university calendar reveals that this is the first Sunday for the second summer session, when a large number of students returned from home after the break between summer sessions, which made the profile of network usage different on that day from the other weekend days. The SVD rotation movie and more analysis of this network data are discussed in Section 2.6.1.

2.3 SVD and PCA

In this section, we give a brief introduction of the mathematical underpinnings of SVD (Section 2.3.1), its relation with PCA (Section 2.3.2), the geometric interpretation of different means and the corresponding decomposition (Section 2.3.3) and a visual aid for choosing among different centerings (Section 2.3.4).

2.3.1 SVD and its properties

Let $X = (x_{ij})_{m\times n}$ with $\text{rank}(X) = r$, $\{\mathbf{r}_i : i = 1, \cdots, m\}$, $\{\mathbf{c}_j : j = 1, \cdots, n\}$ be the row and column vectors of the matrix X respectively. The SVD of X is defined as

$$X = USV^T = s_1\mathbf{u}_1\mathbf{v}_1^T + s_2\mathbf{u}_2\mathbf{v}_2^T + \cdots + s_r\mathbf{u}_r\mathbf{v}_r^T$$

where $U = (\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_r)$, $V = (\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_r)$, and $S = \text{diag}\{s_1,\ s_2, \cdots, s_r\}$ with $s_1 \geq s_2 \geq \cdots \geq s_r > 0$. These $\{\mathbf{u}_i\}$ are orthonormal basis for the column space spanned by the column vectors $(\{\mathbf{c}_j\})$, and the $\{\mathbf{v}_j\}$ form orthonormal basis for the row space spanned by the row vectors $(\{\mathbf{r}_i\})$. The vectors $\{\mathbf{u}_i\}$ and $\{\mathbf{v}_i\}$ are called *singular columns* and *singular rows* respectively (Gabriel and Odoroff, 1984); the scalars $\{s_i\}$ are called *singular values*; and the matrices $\{s_i\mathbf{u}_i\mathbf{v}_i^T\}(i = 1, \cdots, r)$

are referred to as the *SVD components*.

The SVD factorization has an important approximation property. Let A be a rank k ($k \leq r$) (approximation) matrix, and define $R = X - A = (r_{ij})_{m \times n}$ as its residual matrix. We then define the Residual Sum of Squares (RSS) of the matrix A as the Sum of Squares (SS) of the elements in R, i.e.,

$$\mathrm{RSS}(A) = \mathrm{SS}(R) = \sum_{i=1}^{m} \sum_{j=1}^{n} r_{ij}^2. \tag{2.1}$$

Householder and Young (1938) showed that

$$\underset{A:\mathrm{rank}(A)=k}{\arg\min}\ \mathrm{RSS}(A) = A_k = \sum_{l=1}^{k} s_l \mathbf{u}_l \mathbf{v}_l^T,$$

and the corresponding residual matrix R is $R = X - A_k = \sum_{l=k+1}^{r} s_l \mathbf{u}_l \mathbf{v}_l^T$. In other words, the SVD provides the best rank k approximation of the data matrix X.

2.3.2 Four types of centerings for SVD

As noted above, SVD and PCA are closely related. SVD, as defined above, provides a decomposition of X. PCA is very similar with the difference of being column mean centered. Our matrix view raises the question of: “why not do row mean centering?” We will study and compare four possible types of centerings: no centering (SSVD), column centering (CSVD), row centering (RSVD) and centering in both row and column directions, which is referred to as *double centering* (DSVD).

Another natural choice of centering is to remove the overall mean and then apply SVD. When the overall mean of the data matrix is far away from the origin, removing the overall mean will decrease the magnitudes of the observations, and thus can improve the numerical stability of the SVD. However, Gabriel (1978) mentioned that, in the case of a model with an overall constant plus multiplicative terms, the least squares estimation of the components is not equivalent to fitting the overall mean and then applying SVD to the residual part. With a functional data set, the overall mean does not provide useful information about the curves in the data set. There are cases where removing the overall mean will cause the data to lose some useful properties, for example, the orthogonality of the curves (see Zhang (2006b)). Note that the column/row/double centerings automatically remove the overall mean. Thus, we will not discuss the case of just removing the overall mean. However, our programs do provide the option of applying SVD after just removing the overall mean.

Let $\overline{x}$ be the sample overall mean of all the elements in a $m \times n$ dimensional data matrix X, $\overline{x}_c$ be the $n \times 1$ column mean vector (the elements are the means of corresponding columns), and $\overline{x}_r$ be the $m \times 1$ row mean vector (the elements are the means of corresponding rows). The mathematical definition of these means is

$$\begin{aligned}
\overline{x} &= \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}, \\
\overline{x}_c &= \left(\frac{1}{m}\sum_{i=1}^{m} x_{i1}, \frac{1}{m}\sum_{i=1}^{m} x_{i2}, \cdots, \frac{1}{m}\sum_{i=1}^{m} x_{in}\right)^T = \frac{1}{m}\sum_{i=1}^{m} \mathbf{r}_i, \qquad (2.2)\\
\overline{x}_r &= \left(\frac{1}{n}\sum_{j=1}^{n} x_{1j}, \frac{1}{n}\sum_{j=1}^{n} x_{2j}, \cdots, \frac{1}{n}\sum_{j=1}^{n} x_{mj}\right)^T = \frac{1}{n}\sum_{j=1}^{n} \mathbf{c}_j,
\end{aligned}$$

where as at the beginning of Section 2.3.1, the $\{\mathbf{r}_i\}$ are the row vectors of X, and they correspond to the data points in the column space. Meanwhile, the $\{\mathbf{c}_j\}$, the column vectors of X, are the data points in the row space. Note that the overall mean is a scalar, while the column mean and the row mean are vectors. The above definitions directly show that, the column mean is the center of the data point cloud in the column space, and the row mean is the center in the row space. We define the sample overall mean matrix (OM), sample column mean matrix (CM), sample row mean matrix (RM) and the sample double mean matrix (DM) as follows

$$\mathrm{OM} = \overline{x}1_{m\times n}, \quad \mathrm{CM} = 1_{m\times 1}\overline{x}_c^T, \quad \mathrm{RM} = \overline{x}_r 1_{1\times n}, \text{ and } \mathrm{DM} = \overline{x}_r 1_{1\times n} + 1_{m\times 1}\overline{x}_c^T - \overline{x}1_{m\times n}. \quad (2.3)$$

It is straightforward that these mean matrices are uniquely defined for a given data matrix.

Removing these three mean matrices, along with no centering, provide four types of SVD. All four of these decompositions have the same form:

$$X = A + R$$

where A is the approximation matrix, and R is the residual matrix. There are several different forms of the approximation matrix A and the corresponding residual matrix R.

- SSVD: As noticed above, for no centering, $A = A^{(s)}$ is the sum of the first several SVD components of X, which provides the best approximation of the corresponding rank.

- CSVD, RSVD and DSVD: For column centering, $A = \mathrm{CM} + A^{(c)}$ where $A^{(c)}$ is the sum of the first several SVD components of $X - \mathrm{CM}$ (i.e., the first several CSVD components). Similarly, for row centering and double centering, we have $A = \mathrm{RM} + A^{(r)}$ and $A = \mathrm{DM} + A^{(d)}$, where $A^{(r)}$ is the sum of the first several SVD components of $X - \mathrm{RM}$ (i.e., the first several RSVD components) and $A^{(d)}$ is the sum of the first several SVD components of $X - \mathrm{DM}$ (i.e., the first several DSVD components).

Note that A here is not the same for the four centerings, we just use general notation for convenience. Also note that CM and RM usually have rank 1, and DM is at most rank 2.

In terms of approximation performance (i.e, comparison of the RSS, defined in equation (2.1)), we have a diagram to visualize the comparisons among these four types of centerings, as shown in Figure 2.4. Theoretical comparisons are reported in equations (2.4)-(2.11).

Approximation performance and model complexity diagram

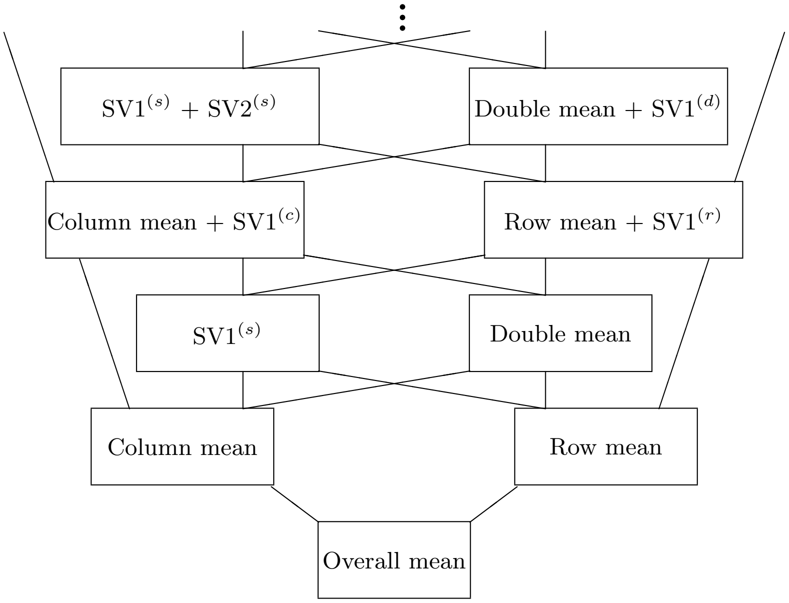
SV1$^{(s)}$ + SV2$^{(s)}$
Double mean + SV1$^{(d)}$
Column mean + SV1$^{(c)}$
Row mean + SV1$^{(r)}$
SV1$^{(s)}$
Double mean
Column mean
Row mean
Overall mean

Figure 2.4: *Relationship between approximations using the four types of SVD factorization. A lower level approximation is always worse than an upper level one. Notice that the lower level also provides a simpler model than the upper level. Lines show direct comparisons between two rectangles, where lower rectangles correspond to larger RSS and less model complexity. More discussion is in the text.*

Figure 2.4 shows the relative approximation relations among the four types of centerings. Theoretical comparisons are reported in equations (2.4)-(2.11). The boxes on the same horizontal level have non-comparable RSS (i.e., either can give a better approximation). Lower level boxes always have larger RSS than upper level boxes. The overall mean ($\overline{x}$) provides the worst approximation (i.e. the largest RSS) among RM, CM and DM. The line segments show direct comparisons given in equations (2.4)-(2.11). For example, the column mean provides a larger RSS than both the double mean and the first SSVD component, and so does the row mean (as shown in equations (2.4)-(2.7) with $\mathrm{rank}(A^{(s)}) = 1$). The first SSVD component has larger RSS than either RSVD or CSVD with mean matrix plus one SVD component, and so does the double mean.

This diagram also shows the level of complexity, i.e., the same level shares similar complexity of the model, and the lower level models are simpler than the upper level models. For example, As shown in Figure 2.4, a model using either the column mean or the row mean is simpler than using the double mean or the first SSVD component. In fact, the double mean can be viewed as the sum of the row mean and the column mean (i.e., an additive model), while the first SSVD component can be treated as the product of the row and the column mean (i.e., a multiplicative model). This is why we treat them as having the same level of complexity.

Theoretical comparisons among the four types of centerings

The following propositions provide the mathematical underpinnings of the comparisons between the four types of centerings in terms of model approximation, which are represented by connecting lines in Figure 2.4.

The first result gives a theorem about the approximation relations between the CSVD model (or the RSVD model) and the DSVD model, when they have the same number of multiplicative terms. These two inequalities show that the under this condition, the double centering gives a better approximation than either the column centering or the row centering. For example, as shown in Figure 2.4, the column mean matrix (or the row mean matrix) of the data matrix X has larger RSS than the double mean matrix (this is the case of $\mathrm{rank}(A^{(c)}) = 0$, i.e., no multiplicative term).

Proposition 2.3.1. *If* $\text{rank}\left(A^{(c)}\right) = \text{rank}\left(A^{(r)}\right) = \text{rank}\left(A^{(d)}\right)$*, we have*

$$\text{RSS}\left(\text{CM} + A^{(c)}\right) \geq \text{RSS}\left(\text{DM} + A^{(d)}\right), \tag{2.4}$$

$$\text{RSS}\left(\text{RM} + A^{(r)}\right) \geq \text{RSS}\left(\text{DM} + A^{(d)}\right). \tag{2.5}$$

Proof: This proposition can be directly derived from the fact that the CSVD or the RSVD model are nested models of the DSVD model, under the above assumptions.

Remark: Note that the rank of the double mean matrix is usually 2. Thus, under the above conditions, the approximation matrix of the DSVD model (i.e., $\text{DM} + A^{(d)}$) usually has one larger rank than that of the CSVD (or RSVD) model (i.e., $\text{CM} + A^{(c)}$ or $\text{RM} + A^{(r)}$).

The second result says that when the approximation matrices of the SSVD, the CSVD and the RSVD models have the same rank, the SSVD has the smallest RSS. For instance in Figure 2.4, the first SSVD component has smaller RSS than only the row mean matrix or the column mean matrix, which is the case of $\text{rank}(A^{(c)}) = 0$ in the proposition.

Proposition 2.3.2. *If* $\text{rank}\left(A^{(c)}\right) = \text{rank}\left(A^{(r)}\right) = \text{rank}\left(A^{(s)}\right) - 1$ *, we have*

$$\text{RSS}\left(\text{CM} + A^{(c)}\right) \geq \text{RSS}\left(A^{(s)}\right), \tag{2.6}$$

$$\text{RSS}\left(\text{RM} + A^{(r)}\right) \geq \text{RSS}\left(A^{(s)}\right). \tag{2.7}$$

Proof: The approximation matrices under these assumptions have the same rank. This proposition can be directly derived from the fact that the SSVD provides the best rank k approximation, when the approximation matrices have the same rank.

The third result shows that when the SSVD, the CSVD and the RSVD models have the same number of multiplicative terms, the SSVD model has a larger RSS (i.e., worse approximation performance) than the other two. For example, in the diagram of Figure 2.4, the first SSVD component has larger RSS than the sum of the row mean (/the column mean) and the first RSVD (/CSVD) component, which is the case of $\mathrm{rank}(A^{(c)}) = 1$ in the proposition.

Proposition 2.3.3. *If* $\mathrm{rank}(A^{(c)}) = \mathrm{rank}(A^{(r)}) = \mathrm{rank}(A^{(s)})$, *we have*

$$\mathrm{RSS}\left(\mathrm{CM} + A^{(c)}\right) \leq \mathrm{RSS}\left(A^{(s)}\right), \tag{2.8}$$

$$\mathrm{RSS}\left(\mathrm{RM} + A^{(r)}\right) \leq \mathrm{RSS}\left(A^{(s)}\right). \tag{2.9}$$

Proof: Assume that $\mathrm{rank}(A^{(c)}) = \mathrm{rank}(A^{(r)}) = \mathrm{rank}(A^{(0)}) = k$. If we have the following models

$$X(i,j) = c_j + \sum_{l=1}^{k} \lambda_l \mathbf{u}_{i,l} \mathbf{v}_{j,l} + \varepsilon(i,j),$$
$$X(i,j) = r_i + \sum_{l=1}^{k} \lambda_l \mathbf{u}_{i,l} \mathbf{v}_{j,l} + \varepsilon(i,j).$$

Gabriel (1978) showed that the least square estimations of the above two models are exactly the CSVD model and RSVD model respectively, while SSVD can be viewed as the cases $c_j = 0$ and $r_i = 0$ for all i and j. Thus we have the proposition.

Remark: Note that in this context, the approximation matrices of the RSVD and CSVD usually have a rank that is one larger than that of the SSVD.

The fourth result describes that when the approximation matrices of the CSVD, the RSVD and the DSVD have the same rank, the DSVD is the worst model in terms of the approximation. For example, in the diagram of Figure 2.4, the double mean matrix has larger RSS than the sum of the column mean matrix (/row mean matrix) and the first CSVD (/RSVD) component, which is the case of rank$(A^{(c)}) = 1$ in the proposition.

Proposition 2.3.4. *If* rank$(A^{(c)}) = $ rank$(A^{(r)}) = $ rank$(A^{(d)}) + 1$*, we have*

$$\text{RSS}\left(\text{CM} + A^{(c)}\right) \leq \text{RSS}\left(\text{DM} + A^{(d)}\right), \tag{2.10}$$

$$\text{RSS}\left(\text{RM} + A^{(r)}\right) \leq \text{RSS}\left(\text{DM} + A^{(d)}\right). \tag{2.11}$$

Proof: Here we just prove the relation between the CSVD and the DSVD. The proof for between the RSVD and the DSVD will be very similar. Assume that rank$(A^{(c)}) = $ rank$(A^{(d)}) + 1 = k$. If we have the following model

$$X(i,j) = c_j + \sum_{l=1}^{k} \lambda_l \mathbf{u}_{i,l} \mathbf{v}_{j,l} + \varepsilon(i,j),$$

the DSVD model under this context can be viewed as $c_j = \overline{x}_c$, $\lambda_1 = 1$, $\mathbf{u}_l = (u_{1,1}, \cdots, u_{m,1})^T = \overline{x}_r$, $\mathbf{v}_1 = (v_{1,1}, \cdots, v_{n,1})^T = 1_{n\times 1}$, and $\sum_{l=2}^{k} \lambda_l \mathbf{u}_l \mathbf{v}_l^T$ as the first DSVD. Because of the result that CSVD is the least square estimation of the above model, we know the CSVD has smaller residual than the DSVD model.

Non-comparable relations among the four types of centerings

In terms of the approximation performance, i.e. smaller RSS, there is no clear relationship between column centering and row centering when they have the same number of SVD components (either could be better), nor between double centering and no centerings (when the number of SSVD components is one larger than that of the DSVD). There are cases that any of these models can be better than any other, as shown in Section 2.5.

2.3.3 Geometric interpretations of the means and centerings

As discussed in Section 2.3.2, for a two-way data matrix X, we have four different types of mean: the overall mean, the column mean, the row mean, and the double mean (matrix). Note that, the column mean or the row mean is usually referred to as a vector, the overall mean is a scalar, and the double mean is a matrix. When the (matrix) approximation is considered, all of them are treated as matrices, as defined in (2.3) at the beginning of Section 2.3.2. In this section, we explore the geometric interpretations of these different means and the corresponding SVDs. The same notation is used in this section: The column vectors of the matrix X are denoted as $\{\mathbf{c}_j\}$, and $\{\mathbf{r}_i\}$ are the row vectors. In the column space, the points of a data matrix are $\{\mathbf{r}_i\}$, while the points of a data matrix in the row space are $\{\mathbf{c}_j\}$.

In this subsection, a simulated 150×2 data matrix is used to illustrate the geometric interpretations of the means and the centerings in the column space. The data matrix is

$$\begin{pmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \vdots & \vdots \\ x_{150} & y_{150} \end{pmatrix}$$

where $\{x_1, \cdots, x_{150}\}$ and $\{y_1, \cdots, y_{150}\}$ are 150 independent observations of two independent random variables, X and Y, respectively. Here, $X \sim N(4, 1)$ and $Y \sim N(15, 4)$.

Figure 2.5 visualizes this data set in the 2-dimensional space spanned by X and Y. The left column shows different means and the first SSVD component. The middle column visualizes the residual after removing the means or removing the first SSVD component. The corresponding further decompositions are displayed in the right column in Figure 2.5. Note that in this space, each pair (x_i, y_i) $(i = 1, \cdots, 150)$ corresponds to one data point (the blue circles in the plots of Figure 2.5) in the column space.

In the row space, the geometric interpretation of these means and the corresponding decompositions will be similar, where the term “column” should be exchanged with the term “row”, and vice versa. Note that, for this example, the row space is a 150-dimensional space (it is not possible to visualize it), and this data set corresponds to only two data points in it. We skip the discussion of row space because it adds no additional insights.

Column mean and CSVD

The first row of Figure 2.5 visualizes this 150×2 data set, the column mean, the residual after removing the column mean, and the next SVD terms. The blue circles in the left panel of the first row shows these 150 data points, which are the rows of the data matrix (i.e., $\{\mathbf{r}_i = (x_i, y_i)\}$ $i = 1, \cdots, 150$). The data points form an elliptical point cloud, where its long axis is parallel to the y-axis, and the short axis is parallel to the x-axis.

The column mean vector, $(\sum_{i=1}^{150} x_i/150, \sum_{i=1}^{150} y_i/150)$, the element-wise mean of $\mathbf{r}_i$, is the center of these points, shown as the red dot in the subplot. Because the rows of the column mean matrix (i.e., the 150×2 matrix which represents the 1×2 column mean vector, as defined in (2.3)) are the same, this red dot (of the column mean vector) also corresponds to all the rows of the column mean matrix. The fact that $\overline{x}_c$ is the center of all points in the column space is the motivation for calling it the column mean.

The middle panel of the first row shows the rows of the residual matrix, after removing the column mean. The blue pluses in this subplot are a rigid translation of the blue circles in the left panel. Here, the rows of the residual matrix are centered at the origin.

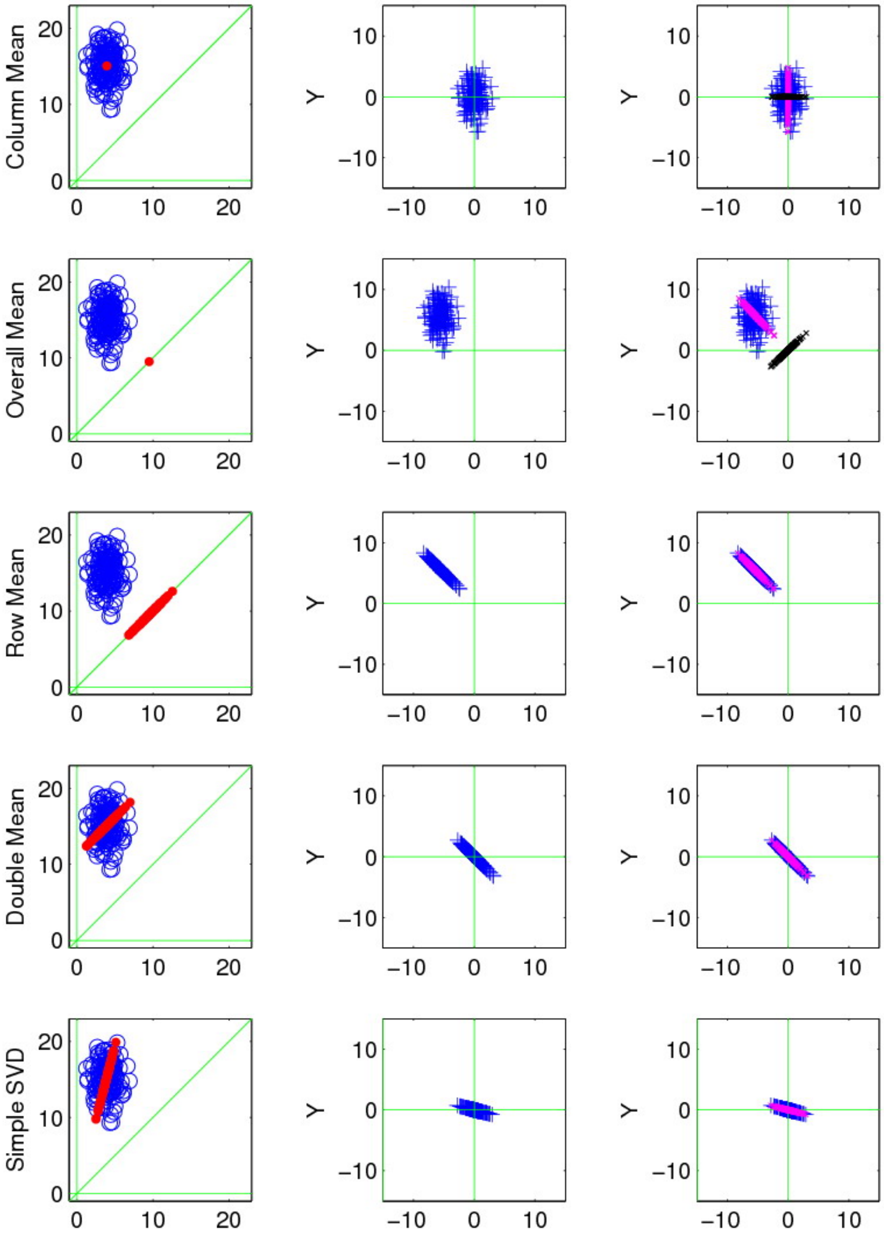

Figure 2.5: *Geometric diagram of the four types of centerings for a two-dimensional point cloud. The first column shows the original data points, while the red dots in them correspond to the overall/column/row/double mean and the first Simple SVD component. The second column shows the residual after fitting the above components. The third column shows the remaining SVD components (the color purple shows the first component, and the black corresponds to the second component).*

The next SVD component of the residual matrix, is to find the projections of the data points of this residual matrix, onto a line through the origin, where the projections have the maximum sum of squares. Because the residual matrix is centered at the origin, the points of the next SVD component are the same as the projections onto the first PC direction (of the sample covariance matrix), i.e., Column SVD is the usual PCA. The right panel of the first row visualizes the next two SVD components. It shows the first SVD component (i.e., the first PC projections) in purple, and the second SVD component (i.e., the second PC projections) in black. Note that, the first SVD component is almost the y-axis, and the second component is nearly the x-axis. These facts are not surprising from the usual geometric interpretation of the PCA method.

Overall mean and the next SVD

The second row of Figure 2.5 shows the overall mean and the next SVD components of this simulated data set. In the column space, the overall mean of the above simulated data matrix, as defined in (2.2), is the projection of the column mean onto the 45 ° line, as shown as the red dot in the left panel of the second row. This can also be seen from the following equations.

$$\begin{aligned}\overline{x} &= \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij} \\ &= \frac{1}{n}\sum_{j=1}^{n}\left(\frac{1}{m}\sum_{i=1}^{m} x_{ij}\right) = \frac{1}{n}1_{1\times n}\overline{x}_c \\ &= \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{n}\sum_{j=1}^{n} x_{ij}\right) = \frac{1}{m}1_{1\times m}\overline{x}_r.\end{aligned}$$

Thus in both the column and the row spaces, the overall mean is the projection of the center point (i.e., the column or row mean) onto the 45 ° line.

If the data points in the column space are near the 45 ° line, the overall mean can be regarded as a sensible notion of center for the observations. However, when the row mean or the column mean is far away from the 45 ° line, it may not make sense to use the overall mean as a center measurement.

The middle panel of the second row in Figure 2.5 shows the point cloud formed by the residual matrix after removing the overall mean. It shows that these points (of the residual matrix) are much closer to the origin than the original data points (i.e., before removing the overall mean). However, the residual point cloud usually does not have the origin as the center (i.e., the column mean of the residual matrix usually is not zero). Thus, the next SVD components usually are not related to the usual PCA method. The right panel of the second row in Figure 2.5 shows the two SVD components of the residual matrix after removing the overall mean. The purple one is the first component, and the black one is the second, which maximize the sum of squares of the projections in order. Note these directions are very different from the CSVD components (the right panel in the first row).

Row mean and RSVD

The third row in Figure 2.5 shows the row mean and the next RSVD component of this simulated data set. Those red dots on the 45 ° are the points of the row mean matrix in the column space. From the definition of the row mean matrix in 2.3.3,

$$\mathrm{RM} = \overline{x}_r 1_{1\times n} = \begin{pmatrix} (\mathbf{r}_1 1_{n\times 1}/m)1_{1\times n} \\ (\mathbf{r}_2 1_{n\times 1}/m)1_{1\times n} \\ \vdots \\ (\mathbf{r}_m 1_{n\times 1}/m)1_{1\times n} \end{pmatrix} = \begin{pmatrix} \overline{x}_r(1)1_{1\times n} \\ \overline{x}_r(2)1_{1\times n} \\ \vdots \\ \overline{x}_r(m)1_{1\times n} \end{pmatrix},$$

where $\overline{x}_r(i)$ is the ith element of the row mean vector. The above equation shows that these points are the projections of the original data onto the 45 ° line. Since the row mean matrix corresponds to a series of points on a line in the column space, it does not make sense to think of them as any type of center in this space. By the definition of the overall mean, these projection points always are centered at the overall mean (as shown in the left panel of the second row).

The residual matrix after removing the row mean matrix for this data set is illustrated in the middle panel of the third row. The points of the residual matrix are in the subspace, which is orthogonal to the 45 ° line. In this subspace, those points are usually not centered at the origin (i.e., their column mean usually is not 0). Thus the remaining SVD is to find the projections, onto a one dimensional subspace, i.e., a straight line, which maximize the sum of squares. The right panel of the third row in Figure 2.5 shows the SVD component for this data set. Note that because the residual matrix is already one-dimensional in this space, these projections are the rows of the residual matrix themselves.

Double mean and DSVD

The fourth row of Figure 2.5 shows the double mean and the next DSVD component of this simulated data set. The left panel of the fourth row uses red dots to visualize the double mean matrix. It shows that the double mean points form a straight line in this space, where this line is parallel to the 45 ° line. In addition, this line is through the column mean point (shown in the left panel of the first row). Note that in the row space, the double mean points also form a straight line, which is parallel to the 45 ° line there, but through the row mean. Compared to the other three means discussed above, these points are the rigid translation of the row mean points, where the translation is exactly moving the overall mean to where the column mean locates. This can be confirmed from the definition of the double mean matrix.

$$\begin{aligned}
\text{DM} &= RM + CM - OM \\
&= \overline{x}_r 1_{1\times n} + 1_{m\times 1}\overline{x}_c^T - \overline{x}1_{m\times n} \\
&= \begin{pmatrix} \overline{x}_r(1)1_{1\times n} + (\overline{x}_c^T - \overline{x}1_{1\times n}) \\ \overline{x}_r(2)1_{1\times n} + (\overline{x}_c^T - \overline{x}1_{1\times n}) \\ \vdots \\ \overline{x}_r(m)1_{1\times n} + (\overline{x}_c^T - \overline{x}1_{1\times n}) \end{pmatrix}.
\end{aligned}$$

This equation proves the former interpretation.

