9-2021

New Methods in Wavelet Analysis for Applications of the Wavelet Transform

Jeffrey D. Williams

New Methods in Wavelet Analysis for Applications of the Wavelet Transform

THESIS

Jeffrey D. Williams, Capt, USAF

AFIT-ENS-DS-21-S-055

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENS-DS-21-S-055

NEW METHODS IN WAVELET ANALYSIS FOR APPLICATIONS OF THE WAVELET TRANSFORM

THESIS

Presented to the Faculty

Department of Operational Sciences

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Doctorate of Operations Research

Jeffrey D. Williams, BS, MS

Capt, USAF

March 26, 2021

AFIT-ENS-DS-21-S-055

NEW METHODS IN WAVELET ANALYSIS FOR APPLICATIONS OF THE WAVELET TRANSFORM

THESIS

Jeffrey D. Williams, BS, MS
Capt, USAF

Committee Membership:

Dr. Raymond R. Hill, PhD
Chair

Dr. Joseph Pignatiello, Jr, PhD
Member

Dr. Edward D. White, PhD
Member

Dr. Eric Chicken, PhD
Member

Adedeji B. Badiru, PhD
Dean, Graduate School of Engineering and Management

AFIT-ENS-DS-21-S-055

Abstract

Common in signal processing (SP) is the detection of events. For instance, seismologists seek to detect abnormalities in an electromagnetic (EM) signal to detect the occurrence of an earthquake. Since many signals are noisy, such as those produced by a seismograph, it can be challenging to distinguish a significant EM pulse from incident noise. In SP, smoothing is used to remove the rough portions of a signal representing noise such that events are more obvious in a signal. This research applies and improves wavelet analysis methods across multiple domains and applications of signals since Wavelet analysis smooths signals while preserving important signal artifacts such as a large EM pulse representing an earthquake. Further, there are several useful properties of wavelet analysis such as time localization and sparsity which improve detection ability in SP. In this dissertation, we explore several applications, and domains of SP such as classical data, functional data, and graph data. We improve event detection such as outliers, and introduce new methods to detect and remove noise across these domains to improve SP analysis.

Table of Contents

Page

List of Figures

Figure Page

Figure Page

List of Tables

NEW METHODS IN WAVELET ANALYSIS FOR APPLICATIONS OF THE WAVELET TRANSFORM

I. Introduction

In the many statistical applications of wavelets in literature, the research focus is often integrating wavelet analysis into new areas of research. In this paper we review three of these areas, classical, functional, and graphical signal processing. Wavelet analysis is valuable to these modes for the many attractive wavelet properties which result in smooth and compact signal reconstructions. However, little attention is given to wavelet thresholding which is an important area of study in wavelet analysis for classical signal processing. Thresholding or wavelet shrinkage consistently improves the extraction of artifacts from a noisy signal Donoho and Johnstone (1994). In this paper we improve the application of wavelet analysis to signal processing domains through analysis, and novel application of wavelet thresholding in signal processing domains.

Classical signal processing problems extend to radar, communications, speech, SONAR, image processing, biomedical, control, and seismology; where a key concern is discerning whether an approaching aircraft or earthquake appear or do not appear in noisy electromagnetic signals Kay (1993). These signals are often one dimensional times series signals. In the classical SP, wavelet analysis is a popular method for smoothing signals and recently for event detections in Atkinson et al. (2018b). Debauchies (1992) shows for many SP applications wavelet analysis outperforms spline models and Fourier analysis for classical signal smoothing. Donoho and Johnstone (1994) present wavelet thresholding methods which further improve upon intrinsic

smoothing of wavelet analysis. Thresholding, which removes or shrinks noisy wavelet data, has been refined significantly to improve noisy signal reconstruction. Our first contribution uses a Hidden Markov Model (HMCV) thresholding approach to block similar wavelet coefficients together for thresholding. Typically block thresholding is more effective than term-by-term thresholding yet methodically grouping wavelet data using HMM is shown in our first paper to improve signal reconstruction and shown to outperform current methods for a detections application.