The middle panel of the fourth row in Figure 2.5 shows the point cloud of the residual matrix, after removing the double mean matrix. It shows that the residual point cloud is in the subspace orthogonal to the 45 ° line. Moreover, these points are currently centered at the origin. This is also true in the row space, because the double mean has a similar interpretation in both the column space and the row space. Thus, the next SVD component is the PCA (of the residual matrix) in the column space, and also the PCA in the row space. The right panel of the fourth row shows the DSVD component for this toy data set. In this case, the next DSVD component is also the same as the residual matrix.

Without removing any means and SSVD

The fifth row in Figure 2.5 visualizes the SSVD of this simulated data set. The left panel shows the original data as blue circles, and the first SSVD component as red dots. Because the first k SSVD components provide the best k-subspace approximation, the points of the first SSVD component always form a straight line through the origin. The direction is driven by the geometry of the data points in the space, which maximize the sum of squares of the projections.

The points of the residual matrix, after subtracting the first SSVD component, lie in the subspace orthogonal to the first SSVD direction. In that subspace, the further SSVD components are constructed in a similar fashion. The middle panel of the fifth row shows the residual matrix after removing the first SSVD component, and the right panel plots the second SSVD component. In this case, the residual matrix and the second component will be the same, because of the data are only two dimensional.

Summary of the geometric interpretation

In summary, different centerings have different geometric interpretations. Removing the overall mean translates the point clouds in both the row space and the column space. Usually the next SVD is neither the PCA in the column space, nor the PCA in the row space. Removing the column mean or the row mean translates the point clouds in one space to have the origin as the center. At the same time, the operation projects the points in the other space onto the 45° line. In terms of approximation, these two mean matrices provide two special one dimensional subspace (i.e, rank 1) approximation of the data matrix. Among all the rank 1 subspace approximations, the first SSVD component provides a data driven approximation, which maximizes the sum of squares of the projections (in either/both of these two spaces). The double mean matrix corresponds to another interesting approximation of the original data matrix. In each space, it projects the points onto the 45° line, and then moves them to be through the center of the point cloud. Note that these points form a straight line in both spaces, however, they usually are not in a one dimensional subspace in each space. After removing the double mean, the SVD of the residual matrix is the same as the PCAs of the residual matrix, in both the column and the row spaces. Thus, if there are some interesting features in the residual matrix after removing the double mean. Exploration in one space is enough to find them.

When a data set is being explored, sometimes the context suggests the most appropriate model. Otherwise, we suggest to try all different centerings and decide which one is preferable. Some criteria are considered below: the model should have small RSS, few components and be easy to interpret. In some situations, the aim of the problem and the constraints of the related context should also be considered. Sometimes other considerations might overrule those criteria, as seen in Section 2.4.

2.3.4 Model selection and a Generalized scree plot

In this subsection, a generalized scree plot is proposed as a visual aid to determine the appropriate centering and the number of components. In the context of PCA, the scree plot (Cattell, 1966) (ref. Section 6.1.3 in Jolliffe (2002) for a good introduction) is widely recommended to attempt to determine an "appropriate" number of PC components. One type of scree plot shows $(i, \lambda_i / \sum_j \lambda_j)$ in a plane, where $\lambda_i = s_i^2$, and the s_i's are the singular values of X, when X is column centered. $\lambda_i / \sum_j \lambda_j$ calculates the relative proportion of the variance that is explained by the ith PC. Because λ_i is nonincreasing in terms of i for PCA, the number of components is commonly chosen to be the i where the line "goes rather flat" (the "elbow" point). Note that in some contexts, the log-scale scree plot might convey an entirely different impression.

Here we define the scree plot for the four types of centerings in a novel way, since the mean matrices and their degree of the approximation need to be incorporated into the plot. Let A_k be the approximation matrix of the four types of centerings with rank k ($k+1$ for DSVD), i.e. the sum of the first k SSVD components; or column (/row/double) mean matrix plus the first

$k-1$ CSVD (/RSVD/DSVD) components. R_k is the residual matrix corresponding to A_k. We denote $l_k = \mathrm{SS}(R_k)/\mathrm{TSS}$ (where the Total Sum of Squares (TSS) is $\mathrm{TSS} = \mathrm{SS}(X)$) as the *residual proportion* of the TSS. The plot of (k, l_k) is called the residual proportion plot. It is obvious that l_k is non-increasing, and we can use similar rules (of the usual scree plot) to decide the appropriate number of components.

As noticed in Section 2.3.2, if the approximation matrices of CSVD/RSVD/SSVD have the same rank k, and DSVD has rank $k+1$, we know SSVD is the best rank k approximation, while the CSVD/RSVD has larger RSS than the first k SSVD components, but has smaller RSS than the first $k-1$ SSVD components (Equations (2.4)-(2.11) showed these comparisons). These comparisons also hold in terms of model complexity, where smaller can be replaced by simpler. If we plot (k, l_k) for the SSVD in the integer grid of k, the residual proportion of the RSVD and the CSVD should be in between that of the SSVD with rank k and $k-1$. Thus we use the half-grid $k-1/2$ here to show the approximation performance of the RSVD/CSVD (i.e., plot ($k-1/2$, l_k) for RSVD/CSVD). The DSVD with one larger rank has non-comparable RSS with SSVD, so we plot it at the same level as SSVD.

The resulting plot described above is defined as our *generalized scree plot*, where simpler models are always to the left. The above special treatment (i.e., plotting $(k-1/2, l_k)$ for the CSVD and RSVD models) makes the points in the plot correspond to (the models of) the rectangles in the approximation diagram of Figure 2.4. And the levels from the bottom to the top in Figure 2.4 correspond to the grids on the horizontal axis from the left to the right. Thus, the generalized scree plot simultaneously visualizes the approximation performance and model complexity. In order to choose the appropriate centering and number of components, one possible way to decide the number of components uses the usual interpretation of scree plot. After this, one can select the one which is the leftmost. Zhang (2006b) provides a MATLAB function, `gscreeplot.m`, to generate the generalized scree plot, which allows various options, including log-scale scree plot. Note that the scree plot assumes a strong signal, with large variation relative to the noise component. If this assumption is violated, other considerations should be used to select the optimal model.

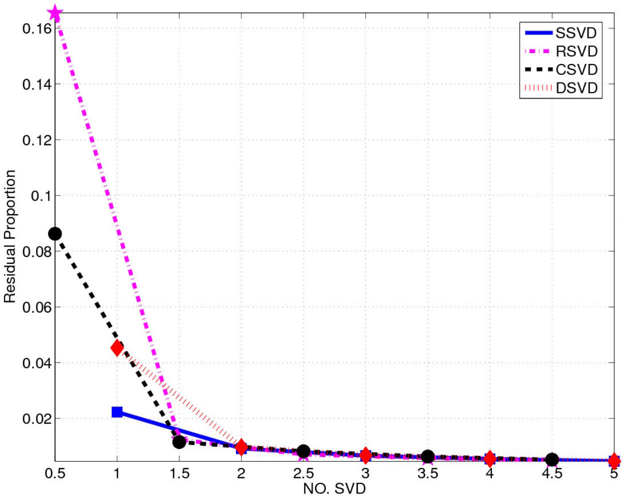

Figure 2.6: *Generalized scree plot for the network traffic data. It shows that all four types of centering use two components to explain the major modes of variation. Thus the RSVD/CSVD model with two components is the best in terms of model complexity and approximation performance. By looking at all the surface plots, we choose SSVD with three components as the model to analyze the network data, because it provides the best interpretability in this context.*

Figure 2.6 shows the generalized scree plot for the network data set in Section 2.2. From the plot, we find that all the models use two components for the major modes of variation, and they have similar approximation performance. In terms of model complexity, we might use either RSVD or CSVD as the final model to find underlying features of the network data set. By looking at the surface plots (the SSVD surface plot is in Figure 2, and the other three can be viewed at Zhang (2006b)) of all four types of centerings, we find the SSVD model with three components provides the best interpretability among the four centerings, which is why this one was chosen in Section

2.2. This is an example of overruling the usual interpretation of the scree plot. However, viewing the generalized scree plot can be a good starting point in terms of model selection. See Section 2.5 for more discussion.

2.4 Generation of the surface plots and the SVD movies

Details of the generation of the surface plots, image plots and two SVD movies (SVD rotation movie and SVD curve movie) are provided below.

Surface plots: The surface plots consist of a set of $k+3$ ($k < \text{rank}(X)$) subplots. The first subplot is the mesh plot for the original data matrix, the next k subplots are the mesh plots for the first k SVD components ($s_i\mathbf{u}_i\mathbf{v}_i^T$), the $(k+2)$nd subplot is the mesh plot for the k-component approximation of the data matrix, i.e. the summation matrix of the first k components. And the last one is the mesh plot for the residual. The surface plots can be generated using the MATLAB function `svd3dplot.m`, which can be obtained from Zhang (2006b).

Image plots: The image plots provide a special view angle of the SVD components. They show the image view of the original data matrix, the first k SVD components, the reconstruction of the k components, and the residual after the approximation. In our design, we let the minimal value of each component share a cool color (blue), and maximum value share a hot color (red). The SVD images show a good view angle to highlight the local variations, data subgroups and interactions between columns and rows. The program `svd3dplot.m` in Zhang (2006b) with option (`'iimage', 1`) generates the image plots. See the demographical data (Section 2.6.3) as an example.

SVD rotation movie: Viewing the surface plots from different angles helps to find data features. Another program, `svdviewbycomp.m` in Zhang (2006b), generates movies for different SVD components with different angles of view. It rotates the surface plot of an SVD component, so that

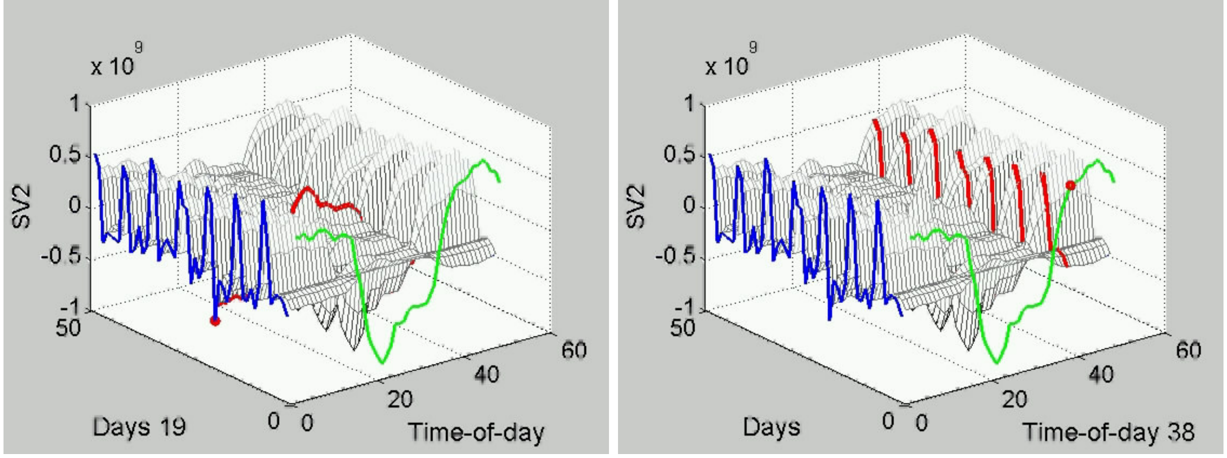

Figure 2.7: *Two snapshots of the SVD curve movie for the second SSVD component of the network traffic data set. (a) shows the 19th row, which corresponds the Friday, June 27. As discussed in the text, it was the late registration day of the UNC summer school. (b) shows the 38th column, which corresponds the (contrast) time series around 19:00 in the evening across days.*

different data features are more clear from different view angles. The SVD rotation movie for the above network data is explained in Section 2.6.1.

SVD curve movie: The SVD curve movie illustrates how the SVD components relate to classical Functional Data Analysis. The curve movie of the ith SVD component $(s_i\mathbf{u}_i\mathbf{v}_i^T)$ is based on the mesh plot of the component. Within each mesh plot, two reference curves with different colors are used to indicate how the surface is generated from the singular row and singular column vectors. The blue curve (in the row direction) is a scaled singular column $(C_{\mathbf{u}_i}\mathbf{u}_i)$, where $c_{\mathbf{u}_i} = \max_{kl}((\mathbf{u}_i\mathbf{v}_i^T)_{kl})/\max_k(|\mathbf{u}_{ik}|)$; and the green curve (in the column direction) is a scaled singular row $(c_{\mathbf{v}_i}\mathbf{v}_i)$, where $c_{\mathbf{v}_i} = \max_{kl}((\mathbf{u}_i\mathbf{v}_i^T)_{kl})/\max_k(|\mathbf{v}_{ik}|)$. The scaled constants are chosen so that the reference curves share a comparable vertical range to the surface. A red line varies on the surface, from the first row to the last and then back to the first row, while a big red dot moves along the singular column (the blue curve) with respect to the red line. Then the red line varies from the first column to the last and then back to the first column, while the corresponding red dot varies along the singular row (the green curve). The motion shows how the curves change in both directions, and highlights features of interest, such as outliers.

Figure 2.7 shows two snapshots of the SVD curve movie for the second SSVD component of the network data set we have discussed in Section 2.2. The left one displays a red line on the 19th row of the second component, which highlights one special day, Friday, June 27. See detailed discussion of this day in Section 2.6.1. The right one visualizes red line on the 38th column, which shows the contrast time series of weekdays and weekends at 19:00 across days. It suggests strong contrast at that time for weekdays and weekends. Two functions `svd3dplot.m` and `svd3dzoomplot.m` in Zhang (2006b) are used to generate the SVD curve movie and a zoomed version, with appropriate options. For large data sets, it is helpful to restrict the range for rows or columns to get a zoomed version of the SVD movie, which demonstrates local features. The chemometrics data set in Section 2.6.2 is used to illustrate the zoomed curve movie.

2.5 Four types of SVD and toy examples

In this section, we use simulated examples to illustrate model selection among the four types of SVD. These examples make it clear that sometimes we do not have a "best" choice. Also for real applications, it is not enough to use only the generalized scree plot to select appropriate models. However, the generalized scree plot is still useful to give an initial impression of which model might be a better candidate. It is also useful when the user does not have time for deep exploration of the data sets. If time permits, we strongly recommend the application of the four types of centerings simultaneously, and the use of some visualization methods, including the matrix views or other information, to select the most interpretable model.

We designed several simulated data sets to illustrate the above idea of model selection. They also show that each of the four types of centerings can be the most appropriate model. In this section, we show four interesting data sets. These simulated data sets are designed as 49×48 matrices, the same as the network traffic data set we discussed in Section 2.2. In this setting, each row can be viewed as one daily usage profile, and each column can be treated as a cross-day times series of one specific time in a day. In addition, these toy examples are designed to have clear weekly patterns. A large number of plots and other simulated examples, similar to those actually shown in this Section, are available at Zhang (2006b).

2.5.1 Example 1

This example is used to illustrate a case that the SSVD can provide the best model among the four centerings. This example is designed to be a multiplicative model (of two vectors, i.e., $f_1(i)g_1(j)$) plus noise. The underlying mathematical model for Example 1 is defined as

$$h_1(i,j) = f_1(i)g_1(j) + \varepsilon(i,j), \tag{2.12}$$

where

$$f_1(i) = \begin{cases} 1, & \text{mod}(i,7) \neq 0 \text{ and } 6, \\ 2, & \text{o.w.} \end{cases}, \quad g_1(j) = \sin\left(\frac{j\pi}{24}\right),$$

and $\varepsilon(i,j) \overset{\text{iid}}{\sim} N(0, .04)$. Note that we will use the same notation ε for different realizations of all the simulated examples. From the model defined in (2.12), it is expected that the SSVD with one component is the best model among the four types of centerings. If the data is thought of as network data usages, all days share the same usage pattern but have different magnitudes.

Figure 2.8 shows the log-scale generalized scree plot for this simulated data set. It suggests that one SSVD component explains the major modes of variations, while the other three centerings use

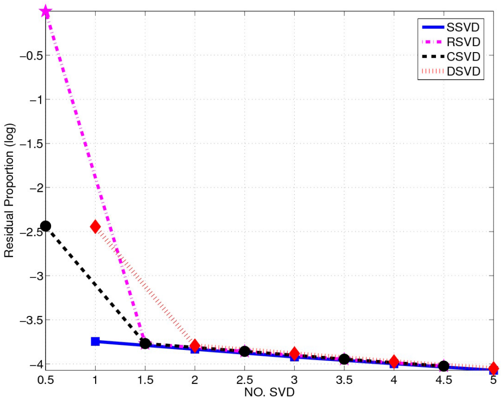

Figure 2.8: *Log-scale generalized scree plot for the first toy example, which is simulated from Equation (2.12). It shows that the SSVD uses one component for the major modes of variations, while the other three centerings use at least two components. The surface plots of the four centerings show that the SSVD has the best interpretability among the four types, illustrated in Figure 2.9.*

at least two components for the major parts. This validates that the SSVD model should be the best among the four centerings in this context.

By comparing the surface plots of the four different centerings (the SSVD surface plots are in Figure 2.9, and all others can be viewed at Zhang (2006b)), we find the SSVD model uses one component to show the weekly pattern and the common daily shape. It also suggests that the major contrast between weekdays and weekends is the daily usage magnitude, while the daily profiles (i.e. the daily usage patterns) stay the same across days. Note that this is the way we designed the toy data set. All other three decompositions show the contrast between the weekdays and weekends. However, they do not directly suggest the fact that weekdays and weekends share the same usage pattern but have different magnitudes.

Figure 2.10 shows the surface plots for the double mean plus one DSVD component. Note that the double mean matrix (the left panel in Figure 2.10) is very similar to the column mean matrix. This is due to the low proportion of TSS explained by the row mean matrix. So the double mean

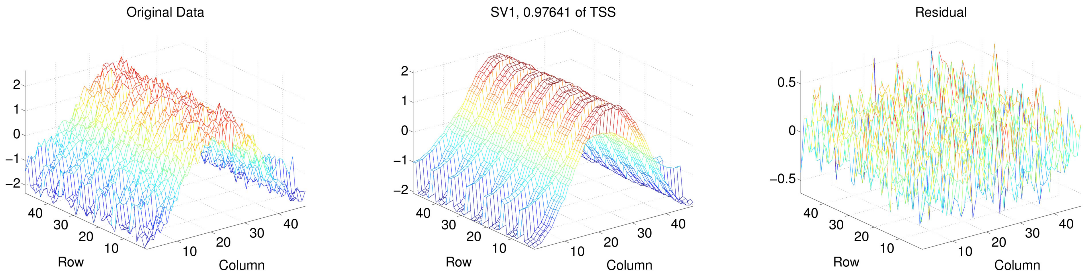

Figure 2.9: *The surface plot of the SSVD for the toy example, which is generated from Equation (2.12). The left panel shows the original simulated data, the middle one shows the first SSVD component, while the right panel is the corresponding residual. It uses one SSVD component to explain the major modes of variations. It suggests that weekdays and weekends share the same usage pattern (i.e., the same daily shape), but have different daily usage magnitudes.*

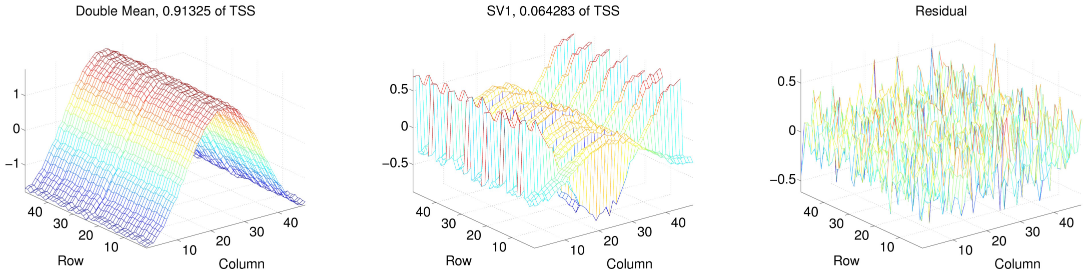

Figure 2.10: *The surface plot of the DSVD model for the first toy example, which is generated from Equation (2.12). The left panel shows the double mean matrix, the middle one shows the first DSVD component, while the right panel is the corresponding residual. The double mean matrix is very similar to the column mean matrix, which shows the common usage pattern for all days. The first DSVD component shows the contrast between weekdays and weekends. We can not tell whether weekdays and weekends share the same usage pattern or not from these plots.*

matrix shows approximately the average usage profile of all days. The first DSVD component (the middle panel in Figure 2.10) shows the contrast between weekdays and weekends. It suggests that the double mean is mostly driven by the weekdays, by showing nearly flat in the first DSVD component. We can not tell whether the weekdays and the weekends share the same usage pattern or not.

Note that this example is designed as one multiplicative component plus some noise. We find that the double mean matrix is significantly different from the first SSVD component. Note that the double mean can be viewed as an additive model, and it is not enough to explain the major modes of variation for this simulated data. So an additive model is not suitable here. We notice that when the data matrix, in terms of point cloud, is far away from the origin in both the column space and the row space, the additive model is similar to the multiplicative model, i.e., the double mean is similar to the first SSVD component (for example, the network data we discussed in Section 2.2. See plots of the DSVD model for the network data set at Zhang (2006b)). However, when the data matrix, when considered as point clouds, are close to the origin in both the column space and the row space, the additive model is significantly different from the multiplicative model (for instance the above simulated data). See Gabriel (1978) for more comparisons between these two models.

From the above discussions, under the setting of the Equation (2.12), SSVD is the best model among all, in terms of model complexity, approximation performance and interpretability.

2.5.2 Example 2

The second example is used to illustrate a situation where CSVD gives the best approximation performance. This example is designed to be the sum of a column mean matrix (i.e. $\mu_c(j)$ in equation (2.13)), a multiplicative component (i.e., $f_2(i)g_2(j)$ in equation (2.13)), and some noise. The model can be written as

$$h_2(i,j) = \mu_c(j) + f_2(i)g_2(j) + \varepsilon(i,j) \qquad (2.13)$$

where

$$\mu_c(j) = \sin\left(\frac{j\pi}{24}\right), \quad g_2(j) = -\cos\left(\frac{j\pi}{24}\right), \quad f_2(i) = \begin{cases} 1, & \text{mod}(i,7) \neq 0 \text{ and } 6, \\ 2, & \text{otherwise} \end{cases}.$$

In terms of network usages, the weekdays and weekends do not have the same usage magnitudes (due to the multiplicative component, $f_2(i)g_2(j)$), nor the same usage patterns (because of the column vector component, $\mu_c(j)$).

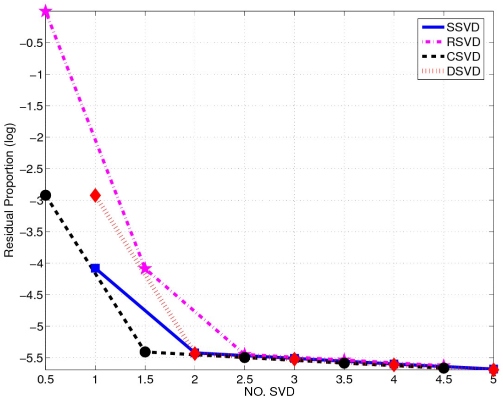

Figure 2.11: *Log-scale generalized scree plot for the second toy example. By using the scree rule mentioned in the text, one is suggested that there are two components for CSVD/SSVD/DSVD, and three components for RSVD. CSVD is the leftmost one, thus it is the appropriate model for the second toy example.*

Figure 2.11 shows a log-scale generalized scree plot for this simulated data set. It suggests that there are 2 components for the CSVD/DSVD/SSVD models, while the RSVD model uses 3 components. The RSVD is the worst model among these four, because the row mean matrix explains a very low proportion of the TSS. It also suggests the CSVD model with two components is the best model among the four types of centerings, in terms of model complexity and approximation performance. We will use the surface plots to compare the interpretabilities of these four centerings.

By examining the surface plots of all four centerings, we find the RSVD and SSVD provide

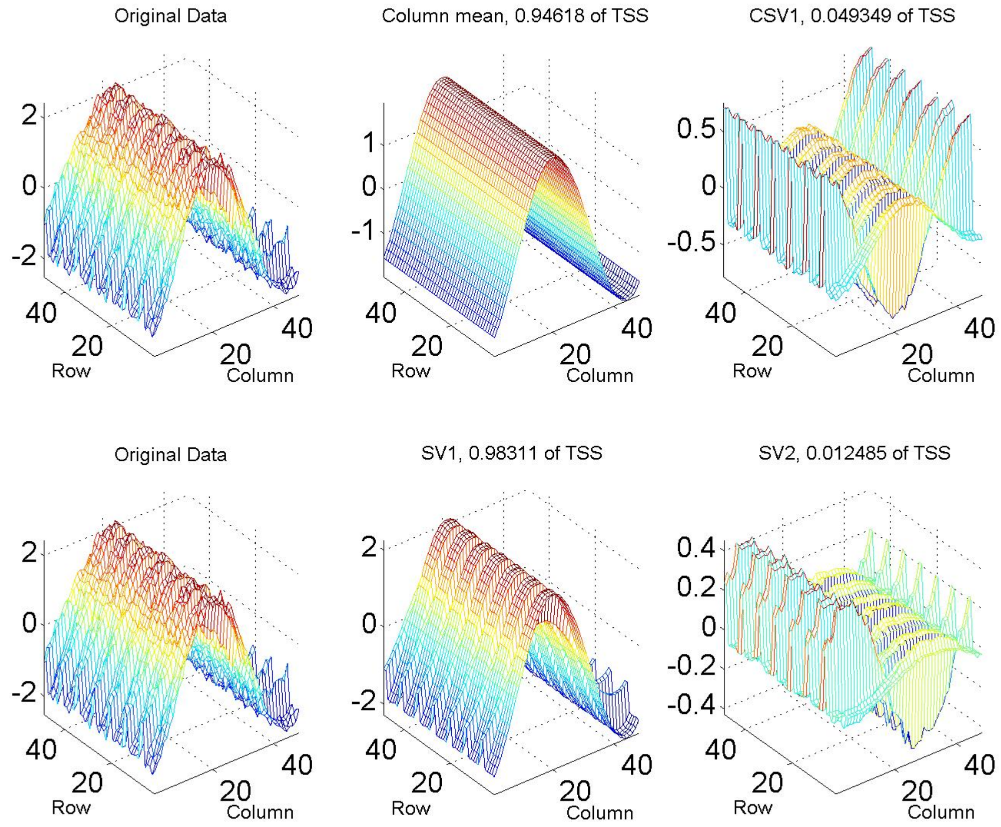

Figure 2.12: *The first row shows the surface plots of the CSVD for model* (2.13), *and the second row provides the surface plots of the SSVD. For this example, the SSVD and the CSVD are candidates of good approximation performance. The CSVD is better than the SSVD, because it contains a simpler model. However, the SSVD shows that weekdays and weekends have different daily shapes and different magnitudes of daily usages.*

similar decomposition, except, the RSVD has an additional row mean matrix. The second row in Figure 2.12 shows the surface plots for the SSVD. See Zhang (2006b) for the RSVD surface plots. Besides, the CSVD model and the DSVD model seem to have the same decomposition as well. The above similarity is due to the fact that the row mean matrix explains a very low proportion of the TSS, as discussed earlier . The surface plots of the CSVD model (the first row in Figure 2.12) show the common usage pattern (i.e., the column mean) in the top middle panel. The first CSVD component (the top right panel) shows the contrast between weekdays and weekends. This also shows that after removing a common daily usage profile, the contrast curves between weekdays and weekends have the same shape, but have different contrast magnitudes. It is hard to answer the question of whether the contrast between them is different in daily shapes or in different usage magnitudes.

Meanwhile, the surface plots for SSVD (the second row in Figure 2.12) use two components for the major variation of the data matrix. If the daily usages share the same usage pattern but with different magnitudes, one SSVD component should be enough for the major modes of variations, as shown in Example 1. Thus, the two SSVD components for this example suggest that weekdays and weekends do have different usage patterns and different magnitudes. This suggests that the SSVD model for this example has better interpretation than the CSVD model.

2.5.3 Example 3

The third example is designed to illustrate that RSVD has the best approximation performance among the four, in a context that is much different from just the transpose of Example 2. The model is set up as the sum of a row mean matrix ($f_3(i)$ in equation (2.14)), a multiplicative component ($f_4(i)g_4(j)$ in equation (2.14)) and noise. It can be written mathematically as

$$h_3(i,j) = f_3(i) + f_4(i)g_4(j) + \varepsilon(i,j), \tag{2.14}$$

where

$$f_3(i) = \cos(\frac{i\pi}{24}), \quad f_4(i) = \sin(\frac{i\pi}{24}), \quad g_4(j) = \begin{cases} 1, & 1 \le j \le 12 \text{ or } 25 \le j \le 36, \\ -1, & 13 \le j \le 24 \text{ or } 37 \le j \le 48. \end{cases}$$

The model of the third toy example is different from the second one in an important way. The function $g_4(j)$ in the multiplicative component is orthogonal to the constant vector $\mathbf{1}_{48\times1}$, and $f_4(i)$ is orthogonal to $f_3(i)$ as well. These make the row mean matrix $f_3(i)$ and the multiplicative component $f_4(i)g_4(j)$ orthogonal to each other in both the column space and the row space. In this example, the rows of the simulated data are not useful models for daily network traffic.

The log-scale generalized scree plot is in Figure 2.13, which shows that the CSVD is the worst model, because of the low proportion of TSS explained by the column mean matrix. The generalized scree plot also suggests that RSVD and SSVD are better models. In terms of approximation performance and model complexity, the generalized scree plot suggests RSVD will be the "optimal" model among the four centerings.

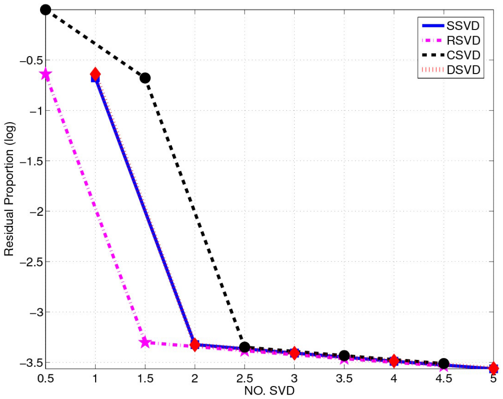

Figure 2.13: *Log-scale generalized scree plot for the third toy example. By using the scree rule mentioned in the text, one is suggested that there are two components for DSVD/RSVD/SSVD, and three components for CSVD. RSVD is the leftmost one, thus it is the most appropriate model for the third toy example in terms of approximation performance and model complexity.*

The first row in Figure 2.14 shows the surface plots of SSVD model with two components. and the second row in Figure 2.14 shows the RSVD surface plots of the simulated data set. By looking at these surface plots, we find that SSVD essentially picks the second part $f_4(i)g_4(j)$ as the first SVD component, and the first part $f_3(i)$ as the second component. In fact, the variation of the

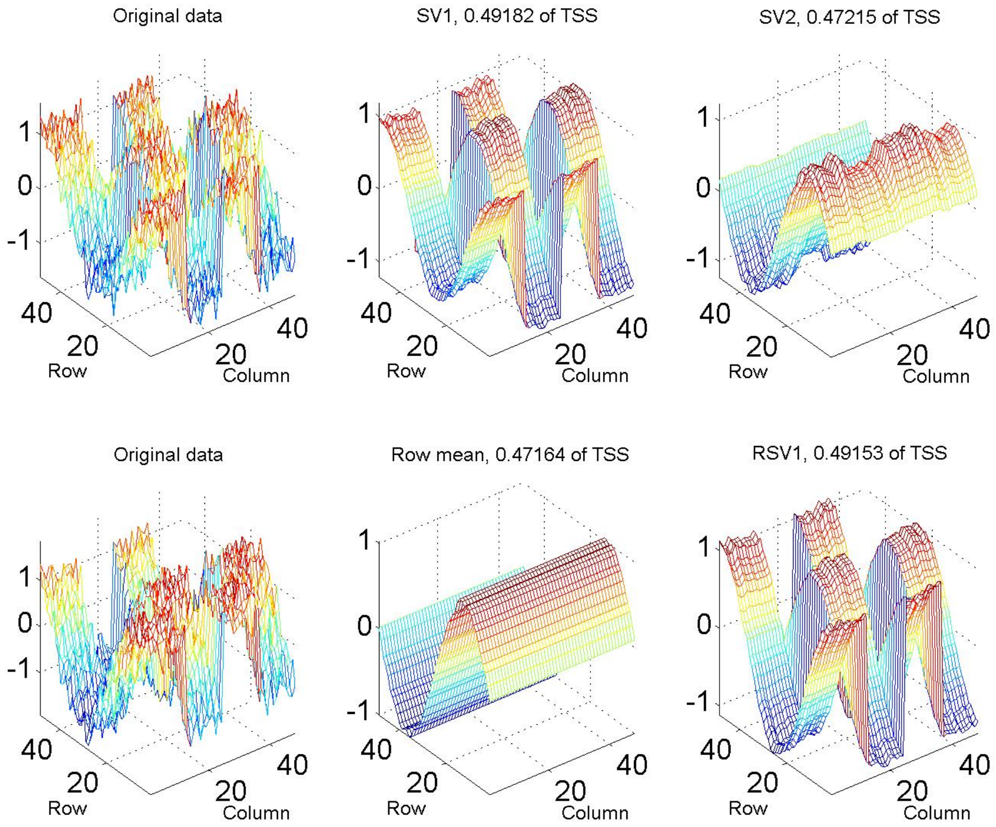

Figure 2.14: *The first row provides the surface plots of the SSVD model; and the second row displays the model of the RSVD model, for Example 3 defined in equation* (2.14)*. The SSVD picks the multiplicative component in the model as the first SVD component. And its second SSVD component is closed to the row mean matrix. The RSVD also picks these two components, but exchanges their order. The reason for this is because they are orthogonal to each other in both the row and the column space, and the multiplicative model has larger variations than the row mean matrix.*

second part $f_4(i)g_4(j)$ is larger than the first part, thus the relative energy (i.e. the proportion of TSS) in the second part dominates the first one, such that the SSVD picks it as the leading SVD component. The RSVD gives the opposite decomposition with the data matrix almost the same as the model described in equation (2.14).

In summary, for this data set, SVD, RSVD and DSVD all provide good separation. We suggest to use the most parsimonious one (the lowest level in the diagram (Figure 2.4)), the RSVD model.

This is close to the true model (2.14). Note that the interpretabilities for the SSVD model and the RSVD model are essentially the same for this example.

2.5.4 Example 4

The fourth example is used to illustrate a case that the CSVD is the best model in terms of model complexity and approximation performance. On the other hand, the worst model in terms of the above two criteria, the RSVD, provides the best model when model interpretability is concerned. This model of this example contains an overall mean level shift (μ) and two multiplicative components. The mathematical description of the model is

$$h_4(i,j) = \mu + f_5(i)g_5(j) + f_6(i)g_6(j) + \varepsilon(i,j), \tag{2.15}$$

where

$$f_5(i) = \begin{cases} 2, & \text{mod}(i,7) \neq 0 \text{ and } 6, \\ 0, & \text{o.w.}, \end{cases} \quad , \quad g_5(j) = \sin(\frac{j\pi}{48}),$$

$$f_6(i) = \begin{cases} 1, & \text{mod}(i,7) = 0 \text{ or } 6, \\ 0, & \text{o.w.}, \end{cases} \quad , \quad g_6(j) = \cos(\frac{j\pi}{48}).$$

Here the two multiplicative components are orthogonal to each other in both column and row spaces. In this example, we use $\mu = 5$, such that all the elements in h_4 are positive.

The generalized scree plot in Figure 2.15 shows that the RSVD uses three components for the major modes of variation, while the other three use two components, which suggest that the RSVD is the "worst" model among the four types of centerings. The CSVD model is the best in terms of model complexity and approximation performance. However, after comparing all the surface

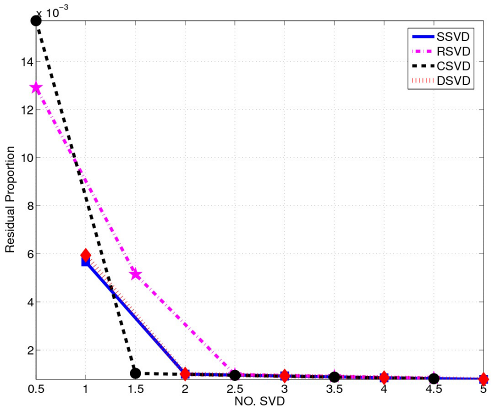

Figure 2.15: *The generalized scree plot for the fourth toy example. By using the scree rule mentioned in the text, one is suggested that there are two components for CSVD/DSVD/SSVD, and three components for RSVD. CSVD is the leftmost one, thus it is the most appropriate model for this toy example in terms of approximation performance and model complexity. However, by looking at the surface plots of the four different centerings, RSVD provides the "optimal" interpretation among them.*

plots (of the four types of decomposition), we find that the RSVD (Figure 2.16), the "worst" model in terms of the approximation and complexity, provides the clearest decomposition using three components. In Figure 2.16, the row mean matrix (the top middle panel) grasps the weekly pattern. The first RSVD component (the top right panel) shows sine curves for the weekdays, and is nearly flat in the weekends. And the second RSVD component (the bottom left panel) picks the cosine curves in the weekends, and stays close to zero for the weekdays. This gives a perfect separation of the curves in model (2.15). For this toy example, we might choose CSVD as the final model, because it is the best model in terms of approximation performance and complexity. On the other hand, RSVD can also be selected as the best choice, because it provides the best separation of the curves, thus it has the best interpretability.

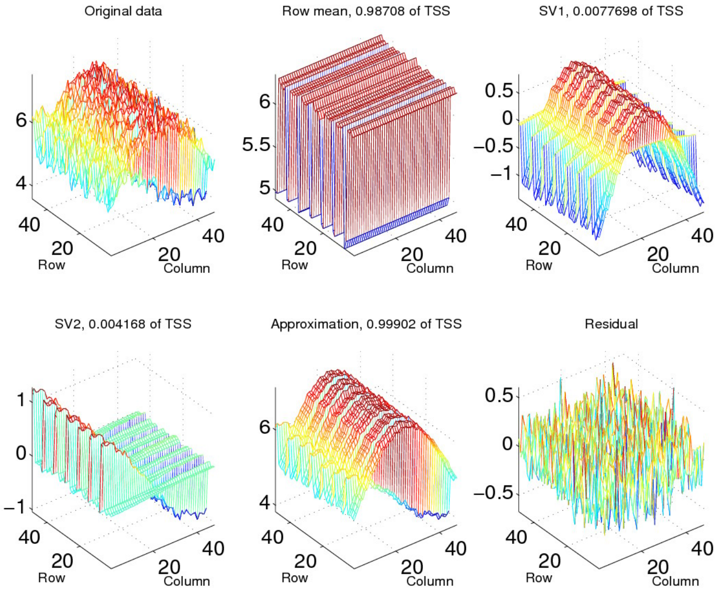

Figure 2.16: *The surface plot of the RSVD for the 4th toy example defined in equation (2.15). The RSVD is the worst model in the sense of the approximation and complexity. But it correctly picks up the weekly pattern (row mean matrix), the shapes of the weekdays (SV1) and the shapes of the weekends (SV2).*

2.6 Real applications

In this section, we further illustrate the utility of our visualization methods for exploration of data. Section 2.6.1 further analyzes the network data mentioned in Section 2.2, and illustrates the usefulness of the SVD rotation movie. The chemometrics data set in Section 2.6.2 is used to show the utilities of zoomed plots and zoomed movies. The image plot is illustrated in Section 2.6.3 to highlight the interactions of age groups and years in a Spanish mortality data set.

2.6.1 Further analysis of the Internet traffic data set

Here we further analyze the network traffic data set in Section 2.2 by using the SVD rotation movie. The surface plot gives an insightful visualization of the SVD components in Section 2.2. Viewing

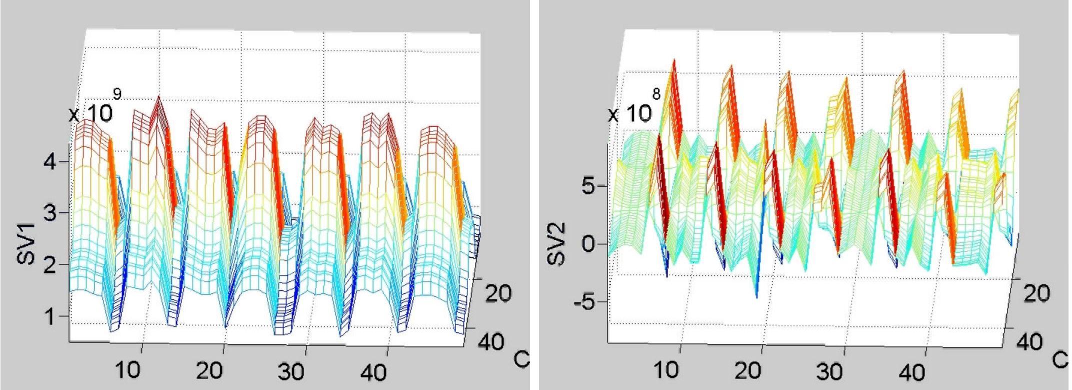

Figure 2.17: *Snapshots of the SSVD rotation movie with carefully chosen angles of view. (a) The first SSVD component of the network traffic data set, showing the 4th weekend (the fourth valley from the left is larger than other valleys) is a long weekend. (In fact, it contains the July 4th holiday) (b) The second SSVD component of the network traffic data set, showing the 19th row is a special weekday (There is one medium bump between the second big bump and the third big bump, which corresponds to this outlying day), which is similar to the weekends but with a smaller bump. (In fact, it is the last registration day of UNC summer school.)*

the surface plot from different angles will help to highlight different interesting features. Here we recommend viewing the full SVD rotation movie for the SVD components. The left panel in Figure 2.17 shows a carefully chosen snapshot of the rotation movie for the first SSVD component. A careful examination shows that besides the typical weekly pattern, the fourth weekend seems to be a long weekend, and the third weekend is kind of short. In fact, Friday July 4 makes the fourth weekend special, and Sunday June 29 is in the third weekend, which has been discussed in Section 2.2.

The right panel in Figure 2.17 is a carefully selected snapshot of the rotation movie for the second SVD component. We find that the 19th row of the second SVD is unusual because although it is a weekday, it looks more like a weekend, but with a smaller bump. The 19th row was Friday June 27, the last day for late registration of the UNC summer school second session. Checking the original data set, we found that the network usage on that morning oscillates a lot, which might be explained by bursts of usage after the end of each class time, followed by rapid departure of the students.

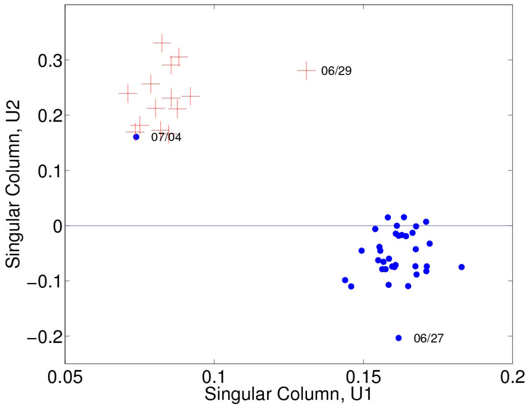

Figure 2.18: *Scatter plots between singular columns* $\mathbf{u}_1$ *vs.* $\mathbf{u}_2$ *for the Internet data, which is useful in detection of outlying days.*

The above, along with the discussion in Section 2.2, shows that the SVD rotation movie and SVD curve movie are useful for outlier identification. Note that the above outlying days can also be identified by other visualization methods. For example, scatter plots between singular rows can be very helpful in finding outlying days. Figure 2.18 shows the scatter plot between $\mathbf{u}_1$ and $\mathbf{u}_2$. From the scatter plot we find two groups of points. Investigation suggests that they correspond to the weekdays and weekends, as indicated using dots (for weekdays) and plus (for weekends) respectively. Friday, July 4, was a holiday, and it falls into the weekend group, which is not surprising because on that day most students and school employees were away. Sunday, June 29, is between those two groups, for reasons discussed above. Friday, July 27, might also be found through this scatter plot, with an unusually low $\mathbf{u}_2$ value. But the confidence here might not be high, because this point is not that far away from the weekday data points. Scatter plot of $\mathbf{u}_1$ and $\mathbf{u}_3$ can also show some outlying days, which is skipped because no new outliers were found.

For this data set, although the scatter plots among singular columns are useful to find clusters

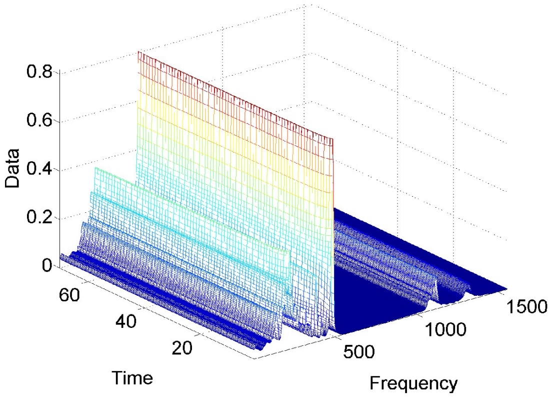

Figure 2.19: *Mesh plot of 70 spectra of the Chemometrics data.*

of days and some outlying days, a limitation is that they cannot provide a functional view of the daily shapes and cannot directly show what drives those days to be outliers. When the user uses SVD or PCA methods for a functional data set, we strongly recommend viewing surface plots and the two types of SVD movies during the exploration.

2.6.2 A Chemometrics data set

The chemometrics data considered here consist of 70 Infrared (IR) spectra of various samples of a polymeric material measured over a 27-day cooling period. Each IR spectrum has 1556 measurements representing integer "frequency numbers". Hence, the data matrix is 70×1556. More details about this data can be found in Marron et al. (2004), which examined it using PCA. The IR spectra ideally can be expressed as a sum of "real" spectra, whose relative intensities change over time. An important objective for studying the IR spectra is to find the underlying chemical process, which is an open problem. In this subsection, we use CSVD to analyze these data, which is suggested by the generalized scree plot. Figure 2.19 shows the mesh plot for the 70 IR spectra. The mesh plot in the time direction is rather flat, which indicates that the variation between spectra is significantly

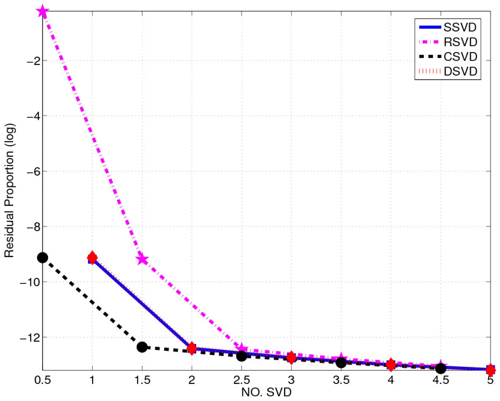

Figure 2.20: *Log-Scale generalized scree plot for the Chemometrics data. It suggests that the CSVD/DSVD/SSVD uses two components for major modes of variations, while the RSVD uses three components. In terms of model complexity, CSVD is recommended, which is exactly the functional PCA method used in Marron et al. (2004).*

smaller than the mean spectrum, and that most information is hidden in the deviations from the mean.

Figure 2.20 shows the log-scale generalized scree plot. In fact, both the original scale and the log scale scree plots suggest the same model. We use log-scale one here for a better visual effect. The plot shows that the CSVD/DSVD/SSVD uses two components for the major modes of variations, while the RSVD needs to use three components. In terms of model complexity, a CSVD model is recommended. This is essentially the functional PCA in Marron et al. (2004). The visualization tools discussed in Section 2.4 are used to explore special features of the data in the spectral direction, which may offer some clue for addressing that open problem.

The column mean matrix explains almost all (99.99%) of the total SS, and we need to examine more CSVD components for information on the underlying chemical process. The first CSVD component (Figure 2.21(a)) explains 96.05% of the residual after removing the column mean. Its surface plot shows some interesting features at the spectra range 300-500, which is particularly

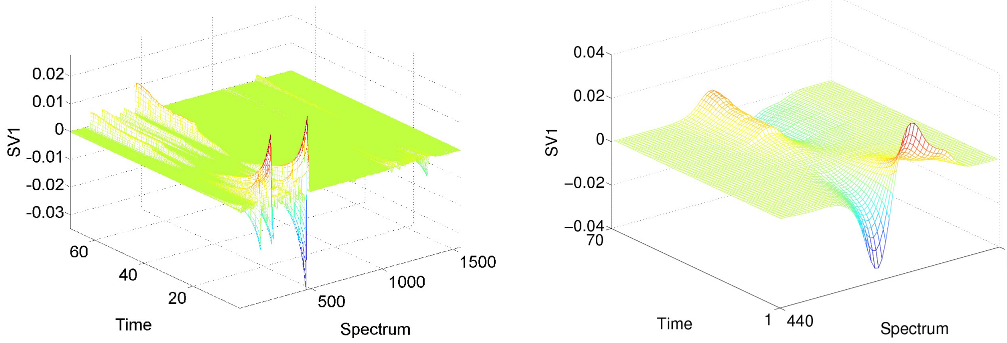

Figure 2.21: *(a) First CSVD component mesh plot for chemometrics data; (b) Zoomed First CSVD component (440-500 spectra) mesh plot.*

interesting from a chemical perspective. At some frequencies the curves change from bottom to top, an indication of an increase of the percentage of chemical materials with high spectrum at those frequencies during the cooling process, or a decrease of chemical materials with low spectrum. At some other frequencies the opposite occurs. We can highlight those frequencies by zooming in on the surface plot of the first CSVD component to the frequencies [300, 500]. Figure 2.21(b) shows the zoomed surface plot at the frequency range from 440 to 500. The zoomed SVD curve movie for the first CSVD component can be used to view the local changing patterns.

The zoomed curve movie of the first CSVD component for frequencies between 440-500 is available at Zhang (2006b). Figure 2.22 shows some snapshots from this movie. For instance, we find that at 467, the spectrum varies from lower (than average) to higher (than average); and at 476, the spectrum varies from higher (than average) to lower (than average). This means that the chemical material corresponding to the frequency 467 changes from below the average to above the average. Similar peaks representing shifts in material types can also be found in other movies in Zhang (2006b).

Besides highlighting signals in the spectral direction, the zoomed SVD movie also helps to

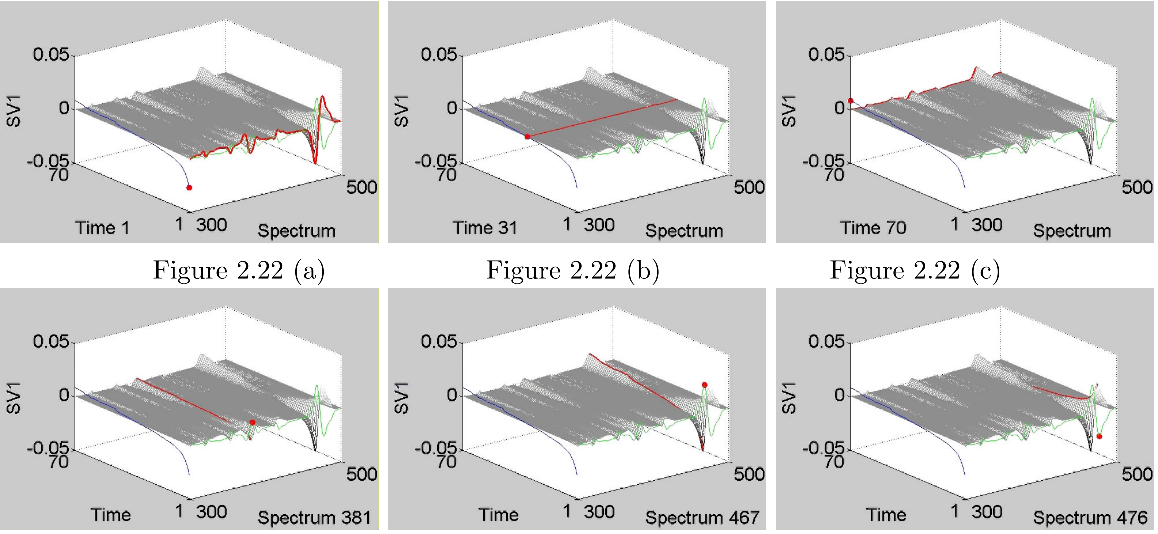

Figure 2.22 (a) Figure 2.22 (b) Figure 2.22 (c)

Figure 2.22 (d) Figure 2.22 (e) Figure 2.22 (f)

Figure 2.22: *Snapshots of the zoomed version of the SVD curve movie of the first CSVD component for the Chemometrics data. The first row shows changing pattern in the time direction, while the second row highlights some significant frequencies in the spectra direction.*

understand the time-varying features. The three plots of the second row in Figure 2.22 show some snapshots of changing patterns in the time direction. The curve changes from a clear shape to a rather flat line and then to a flipped version of the clear shape. This phenomenon shows that the change is smooth with respect to time at all points in the spectra, which is also noted by using colored curves in Marron et al. (2004).

The higher order CSVD components also show some interesting features at frequencies less than 500. The zoomed SVD surface plots and the zoomed curve movies highlight those features. The analysis is skipped here to save space, and the pictures are available at Zhang (2006b).

2.6.3 A Demographical data set

Understanding mortality is an important research problem in demography. The following analysis is trying to understand the variation of Spanish mortality among different age groups and across

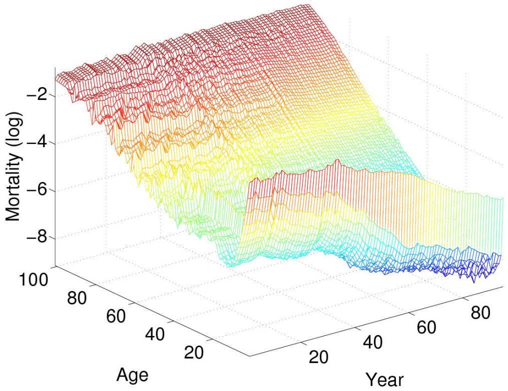

Figure 2.23: *Mesh plot for the Spain mortality data set (after the logarithm transformation). There is a general decreasing trend across years for all the age groups. Also notice that the mortality of younger age groups decrease more significantly than that of older people.*

the years, following a first functional data analysis by Dr. Andrés M. Alonso Fernández. The Spanish mortality data set (after a logarithm transformation) used here was collected such that each row represents an age group from 0 to 110, and each column represents a year between 1908 and 2002 (HMD, 2005). We can view each column vector as a mortality curve of different age groups; or each row vector as a time series of mortality for a given age group. A mesh plot of this mortality data is in Figure 2.23. In the year direction, we find the mortality decreases when the year increases. It seems likely that improvements in medical care and in life quality made this happen. We also notice younger people benefit more than older people, shown as larger decreasing magnitudes in the surface plot. In order to understand the variation from 1908 to 2002 (95 years), and also the variation of mortality among different age groups, a natural choice for the analysis is the SSVD method. The generalized scree plot (shown in Figure 2.24) for this data set, shows that the SSVD and the DSVD have similar approximation performance and complexity. However, the image plots for them show that the SSVD model has better interpretability, as discussed below. Note that the logarithm transformation is monotone, so the variation information will be kept after

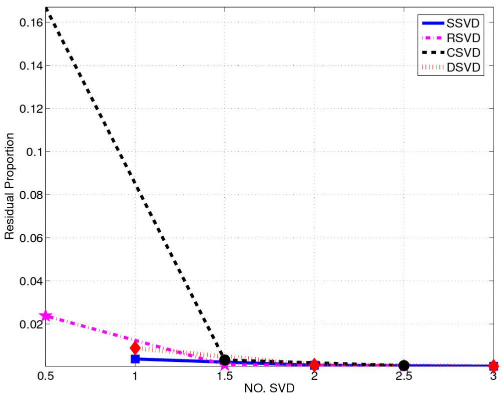

Figure 2.24: *Generalized scree plot for the Spanish mortality data. It suggests that SSVD and DSVD are better models in terms of approximation performance and model complexity. By comparing the surface plots of these two types, SSVD model provides better interpretability, as shown in the surface plot in Figure 2.25 and the image plot in Figure 2.26.*

the transformation.

The surface plots of all four types of centerings for this data set can be found at Zhang (2006b). From the surface plot of the SSVD model, as shown in Figure 2.25, we find the first SSVD (the upper left panel in Figure 2.25) is a smoothed version of the original data. From it, we find the shape of mortality curve (at different ages) remains mostly the same, but there is a decreasing trend over time. There are two medium bumps in the year direction. Checking the world history, we find these two outlying periods correspond to the 1918 world influenza pandemic, and the 1936-1939 Spanish civil war, which killed millions of people (in unusual age distribution) in Spain. The second component (the upper right panel in Figure 2.25) highlights an infant effect, because all the other age groups look relatively flat in the surface. This shows that the mortality of infants decreases significantly (compared to the average). The third component (the lower left panel in Figure 2.25) seems to give an interesting clustering of the data. For example, in the age direction, there are four prominent groups, over 45, from 18 to 45, from 3 to 18 and lower than 3. In addition, in the

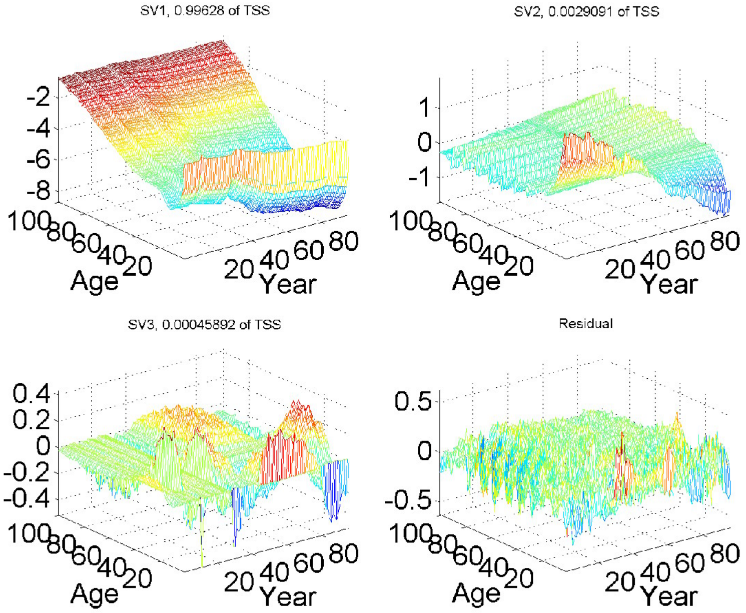

Figure 2.25: *The surface plots for the SSVD of the Spanish mortality data set. The first SSVD components is essentially the smoothed version of the original data set. It shows the general decreasing trend for the same age groups, and the general contrast of different age groups. The second SSVD component shows the infant effect by showing large bump at the corresponding age group. The third SSVD component shows several blocks in the surface, which suggest some grouping information both in the age direction and in the year direction, which is more straightforward in the image plot, as shown in Figure 2.26.*

year direction, there are four major groups, which are from 1908-1935, 1936-1950, 1951-1985 and 1985-2002. For these year groups, the first three years in 1936-1950, as the local larger bumps in the surface, corresponds to the Spanish civil war. The year group, from 1985 to 2002, corresponds to the increasing modern traffic fatalities. There is also a large bump in the year group 1908-1935, which corresponds to the 1918 influenza pandemic. The surface plot for the third component shows several blocks in the matrix, which suggests that the mortalities of different age groups were affected differently by the special events. For example, the 1918 influenza pandemic, civil war and the modern traffic fatalities affected younger people more than other age groups, shown as the higher blocks in the picture.

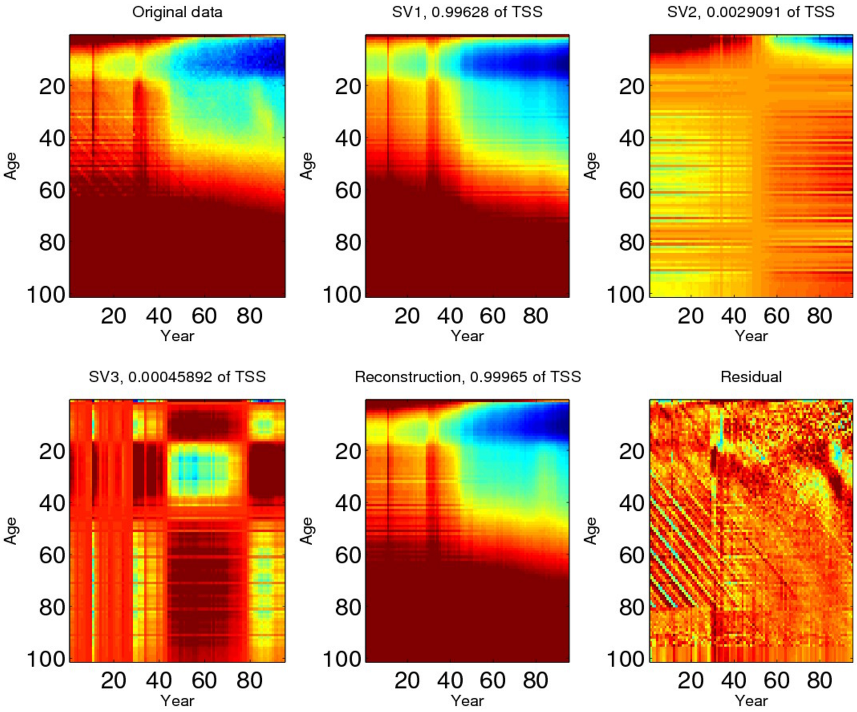

Figure 2.26: *The image plot of the SSVD for the Spain mortality data (after a logarithm transformation). The firs three components provides similar underlying features as shown in Figure 2.25. However, the local variations here are more straightforward than the surface plot. The residual part shows blue diagonals, which correspond cohort effects. One possible reason for this is the rounding error when collecting data in the early 20th century.*

In order to highlight the local variation of each SSVD component, we look at the surface plots from a special angle, directly above the surface plots, and indicate the height of the surface in color. This leads to the image view of the matrices. The image plot for the mortality data set is in Figure 2.26. As discussed in Section 2.4, we assign the maximum value of each matrix to the color red, and the minimum value to the color blue. The image view of the matrix reveals relative variation within each component.

From Figure 2.26, we find that the image view of the first SVD component, reveals the decreasing trend across years. It also suggests that the decreases among younger people is more significant than older people. In addition, this image highlights the above two outlying periods in years, by showing two dark red regions in the years 1918 and 1936-1945. Note that these two outlying periods are more apparent in the image plot than in the surface plot.

The second component in the image view is almost the same as the surface plots, showing the infant effect discussed above. The third component in the image view reveals the grouping information for the ages or possibly also for the years. The interpretation of this is similar to the surface plots, and is not reported here to save space, although the image view is more straightforward than the surface plots. The image view of the residual highlights some interesting features, which might not be seen from other visualization methods. Instead of appearing to show only noise as the surface plot seems to do, the image view of the residual has some blue diagonal lines in the left bottom area. These diagonal lines turn out to be the same population group (i.e., *cohort*) among the years. The blue lines are evenly distributed among the age groups, are 10 years apart, and all end at around 80 years age of old. This strange behavior exists for about 40 years, i.e., until 1958. A very likely reason for this is the rounding errors in the early years. This believed to be due to imprecise death record, e.g., if somebody died at age 39, the age at death was sometimes appeared as 40 in the record. It is clear that the image plot highlights this cohort effect (the interaction between age groups and years), which is harder to find using other existing PCA visualization methods.

2.7 Discussion

For noisy data, the SVD components can contain a lot of noise. Both of the two reviewers of *Journal of Computational and Graphical Statistics* suggest incorporating some smoothing techniques to reduce the noise in the component matrices. There is a lot of discussion between pre-smoothing before applying FDA methods or applying smoothing-incorporated FDA methods, for example, see Chapter 10 of Ramsay and Silverman (2005). For our functional SVD methods, it is possible to pre-smooth the data matrix, and then apply our programs to gain additional insights. Users can use their own favorite smoothers in the pre-smoothing step. We are working on incorporating some regularized SVD methods (for example, Huang et al. (2007)) into our program, which gives a more natural incorporation of smoothing.

One reviewer of *Journal of Computational and Graphical Statistics* observed that our notion of the two-way data matrix requires that the data are observed on common grid points. In cases with irregularly observed data points or missing data, one could preprocess the data using some smoothing or interpolation methods to obtain regular observed data matrices, before applying our visualization methods. However, we did not proceed in this Chapter, because our data sets are not irregular and do not have any missing values. We plan to add options for interpolation into our program after my graduation, such that the program can be directly applied to irregular-grid data matrices, and also to matrices containing missing values.

Chapter 3

Anomaly Detection for Time Series with Long Range Dependence

In the context of Internet traffic anomaly detection, we will show that some outliers in a time series can be difficult to detect at one scale while they are easy to find at another scale. In this chapter, we explore an outlier detection method for a time series with long range dependence, and conclude that testing outliers at multiple time scales helps to reveal them. We propose a MultiResolution Anomaly Detection (MRAD) procedure for time series with long range dependence. In addition, several theoretical properties of the proposed MRAD procedure are proven, especially when outliers appear as a slight local mean level shift with a rather long duration, e.g., as generated by a port scan, which will be discussed in Section 3.6. A novel MRAD outlier map is proposed to visualize the location of the outliers, and also to suggest the significance probabilities (p values) for them.

3.1 Introduction

Internet intrusions, such as Distributed Denial of Service (DDoS) attacks, have become a large problem on the Internet. Available network intrusion detection methods are classified in a useful way in the course of Jeffay (2005). These include many different classes: signature-based detection (e.g. Bro (Paxson, 1999) or Snort (Roesch, 1999)), signal processing based methods (e.g. Barford et al. (2002)), multivariate data analysis methods (e.g. Lakhina et al. (2004b)), data mining (e.g. Lee et al. (1999); Stolfo et al. (2001)) etc. One of the common approaches for detection of network intrusions is to view them as types of network anomalies. See for example McHugh (2001) for a good summary of intrusions and related detection methods.

Anomaly detection methods classify network traffic into two major groups, the “normal” traffic and the anomalies. The anomalies are possible candidates for network intrusions. This type of separation is similar to outlier detection in statistics, where anomalies correspond to statistical outliers, and the data of normal traffic are regular observations.

Outlier detection is a classical topic in Statistics. See for example Hawkins (1980), Barnett and Lewis (1994) for good overview. Internet measurements collected at a single location, for example the packet counts per time interval, form time series with Long Range Dependence (LRD) (Leland et al., 1994) and Self-Similarity (SS) (Willinger et al., 1997). In the time series context, Fox (1972) introduced the notions of additive outlier and innovation outlier. Detection methods include the intervention analysis method, which was first introduced by Box and Tiao (1975). See also in Tiao and Tsay (1983), Martin and Yohai (1986), Chang et al. (1988), Tsay (1988), etc. The detection methods in the literature usually assume the observations are independent or short range dependent (i.e., the autocorrelation function of the time series decays exponentially as the lag goes to infinity). For detecting outliers in LRD time series, these methods may not be suitable, because the artifacts that are naturally generated by the LRD may cause these methods to flag more false alarms. Another characteristic of network traffic data is that different types of network anomalies show statistical abnormal signals in different time scales (Barford et al., 2002).

The multiscale property of the normal traffic and the anomalies motivate us to find detection methods which effectively use these multiresolution characteristics. In this chapter, we propose a MultiResolution Anomaly Detection (MRAD) method for time series with long range dependence, and present a corresponding visualization tool, the MRAD outlier map, which can highlight outliers in different time scales. For theoretical development, we look at a special case where outliers are in the form of a local mean level shift with an unknown duration.

A critical issue for network intrusion detection is that the detection method should be implemented as a real time algorithm, and the test statistic can be updated iteratively. Thus it is possible to flag and stop the current Distributed Denial of Service (DDoS) attacks in or close to real time. One of the proposed MRAD procedures, which is based on sliding window aggregation, can be easily adapted to be an online detection approach, as discussed in Section 3.3.

The remaining part of this chapter is arranged as follows. A motivation example is discussed in Section 3.2, to illustrate the idea of the MRAD method. Section 3.3 discusses two aggregation methods, which are used in this chapter. Section 3.4 reviews fractional Gaussian noise, and some results in extreme value theory for stationary processes. Hypothesis test problem for the MRAD method is formulated, and several important theoretical properties are discussed in this section as well. Section 3.5 develops an improved test threshold for this MRAD method. Section 3.6 uses several simulated data sets, and some semi-experiments to evaluate the usefulness of the MRAD method. Further analysis of the motivating example, and variations of the MRAD outlier maps are discussed as well in this section. Section 3.7 summarizes our work and gives related discussion. All theoretical justifications are reported in Section 3.8.

3.2 Motivating example and the MRAD map

In this section, we use one real network trace to illustrate the idea of our MRAD method. Figure 3.1 shows one time series of byte counts (per $10ms$), which was collected at the main Internet link between the UNC campus and outside. The data were collected on Wednesday April 10, 2002, starting from 1:00 pm, with a 2 hours duration. This time series has been studied by the UNC Internet Data Study Group (Le and Hernández-Campos, 2004). The estimated Hurst parameters (see details of Hurst parameter in Section 3.4.1), mostly are larger than 0.9 by using different estimation methods, i.e., this series does have LRD. A quick look at the time series reveals that there might be network anomalies at a number of locations. Some have a very short duration (e.g. the single spike locates around 5.8 ($\times 10^5$)), while others may last for a few minutes (e.g. the medium bump cluster near 3 ($\times 10^5$)).

As discussed in Section 3.1, the multiscale properties of "normal" network traces and network anomalies motivate our MRAD method. An MRAD method for a time series (at the finest scale) has the following steps.

1. Form time series at different scales.

2. Test whether the observations at each time and scale are outliers or not.

3. Report or visualize the testing results.

Section 3.3 describes the method of forming time series at different scales in detail. Discussion of the tests based on these aggregation methods is in Section 3.4. Here assume that the (standardized) time series at different scales have been generated, and the tests are performed at each time location and each scale. Figure 3.2 shows a novel visualization to report the test results, the MRAD outlier map. For this particular data set, the multiscale time series are generated by non-overlapping

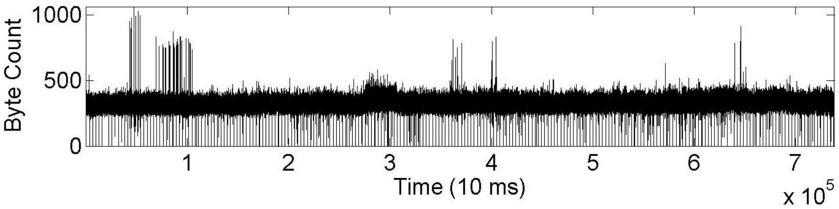

Figure 3.1: *Original byte count time series. Some anomalies may exist in this time series, by showing huge spikes or clusters of medium level shift.*

window aggregation (see Section 3.3 for details). The horizontal axis corresponds to the time locations, and the vertical axis shows different scales. In this plot, finer scales are in the top region, while coarser scales are at the bottom. Detailed definition of these scales are described in Section 3.3. The map visualizes the significance probability (p value) simultaneously over scale and time location. Note that under suitable assumptions (see Section 3.4.5 for details), the marginal distribution of each pixel (in the map) is the same, when there is no outlier at all in the whole time series. We use hotter colors (red) to show small p values, i.e., they correspond to higher chance to be outliers. And cooler colors (blue) are used to display large p values, i.e., they are less likely to be anomalies.

From this plot, we find several important outlying regions. For example, around the location $3(\times 10^5)$, and the scale region $5-15$, there is one red zone, which suggests that there are outliers in the region between $(2.8(\times 10^5), 3.1(\times 10^5))$. This group of anomalies might be directly detected from the original time series, as discussed earlier, but it is not easy to visually distinguished signal from noise. In addition, around the locations $6.2(\times 10^5)$ or $6.8(\times 10^5)$ and the scale 15, there are some orange zones, which also suggest outliers in these regions. Other anomalies might not be obvious in this map, since the display resolution is too small compared to the number of locations (about 7×10^5 observations in this time series) in the series. Zoomed version of this outlier map or a

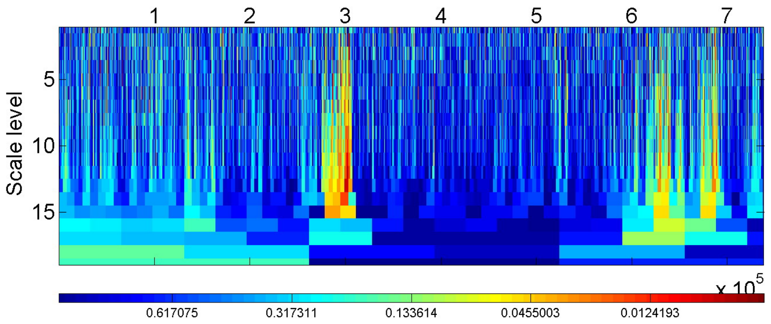

Figure 3.2: *The MRAD outlier map based on the non-overlapping window aggregation method. The relative hot colors correspond to small p values, which suggests a high chance to be outliers. The relative cool colors stand for large p values, which mean a low possibility to be anomalies.*

dynamic sliding outlier movie displays the outlier map locally to overcome this resolution problem, see Section 3.6.6 for more discussion.

The above outlier map visualizes the significance probability of all time locations at all scales. It is natural to focus on those locations and scales that are more likely to be outliers. This motivates the following thresholded outlier map. Figure 3.3 shows the thresholded outlier map for the same data set. Here we use color red to highlight those locations, where their corresponding p values are smaller than $2(1 - \Phi^{-1}(3))$ (the absolute values of the normalized observations are larger than 3), i.e. they are more likely to be network anomalies. Figure 3.3 highlights those outlying regions shown in Figure 3.2, and they are more obvious than the former map. Note that, the users are required to select their own threshold. A MATLAB program, `mradvisual.m`, which is available at Zhang (2006a), provide a default asymptotic threshold (see Section 3.4.5 for details), and allows alternative thresholds.

Note that the above maps do not rely on the way the multiscale time series are formed. For any multiscale outlier detection method, these map can be viewed as a general way to display the test results. However, the interpretation of these outlier maps may be different for various multiscale aggregation methods. Another outlier map based on sliding window aggregation is discussed in

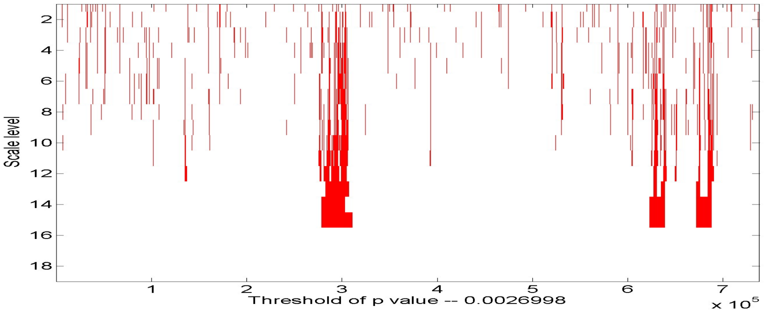

Figure 3.3: *Thresholded MRAD outlier map for the byte count data. Here we highlight the p values which are smaller than* $2(1-\Phi^{-1}(3))$ *(i.e.,* $|Y_L(i)| \geq 3$*).*

Section 3.3.3.

3.3 Two types of aggregation

There are many methods to form time series at different scales, including simple aggregation, the kernel method, the wavelet method, etc. In this chapter, we use simple aggregation to illustrate the MRAD method, because of its tractability for theoretical analysis (details of the theoretical properties are discussed in Section 3.4.5). In this section, two types of general aggregation method will be introduced. One is called Non-Overlapping Window Aggregation (NOWA), and the other one is Sliding Window Aggregation (SWA).

In this chapter, we will use a dyadic-like structure to construct multiscale time series, so that the window sizes with respect to different scales increase exponentially. This treatment is used to avoid over-using scales (i.e, to avoid viewing too many scales). The size of aggregation window for scale k is 2^{k-1}. In other words, the observation at scale k is a function of consecutive 2^{k-1} observations at scale 1. This dyadic structure is motivated by the Haar Wavelet bases (see an introduction of Haar Wavelet in Ogden (1997) or Vidakovic (1999)). Note that the base 2 can be replaced by any positive integer. A general definition of these aggregations based on general bases is given in Section 3.4.4. In the following text, without specification, the aggregation base is always 2.

The classical outlier detection methods in time series usually use the whole time series to estimate the model, and then perform hypothesis tests on the residuals. To test whether the ith observation is an outlier, the usual methods, such as the intervention models in Tsay (1988), exploit the future information after the time i. This is common when outlier detection is performed when all the data are available (from an experiment). The following non-overlapping window aggregation is within this type. However, the realtime detection algorithm cannot use the future observations. In this section, we also develop a one-side sliding window aggregation in this purpose. After selecting appropriate aggregation constants, we will find the aggregation vectors at some particular time locations based on these two aggregation methods are the same (see details in the Sections 3.3.1 and 3.3.2). However, the outlier map from different aggregation has a different interpretation, as discussed in Section 3.3.3.

3.3.1 Non-Overlapping Window Aggregation

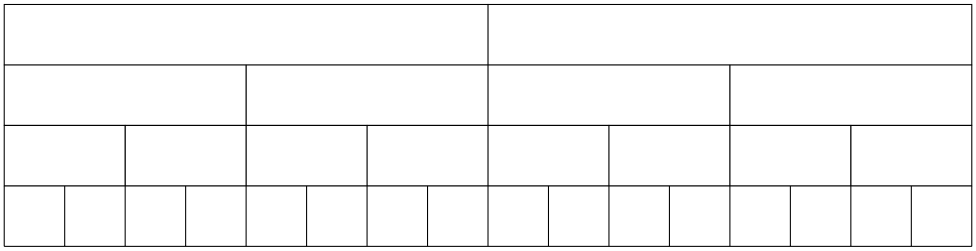

Figure 3.4: *The idea of non-overlapping window aggregation. Every two observations at scale k are aggregated to form one data point in a relatively coarser scale (scale $k+1$). Note that lower rows are finer scales in this diagram.*

Figure 3.4 shows the idea of Non-Overlapping Window Aggregation (NOWA), which is a natural way to aggregate a given time series over a range of scales. Let us assume the bottom row in Figure

3.4 is the observed time series (in the finest scale, or scale 1). To form a relatively larger (coarser) scale time series (scale 2), we can take the sum of every pair observations (without overlap) to form one data point in the new scale, as displayed in the second lowest row in the diagram. Similarly, we can sum every 4 ($= 2^{3-1}$) observations of the bottom row to form an even larger scale time series (scale 3), which is shown in the second row from the top in the diagram. Note that the scale 3 aggregation (the second row from the top) can also be viewed as the sum of every 2 observations at scale 2 (the second bottom row). It is straightforward to show that the window size (of observations) increases at a exponential rate as the scale increases. After aggregation, a constant is usually chosen to normalize each time series, such that all the values in these cells are comparable. In fact, under suitable conditions, these cells are identical distributed. See detailed discussion is in Section 3.4.5.

We can define the above procedure explicitly using mathematical notation. Let $Y_1(i)$ ($i = 1, \cdots, N$) be a time series at the finest scale. Let $Y_k(i)$, $i = 1, \cdots, \lceil N/2^{k-1} \rceil$ be the corresponding k-scale time series, where $\lceil x \rceil$ returns the smallest integer which is greater than or equals x. The above diagram shows how to aggregate the time series from scale k to scale $k+1$ as

$$Y_{k+1}(i) = w_1 Y_k(2i-1) + w_2 Y_k(2i),$$

where w_1 and w_2 are chosen in order to normalize the series at scale $k+1$. This formula shows that for NOWA, one observation at scale $k+1$ is a scaled weighted average of two observations at scale k. In terms of the finest scale, the definition will be

$$Y_k(i) = \sum_{j=1}^{2^{k-1}} \widetilde{w}_k(j) Y_1((i-1)2^k + j),$$

where $\{\widetilde{w}_k(j)\}$ are chosen for the same purpose. In this chapter, $w_1 = w_2 = 1/2^H$, $\widetilde{w}_k(j) = 1/L^H$, where H is the Hurst parameter (see Section 3.4.1 for details) and $L = 2^{k-1}$. These (equal) weights make the marginal distribution of each cell in Figure 3.4 is the same, if there is no outlier in the original time series. From the above definition, we find that one observation $Y_1(i)$ on the finest scale, can only be used once to form the time series at scale k. To form observations at that scale, the finest scale time series are divided by non-overlapping sections, where each section has length (/duration/window size) 2^{k-1}. The observations at scale k are a function of the observations within each section. That is why we call this type of aggregation as non-overlapping window aggregation. Note that the approximation component of most (discrete) wavelet methods can be viewed as one special aggregation method in this formula.

3.3.2 Sliding Window Aggregation

Another important aggregation method is overlapping window aggregation, such as traditional kernel methods (see a good introduction of kernel regression in Chapter 5 of Wand and Jones (1995)). In this chapter, we consider a special one-sided sliding window method (called *Sliding Window Aggregation* (SWA)), which is motivated by detection of network anomalies in real time. Figures 3.5 - 3.7 show the idea of sliding window aggregation. The bottom row in these three figures shows 16 observations at the finest scale. To form a relatively coarser scale, we use a 2-observation aggregation, which is the same as the NOWA method. However, this new aggregation allows overlapping window, as discussed below. The window sizes with respect to scale for SWA also increase exponentially.

Figure 3.5 illustrates how to form scale 2 time series (the top row) from scale 1 (the bottom row). We will sum two consecutive observations at scale 1 to get an observation at scale 2. For example, we will aggregate the first 2 observations in the bottom row, as highlighted by two blue

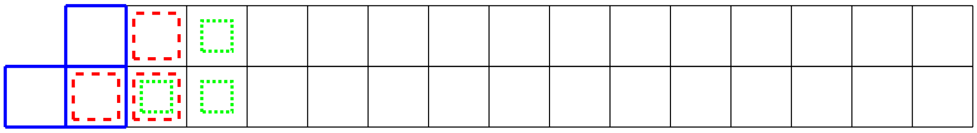

Figure 3.5: *The idea of the sliding window aggregation. This figure illustrates how to form time series at scale 2. The first data point at scale 2 can only be observed at the second time point, as shown as the leftmost observation (blue square box) in the top row. It is the sum of the first two observations in the finest scale (i.e., the two blue square boxes in the bottom row). The red dashed boxes and the green dotted boxes are used to illustrate the aggregation of the 2nd and the 3rd observations at scale 2, as the aggregation boxes slide one observation to the right.*

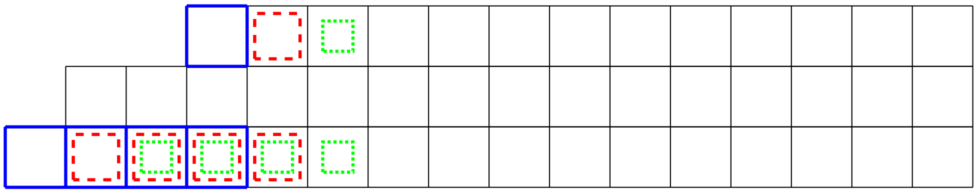

Figure 3.6: *This figure shows how to form time series at scale 3 by using SWA. The interpretations of different color boxes are the same as in Figure 3.5, and the aggregation boxes also slide one observation each time to the right.*

square boxes in Figure 3.5, to form the first observation at scale 2, shown as the blue square box in the top row. After this, we will use the second and the third observations at scale 1, as shown as the two red dashed square boxes in the bottom row, to form the second observation at scale 2, which is displayed as a red dashed square box in the top row. Note that the aggregation window for these two aggregations slides one observation to the right (toward new observations), and these two windows overlap each other. This is why we called this aggregation the *sliding window aggregation*. In Figure 3.5, we also use two green dotted square boxes in the bottom row and a green dotted square box in the top row to show the generation of the third observation at scale 2.

Figures 3.6 and 3.7 show how the scale 3 and 4 time series (the top row in each Figure) are aggregated from the scale 1 series (the bottom row). To form observations of scale 3 (or a even larger scale k), the window size becomes $2^{3-1} = 4$ (or 2^{k-1}), we will start from the first 4 (or 2^{k-1}) observations (the blue square boxes in the bottom row in both figures) at scale 1 to form the first

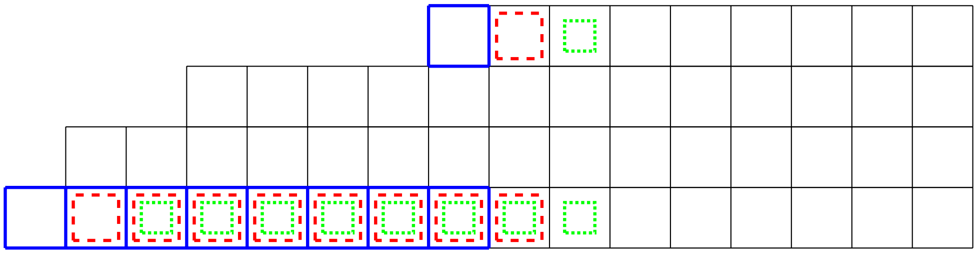

Figure 3.7: *This figure shows how to form time series at scale 4 by using SWA. The first data point at scale 4 can only be observed at the eighth time point, as shown as the leftmost observation (blue square box) in the top row. It is the sum of the first 8 observations in the finest scale (i.e., the blue square boxes in the bottom row). Generation of the next two observations is illustrated by the red dashed boxes and the green dotted boxes, which have similar interpretation as in Figures 3.5 and 3.6.*

observation at scale 3 (or k) (the blue box in the top row in Figures 3.6 and 3.7)), and then slide the window by one observation at scale 1 to the right to form the second observation at scale 3 (or k), etc. In Figures 3.6 - 3.7, this means sliding the window of the blue boxes in the bottom row to the window of the red dashed boxes. Continuing the sliding by one observation to the right, we can form all the observations at different scales. In Figures 3.6 and 3.7, we use red dashed and green dotted boxes to illustrate how to form the 2nd and the 3rd observations at coarser scales. Note that the index i for each observation at scale k formed by windows of size 2^{k-1} at scale 1, will be the time stamp of the last observation at scale 1. Thus, the time series at scale k generated by the SWA, does not have the first $2^{k-1} - 1$ observations, as shown in Figure 3.7. Note that the relationship between observations at scale k and $k + 1$, based on SWA, are no longer as simple as those based on NOWA.

The following provides the mathematical definition of SWA. Assume that $\{Y_1(i)\}$ is the time series at the finest scale (scale 1). The observation at time i and scale k is defined as

$$Y_k(i) = \sum_{j=0}^{2^{k-1}-1} w_k^*(j) Y_1(i-j),$$

where $w_k^*(j)$ are chosen to make $\{Y_k(i)\}$, for all i and k, share the same marginal distribution, when there are no outliers in $\{Y_1(i)\}$.

Note that the sliding window aggregation can be easily implemented as an iterative (over time) algorithm, and thus is well suited for realtime outlier detection. If a new data point at the finest scale arrives, all the observations of the current time point at different scales can be easily updated. Let $\mathbf{Y}(i) = (Y_1(i), Y_2(i), \cdots, Y_k(i))^T$ be the current observation vector over different scales, and $Y_1(i+1)$ be the new arrived observation at scale 1 (after normalization). The new observation vector (over scales) at time $i+1$, $\mathbf{Y}(i+1) = (Y_1(i+1), Y_2(i+1), \cdots, Y_k(i+1))^T$, will be

$$\mathbf{Y}(i+1) = f_1(\mathbf{Y}(i)) + f_2(Y_1(i+1)) - f_3(\widetilde{\mathbf{Y}}(i)),$$

where $\widetilde{\mathbf{Y}}(i) = (Y_1(i), Y_1(i-1), \cdots, Y_1(i-2^k+1))^T$. Functions f_1, f_2, and f_3 are predefined from the above $w_k^*(j)$ (see details about these weights in Section 3.4.5). Assume the above aggregation is simply the summation of the observations within the window, the above updating function is

$$\mathbf{Y}(i+1) = \mathbf{Y}(i) + Y_1(i+1) - \widetilde{\mathbf{Y}}(i),$$

i.e.,

$$\begin{pmatrix} Y_1(i+1) \\ Y_2(i+1) \\ \vdots \\ Y_k(i+1) \end{pmatrix} = \begin{pmatrix} Y_1(i) \\ Y_2(i) \\ \vdots \\ Y_k(i) \end{pmatrix} + Y_1(i+1) - \begin{pmatrix} Y_1(i) \\ Y_1(i-1) \\ \vdots \\ Y_1(i-2^k+1) \end{pmatrix}.$$

It is straightforward that the complexity of this update depends on the number of scales.

Note that the SWA defined above can be viewed as a special case of the one-sided kernel method (see Gijbels et al. (1999) for a usage of one-sided kernels). Thus, SWA can be extended to allow use of a general (as apposed to the uniform kernel method here) one-sided kernel method for detection. If the detection is not necessarily required to be realtime, this method can be modified to other forms, such as a symmetric sliding window. Thus, the usual kernel method can be adapted in this framework for outlier detection, although theoretical properties based on kernel methods might be different.

These two aggregation methods have a strong relationship. For example, at the 16th time slot, the observation vector generated by NOWA is the same as the vector from SWA, if appropriate $\widetilde{w}$ and w^* are used. Thus, the test at each location designed for NOWA is very similar to that for SWA. The theoretical properties in Section 3.4.5 are mainly based on the sliding window aggregation, while all the visualizations are based on the non-overlapping window aggregation, because the NOWA provides a good interpretation, as discussed below.

3.3.3 MRAD outlier map based on SWA

After forming time series at different scales, we can construct hypothesis tests for observations over time locations and scales, and report the test results. These results can be whether they are outliers or not, or the possibilities of these locations and scales to be outliers. A multiple test at each location over scales will be explored in detail (Section 3.4.5), along with some theoretical properties. The following discusses the MRAD outlier map based on SWA. Figure 3.8 shows the MRAD outlier map based on SWA for the network data set discussed in Section 3.2. The MRAD outlier map, based on NOWA, has been discussed in Section 3.2 (Figure 3.2). These two types of outlier maps display the testing significance probability (p value), based on the marginal distribution of each observation, at each time location and scale. The rows in the map visualize the results at different

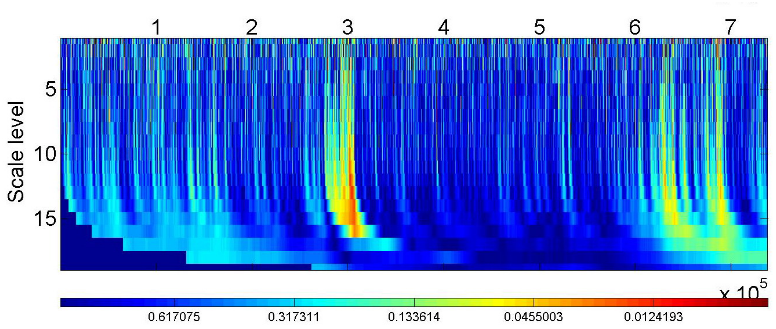

Figure 3.8: *The MRAD map for the network byte counts data (sliding window aggregation) on April 10, 2002. It shows network anomalies at time locations around 3, 6.5 and 6.7 ($\times 10^5$), and scales from 10 to 15, by showing hotter regions.*

scales, and the columns are the results for different time locations over scales. As discussed in Section 3.2, hot colors (red) are used to show small p values (i.e. higher chance to be outliers), and cool colors (blue) display large significance probabilities (i.e., lower chance to be outliers). Note that the time series at scale k generated by SWA, does not have the first $2^{k-1}-1$ observations, so that we cannot perform hypothesis tests at these regions. We display them in the color of dark blue, the same color as for the testing result of $p=1$.

The interpretation of the MRAD outlier map relies on the method of aggregation. The MRAD outlier map based on SWA has a significantly different interpretation from the map based on NOWA. For the method of SWA, let us assume that the observation at time i and scale k is flagged as an outlier. This flag means that there exists network anomalies at locations from $i-2^{k-1}+1$ to i (i.e., the last 2^{k-1} time locations) at the finest scale. For the method of NOWA, the flag at time i and scale k means the corresponding block (i.e., from $(i-1)\times 2^{k-1}+1$ to $i\times 2^{k-1}$) at the finest scale contains outliers. Note that 2^{k-1} is the window size of aggregation (of scale 1 observations) at scale k. The above shows that NOWA can involve information after time i. There is a characteristic pattern in the MRAD outlier map: a single huge spike may cause the map to be flagged (i.e., to be thought of as possible outliers) at larger scales. In fact, based on the tests we defined in Section

3.4.4, the flag will first appear in a relatively finer scale, then a huge spike will be flagged shortly (in time) after it comes up. Because the detection method, based on SWA, is actually iterative in time, we can either remove the detected anomalies, or restart a new iterative procedure, once one observation is flagged as an outlier. These treatments (i.e., removing the detected anomalies or restarting a new procedure) will avoid the map showing the above special pattern (i.e., one significant anomaly at a finer scale and current time might cause the map flag anomalies at coarser scales later).

Figure 3.8 shows the MRAD outlier map for the whole time series based on SWA. It also highlights the three regions (locations around 3, 6.2, and 6.8($\times 10^5$), and scales around 15), which were discussed in Section 3.2. However, compared to the NOWA map, these regions are moved slightly to the right, which reflects the above effect. Note that in real applications, the MRAD outlier map based on SWA for the whole time series usually is not available, unless there are no outliers at all. The reason is that the iterative procedure is usually stopped, once an anomaly is flagged.

To form time series at different scales, we need to standardized the original series before the aggregation. In our exploration, a robust method is used to normalize the time series at the finest scale. After that, appropriate constants are chosen to aggregate multiscale time series. The robust estimation of the mean we used in this paper is the median, and the robust estimation for the standard deviation uses the median absolute deviation times a constant (1.4826) (see page 106 in Hampel et al. (1986)). We use these as a starting point to explore the usefulness of the MRAD method. Exploration of better normalization procedures, especially when there are outliers in the time series, is proposed as future work.

3.4 Theoretical properties for the MRAD method

In this Section, we explore some theoretical properties of the MRAD method. Section 3.4.1 reviews some background on fractional Gaussian noise, which is one of our model assumptions. Section 3.4.2 reviews one important result in extreme value theory, which will be used to develop an asymptotic threshold of our MRAD method. The formal description of the testing problem we are dealing with is given in Section 3.4.3. In order to prove the theoretical properties, particular NOWA and SWA methods are defined in Section 3.4.4. Most of the theoretical properties are the same for these two aggregation methods, as discussed in Sections 3.4.4 and 3.4.5. One major property is that the power of a two-scale MRAD method based on this procedure, is larger than the average power of testing methods at the two scales, which is proven in Section 3.4.5. Asymptotic test threshold is developed as well in Section 3.4.5.

3.4.1 Background on LRD and fractional Gaussian noise

Let $\gamma(h) = EX_iX_{i+h}$ be the autocovariance function of a statistical time series. A stationary time series is said to have *long range dependence* (LRD), if $\gamma(h) \propto L(h)h^{-\alpha}$ as $h \to \infty$, where $L(h)$ is a slowly varying function, and $\alpha \in (0,1)$ (see Taqqu (2003)). An important feature of LRD time series is that $\sum_{h=1}^{N} |\gamma(h)| \to \infty$ as $N \to \infty$. Here we give the definition of fractional Brownian motion (fBm) and fractional Gaussian noise (fGn).

Definition 3.4.1. *A stochastic process $\{B_H(t)\}_{t\in\mathcal{R}}$ is called a fractional Brownian motion (fBm), if it is a Gaussian process with mean 0, stationary increments, variance $EB_H^2(t) = t^{2H}\sigma^2$ and covariance*

$$EB_H(s)B_H(t) = \frac{\sigma^2}{2}\left\{s^{2H} + t^{2H} - |s-t|^{2H}\right\},$$

where $0 < H < 1$ is called scaling exponent or Hurst parameter.

Definition 3.4.2. *The increment process of a fractional Brownian motion, $X_i = B_H(i+1) - B_H(i)$, $i \geq 1$ is called a fractional Gaussian noise (fGn).*

Note that fGn is a mean zero, stationary Gaussian time series, with autocovariance function $\gamma(h)$ given by

$$\gamma(h) = \frac{\sigma^2}{2}\{|h+1|^{2H} - 2|h|^{2H} + |h-1|^{2H}\}, \quad h \geq 0. \tag{3.1}$$

For $H \neq 1/2$, $\gamma(h) \sim \sigma^2 H(2H-1)|h|^{2H-2}$ as $h \to \infty$. So when $1/2 < H < 1$, the fGn shows long range dependence. It has been used for modeling network traffic.

See Mandelbrot and Van Ness (1968), Taqqu (2003) for more information about fractional Brownian motion and fractional Gaussian noise.

3.4.2 Background on extreme value theory of stationary processes

There is a large literature in the field of extreme value theory about stationary processes, see for example Leadbetter et al. (1983). The following lemma was first established in Berman (1964), which will be used in this chapter.

Lemma 3.4.1. *Let $\{\xi_n\}$ be a (standardized) stationary normal sequence with the autocovariance function $\{\gamma(n)\}$ satisfying $\gamma(n)\log n \to 0$. Let $M_n = \max\{\xi_1, \xi_2, \cdots, \xi_n\}$ Then*

$$P\{a_n(M_n - b_n) \leq x\} \to \exp(-e^{-x}),$$

where

$$a_n = \sqrt{2\log n}, \quad b_n = \sqrt{2\log n} - (2\sqrt{2\log n})^{-1}(\log\log n + \log 4\pi). \tag{3.2}$$

Here, $\exp(-e^{-x})$ *is called a Gumbel type distribution function.*

As noted in Berman (1964) and Leadbetter et al. (1983), this lemma implies that the distribution of M_n for a stationary process, when the stationary process satisfies the condition in the lemma, asymptotically is the same as the M_n for a independent Gaussian sequence.

3.4.3 The testing problem

To show the theoretical properties of the MRAD method we introduced in Sections 3.2 and 3.3, we define a testing problem as follows. Let $\{Y_1(i)\}$, $i = 1, \cdots, N$ be the observed time series. We assume the underlying model for $\{Y_1(i)\}$ is

$$Y_1(i) = X_1(i) + \delta I_{i \in [a_0, a_1]}(i), \tag{3.3}$$

where $\{X_1(i)\}$ is a fGn with Hurst parameter H.

In the context of intrusion detection, $X_1(i)$ can be viewed as the normal traffic, and the level shift represents a type of network anomaly, such as a DDoS attack. We are interested in detecting the starting time of the attack a_0, which can be formulated as a testing problem, i.e., testing whether the observation $Y_1(i)$ is an outlier or not:

$$H_0 : \mathcal{L}(Y_1(i)) = \mathcal{L}(X_1(i)) \text{ vs. } H_1 : \mathcal{L}(Y_1(i)) = \mathcal{L}(X_1(i) + \delta), \tag{3.4}$$

where $\mathcal{L}(Y_1(i))$ means the distribution of the random variable, $Y_1(i)$. Note that the problem (3.4) is a pointwise testing problem in the time space.

3.4.4 Simple MRAD procedures

Here we propose two formal testing procedures. One is based on NOWA, defined in Section 3.3.1, and the other is based on SWA, defined in Section 3.3.2. We will show that, most of the theoretical properties in both cases are the same.

Let $Y_1(i)$ be the observed time series specified by the model (3.3), N be the number of observations, b be an integer, $M = \lfloor \log_b N \rfloor$ or some specific integer (no larger than M). Here $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ are defined as $\lfloor a \rfloor = \max\{n \in \mathcal{Z} : n \leq a\}$, $\lceil a \rceil = \min\{n \in \mathcal{Z} : n \geq a\}$ and $\mathcal{Z}$ is the set of all integers. Let L be the aggregation level, where $L = b, b^2, \cdots, b^M$. In this chapter, we use $b = 2$ to illustrate the usefulness of the MRAD method. Other values of b may be sometime desirable, so this is allowed in the MATLAB program we provide on the website of Zhang (2006a).

The two MRAD procedures are defined as

- MRAD procedure based on NOWA: we define

$$Y_L^N(i) = \frac{1}{L^H} \sum_{j=1}^{L} Y_1((i-1)L + j), i = 1, 2, \cdots, \lceil N/L \rceil. \tag{3.5}$$

  When $i > N$, $Y_1(i)$ is defined as $Y_1(i) = Y_1(\text{mod}(i, N))$, where $\text{mod}(i, N)$ is the residual integer of i divided by N.

- MRAD procedure based on SWA: we define

$$Y_L^S(i) = \frac{1}{L^H} \sum_{j=i-L+1}^{i} Y_1(j), i = L, L+1, \cdots, N. \tag{3.6}$$

Based on the above definitions, it's straightforward to show that the $\{Y_L^N(i)\}$ and $\{Y_L^S(i)\}$, when $i = L, 2L, \cdots, jL, \cdots$, are the same. We will use $Y_L(i)$ to denote the observations based on either aggregation method.

The rejection region for the testing problem (3.4) at the aggregation level L (when $L = b^k$, we refer to k as the scale level for simplicity) is $\{|Y_L(i)| > C_\alpha\}$, with C_α defined by $P(|Y_L(i)| > C_\alpha|\delta = 0) = \alpha$. It is shown in Section 3.4.5 that the critical value C_α is a function of α, and does not depends on L.

The test based on either aggregation is described as follows. Take the ith observation as an example, the procedure uses $\max_L |Y_L(i)|$ as the test statistic, and $\max_L |Y_L(i)| > C_\alpha^M$ as the rejection region, i.e.,

1. Set a unique threshold C_α^M, so that, under the null hypothesis (i.e., there is no outlier in the time series),
$$P(\bigcup_L \{|Y_L(i)| > C_\alpha^M|\delta = 0\}) = \alpha. \tag{3.7}$$

2. For any $i = 1, \cdots, N$, if any of the observations at different scales exceeds the test threshold, i.e., falls into the following rejection region:
$$R : \max_L |Y_L(i)| > C_\alpha^M, \tag{3.8}$$
we claim that $Y_1(i)$ is a possible outlier.

The theoretical properties of this procedure will be studied in Section 3.4.5.

3.4.5 Theoretical properties of a 2-scale MRAD procedure

In this section, we show that the power of a two-scale MRAD procedure (either based on NOWA or SWA), as described in Subsection 3.4.4, is larger than the average power of the naive outlier detection method based on the two scales. This shows that the MRAD method is better than methods based on a single scale. The two-scale procedure is sufficient to show this feature. We conjecture that this result is also true for larger scale MRAD procedure. The prove of a m-scale procedure is difficult, and is a potential future work.

Since $\{X_1(i)\}_{i=1,\cdots,N}$ is a fGn with Hurst parameter H, it can be shown that $\{X_L^N(i)\}$ defined by the NOWA aggregation, $X_L^N(i) = \frac{1}{L^H}\sum_{j=1}^{L} X_1((\lceil i/L \rceil - 1)L + j)$, is also a fGn with the same Hurst parameter H. Thus, $X_L^N(i)$ has the same marginal distribution as $X_1(i)$. It can also be shown that $\{X_L^S(i)\}$ based on SWA, has the same marginal distribution, as the $\{X_L^N(i)\}$ based on NOWA. So for both aggregation methods, all $\{X_L(i)\}$ share the same marginal distribution (i.e. $N(0, 1)$). We are interested in testing whether the ith observation $Y_1(i)$ is an outlier (i.e. $Y_1(i) = X_1(i) + \delta$, and $\delta \neq 0$) or not, as defined in Equation (3.4).

Let $\{|Y_L(i)| > C_{\alpha,L}\}$ be the rejection region for scale L for the significance level α. Because the marginal distributions of $\{Y_L(i)\}$ are the same ($N(0, 1)$), when there is no outlier within the series, the threshold $C_{\alpha,L}$ does not depend on L. We denote this threshold as C_α. Note that for a given significance level α, $C_\alpha = \Phi^{-1}(1 - \alpha/2)$.

The following propositions and theorems show some important theoretical properties for the MRAD procedure based on SWA (see in Section 3.4.4). Unless otherwise specified, all the following results are based on SWA. Proof of the following propositions and theorems are in Section 3.8.

Proposition 3.4.1. *If $P_0(\cup_L\{|Y_L(i)| > C_\alpha^M\}) = \alpha$, and $P_0(|Y_L(i)| > C_\alpha) = \alpha$, we have $C_\alpha^M \geq C_\alpha$.*

Remark: This proposition shows the MRAD method provides a more conservative threshold than the naive outlier detection method at any scales.

For the remaining part of this section, let C_α^M be the 2-scale MRAD testing threshold, and C_α be the testing threshold based on one single scale.

Proposition 3.4.2. *The time series at the ith location over different scales, $\{Y_k(i)\}$, $k = 1, 2, \cdots$, based on the SWA method with base b, is a stationary process, with autocorrelation function*

$$\begin{aligned}\rho_b(k-1) &= \frac{1}{b^{kH}}\left[1 + \frac{1}{2}(b^{k2H} - (b^k - 1)^{2H} - 1)\right] \\ &= Hb^{k(H-1)} + \frac{b^{-kH}}{2} - \frac{H(2H-1)}{2}b^{k(H-2)} + o(b^{k(H-2)}).\end{aligned}$$

Remark: This proposition shows that at each time location, the observations across scales form a stationary process. The autocorrelation function decays exponentially, i.e., this process over scales is short range dependent. It also shows that when H is large, the decay rate will be slow. We can use some results in the analysis of stationary processes to explore more theoretical properties, and to develop an asymptotic calculation of the threshold (see Theorem 3.4.2 in this subsection).

Proposition 3.4.3. *For a 2-scale MRAD method, let C_α^M be the testing threshold of significance level α, we have*

$$C_\alpha^M = C_0 - \frac{\phi(C_0)C_0^2H^2}{2\sqrt{1-\alpha}}L^{2(H-1)} + o(L^{2(H-1)}),$$

as $L \to \infty$. Here $C_0 = \Phi^{-1}\left(\frac{1+\sqrt{1-\alpha}}{2}\right)$.

Remark: The proposition implies that C_0 is the limit of C_α^M. When L is large, the testing threshold C_α^M is close to C_0. In addition, the convergence rate is of the order $2(H-1)$. Larger H corresponds to lower convergence rates.

Theorem 3.4.1. *Let*

$$\beta_{(1,L)} = P_1(\max_{l=1,L} |Y_l(i)| > C_\alpha^M),$$

$$\beta_1 = P_1(|Y_1(i)| > C_\alpha),$$

$$\beta_L = P_1(|Y_L(i)| > C_\alpha).$$

For any $\delta > 0$, there exists $\alpha_\delta > 0$ and $L_\delta > 0$, when $\alpha \in (0, \alpha_\delta)$ and $L > L_\delta$, the following inequality holds:

$$\beta_{(1,L)} \geq \frac{\beta_1 + \beta_L}{2}.$$

Remark: This theorem shows that for any level shift, when the significance level α is small and L is large, the power of the two-scale MRAD is larger than the average power of the outlier detection method based on a single scale.

Theorem 3.4.2. *Let $M_m = \max_L(Y_L(i))$ be the maximum value of $Y_L(i)$ for an m-scale MRAD procedure, which is defined in* (3.5)*, M_m has a limiting distribution of the Gumbel type*

$$P\{a_m(M_m - b_m) \leq x\} \to \exp(-e^{-x}) \quad as \quad m \to \infty,$$

with

$$a_m = \sqrt{2\log m}, \quad and \quad b_m = a_m - \frac{\log\log m + \log 4\pi}{2a_m}.$$

Remark 1: This theorem shows that the asymptotic distribution of $\max_L(Y_L(i))$ of an m-scale

MRAD procedure is the same as m i.i.d. standard normal random variables. i.e. $P(M_m \leq x) \approx \Phi^m(x)$, when m goes to infinity. Similarly, for $\widetilde{M_m} = \min_L(Y_L(i))$, we have $P(\widetilde{M_m} \geq -x) \approx \Phi^m(x)$.

Remark 2: This theorem implies that one asymptotic test threshold is $C_0^M = \Phi^{-1}((1-\alpha/2)^{1/m})$, by using a simplified two-test Bonferroni type procedure (of M_m and $\widetilde{M_m}$). Davis (1979) showed that when $\gamma(n)\log(n) \to 0$, M_n and $\widetilde{M_n}$ are asymptotically independent. This result leads to an independent asymptotic test threshold $C_I^M = \Phi^{-1}((1-\alpha)^{1/2m})$.

Remark 3: At one particular time location, the test across time scales can be viewed as a multiple test problem (testing whether $\{Y_L(i)\}$ is an outlier ($L = 1, \cdots, m$) or not). Given the significance level as α, a typical Bonferroni type test threshold for this multiple test is $C_b = \Phi^{-1}(1-\alpha/2m)$. Note that $(1-\alpha/2)^{1/m} = 1-\alpha/2m-(m-1)\alpha^2/(8m^2)+o(\alpha^2)$, and $(1-\alpha)^{1/2m} = 1-\alpha/2m-(2m-1)\alpha^2/(8m^2)+o(\alpha^2)$, when $\alpha \to 0$. This yields that $C_I^M \leq C_0^M \leq C_b$. Thus both of the two asymptotic thresholds, have larger power than the usual Bonferroni procedure. In addition, the independent asymptotic threshold has the largest power among these three. Figure 3.9 shows the two asymptotic thresholds at different number of scales, when the significance level (α) is 0.05. It validates the above comparison.

Remark 4: Note that when $m = 2$, the m-scale asymptotic test thresholds (as discussed in Remarks 1 and 2), is different from the two-scale asymptotic threshold, which was developed in Proposition 3.4.3. Note that, in Proposition 3.4.3, we assume the distance between scales goes to infinity, thus, the observations across scale are asymptotic independent. However, in Remarks 1 and 2 in this theorem, the observations across scale are not asymptotic independent. Different conditions and assumptions make the thresholds different from each other.

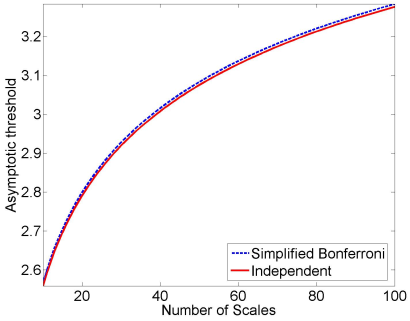

Figure 3.9: *The Comparisons between the simplified Bonferroni asymptotic threshold and the independent asymptotic threshold, when $\alpha = 0.05$. It validates that the independent asymptotic threshold is smaller than the simplified Bonferroni one, thus it has larger power*

3.5 An improved test threshold

The above Theorem 3.4.2 provides two asymptotic test thresholds for the m-scale MRAD method: simplified Bonferroni-type threshold, $C_0^M = \Phi^{-1}((1-\alpha/2)^{1/m})$; and independent asymptotic threshold, $C_I^M = \Phi^{-1}((1-\alpha)^{1/2m})$. , as discussed in the remarks of Theorem 3.4.2. However, these thresholds are not precise, because it is the threshold when m, the number of scales, goes to infinity. Often a rather small number of scales (e.g. 15) is used. An improved test threshold is explored in this section.

When there is no outlier within the time series, Proposition 3.4.2 (in Section 3.4.5) shows that the observations at one particular time location over scales, form a stationary process. The autocovariance functions (given in the proposition) of these stationary processes, over different time locations, are the same. The marginal distribution of these observations are standard Normal, when there is no outlier within the series. Thus, given the Hurst parameter, the significance level, and the number of scales we use, an exact test threshold can be developed based on the multivariate Normal distribution. It is difficult to get the test threshold explicitly, because the complexity of multivariate Normal calculation. We developed a MATLAB function, to approximate the exact threshold based on simulation. We call the estimate of this threshold as *the improved test threshold* for the MRAD method.

For example, let the number of scales be 10, significance level be 0.1, and the Hurst parameter be 0.9, the covariance function of this 10-dimensional Normal random vector is precisely defined. We simulate this Normal random vector 1000 times. For each vector, we will find the maximum of the absolute values. Thus, the 90% sample quantile of these 1000 maximum values provides an estimate of the test threshold. In order to reduce the variability of this estimation, we repeat this estimating procedure several times (for example, 1000 times). The average of these quantiles is a better estimate. And these (1000-time) replications provide a measurement of the error margin of this estimation.

Table 3.1 provides a partial list of the improved thresholds, and their corresponding 95% error margins, under different combinations of parameters: the significance level, Hurst parameter, and the number of scales in the MRAD method. The MATLAB function, `mradtestthreshold.m`, available at the website of Zhang (2006a), calculates the improved threshold, when the above parameter combination is specified.

α	H	number of scales	threshold	95% error margin
0.1	0.5	10	2.4561	0.0024
0.1	0.7	10	2.3899	0.0025
0.1	0.9	10	2.2015	0.0027
0.1	0.99	10	1.8656	0.0029
0.1	0.5	11	2.4915	0.0024
0.1	0.7	11	2.4258	0.0024
		……		

Table 3.1: *The improved test thresholds, for the MRAD method, under different combinations of significance level, α, Hurst parameter, H, and number of scales, m, in the MRAD method. The right most column is the 95% error margins for the improved test thresholds.*

Figure 3.10 shows the relationship between the exact test threshold and the Hurst parameter. In

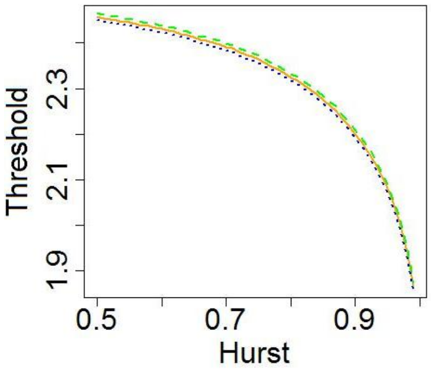

Figure 3.10: *The improved test threshold vs. the Hurst parameter. In this plot, we plot the estimates of the test thresholds (the orange line) at the significance level $\alpha = 0.1$. Two references lines shows the 95% confidence intervals of these estimates. The green dashed line is the upper bond of the confidence interval, and the blue dotted line is the lower bond of the confidence interval. It suggests that the estimates have small variability. This plot also shows that when the Hurst parameter increases, the test threshold decreases.*

the calculation of test thresholds, we set the significance level α to be 0.1, and the number of scales is 10. After that, for each Hurst parameter, we simulate 1000 samples to get the upper 10% quantile (i.e., $\alpha = 0.1$) of the statistics $\max_L |Y_L(i)|$, which is one estimate of our test threshold. Note that in the program of `mradtestthreshold.m`, we repeat this estimation several times (typically 1000 times), and use the average of these estimates, which provides a more correct estimation of the test threshold. The orange line in Figure 3.10 shows the averages of different parameter combinations. The green dashed line is the upper bond of the central 95% confidence interval, and the blue dotted line corresponds to the lower bond of the confidence interval. These two bonds suggest that our estimation has a very small variability. Thus this improved test thresholds are close to the theoretical exact thresholds. From Figure 3.10, we find that, when the Hurst parameter increases, the test threshold decreases. This can be easily verified theoretically from a two-scale MRAD procedure. The increase in the Hurst parameter causes the correlation to become larger. It can be shown that at the same significance level, the test threshold will decrease.

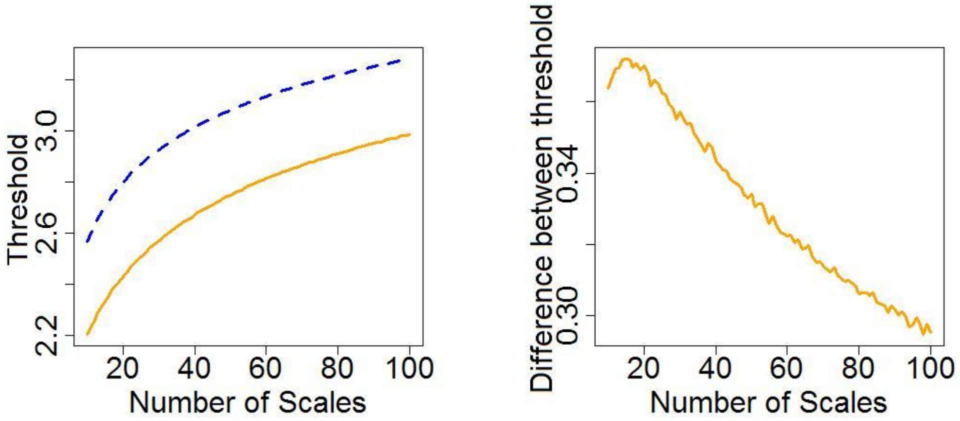

Figure 3.11: *The left panel shows the improved thresholds (the orange line) at different number of scales, when the Hurst parameter is 0.9 and the significance level is 0.1. The blue dashed line in this plot is the simplified Bonferroni type asymptotic threshold. The right panel shows the differences between these two thresholds. It shows that when the number of scales increases, the difference between these two thresholds decreases.*

The left panel in Figure 3.11 shows the relationship between the improved test threshold and the number of scales. The orange line plots the estimates of the exact thresholds. It demonstrates that when the number of scales increases, the improved test threshold also increases, which is not surprising. The blue dashed line in the left panel is the simplified Bonferroni type of asymptotic threshold, $C_0^M = \Phi^{-1}((1-\alpha/2)^{1/m})$ (remind that m is the number of scales used in the MRAD procedure), which is defined in Theorem 3.4.2. It shows that the improved thresholds are smaller than the asymptotic thresholds (at each parameter combination), so the test using the improved threshold has larger power than the test using the later one. The right panel shows the differences between the asymptotic thresholds and the improved thresholds. It shows that when the number of scales becomes larger, the difference will be smaller. Thus, when m, the number of scales, is a large number, it will not lose too much power by using the asymptotic threshold.

3.6 Simulations and applications in network intrusion detection

In this section, we use several toy examples (Sections 3.6.1, 3.6.2, and 3.6.4) semi-experiments (Section 3.6.5) and real applications (Section 3.6.6) to further illustrate the usefulness of the MRAD method. All the figures and movies in this section are produced from the MATLAB function, `MRADvisual.m`, which can be downloaded from the website of Zhang (2006a).

3.6.1 Fractional Gaussian noise and local mean level shift

In this subsection, we use simulation to evaluate the MRAD method. In these simulations, the background time series are simulated from fractional Gaussian noise with several predefined Hurst parameters, such as 0.7, 0.9, and 0.95. As noticed from the UNC Internet Data Study Group (Le and Hernández-Campos, 2004), the Hurst parameters for packet-count and byte-count time series are usually close to 0.9. So the study in this subsection provides a good approximation to those network features.

In the following analysis, we use standard fractional Gaussian noise as the background series, and the anomalies are set to be a local mean level shift. There are several parameters that need to be set (See Section 3.4.3 for definitions of these): the Hurst parameter (H) of the fractional Gaussian noise, the length of the whole series (N); the starting time (a_0), the duration ($K = (a_1 - a_0)$), and the intensity (or the mean of the level shift, δ) of the local level shift.

In this subsection, the length of the background time series, N, is set to be 2^{15}; and the Hurst parameter is set to be 0.9. The intensities (δ) of the level shift are chosen as 1, which means that the mean of the level shift is the same as the standard deviation of the background trace. More challenging (i.e. smaller) intensities are investigated in Section 3.6.2. The starting point of the local level shift, a_0, is randomly simulated from a uniform distribution, $U[0, 2^{14}]$, i.e., the starting point falls into the first half part of the trace. The duration of the level shift, K, is simulated from an exponential distribution with mean duration 4000. In terms of network anomaly detection, these settings correspond to a rather long duration, but relatively low intensity attacks, which are quite challenging to detect by other methods. Note that, the uniform and exponential distributions are chosen as natural starting points of this study. The true distribution of the starting point of a particular type of attacks, and the true distribution of the duration of an attack, are interesting future research topics.

In our simulation, after we simulate the starting point, and the duration of the level shift, a number of background traces based on the same Hurst parameter are simulated to evaluate the performance of the MRAD method. For example, the following shows one particular simulated setting, the starting point of the level shift is 5644, the duration is 6465, and the intensity is 1. Those background traces are 100 fractional Gaussian noises with the Hurst parameter $H = 0.9$. We add the same level shift into these traces. These 100 replications provide a way to calculate some statistics, which help to evaluate the MRAD method. In this subsection, we use the following statistics for evaluation: Detected Outlier Rate (DOR), True Discovery Rate (TDR), False Discovery Rate (FDR) and False Negative Rate (FNR). Another predefined measurement, True Outlier Proportion (TOP), is also reported. The definitions for these statistics are:

- TOP: the true outlier proportion contained in the time series, i.e., the designed anomalies duration divided by the length of the whole time series. In our simulation study, this is a predefined non-random number.

- DOR: the detected outlier rate (i.e. the average detected outlier proportion) contained in the time series, i.e., the expected number of flagged anomalies divided by the total number of observations within the whole time series.

- TDR: among all the designed outliers, the expected proportion of those actually detected.

- FDR: among all the detected outliers, the average ratio of those that fall outside the designed anomaly interval.

- FNR: among all the non-flagged observations, the average ratio of those that fall into the designed anomaly interval.

By using the classical FDR table (from Benjamini and Hochberg (1995)) below, we can define these statistics and measurements explicitly using mathematical notation.

	Declared non-anomaly	Declared anomaly	Total
regular observations	U	V	m_0
true anomalies	T	S	$m - m_0$
	$m - R$	R	m

Table 3.2: The classical FDR definition table.

The random variables in this table are described as

U: the number of observations that are actually anomalies, but not identified.

V: the number of observations that are anomalies, and identified.

T: the number of observations that are not anomalies, and identified as regular observations

S: the number of observations that are not anomalies, but identified as outliers

R: the total number of observations, which are identified as outliers.

For example, in the simulation we discussed above, we have $m = 2^{15}$, and $m - m_0 = 6465$. Note that these random variables vary for different realizations of the background series. Based on these notions, we can formally define the above statistics and measurements.

Starting time	Duration	Intensity	TOP	DOR	TDR	FDR	FNR
5644	6465	1	19.73%	20.80%	94.64%	5.36%	0.08%

Table 3.3: Evaluation of MRAD based on simulation. In these time series, we injected outliers (i.e., a mean level shift) at 19.73% locations. The MRAD method reported 20.80% locations are anomalies. Among these flagged locations, around 5% of them are false alarms, and we detected approximately 95% of the injected locations. This shows good performance of the MRAD method.

TOP: $(m - m_0)/m$.

Note that this is not a random variable. It just provides the outlier proportion.

DOR: $E(R/m)$.

The average proportion of detected anomalies among all observations.

TDR: $E(S/(m - m_0))$.

The average proportion of detected anomalies among the designed anomalies.

FDR: $E(V/R)$.

The average false discovery proportion.

FNR: $E(T/(m - R))$.

The average false negative proportion.

For these 100 simulation sets, we can calculate the proportions for each simulation, and the averages of these proportions give estimate of the above measurements. The results for the above simulation set are listed in Table 3.3.

Table 3.3 shows that the designed outlier proportion is around 20% of the total observations, and our method flags around 21% (on average) within the series. Among all the designed anomalies, we detected 95% of them (on average). Among all the declared anomalies, (on average) about 5% of them are false alarms. In addition, among all the non-declared observations, only less than 0.1% of them are true anomalies. These numbers show that our method is good at detecting level shifts.

Figure 3.12 displays the results of one realization among these 100 simulated traces. The top

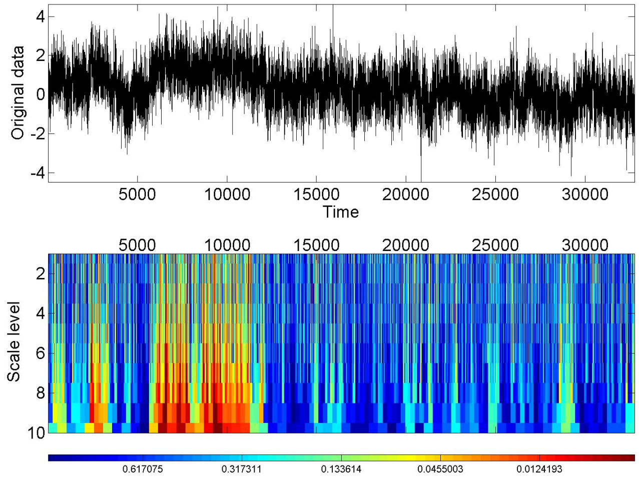

Figure 3.12: *MRAD outlier map based on NOWA for one simulation trace, which is fractional gaussian noise plus a given local mean level shift. This map highlights the injected level shift by showing a hotter region around the time interval (5,500, 12,000), and scales from 6 to 10.*

Starting time	Duration	Intensity	TOP	DOR	TDR	FDR	FNR
10223	11835	1	36.11%	36.41%	98.74%	1.26%	0.26%
1946	3760	1	11.47%	12.03%	93.74%	6.26%	0.24%
9572	4407	1	13.45%	14.17%	93.53%	6.47%	0.25%
3173	4491	1	13.71%	14.62%	93.65%	6.35%	0.03%
......							

Table 3.4: *This table provides more evaluation results by simulation. The starting points here are also simulated from $U[0, 2^{14}]$, and the durations are exponential distributed with the mean of 4000. All background traces under each setting are simulated 100 times. More results are available at Zhang (2006a).*

panel plots the time series containing anomalies. It is rather difficult to conclude whether there are anomalies, and to identify where they are. The bottom panel shows the MRAD outlier map (based on NOWA). It highlights an area at locations from 5,500 to 12,000 (a relatively hotter zone), and scales from 6 to 10. This NOWA outlier map suggests that, these corresponding locations at the finest scale, contain anomalies. These identified locations are close to those locations where we input the level shift.

Table 3.4 provides more simulation results for evaluation of the MRAD method. The starting points of these level shifts are also from the same uniform distribution defined above, i.e., $a \sim$

$U[0, 2^{14}]$. The durations of these level shifts are exponential distributed with the mean 4000 as well. All the intensities are set to be 1 (or a less number such as 0.5). We simulated over 100 different combinations of a_0, K, δ, and H. The full table (of over 100 simulation settings) is available at the website of Zhang (2006a). In Table 3.4, we provide a small portion of the results, because this is sufficient to illustrate the usefulness of the MRAD method. Among each combination of the starting time, the duration and the intensity, we also simulate 100 background traces, and evaluate the method based on the above measurements, such as FDR and FNR.

In Table 3.4, we find that most TDRs are greater than 90%, i.e, we are able to identify most of the designed anomalies. The FDRs and FNRs are relatively small in these simulations. Note that there is an excellent result, as shown in the first row of Table 3.4. The FDR in this row is dramatically smaller than others. One possible reason is that the designed outlier proportion and the declared outlier proportion are really large, which make a significant increase of the denominator part in the definition of the FDR. See more evaluation results at Zhang (2006a).

3.6.2 Intensity of the mean level shift

In the above simulations, we use the level shift intensity $\delta = 1$ to demonstrate the usefulness of the MRAD method. This corresponds to the mean level shift being the same as the standard deviation. See Figure 3.12 for a visual impression of this setting. The previous section shows that when the mean level shift is the same as the standard deviation, the MRAD method performs well in detection of this level shift. In this section, more challenging intensities are investigated.

The left panel of Figure 3.13 shows the distributions of FDRs (as defined in Section 3.6.1) for 10 simulations in each case, under different values of δ; and the right panel of Figure 3.13 displays the distributions of the corresponding FNRs. Both plots show boxplots, representing the population of the observed FDR or FNR respectively, from the top panel to the bottom panel, as δ varies from a large value (0.9) to a small value (0.1).

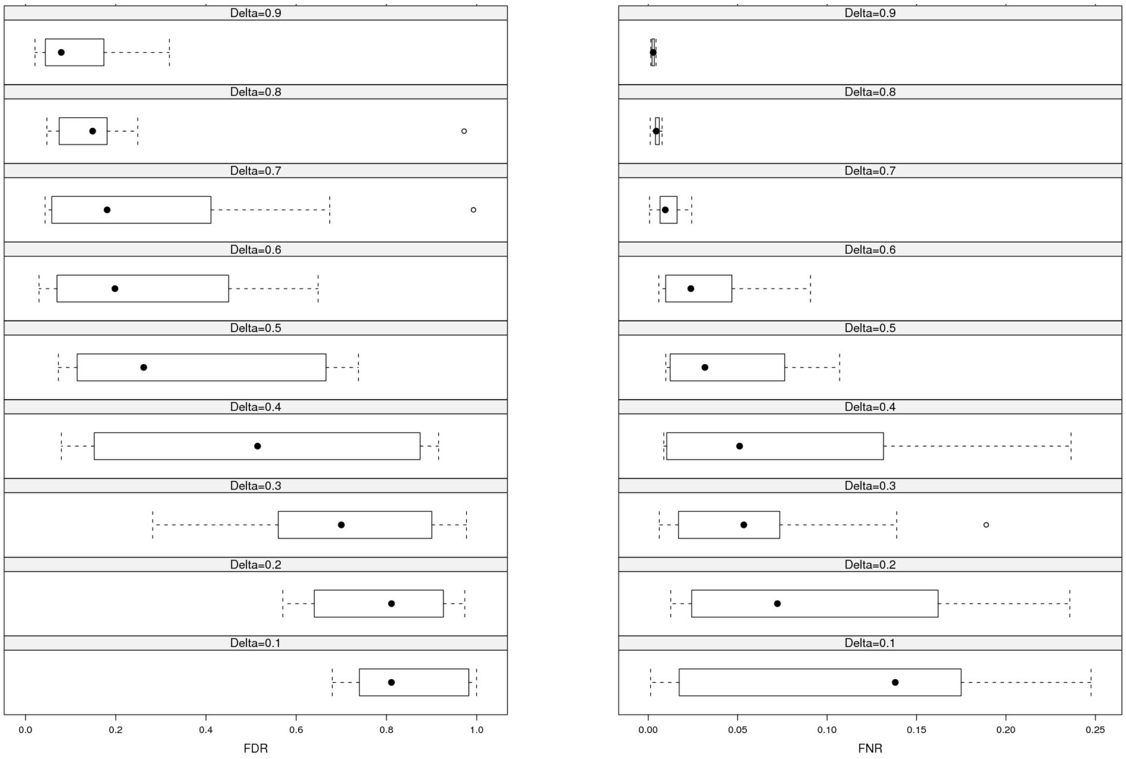

Figure 3.13: *The left panel shows the distribution of FDRs under different intensities, and the right panel shows the distribution of the FNRs. In each panel, δ varies from a large value (top subpanels) to a small value (bottom subpanels). This shows that the FDR and FNR increase as the intensities decreases.*

The boxplot within each subpanel shows the interquartile range (i.e., from the first quartile to the third quartile) as the central box. The black dots within the central boxes are the medians under each setting. The two extension lines, outside the central boxes, essentially reach the minimum data point (to the left) and the maximum data point (to the right). If either of these two lines is longer than 1.5 times the interquartile range, the corresponding minimum data point (or the maximum data point) is shown as a dot (In this plot, the symbol is “o”). And the extension lines end at the 2nd minimum data point (or the maximum data point).

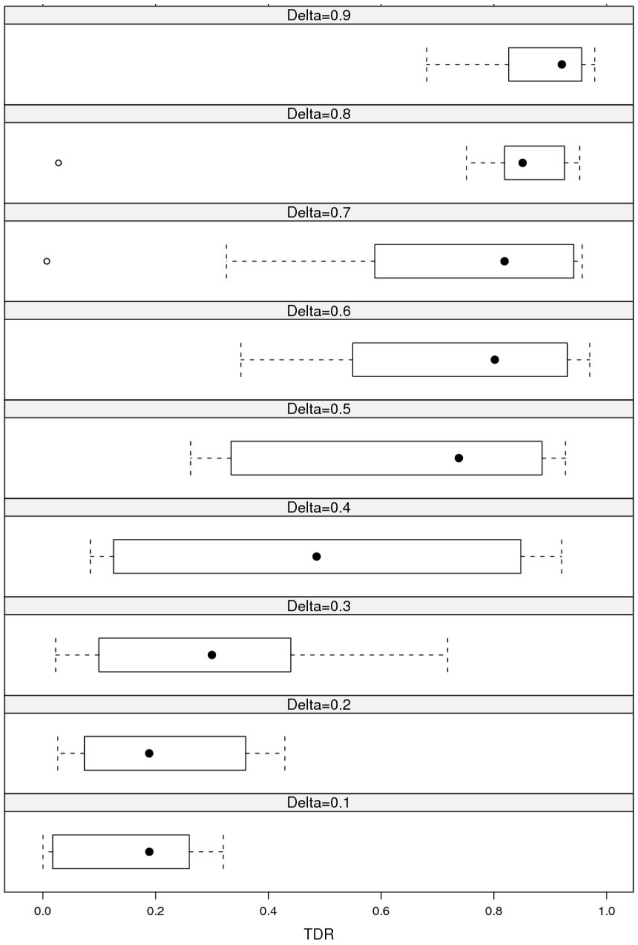

Figure 3.14: *The distribution of TDRs under different intensities. The intensity parameter, δ, varies from a larger value to a smaller value, as the subpanels display from the top to the bottom. This plots show that the TDR decreases while the intensity decreases.*

These subpanels in Figure 3.13 show the distributions of the FDRs and FNRs. We find that the median FDRs and FNRs significantly increase while the intensities of the level shift decrease. In addition, the median FDR and FNR decreases slowly in both plots when δ is larger than 0.4. There is a significant increase (or jump) in the median at $\delta = 0.4$. Note that even when δ is smaller than 0.4, the median FNRs (as shown in the right penal) are still smaller than 0.10, i.e., half of the simulations have FNRs less than 10%, which indicates a really good performance of our MRAD method.

Figure 3.14 shows the distributions of the True Discovery Rate (TDR, defined in Section 3.6.1) under the same challenging intensities. The median TDR decreases when the intensity decreases, as shown in the figure from the top to the bottom. It also shows that the decrease is slow when δ is larger than 0.4, and again there is a significant jump at $\delta = 0.4$. Note that the median TDRs are larger than 0.6 when δ is greater than or equals 0.5, i.e., if the intensity is larger than or equals the half of the standard deviation, the MRAD will detect more than half of the observations within the left shift.

3.6.3 Comparisons between the MRAD method and the intervention analysis

In this subsection, we use simulations to compare the proposed MRAD method and the classical outlier detection method based on intervention analysis.

The intervention analysis method was first introduced by Box and Tiao (1975). Chang et al. (1988), Tsay (1988) etc. developed an iterative algorithm for detecting outliers in time series based on intervention analysis. The detection procedure for outliers within a stationary time series can be summarized as follows

- Assume that there is no outlier within the series, and fit the time series with an autoregressive moving average model (ARMA(p, q)),

$$\phi(B)Y_t = \theta(B)\varepsilon_t,$$

where B is a backshift operator (i.e., $B^k(Y_t) = Y_{t-k}$), and

$$\phi(B) = 1 - \phi_1 B - \cdots - \phi_p B^p,$$

$$\theta(B) = 1 + \theta_1 B + \cdots + \theta_q B^q.$$

More discussion about ARMA models can be found in classical time series textbooks, such as Brockwell and Davis (1991) and Box et al. (1994).

- Get the estimated noise $\widehat{\varepsilon}(t)$
$$\widehat{\varepsilon}(t) = \frac{\phi(B)}{\theta(B)} Y_t$$

- Among all the $\{\widehat{\varepsilon}(t)\}$, test whether the most suspicious observation is an outlier or not. If not, conclude that there is no outlier (anymore) within the series, and stop this procedure. If yes, remove that observation (or those observations) and reprocess this procedure till stop.

Scientific Computing Associates provides an automatic software package, SCA, for comprehensive analysis of time series, including detection of outliers within the time series. However, their automatic programs runs extremely slow for our simulated data set (i.e., fractional Gaussian noise plus a local mean level shift). The new SAS/ETS package (in SAS 9.13) also provides routines for detecting outliers in time series, by using the intervention analysis methods. In the following comparisons, we use the SAS/ETS package to detect the imputed level shift automatically.

In this subsection, we also simulated the fractional Gaussian noises with 2^{15} observations. The level shifts we imputed here are simulated from the same distribution discussed as in Section 3.6.1: the starting point is from $U[0, 2^{14}]$, the duration of the level shift is from exponential distribution with mean 4000, and the mean level shift is 1 for all simulations. In the following plots, we report

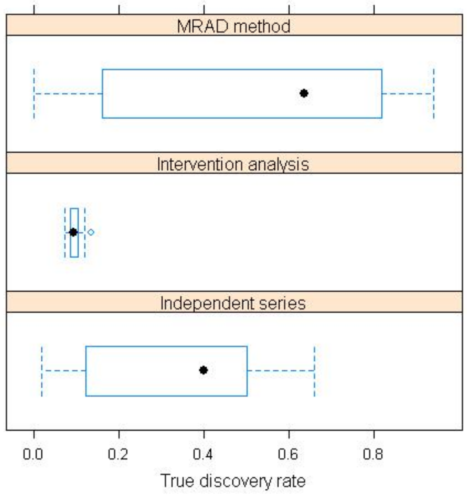

Figure 3.15: *True discovery rate comparisons among the MRAD method (the top subpanel), the naive one-scale method (the bottom subpanel), and the intervention analysis method (the middle subpanel). The MRAD method has the smallest median TDR, and the median TDR for the intervention analysis is the largest. This suggests that our method is better than the intervention analysis, when the time series is long range dependent. Note that the intervention model actually specifies a wrong model.*

the same sets of statistics (i.e., FDR, FNR, TDR) for three types of method: our MRAD method, a naive one-scale method (see details in Section 3.4.5), and the intervention analysis method.

Figure 3.15 shows the distribution (by using boxplot) of the True Discovery Rate (TDR, see details in Section 3.6.1) for these three detection methods. The top panel provides the boxplot for the TDRs using the MRAD method based on 10 sets of simulation. Each simulation set contains 100 simulations of background traces. We calculate the true discovery proportions for each simulations, and show the average of these proportions. The boxplot visualizes the distribution of 10 averages of these proportions. Note that the median TDR of the MRAD method is larger than 0.6. However, the spread of the TDRs is large. The middle panel is the boxplot for the TDRs using the intervention analysis based on the same 10 sets of simulation. It shows that the variation among different sets of simulation is small. However, the median TDR is around 0.1. This shows that it is hard to find the imputed anomalies by using the intervention analysis. Note that the intervention analysis

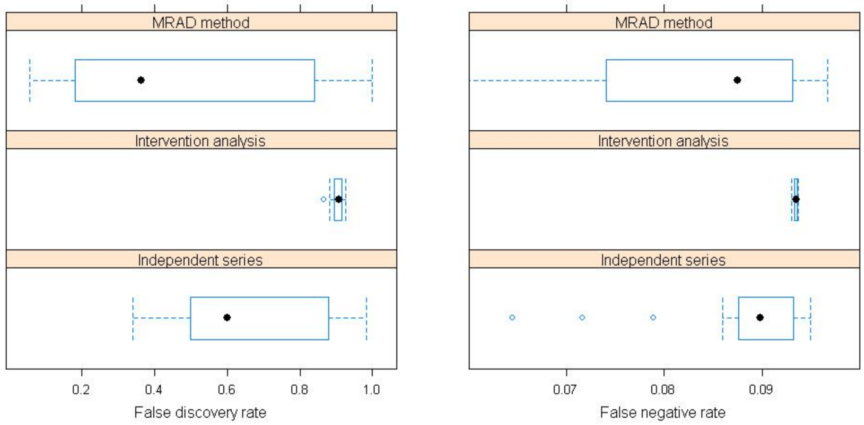

Figure 3.16: *FDR and FNR comparisons among the MRAD method (the top subpanels), the naive one-scale method (the bottom subpanels), and the intervention analysis method (the middle subpanels). These two plots shows that the MRAD method has the smallest median FDR and the smallest median FNR.*

method actually treats the background time series as an ARMA sequence, which is a wrong model in our case. The bottom panel shows the naive one-scale detection method, which is discussed in Section 3.4.5. The median TDR for this method is larger than the intervention analysis, but smaller than the MRAD method. These three plots suggest that our MRAD is the best among these three methods (on average).

Figure 3.16 shows the distribution of the FDRs and FNRs for these three methods using the same 10 sets of simulations. The left panel is the boxplot of the FDRs, and the right panel shows the boxplot of the FNRs. By comparing these three methods, we find that the median FDR of the MRAD method is the smallest one, and the intervention models has the largest median FDR. All three methods have really small median FNRs. In fact, most of these methods report a large proportion of the data observations as regular observations. This will cause the FNRs to be relatively small. In addition, we notice that the median FNR of the MRAD method is the smallest as well, which suggests that the MRAD method provides a better solution in detecting outliers, when the background time series is a long range dependent trace.

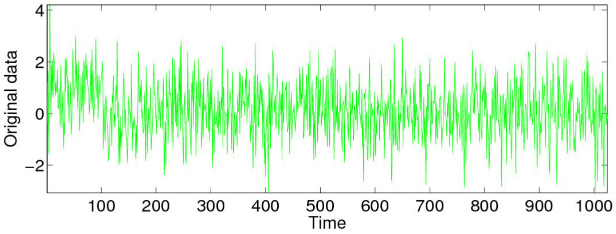

Figure 3.17: *The time series plot for the white noise sequence with one local mean level shift. It is not easy to find the location of the level shift.*

3.6.4 White noise with a local mean level shift and the scale movie

The simulated data set here is designed as a white noise series plus a mean level shift. Although the white noise series is not a series with long range dependence, the MRAD procedure can also be used to find mean level shifts in this context. This section contains an example to illustrate this and a new MRAD scale outlier movie.

The example here is simulated from an independent identically distributed series, where $X_1(i) \sim N(0,1)$, $1 \leq i \leq 1000$. The observation series is given by $Y_1(i) = X_1(i) + \delta I_{i \leq 100}(i)$, where $\delta = 1$. So the example is designed as white noise series plus a local mean level shift. Figure 3.17 shows the time series of a simulated trace, where the designed level shift is not easily visible.

In this section, we use a scale outlier movie to highlight the MRAD test results at a given scale. The *MRAD scale outlier movie* can also be generated by the program `MRADvisual.m`, which is available at the website of Zhang (2006a). Figure 3.18 shows four carefully selected snapshots of the MRAD scale outlier movie for this toy data set. The full movie of this example and all other movies discussed in this chapter can be viewed at Zhang (2006a). The MRAD outlier movie varies its frames when the scale level changes from minimum to maximum, and then back to the minimal level. The top panel shows standardized time series at the finest scale, $\{Y_1(i)\}$, in green as the background, and highlights the aggregated time series, $Y_k(i)$, at a specified scale, k, in

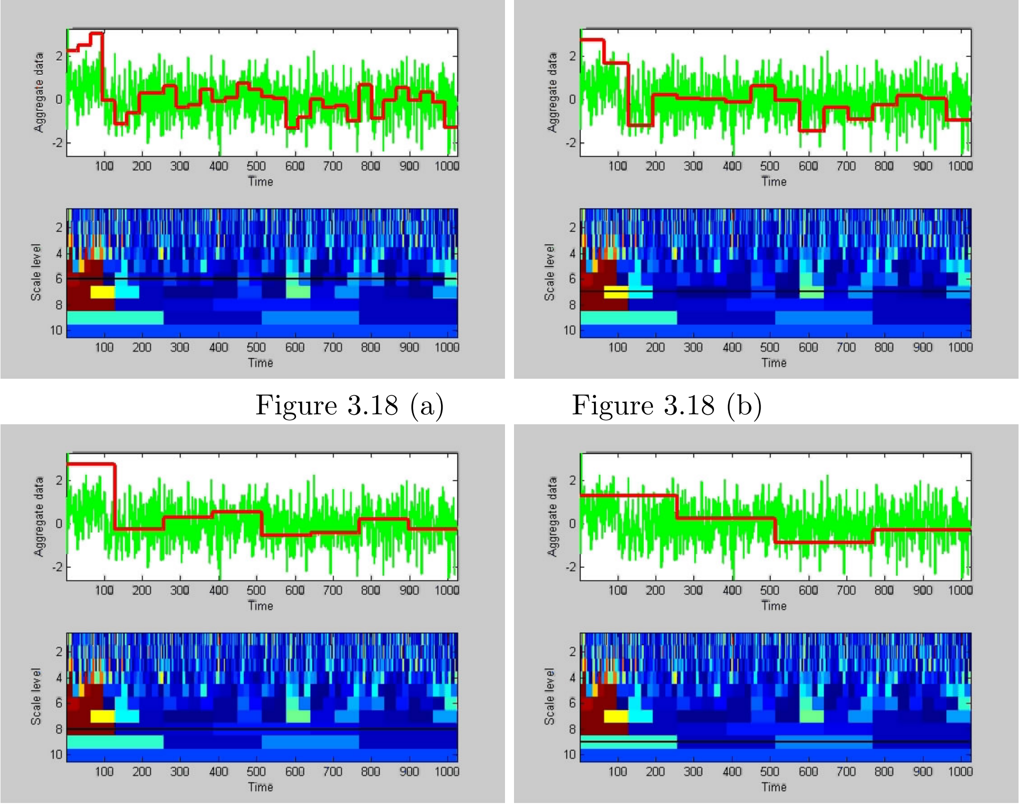

Figure 3.18 (a) Figure 3.18 (b)

Figure 3.18 (c) Figure 3.18 (d)

Figure 3.18: *Four selected snapshots of the MRAD movie for the simulated white noise series. It shows that outliers in this data set are highlighted around aggregation level 6 (Figure 3.18(a)), 7 (Figure 3.18(b)), 8 (Figure 3.18(c)) and 9 (Figure 3.18(d)). It suggests the first 96 observations having a mean level shift. This is not far away from the designed duration* 100.

red (as discussed in Section 3.4.4). The standardization make the series at different scales have a comparable range of values to display. The bottom panel of the movie shows the outlier map (which is the same MRAD outlier map discussed in Section 3.2) in the background. A scale level indicator (i.e., the black line) varies with the scale level. The interpretations of the colors are the same as the MRAD outlier map, as defined in Section 3.2.

Figure 3.18 shows four carefully selected snapshots of the MRAD scale outlier movie for this simulated data set. Figure 3.18(a) highlights the MRAD results at scale 6. It shows the first three blocks are likely to be outliers, which corresponds to the observations from 1 to 96 at the original time series. Figure 3.18(b) shows the results at scale 7. It shows that the first block is an outlier, and the second might also be an outlier, but the significance probability is a relatively larger number, which means the evidence is not strong. At this scale, it suggests that the first 64 observations in the original time series most likely contain outliers. The next 64 observations might contain outliers as well, but the number of locations might be fewer, or the intensity of the level shift is smaller. Figure 3.18(c) highlights the result at scale 8. The first block shows a small significance probability, which suggests that the first 128 observations in the original time series contain outliers. Combining the above interpretations together, it suggests that the first 96 observations at the finest scale contains outliers, or a mean level shift. This almost reveals the underlying structure we used in simulation. Figure 3.18(d) shows the results at scale 9. Note that the corresponding observations (i.e, the first 96 observations at the finest scale) in the outlier map change to the color blue. This suggest that those observations are no longer flagged as outliers at larger scales, which makes sense since this is a smaller scale phenomenon.

3.6.5 Semi-experiments

In this subsection, we use several semi-experiments to illustrate the usefulness of the MRAD method. Because there is a lack of labeled data traces available in the field of statistical anomaly detection, one of the best testbeds for an anomaly detection method is to use semi-experiments for evaluation.

The semi-experiment studied here, for anomaly detection evaluation can be summarized as follows

- Collect a "normal" trace from the Internet,
- Simulate some specific types of network anomalies in the lab,

- Combine these two types of traffic together, and then use detection methods to identify the injected anomalies.

In the following analysis, the UNC Internet Data Study Group collected a real trace from the UNC campus with a duration of three hours, and removed those network flows without either the starting point or the end point within this duration. The remaining flows are treated as the "normal" traffic. We use the center one hour trace as the background traffic, such that the feature time series are stationary. The network anomalies simulated in the lab include portscans, roseattacks and TCP synflood attacks (see Jeffay (2005) for details about these network anomalies). In this section, we will analyze one example of portscan as the injected anomaly. Note that in this analysis, because the way we get the "normal" trace, the MRAD method might also flag some hidden anomalies within the background trace. After combining the anomalies and the background trace, three statistical features were collected: packet-count, byte-count and flow-count per time interval. Here we use $10ms$ as the shortest time interval, which is the same as the motivating example discussed in Section 3.2.

A port scan (see one definition in Search-Security-Definitions (2005)) is a series of small packets, are intended to learn which computer network services, associated with a port number, the target computer provides. The port scan provides information for the attackers as to where to probe for system weaknesses. Note that a high frequency port scan will dramatically increase the number of flows (i.e., a huge local mean level shift). It also increases the number of packets and bytes to a certain degree. If the probing packets are really small in terms of size (in bytes), the increase of byte counts will typically be dominated by the variability of the real trace itself. If the probing frequency is low enough, the increase of packet counts will also be dominated by the variability of the background. Thus, these anomalies are not detectable in the series of packet counts and byte

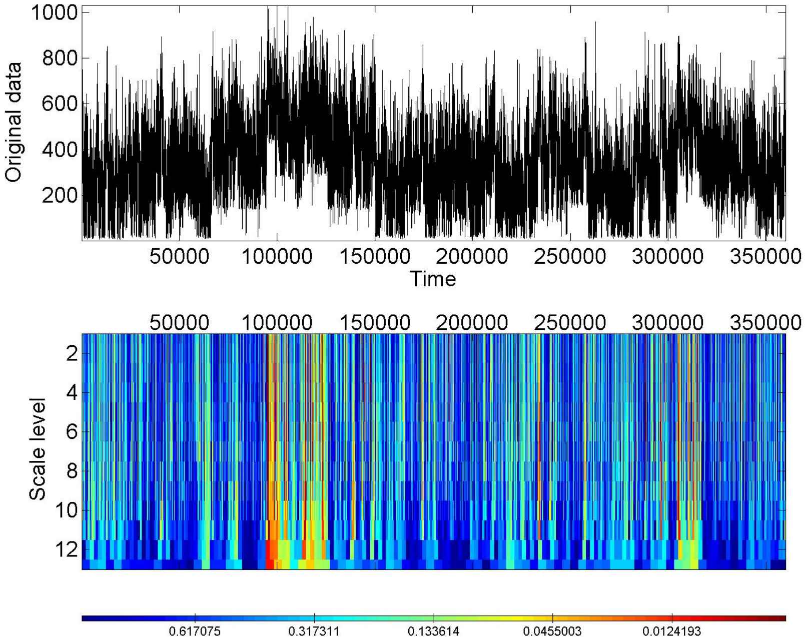

Figure 3.19: *The MRAD outlier map for packet count series of the semi-experiment example. It highlights two possible outlying regions: around 100,000, and around 300,000.*

counts. In this example, a medium frequency is used to send out small probing packets, such that the increase in packet count time series is detectable. See Lee et al. (2003) for more about detection methods and characterization of port scan attacks.

As discussed earlier, the background trace used here lasts for 1 hour. The portscan simulated for this example lasted 6 minutes, i.e., 10% of the total trace. It is generated as a type of medium frequent anomaly. Three statistical features are collected. These injected portscans make a significant increases in the flow counts, but only small changes in the packet counts and byte counts. It is expected that it is easy to detect these anomalies in the flow trace, but rather difficult to detect them in the byte or packet traces.

The top panel in Figure 3.19 shows the time series plot of the packet-count trace. It is hard to tell whether there are network anomalies, and to identify the locations of the anomalies. The estimated

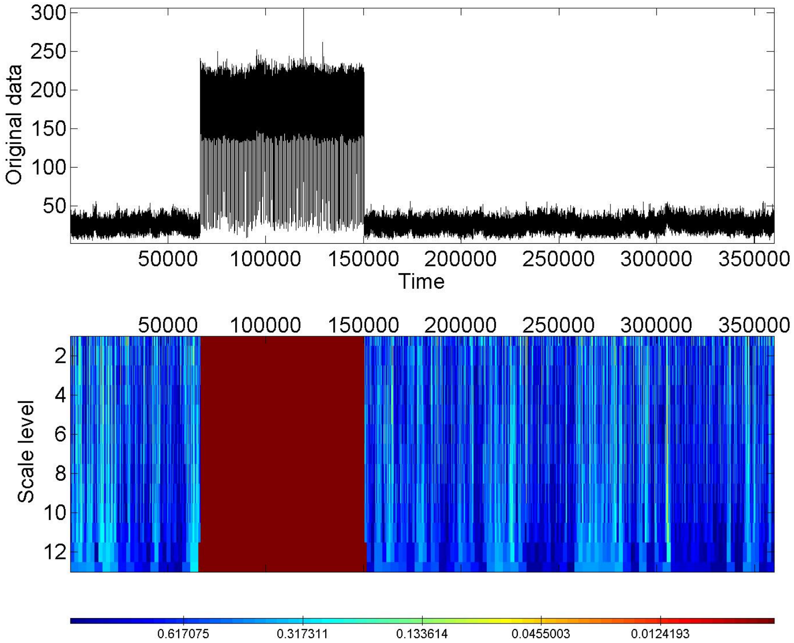

Figure 3.20: *The MRAD outlier map for the flow count series of the semi-experiment example. It reveals the injected anomalies at around 100,000.*

Hurst parameter for this trace is 0.95, which shows a strong long range dependent structure. We use this estimate (0.95) of the Hurst parameter, and perform an MRAD procedure based on NOWA. The bottom panel in Figure 3.19 is the MRAD outlier map based on NOWA. It highlights two hotter zones, around 100,000 and around 300,000. In fact, the zone around 100,000 corresponds to the location where we injected the network anomaly. We conjecture that those anomalies around 300,000 are some hidden anomalies within the "normal" trace, and we are working on identification of the causes of these hidden anomalies.

The top panel in Figure 3.20 is the flow count time series. It shows that the simulated portscan dramatically increases the number of flows in this experiment. Here we use the same estimate of the Hurst parameter 0.95 as a starting point to proceed the MRAD method. In fact, we can detect these anomalies even by eye. The bottom panel in Figure 3.20 shows that at a range of scales (in fact, all scales in this plot), the signals of the anomalies are strong enough to be detected. This shows that when the intensity of the level shift is a relatively larger number (compared to the standard deviation of the normal observations), the level shift is very clearly apparent at all scales. In fact, low intensity level shifts are more challenging, and our method will highlight these anomalies, as discussed in Section 3.6.2.

The analysis of the byte count time series reveals the anomalies around 300,000, but does not provide much insight about the injected anomaly at around 100,000. This confirms that the small-packet portscan does not generate a significant signal in the sense of size (of bytes). The picture is available at the website of Zhang (2006a). We skip it here because it does not provide additional insights. More analysis of other semi-experiments are also available at that website.

3.6.6 More analysis of the motivating example, zoomed maps and movies

In this subsection, we continue to analyze the network trace, which has been studied in Section 3.2, by using a zoomed version of the outlier map and movie.

The MRAD outlier map for the whole trace is in Figure 3.2. Note that the outlier map has smaller resolution, compared to the number of observations of the original time series. Thus it is hard to find some outliers directly from this map (for example, one single outlier at the finest scale). In this section, we develop zoomed versions of the MRAD outlier map (including the thresholded outlier map) to find additional insights from this data set.

Figure 3.21 shows three carefully selected snapshots from the MRAD zoomed outlier movie. When the time series has a large number of observations, such that the resolution of the MRAD outlier map is relatively small, it is natural to view the map locally. The MRAD zoomed outlier movie visualizes the whole map, in a horizontally sliding window, from the first group of observations

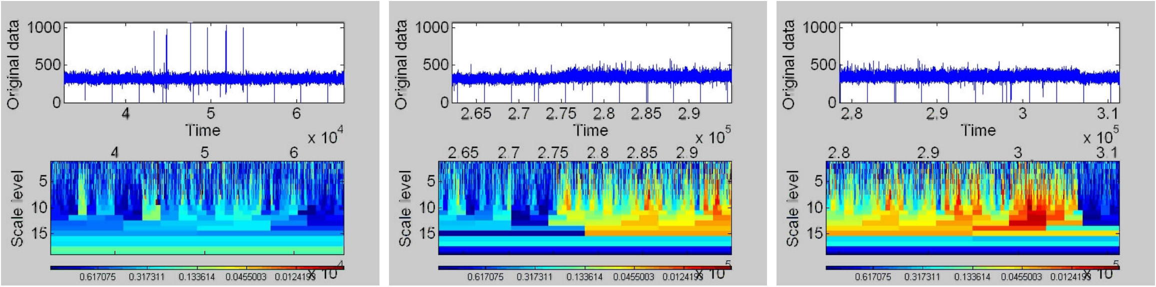

Figure 3.21: *Three carefully selected snapshots of the MRAD zoomed movie for this motivating example. The left panel shows that some spikes can only be highlighted at the finer scales, and will not be flagged at coarser scales. The right two panels highlights the starting point and the end point of a long duration left shift.*

to the last group of observations. Within each group, the zoomed map will have a better resolution to reveal more insights of the time series. The MATLAB function, `mradvisual.m`, can generate this movie with appropriate options. The interpretation of each frame is the same as the whole outlier map, which has been discussed in Section 3.2

The left panel in Figure 3.21 shows a part of the time series in the top subpanel. It shows several spikes within the time series. The bottom subpanel shows dark thin lines at finer scales. However, in relatively coarser scales, the zoomed map shows blue at almost all grids. This suggests those spikes disappear after aggregation. The right two panels highlight the mean level shift at around time 300,000, which has been discussed in Section 3.2. The middle panel highlights the starting point of this level shift, which is around 275,000 in the original time series. At some locations, the map shows a relatively cooler color (blue) at finer scales, but hotter colors (yellow or red) at coarser scales. This suggests that some locations are not shown as anomalies at one scale, but will be flagged at another scale. Viewing multiple scales will boost the (low) intensity of the level shift, and help to identify network anomalies. The right panel shows similar features as the middle panel. Note that these two panels also suggest this part of network anomalies starting gradually, but ending rather rapidly. In the middle panel around 275,000, the color changes gradually from cooler regions to hotter regions. In the right panel around 306,000, the color change from hotter to cooler is more like a jump. These changes are also visible in the zoomed time series plot in the top panels.

We can also use the zoomed thresholded outlier map and movie. Because it does not provide additional insights, we do not report on them here. See these movies and plots at the website of Zhang (2006a).

3.7 Conclusion and discussion

The above theoretical work and examples show that the MRAD method helps to identify outliers, especially when outliers are in the form of a slight mean level shift, even when the level shift is not obvious in the original time scale. We have proved that for a two-resolution MRAD procedure, the MRAD method uses a more conservative threshold, and has larger power on average than detection methods based on a single scale. Two thresholds, one being an asymptotic threshold, and the other being an estimation of the exact threshold, are developed in this chapter.

In this research, we assume that the Hurst parameter is known, or can be correctly estimated from the time series. And we use one type of robust procedure to standardize the original time series. It is natural to explore or develop a robust estimation of the Hurst parameter and a better robust procedure in terms of long range dependent time series.

The rejection region, as defined in (3.8), for the hypothesis testing problem (3.4), takes into account multiple comparisons at different scales. However, we did not consider the multiple comparison in the time space. In my post-doctorial research, we plan to incorporate some multiple testing methods in the time space, for example control the False Discovery Rate (Benjamini and Hochberg, 1995) in the time direction.

The aggregation method used in this chapter is natural and simple. It makes sense to explore other aggregation methods, such as wavelet methods, kernel methods, etc. Other types of outliers should also be explored. It is also natural to consider other types of long range dependent time series as background, such as fractional ARIMA and stable process. More challenging research remains to be done on this aspect.

3.8 Theoretical justifications

In this section, we provide some background on the Bivariate Normal distribution, and some proofs for the propositions in Section 3.4.5.

3.8.1 Background on the bivariate Normal distribution

The following lemma is used to calculate probabilities of a bivariate Normal distribution.

Lemma 3.8.1. *Let $f(x,y)$ be the probability density function of a bivariate normal distribution.*

$$f(x,y) = \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left\{-\frac{1}{2(1-\rho^2)}[x^2 - 2\rho xy + y^2]\right\}$$

we have

$$\begin{aligned}\int_a^b \int_c^d f(x,y)dxdy = {} & [\Phi(d)-\Phi(c)][\Phi(b)-\Phi(a)] + [\phi(c)-\phi(d)][\phi(a)-\phi(b)]\rho \\ & + \frac{[d\phi(d)-c\phi(c)][b\phi(b)-a\phi(a)]}{2}\rho^2 + O(\rho^3)\end{aligned}$$

where

$$\phi(x) = \frac{e^{-x^2/2}}{\sqrt{2\pi}} \quad \text{and} \quad \Phi(x) = \int_{-\infty}^{x} \phi(t)dt$$

Proof: Let $\widetilde{\phi}(x) = x\phi(x)$, the Taylor expansions at $\rho = 0$ of the following two functions are

$$\begin{aligned}
\Phi\left(\frac{c-\rho x}{\sqrt{1-\rho^2}}\right) &= \Phi(c) - \phi(c)x\rho + O(\rho^2), \\
\Phi\left(\frac{-c-\rho x}{\sqrt{1-\rho^2}}\right) &= \Phi(-c) - \phi(-c)x\rho + O(\rho^2),
\end{aligned}$$

which gives

$$\begin{aligned}
\int_a^b \int_c^d f(x,y)dxdy &= \int_a^b \phi(x)\left[\Phi\left(\frac{d-\rho x}{\sqrt{1-\rho^2}}\right) - \Phi\left(\frac{c-\rho x}{\sqrt{1-\rho^2}}\right)\right]dx \\
&= \int_a^b \phi(x)[\Phi(d)-\Phi(c)]dx + (\phi(c)-\phi(d))\rho\int_a^b \phi(x)xdx \\
&\quad + \frac{d\phi(d)-c\phi(c)}{2}\rho^2\int_a^b \phi(x)(1-x^2)dx + O(\rho^3) \\
&= [\Phi(d)-\Phi(c)][\Phi(b)-\Phi(a)] + (\phi(c)-\phi(d))\frac{e^{-a^2/2}-e^{-b^2/2}}{\sqrt{2\pi}}\rho \\
&\quad + \frac{d\phi(d)-c\phi(c)}{2}\frac{-ae^{-a^2/2}+be^{-b^2/2}}{\sqrt{2\pi}}\rho^2 + O(\rho^3) \\
&= [\Phi(d)-\Phi(c)][\Phi(b)-\Phi(a)] + [\phi(d)-\phi(c)][\phi(b)-\phi(a)]\rho \\
&\quad + \frac{[\widetilde{\phi}(d)-\widetilde{\phi}(c)][\widetilde{\phi}(b)-\widetilde{\phi}(a)]}{2}\rho^2 + O(\rho^3).
\end{aligned}$$

Remark 1: A commonly used special case is:

$$\begin{aligned}
\int_{-c}^c \int_{-c}^c f(x,y)dxdy &= [\Phi(c)-\Phi(-c)]^2 + \frac{[\widetilde{\phi}(c)-\widetilde{\phi}(-c)]^2}{2}\rho^2 + O(\rho^3) \\
&= [\Phi(c)-\Phi(-c)]^2 + 2c^2\phi^2(c)\rho^2 + O(\rho^3).
\end{aligned}$$

Remark 2: For non-centered bivariate normal distributions, i.e.,

$$f_{(\mu_x,\mu_y,\rho)}(x,y) = \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left\{-\frac{1}{2(1-\rho^2)}\left[(x-\mu_x)^2 - 2\rho(x-\mu_x)(y-\mu_y) + (y-\mu_y)^2\right]\right\},$$

we have, in the limit $\rho \to 0$,

$$\int_a^b dx \left[\int_c^d dy f_{(\mu_x,\mu_y,\rho)}(x,y)\right] = \int_{a-\mu_x}^{b-\mu_x} dx \left[\int_{c-\mu_x}^{d-\mu_y} dy f(x,y)\right],$$

where $f(x,y)$ as defined earlier. Now we can apply Lemma 3.8.1.

3.8.2 Power at one single scale

Lemma 3.8.2. *Let $\{|Y_L(i)| > C\}$ be the form of a rejection region at scale L. The power of the test at scale level L is given by*

$$P_1(|Y_L(i)| > c) = \begin{cases} \Phi(-C - L^{1-H}\delta) + \Phi(-C + L^{1-H}\delta), & L \le K(i), \\ \Phi(-C - \frac{K}{L^H}\delta) + \Phi(-C + \frac{K}{L^H}\delta), & L > K(i). \end{cases}$$

where $K(i) = i - a_0 + 1 \le K = (a_1 - a_0)$, when $i \in [a_0, a_1]$.

Proof: This comes from the fact that when H_1 is true, $Y_L(i)$ is from a Normal distribution with standard deviation 1, and mean given by $L^{1-H}\delta$ when $L \le K(i)$, or $K\delta/L^H$ when $L > K(i)$.

Remark: Substitute that $C_\alpha = \Phi^{-1}(1 - \alpha/2)$, we have the power at scale level 1 (i.e. the finest scale) is $P_1(|Y_1(i)| > C_\alpha) = \Phi(-C_\alpha - \delta) + \Phi(-C_\alpha + \delta)$.

Note that, in the later proves, K and $K(i)$ are interchangeable for simplicity.

3.8.3 Proof for Proposition 3.4.1

Due to the fact that

$$\{|Y_L(i)| > C_\alpha^M\} \subset \bigcup_L \{|Y_L(i)| > C_\alpha^M\},$$

we have

$$P\{|Y_L(i)| > C_\alpha^M\} \leq P\left(\bigcup_L \{|Y_L(i)| > C_\alpha^M\}\right) = P\{|Y_L(i)| > C_\alpha\},$$

which yields that $C_\alpha \leq C_\alpha^M$.

3.8.4 Proof for Proposition 3.4.2

In this subsection, we calculate a general covariance between $X_1(i)$ and $X_L(i)$, and this yields the result of Proposition 3.4.2 immediately.

Let $\{X_1(i)\}$ and $\{X_L(i)\}$ (based on SWA) be the fractional Gaussian noise series at scales 1 and L, respectively, as defined in Section 3.4.5.

Lemma 3.8.3. $\mathrm{Cov}(X_1(i), X_L(i)) = L^{-H}/2 + HL^{H-1} + O(L^{H-2})$ *as* $L \to \infty$.

Proof: $X_L(i) = L^{-H} \sum_{j=0}^{L-1} X_L(i-j)$, So we have

$$\begin{aligned}
\mathrm{Cov}(X_1(i), X_L(i)) &= \mathrm{Cov}(X_1(i), L^{-H} \sum_{j=0}^{L-1} X_1(i-j)) = L^{-H} \sum_{j=1}^{L} \mathrm{Cov}(X_1(1), X_1(j)) \\
&= L^{-H} \sum_{j=1}^{L} \gamma(j-1) = L^{-H}[1 + \sum_{j=1}^{L-1} \gamma(j)] \\
&= L^{-H}[1 + 2^{-1} \sum_{j=1}^{L-1} \{(j+1)^{2H} + (j-1)^{2H} - 2j^{2H}\}] \\
&= L^{-H}[1 + 2^{-1}(L^{2H} - (L-1)^{2H} - 1)] \\
&= HL^{H-1} + \frac{L^{-H}}{2} - \frac{H(2H-1)}{2} L^{H-2} + o(L^{H-2}),
\end{aligned}$$

where the last equation holds due to the fact that

$$\begin{aligned}
L^{2H} - (L-1)^{2H} &= L^{2H} \left[1 - (1 - \frac{1}{L})^{2H} \right] \\
&= L^{2H} \left[1 - (1 - 2H\frac{1}{L} + H(2H-1)L^{-2} + o(L^{-2})) \right] \\
&= L^{2H} \left[\frac{2H}{L} - H(2H-1)L^{-2} - o(L^{-2}) \right] \\
&= 2HL^{2H-1} - H(2H-1)L^{2H-2} - o(L^{2H-2}).
\end{aligned}$$

Remark 1: When $1/2 \leq H < 1$, we know the slowest term above is HL^{H-1}, so we can rewrite the above as $\mathrm{Cov}(X_1(i), X_L(i)) = HL^{H-1} + o(L^{-1/2}) \to 0$ as $L \to \infty$.

Remark 2: This lemma leads to a direct proof of Proposition 3.4.2:

Let b the aggregation bin size, which is defined in Section 3.4.4, and $l = b, b^2, \cdots, b^k, \cdots$ are the aggregation scales. Define $\xi_b(k) = Y_{b^k}(1)$. Let $L = b^{k+1}$, Lemma 3.8.3 suggests that $\{\xi_b(k), k = 0, 1, 2, \cdots\}$ is also a stationary process, and the covariance structure is given by

$$\begin{aligned}\rho_b(k) &= \mathrm{Cov}(\xi_1, \xi_{k+1}) \\ &= \mathrm{Cov}(X_1(1), X_L(1)) \\ &= \frac{1}{b^{kH}}\left[1 + \frac{1}{2}(b^{k2H} - (b^k - 1)^{2H} - 1)\right] \qquad (3.9) \\ &= \frac{b^{kH}}{2}\left[1 - \left(1 - \frac{1}{b^k}\right)^{2H}\right] + \frac{1}{2b^{kH}} \\ &= Hb^{k(H-1)} + \frac{b^{-kH}}{2} - \frac{H(2H-1)}{2}b^{k(H-2)} + o(b^{k(H-2)}),\end{aligned}$$

which is the proposition.

3.8.5 Proof for Proposition 3.4.3

Assume the correlation coefficient between $Y_L(i)$ and $Y_1(i)$ is ρ. When H_0 is true, we have the following result, by using the remark 1 of the Lemma 3.8.1.

$$\begin{aligned}P_0(\{|Y_L(i)| > C_\alpha^M\} \cup \{|Y_1(i)| > C_\alpha^M\}) &= 1 - P_0(\{|Y_L(i)| \le C_\alpha^M\} \cap \{|Y_1(i)| \le C_\alpha^M\}) \\ &= 1 - [(\Phi(C_\alpha^M) - \Phi(-C_\alpha^M))^2 + 2(C_\alpha^M)^2\phi^2(C_\alpha^M)\rho^2 + O(\rho^3)].\end{aligned}$$

Let $L \to \infty$, we have $\rho \to 0$, the above converges to

$$1 - (\Phi(C_\alpha^M) - \Phi(-C_\alpha^M))^2 = 1 - (1 - 2\Phi(-C_\alpha^M))^2 = \alpha,$$

which leads to

$$C_{\alpha}^{M} = -\Phi^{-1}\left(\frac{1-\sqrt{1-\alpha}}{2}\right).$$

Let $\rho = \mathrm{Cov}(Y_1(i), Y_L(i))$. Lemma 3.8.3 gives $\rho = HL^{(H-1)} + o(L^{(H-1)})$. In addition, let $C_{\alpha}^{M} = C_0 + a\rho^{\gamma} + o(\rho^{\gamma})$.

When $\rho \to 0$, we have

$$P_0(\{|Y_L(i)| > C_{\alpha}^{M}\} \cup \{|Y_1(i)| > C_{\alpha}^{M}\}) = 1 - [(\Phi(C_{\alpha}^{M}) - \Phi(-C_{\alpha}^{M}))^2 + 2(C_{\alpha}^{M})^2\phi^2(C_{\alpha}^{M})\rho^2 + O(\rho^3)].$$

Using Taylor expansion as $\rho \to 0$, We have

$$\begin{aligned}
\Phi(C_{\alpha}^{M}) &= \Phi(C_0 + a\rho^{\gamma} + o(\rho^{\gamma})) = \Phi(C_0) + \phi(C_0)a\rho^{\gamma} + o(\rho^{\gamma}),\\
\Phi(-C_{\alpha}^{M}) &= \Phi(-C_0 - a\rho^{\gamma} - o(\rho^{\gamma})) = \Phi(-C_0) - \phi(C_0)a\rho^{\gamma} + o(\rho^{\gamma}),\\
\Phi(C_{\alpha}^{M}) - \Phi(-C_{\alpha}^{M}) &= \Phi(C_0) - \Phi(-C_0) + 2\phi(C_0)a\rho^{\gamma} + o(\rho^{\gamma})\\
&= \sqrt{1-\alpha} + 2\phi(C_0)a\rho^{\gamma} + o(\rho^{\gamma}),\\
\left[\Phi(C_{\alpha}^{M}) - \Phi(-C_{\alpha}^{M})\right]^2 &= (1-\alpha) + 4\phi(C_0)a\sqrt{1-\alpha}\rho^{\gamma} + o(\rho^{\gamma}),\\
(C_{\alpha}^{M})^2 &= (C_0 + a\rho^{\gamma} + o(\rho^{\gamma}))^2 = C_0^2 + 2C_0a\rho^{\gamma} + o(\rho^{\gamma}).
\end{aligned}$$

We also have

$$\phi(C_{\alpha}^{M}) = \phi(C_0 + a\rho^{\gamma}) = \phi(C_0) - C_0\phi(C_0)a\rho^{\gamma} + o(\rho^{\gamma}),$$

which leads to

$$\phi^2(C_{\alpha}^{M}) = (\phi(C_0) - C_0\phi(C_0)a\rho^{\gamma} + o(\rho^{\gamma}))^2 = \phi^2(C_0) - 2C_0\phi^2(C_0)a\rho^{\gamma} + o(\rho^{\gamma}).$$

The above yields

$$2(C_\alpha^M)^2\phi^2(C_\alpha^M)\rho^2 = 2\phi^2(C_0)\left[C_0^2\rho^2 + 2C_0 a(1-C_0^2)\rho^{2+\gamma} + o(\rho^{2+\gamma})\right],$$

thus we have

$$\begin{aligned} P_0(\{|Y_L(i)| > C_\alpha^M\} \cup \{|Y_1(i)| > C_\alpha^M\}) &= 1 - P_0(\{|Y_L(i)| \le C_\alpha^M\} \cap \{|Y_1(i)| \le C_\alpha^M\}) \\ &= 1 - [(\Phi(C_\alpha^M) - \Phi(-C_\alpha^M))^2 + 2(C_\alpha^M)^2\phi^2(C_\alpha^M)\rho^2 + O(\rho^3)] \\ &= \alpha - 4\phi(C_0)a\sqrt{1-\alpha}\rho^\gamma - 2\phi^2(C_0)C_0^2\rho^2 + O(\rho^2). \end{aligned}$$

From the above, we have $\gamma = 2$, and

$$a = -\frac{\phi(C_0)C_0^2}{2\sqrt{1-\alpha}}.$$

In summary, we have

$$C_\alpha^M = C_0 - \frac{\phi(C_0)C_0^2}{2\sqrt{1-\alpha}}\rho^2 + o(\rho^2).$$

Substituting with $\rho = HL^{H-1} + o(L^{H-1})$, we have

$$C_\alpha^M = C_0 - \frac{\phi(C_0)C_0^2H^2}{2\sqrt{1-\alpha}}L^{2(H-1)} + o(L^{2(H-1)}).$$

3.8.6 Power for a two-scale MRAD procedure

Lemma 3.8.4. *Let* $\beta_{(1,L)} = P_1(\max_{l=1,L}\{|Y_l(i)| > C_\alpha^M\})$, *we have*

$$\beta_{(1,L)} = 1 - \sqrt{1-\alpha}[\Phi(C_0-\delta) - \Phi(-C_0-\delta)] - 2Hk\delta C_0\phi(C_0)[\phi(C_0-\delta) - \phi(-C_0-\delta)]L^{-1} + o(L^{-1}).$$

when $L \to 0$. *Here* $C_0 = \Phi^{-1}(\frac{1+\sqrt{1-\alpha}}{2})$.

Proof: When k and δ are fixed, and L is a fixed larger number, we have $Y_1(i) \sim N(\delta, 1)$, $Y_L(i) \sim N(\mu_L, 1)$, and $\rho = \text{Cov}(Y_1(i), Y_L(i)) = HL^{(H-1)} + o(L^{(H-1)})$ is closed to zero. Here $\mu_L = \frac{K\delta}{L^H}$, $K = 1$, 2, $\cdots$, L. Using Lemma 3.8.1, we have

$$\begin{aligned}
\beta_{(1,L)} &= 1 - P_1(\{-C_\alpha^M - \delta \le X_1(i) \le C_\alpha^M - \delta\} \cap \{-C_\alpha^M - \mu_L \le X_L(i) \le C_\alpha^M - \mu_L\}) \\
&= 1 - P_1(\{-C_0 - \delta + a\rho^2 - o(\rho^2) \le X_1(i) \le C_0 - \delta - a\rho^2 + o(\rho^2)\} \\
&\quad \cap \{-C_0 - \mu_L + a\rho^2 - o(\rho^2) \le X_L(i) \le C_0 - \mu_L - a\rho^2 + o(\rho^2)\}) \\
&= 1 - [\Phi(C_0 - \delta) - \Phi(-C_0 - \delta)][\Phi(C_0) - \Phi(-C_0)] \\
&\quad + [\phi(C_0 - \delta) - \phi(-C_0 - \delta)][2C_0\phi(C_0)(k\delta L^{-H})HL^{(H-1)}] + o(L^{-1}) \\
&= 1 - [\Phi(C_0 - \delta) - \Phi(-C_0 - \delta)]\sqrt{1-\alpha} + 2Hk\delta C_0\phi(C_0)[\phi(C_0 - \delta) - \phi(-C_0 - \delta)]L^{-1} + o(L^{-1}),
\end{aligned}$$

where

$$a = \frac{\phi(C_0)C_0^2}{2\sqrt{1-\alpha}},$$

which is calculated from the proposition 3.4.3.

3.8.7 Proof for Theorem 3.4.1

Let $\rho = \text{Cov}(Y_1(i), Y_L(i))$. From the equation 3.3, we know when the alternative hypothesis is true, we have the marginal distribution of $Y_1(i)$, and $Y_L(i)$ are $Y_1(i) \sim N(\delta, 1)$, $Y_L(i) \sim N(\mu_L, 1)$, where $\mu_L = \frac{K\delta}{L^H}$, and $K = 1, 2, \cdots, L$.

From the Lemma 3.8.2, the power at scale 1 and L is given by

$$\beta_1 = 1 - [\Phi(C_\alpha - \delta) - \Phi(-C_\alpha - \delta)],$$

$$\beta_L = 1 - [\Phi(C_\alpha - \mu_L) - \Phi(-C_\alpha - \mu_L)].$$

When K and δ are fixed, let $L \to \infty$, we have $\rho \to 0$, $\mu_L \to 0$, and

$$\beta_L = \alpha + O(L^{-2H}),$$

$$\beta_{(1,L)} = 1 - [\Phi(C_0 - \delta) - \Phi(-C_0 - \delta)]\sqrt{1-\alpha} + O(L^{-1}),$$

which can be directly derived from Lemma 3.8.2 and Lemma 3.8.4.

Define the power difference function $f(\alpha, \delta)$ as

$$\begin{aligned} f(\alpha,\delta) = {} & \left[\Phi\left(\Phi^{-1}\left(1-\frac{\alpha}{2}\right)-\delta\right)-\Phi\left(-\Phi^{-1}\left(1-\frac{\alpha}{2}\right)-\delta\right)\right]+[1-\alpha] \\ & -2\sqrt{1-\alpha}\left[\Phi\left(\Phi^{-1}\left(\frac{1+\sqrt{1-\alpha}}{2}\right)-\delta\right)-\Phi\left(-\Phi^{-1}\left(\frac{1+\sqrt{1-\alpha}}{2}\right)-\delta\right)\right]. \end{aligned}$$

We have $2\beta_{(1,L)} - (\beta_1 + \beta_L) = f(\alpha, \delta) + O(L^{-1})$, when $\alpha \to 0$. Thus we only need to show $f(\alpha, \delta) > 0$ as $\alpha \to 0$.

Note that $\alpha = 0$, $f(\alpha, \delta) = 2 - 2 = 0$.

$$\begin{aligned} \frac{\partial f(\alpha,\delta)}{\partial\alpha} = {} & -\frac{1}{2}\exp\left\{-\frac{\delta^2}{2}\right\}[\exp\{\delta C_\alpha\}+\exp\{-\delta C_\alpha\}] - 1 + \frac{1}{\sqrt{1-\alpha}}[\Phi(C_0-\delta)-\Phi(-C_0-\delta)] \\ & + \frac{1}{2}\exp\{-\frac{\delta^2}{2}\}[\exp\{\delta C_0\}+\exp\{-\delta C_0\}]. \end{aligned}$$

By the *mean value theorem* in Calculus (e.g., page 43 in Beals (1973)),

$$\begin{aligned} C_0 - C_\alpha &= \Phi^{-1}\left(\frac{1+\sqrt{1-\alpha}}{2}\right) - \Phi^{-1}\left(1-\frac{\alpha}{2}\right) \\ &= \frac{1}{\phi(\Phi^{-1}(\xi))}\left[\frac{\sqrt{1-\alpha}-(1-\alpha)}{2}\right], \end{aligned} \tag{3.10}$$

where $\xi \in (1-\alpha/2, (1+\sqrt{1-\alpha})/2)$.

Equation (3.10) implies

$$\frac{1}{\phi(C_\alpha)}\left[\frac{\sqrt{1-\alpha}-(1-\alpha)}{2}\right] \le C_0 - C_\alpha \le \frac{1}{\phi(C_0)}\left[\frac{\sqrt{1-\alpha}-(1-\alpha)}{2}\right]. \tag{3.11}$$

Since

$$\begin{aligned} \lim_{\alpha\to 0+}\frac{1}{\phi(C_0)}\left[\frac{\sqrt{1-\alpha}-(1-\alpha)}{2}\right] &= \lim_{\alpha\to 0+}\frac{\frac{1}{2}\times[\frac{1}{2}(1-\alpha)^{-1/2}\times(-1)+1]}{\Phi^{-1}(\frac{1+\sqrt{1-\alpha}}{2})\times\frac{1}{4}\times(1-\alpha)^{-1/2}} = 0, \\ \lim_{\alpha\to 0+}\frac{1}{\phi(C_\alpha)}\left[\frac{\sqrt{1-\alpha}-(1-\alpha)}{2}\right] &= \lim_{\alpha\to 0+}\frac{\frac{1}{2}\times[\frac{1}{2}(1-\alpha)^{-1/2}\times(-1)+1]}{\Phi^{-1}(1-\frac{\alpha}{2})\times\frac{1}{2}} = 0, \end{aligned}$$

we have

$$\lim_{\alpha\to 0+}(C_0 - C_\alpha) = 0.$$

Thus, when $\alpha \to 0$,

$$\exp\{\delta(C_0 - C_\alpha)\} - 1 = \delta(C_0 - C_\alpha) + o(\delta(C_0 - C_\alpha)),$$

which leads to

$$\begin{aligned}\exp\{\delta C_0\} - \exp\{\delta C_\alpha\} &= \exp\{\delta C_\alpha\}\exp\{\delta(C_0 - C_\alpha) - 1\} \\ &= \exp\{\delta C_\alpha\}[\delta(C_0 - C_\alpha) + o((C_0 - C_\alpha))].\end{aligned}$$

By equation (3.11),

$$\begin{aligned}\lim_{\alpha\to 0+} \exp\{\delta C_\alpha\}(C_0 - C_\alpha) &\geq \lim_{\alpha\to 0+} \frac{(\sqrt{1-\alpha} - (1-\alpha))/2}{\exp\{-\delta C_\alpha\}\phi(C_\alpha)} \\ &= \lim_{\alpha\to 0+} \frac{\frac{1}{2} \times [1 - \frac{1}{2}(1-\alpha)^{-1/2}]}{\exp\{-\delta C_\alpha\}[\frac{\delta + C_\alpha}{2}]} \\ &= +\infty.\end{aligned}$$

This yields that $\exp\{\delta C_0\} - \exp\{\delta C_\alpha\} \to +\infty$, when $\alpha \to 0+$, i.e.

$$\frac{\partial f(\alpha,\delta)}{\partial \alpha}|_{\alpha\to 0+} > 0.$$

From the above, we know that for any $\delta > 0$, there exists α_δ, such that for any $\alpha \in (0, \alpha_\delta)$, we have

$$f(\alpha,\delta) > 0,$$

hence the theorem holds.

3.8.8 Proof for Theorem 3.4.2

Let b the aggregation bin size, which is defined in Section 3.4.4, and $l = b, b^2, \cdots, b^k, \cdots$ are the aggregation scales. Define $\xi_b(k) = Y_{b^k}(1)$. Proposition 3.4.2 shows that

$$\begin{aligned}\rho_b(k) &= \mathrm{Cov}(\xi_1, \xi_{k+1}) \\ &= Hb^{k(H-1)} + \frac{b^{-kH}}{2} - \frac{H(2H-1)}{2} b^{k(H-2)} + o(b^{k(H-2)}).\end{aligned}$$

This follows directly from Lemma 3.8.3. Note that when $k \to \infty$, the leading term of the above is $Hb^{k(H-1)}$, which decays exponentially. Thus, it is straightforward that

$$\lim_{k\to\infty} \rho_b(k) \log k = 0$$

Lemma 3.4.1 (Berman (1964), see also Leadbetter et al. (1983)) leads the result.

Chapter 4

Summary and Comments

In this chapter, we summarize all the work we have already done, and comment about some potential future work.

4.1 Functional SVD

For the SVD, PCA and related Visualizations, we obtained the following results in Chapter 2:

1. In Section 2.3, we explored the connections and differences between the SVD and PCA methods, especially from an FDA viewpoint. We have extended the usual SVD to include four types of centerings: SSVD, CSVD, RSVD and DSVD. Recall that the usual PCA method is exactly the CSVD in this general framework.

2. For selection of the four types of centerings, three criteria are recommended, including approximation performance, model complexity and model interpretability. A generalized scree plot has been proposed as a starting point for model selection, which provides simple understanding of the tradeoff between approximation performance and model complexity. To get the most appropriate model, users should also consider interpretability. Other considerations might overrule the scree rules based on the generalized scree plot, as discussed in Section

2.3.4. Several toy examples have been designed in Section 2.5 to illustrate important points about the model selection.

3. We explored the geometric interpretations for the overall mean, the column mean, the row mean, the double mean and their further SVD components, by using a 150×2 data set, in Section 2.3.3.

4. Several matrix views of the SVD components have been proposed to explore underlying features of the data matrix, including the surface plots, image plots, curve movies and rotation movies. All these novel visualization methods can be used to find underlying features of a two-way data matrix. Several real applications have been analyzed using these visualization methods, and presented in Section 2.2 and Section 2.6.

4.2 The MRAD method

For the MRAD method for a time series with long range dependence, we have the following results in Chapter 3.

1. The MRAD procedure has been proposed in Section 3.4.4. The MRAD procedure described there is a pointwise testing method. We used two types of simple aggregation method (i.e., non-overlapping window aggregation and sliding window aggregation, discussed in Section 3.3) to form multiscale time series. After that, we use the observations at the same time locations but at different scales to form a hypothesis testing problem. When the series does not contain any outliers, the marginal distribution of the observations over difference scales and locations are standard Normal. The testing threshold based on the MRAD method has been proved to have a larger threshold than the detection method based on one single scale,

i.e., it provides a more conservative testing threshold.

2. In Section 3.4.5, we have provided the theoretical upper limit of the testing threshold for a two-scale MRAD procedure. We also provided the convergence rate of the threshold to the upper limit. This leads to an approach to calculate the testing threshold theoretically.

3. We have proved theorem 3.4.1 which shows that a two-scale MRAD method has larger power on average than tests based on any single scale, under suitable conditions (see Chapter 3.4.5 for details, and Section 3.8 for the justifications). In order to prove the above theorem, we have explored the correlations between the observations at the same time location but at different scales.

4. By using some results in Extreme value theory, we explored the asymptotic distribution of our test statistics and developed a simplified Bonferroni type asymptotic threshold (Theorem 3.4.2 in Section 3.4.5). An improved test threshold, the estimate of the exact test threshold, is developed in Section 3.5. This threshold depends only on the Hurst parameter of the time series, significance level and number of scales used in the procedure.

5. We have proposed an MRAD outlier map to visualize the significance probabilities of the observations over different locations and scales. In Section 3.6, we used simulations, semi-experiments and real applications to illustrate the MRAD method and the MRAD outlier map. This map is first introduced in Section 3.2 along with the motivation example. Several summaries, including FDR, FNR, and TDR, are reported to evaluate this proposed method. To highlight results at each scale, we also proposed an MRAD outlier movie, which highlights each scale in turn. A zoomed version of the MRAD map is useful for viewing the outliers locally.

4.3 Future work

There are many potential future research problems related to functional SVD and the MRAD method. Here we report some of them as my future research topics.

SVD and PCA

1. For the visualization method we proposed in Chapter 2, we have provided some MATLAB functions, which can be downloaded at Zhang (2006b). We plan to develop an R package as well, to incorporate our visualization methods and other existing methods.

2. Low dimensional projections of high-dimensional data sets have attracted much attention in Statistics in recent years. We plan to explore more visualization methods for high-dimensional data sets based on SVD and PCA.

3. SVD is in fact a least square estimation of a bilinear model (Gabriel and Odoroff, 1984; Gabriel, 1978). Thus the estimation of the SVD components will be sensitive to outliers. We plan to explore the different types of outliers (especially outliers from the FDA view point) in a two-way data matrix, and study how those outliers affect the estimation. We also intend to develop a new robust version of SVD, (for example, the extension of the robust PCA method in Locantore et al. (1999)) and compare it with some current methods (for example Hawkins et al. (2001); Liu et al. (2003)).

4. Asymptotic study of the estimation of the singular columns and singular rows is expected to give many interesting and insightful results, when the number of columns or the number of rows goes to infinity (or both).

The MRAD method

1. As discussed in Section 3.2, we will have to standardize the time series before we form multiresolution time series. We intend to develop or incorporate some robust procedures for standardization, such that the outliers will have minor impact in the normalization step.

2. The aggregation method here assumes a known Hurst parameter. In real applications, we need to incorporate some robust estimation methods for the Hurst parameter (for example Shen et al. (2005)). We plan to explore and incorporate robust estimation method of the Hurst parameter into our program, and provide a solution when the Hurst parameter is unknown.

3. In the pointwise testing problem, we will explore other testing statistics from different time scales, for example, the likelihood ratio test statistics. The related testing thresholds will also be explored, both theoretically and empirically.

4. The MRAD method can be viewed as the approximation part of a Haar wavelet. Thus we can naturally extend our MRAD method into the wavelet framework. Theoretical properties will be explored.

5. The outlier format in Chapter 3 is level shift. We plan to explore other types of outlier, including additive outlier, innovation outlier, etc (Chang et al., 1988; Tsay, 1988).

6. Multiple comparison issues in the time space should be properly accounted for, and incorporated into the MRAD method. The usual FDR context does not assume a strong dependence structure. We plan to explore the FDR method or develop new methods to perform multiple comparison for the LRD observations.

Bibliography

Barford, P., Kline, J., Plonka, D., and Ron, A. (2002), "A Signal Analysis of Network Traffic Anomalies," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement*, pp. 71–82.

Barnett, V. and Lewis, T. (1994), *Outliers in Statistical Data*, John Wiley & Sons, 3rd ed.

Beals, R. (1973), *Advanced Mathematical Analysis*, Springer-Verlag.

Benjamini, Y. and Hochberg, Y. (1995), "Controlling the False Discovery Rate: a Practical and Powerful Approach to Multiple Testing," *Journal of Royal Statistical Society, Series B*, 57, 289–300.

Berman, S. M. (1964), "Limit theorems for the maximum term in stationary sequences," *Annals of Mathematical Statistics*, 35, 502–516.

Box, G. E. P., Jenkins, G. M., and Reinsel, G. C. (1994), *Time Series Analysis, Forecasting and Control (3rd Edition)*, Prentice Hall, 3rd ed.

Box, G. E. P. and Tiao, G. C. (1975), "Intervention Analysis with Applications to Economic and Environmental Problems," *Journal of the American Statistical Association*, 70, 70–79.

Brockwell, P. J. and Davis, R. A. (1991), *Time Series: Theory and Methods*, Spinger-Verlag.

Cao, J., Cleveland, W. S., Lin, D., and Sun, D. X. (2002), "Internet traffic tends toward Poisson and Independent as the load increases," in *Nonlinear Estimation and Classification*, Springer.

Cattell, R. B. (1966), "The scree test for the number of factors," *Multivariate Behavioral Research*, 1, 245–276.

Chang, I., Tiao, G. C., and Chen, C. (1988), "Estimation of Time Series Parameters in the Presence of Outliers," *Technometrics*, 30, 193–204.

Cheng, C., Kung, H. T., and Tan, K. (2002), "Use of Spectral Analysis in Defense Against DoS Attacks," in *Proceedings of IEEE GLOBECOM 2002*.

Davis, R. A. (1979), "Maxima and Minima of Stationary Sequences," *The Annals of Probability*, 7, 453–460.

Embrechts, P. and Maejima, M. (2002), *Selfsimilar Processes*, Princeton University Press, ISBM 0-691-09627-9.

Fox, A. J. (1972), "Outliers in Time Series," *Journal of the Royal Statistical Society. Series B*, 34, 350–363.

Gabriel, K. R. (1971), "The biplot graphic display of matrices with application to principal component analysis," *Biometrika*, 58, 453–467.

— (1978), "Least Squares Approximation of Matrices by Additive and Multiplicative Models," *Journal of Royal Statistics Society, Series B*, 40, 186–196.

Gabriel, K. R. and Odoroff, C. L. (1984), "Resistant lower rank approximation of Matrices," in *Data analysis and Informatics*, pp. 23–30.

Gijbels, I., Hall, P., and Kneip, A. (1999), "On the Estimation of Jump Points in Smooth Curves," *Annals of the Institute of Statistical Mathematics*, 51, 231–251.

Golyandina, N., Nekrutkin, V., and Zhigljavsky, A. (2001), *Analysis of Time Series Structure*, Chapman and Hall.

Gower, J. C. and Hand, D. J. (1996), *Biplots*, Chapman and Hall.

Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., and Stahel, W. A. (1986), *Robust Statistics: the approach based on Influence Functions*, John Wiley & Sons.

Hannig, J., Marron, J. S., and Riedi, R. H. (2001), "Zooming statistics: Inference across scales," *Journal of the Korean Statistical Society*, 30, 327–345.

Hawkins, D. M. (1980), *Identification of Outliers*, Chapman and Hall.

Hawkins, D. M., Liu, L., and Young, S. S. (2001), *Robust Singular Value Decomposition*, technical report, NISS 122.

HMD (2005), "Human Mortality Database," University of California, Berkeley (USA), and Max

Planck Institute for Demographic Research (Germany). Available at http://www.mortality.org or http://www.humanmortality.de.

Householder, A. S. and Young, G. (1938), "Matrix approximations and latent roots," *American Mathematical Monthly*, 45, 165–171.

Hu, W., Liao, Y., and Vemuri, V. R. (2003), "Robust Support Vector Machines for Anomaly Detection in Computer Security," in *Proceedings of the 2003 International Conference on Machine Learning and Applications*, pp. 168–174.

Huang, J. Z., Shen, H., and Buja, A. (2007), *Principal Component Analysis of Two-Way Functional Data Using Two-Way Regularized Singular Value Decompositions*, technical report.

Jeffay, K. (2005), "Course Website for COMP 290: Network Intrusion Detection," Course of Department of Computer Science in the University of North Carolina at Chapel Hill. Materials are available at http://www.cs.unc.edu/˜jeffay/courses/nidsS05/.

Jolliffe, I. T. (2002), *Principal Component Analysis*, Springer-Verlag.

Jung, J., Paxson, V., Berger, A. W., and Balakrishnan, H. (2004), "Fast Portscan Detection Using Sequential Hypothesis Testing," in *Proceedings of 2004 IEEE Symposium on Security and Privacy.*

Kluger, Y., Basri, R., Chang, J. T., and Gerstein, M. (2003), "Spectral Biclustering of Microarray Data: Coclustering Genes and Conditions," *Genome Research*, 13, 703–716.

Kurose, J. F. and Ross, K. W. (2001), *Computer Network, A Top-Down Approach Featuring the Internet*, Addison Wesley.

Lakhina, A., Crovella, M., and Diot, C. (2004a), "Characterization of Network-Wide Anomalies in Traffic Flows," in *Proceedings of the ACM/SIGCOMM Internet Measurement Conference*, pp. 201–206.

— (2004b), "Diagnosing Network-Wide Traffic Anomalies," in *ACM SIGCOMM Computer Communication Review*, vol. 34, pp. 219–230.

Lakhina, A., Papagiannaki, K., Crovella, M., Diot, C., Kolaczyk, E. D., and Taft, N. (2004c), "Structural Analysis of Network Traffic Flows," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 32, pp. 61–72.

Lane, T. and Brodley, C. E. (1997), “Detecting the Abnormal: Machine Learning in Computer Security,” .

Le, L. and Hernández-Campos, F. (2004), “UNC Network Data Analysis Study Group: Summary Page for LRD Project,” Website available at http://www-dirt.cs.unc.edu/net_lrd/, created and maintained by the two authors.

Leadbetter, M. R., Lindgren, G., and Rootzén, H. (1983), *Extremes and Related Properties of Random Sequences and Processes*, Springer-Verlag.

Lee, C. B., Roedel, C., and Silenok, E. (2003), *Detection and Characterization of Port Scan Attacks.*

Lee, W. and Fan, W. (2001), “Mining system audit data: Opportunities and challenges,” in *ACM SIGMOD Record*, vol. 30, pp. 35–44.

Lee, W., Stolfo, S., and Mok, K. (1999), “A Data Mining Framework for Building Intrusion Detection Models,” in *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, pp. 120–132.

Leland, W. E., Taqqu, M. S., Willinger, W., and Wilson, D. V. (1994), “On the Self-Similar Nature of Ethernet Traffic (Extended Version),” *IEEE/ACM Transactions on Networking*, 2, 1–15.

Liu, L., Hawkins, D. M., Ghosh, S., and Young, S. S. (2003), “Robust singular value decomposition analysis of Microarray data,” *PNAS*, 100, 13167–13172.

Locantore, N., Marron, J. S., Simpson, D. G., Tripoli, N., Zhang, J. T., and Cohen, K. L. (1999), “Robust Principal Component Analysis for functional data,” *Test*, 1–73.

Long, C. (1983), “Visualization of Matrix Singular Value Decomposition,” *Mathematics Magazine*, 56, 161–167.

Mandelbrot, B. B. and Van Ness, J. W. (1968), “Fractional Brownian motions, fractional noises and applications,” *SIAM Review*, 10, 422–437.

Marron, J. S., Wendelberger, J. R., and Kober, E. M. (2004), “Time Series Functional Data Analysis,” Los Alamos National Lab, Technical Report Number: LA-UR-04-3911.

Martin, R. D. and Yohai, V. J. (1986), “Influence Functionals for Time Series,” *The Annals of Statistics*, 14, 781–818.

McHugh, J. (2001), "Intrusion and Intrusion Detection," *International Journal of Information Security*, 1, 14–35.

Muller, N., Magaia, L., and Herbst, B. M. (2004), "Singular Value Decomposition, Eigenfaces, and 3D Reconstructions," *SIAM Review*, 46, 518–545.

Ogden, R. T. (1997), *Essential Wavelets for Statistical Applications and Data Analysis*, Boston: Birkhäuser.

Okamoto, M. (1972), "Four techniques of Principal Component Analysis," *Journal of Japanese Statistical Society*, 2, 63–69.

Paxson, V. (1999), "Bro: A System for Detecting Network Intruders in Real-Time," *Computer Networks*, 31, 2435–2463.

Paxson, V. and Floyd, S. (1995), "Wide-Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking*, 3, 226–244.

Ramsay, J. O. and Silverman, B. W. (2002), *Applied Functional Data Analysis, Methods and Case Studies*, Springer-Verlag.

— (2005), *Functional Data Analysis (Second Edition)*, Springer-Verlag.

Roesch, M. (1999), "Snort - Lightweight Intrusion Detection for Networks," in *Proceedings of LISA 9́9: 13th Systems Administration Conference*, pp. 229–238.

Rolls, D. A., Michailidis, G., and Hernández-Campos, F. (2005), "Queueing analysis of network traffic: methodology and visualization tools," *Computer Networks*, 48, 447–473.

Search-Security-Definitions (2005), "Search Security Definitions: Port Scan," Available at http://searchsecurity.techtarget.com/sDefinition/0,290660,sid14_gci214054,00.html. Contributor is Stephanie Ireland.

Seely, D. (1989), "A Tour of the Worm," in *Proceedings of the Winter 1989 Usenix Conference, San Diego, CA*, pp. 287–304.

Shen, H. and Huang, J. Z. (2005), "Analysis of Call Center Arrival Data Using Singular Value Decomposition," *Applied Stochastic Models in Business and Industry*, 21, 251–263.

Shen, H., Zhu, Z., and Lee, T. C. M. (2005), *Robust Estimation of Self-Similarity parameter in Network Traffic using Wavelet Transform*, technical report No. UNC/STOR/05/04.

Sommer, R. and Paxson, V. (2003), "Enhancing Byte-Level Network Intrusion Detection Signatures with Context," in *Proceedings of the 10th ACM conference on Computer and Communications Security*, pp. 262–271.

Spafford, E. H. (1989), "The Internet Worm Program: An Analysis," in *ACM SIGCOMM Computer Communication Review*, vol. 19, pp. 17–57.

Specht, S. and Lee, R. (2004), "Distributed Denial of Service: Taxonomies of Attacks, Tools, and Countermeasures," in *Proceedings of the 17th International Conference on Parallel and Distributed Computing Systems*, pp. 543–550.

Stevens, W. R. (1994), *TCP/IP illustrated, Vol. 1: The Protocols*, Addison-Wesley.

Stolfo, S., Lee, W., Chan, P., Fan, W., and Eskin, E. (2001), "Data Mining-based Intrusion Detectors: An Overview of the Columbia IDS Project," in *ACM SIGMOD Record*, vol. 30, pp. 5–14.

Taqqu, M. S. (2003), "Fractional Brownian Motion and Long-Range Dependence," in *Theory and Applications of Long-Range Dependence*, Birkhäuser, pp. 5–38.

Terrell, J., Zhang, L., Jeffay, K., Nobel, A., Shen, H., Smith, F. D., and Zhu, Z. (2005), "Multivariate SVD Analyses For Network Anomaly Detection," Refered Poster of ACM SIGCOMM Philadelphia, Aug 2005.

Tiao, G. C. and Tsay, R. S. (1983), "Consistency properties of least squares estimates of autoregressive parameters in ARMA models," *The Annals of Statistics*, 11, 856–871.

Tsay, R. S. (1988), "Outlier, Level Shifts, and Variance Changes in Time Series," *Journal of Forecasting*, 7, 1–20.

Vidakovic, B. (1999), *Statistical Modeling By Wavelets*, Wiley.

Wand, M. P. and Jones, M. C. (1995), *Kernel Smoothing*, Chapman and Hall.

Wang, H. and Marron, J. S. (2006), "Object Oriented Data Analysis: Sets of Trees," *Annals of Statistics*, submitted.

Willinger, W., Taqqu, M. S., and Erramilli, A. (1996), “A Bibliographical Guide to Self-Similar Traffic and Performance Modeling for Modern High-Speed Networks,” in *Stochastic Networks: Theory and Applications*, eds. Kelly, F. P., Zachary, S., and Ziedins, I., Oxford University Press, pp. 339–366.

Willinger, W., Taqqu, M. S., Sherman, R., and Wilson, D. V. (1997), “Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level,” *IEEE/ACM Transactions on Networking*, 3, 71–86.

Zhang, L. (2006a), “Multiresolution Anomaly Detection Programs, Images and Movies,” Available at http://www.unc.edu/˜lszhang/research.

— (2006b), “SVD movies and plots for Singular Value Decomposition and its Visualization,” Available at http://www.unc.edu/˜lszhang/research/network/SVDmovie/.

Zhang, L., Marron, J. S., Shen, H., and Zhu, Z. (2006), “Singular Value Decomposition and its Visualization,” *Journal of Computational and Graphical Statistics*, accepted.