Ramsay and Silverman (2007) present several functional SP examples; Law enforcement organizations are interested in analyzing handwriting samples, paleontologists are interested in analyzing bone densities in X-Rays, and meteorologists are interested in comparing yearly Sea Surface Temperatures for an El Nino Regions. These data differ from classical SP in that each sample or piece of data is a function. This is called functional data analysis. In the second paper we explore outlier analysis in functional data. Outliers are prevalent in nature, and can influence statistical inferences sought from the populations that contain them. Functional data do not have a well established statistical method for detecting the presence of outliers nor a measure of their potential influence. In some functional data, such as time series, observations that might be classified as outliers present themselves over a given period, cycle, or frequency, that are irregular for their spatial location. Ramsay and Silverman (2007) present several functional methods which are classical methods repurposed for functional data; these methods include functional principal components, and functional discriminant analysis. James and Hastie (2001) uses functional principal components to determine outliers in functional data, but these methods are not effective for functional data with non-Gaussian noise. Sun and Genton (2011) present the functional box plot which uses robust principal components to determine the location of outliers in functional data. We present the Wavelet Analysis of Variance (WANOVA) box

plot for outlier detection. Our second paper compares the WANOVA box plot with the functional box plot of Sun and Genton (2011), the rainbow plots of Hyndman and Shang (2010a), and methods presented in Ramsay and Silverman (2007) and show it is a more robust method for functional outlier detection and analysis.

Many signals have an underlying data structure which effect how the signals behave, such that we need to include the structure in the analysis. Ortega et al. (2018) mentions that data flows on social networks like virtual highways, moving rapidly between connected users. Traffic data analysis, is guided by the physical highways on which vehicles travel. Graphical signal processing (GSP) is a mode of signal processing where the structure of the data is integrated into the analysis. Signals on the graph domain are often noisy and need to be smoothed. Hammond et al. (2011) presents the spectral graph wavelet transform (SGWT) which generates wavelet operators based on the structure of the graph and smooths the graph signal. de Loynes et al. (2021) presents a SUREshrink thresholding method from Donoho and Johnstone (1995)for graph wavelets which further smooths the graphs signal. In our third work we present a BTWave thresholding method based on the Neighblock method presented by Cai and Silverman (2001). BTWave outperforms the SGWT and the SGWT with SUREshrink for smoothing graph signals.

In chapter three we present wavelet analysis and discuss the relevant signal processing domains and associated concepts to provide a background for our three contributions to wavelet research. In chapter four we present the WANOVA Box plot, which uses a thresholding technique and the WANOVA method to detect outliers in functional data. In Chapter five, we present a wavelet thresholding method for classical and functional signal processing using Hidden Markov Processes. In Chapter six we present a wavelet thresholding technique for GSP using graph partitioning. In chapter seven we conclude and present our main contributions to wavelet analysis.

II. Background

2.1 Signal Processing

In this paper we present wavelet analysis methods that improve upon signal processing techniques, with particular focus on more recent applications of signal processing.

Signal processing originated from communication experiments in the 19^{th} century and emerged into several well established signal processing functional areas in the early to mid 20^{th} century. These areas include radar, communications, speech, sonar, image processing, biomedicine, control and seismology. The common goal of these classic signal process areas is detection; that is discovering whether an event of interest occurs and eventually extracting that event from noise (Kay, 1993). Kay (1993) presents the scenario of a radar determining whether an airplane is approaching or not. Detection through signal processing would determine that there was actually a plane approaching and that portion of the signal would be extracted.

2.2 Wavelet Preliminaries

This primary technique that ties this research together is the application of wavelet analysis across multiple domains. Wavelets are carefully constructed orthonormal functions which through expansion and dilation are used to approximate a signal (Debauchies, 1992). Wavelets are robust in their approximation applications. They are used for compression, statistical process control (SPC), denoising, image analysis, density analysis, and nonparametric regression Chicken (2003). One well known example of wavelet data compression is JPEG 2000 which uses wavelets to compress photos into smaller components for electronic transmission. This research focuses on a few of the wavelet applications such as denoising images and signals. The domain

of the data to which wavelets are applied in this research are the functional domain to include time series and performance curve data as well as the graph domain. Techniques from classical signal processing using wavelets are improved and extended to these doamins.

2.3 Introduction to Wavelets

Wavelets are mentioned in a few instances as designed for functional data. This is because wavelets were originally designed for signal processing and profile monitoring which are inherently functional. This nuanced difference between wavelets and multivariate methods extends to the functional domain make wavelets especially useful for functional analysis. Wavelets transformations filter data into two filter banks which measure general and detailed features of a signal. This two factor characteristic of wavelets is effective at highlighting both the magnitude and shape features persistent in functional outliers.

2.4 Wavelets

Wavelet transforms, an extension from the Fourier transform, are adept for functional analysis and removing noise from signals.

> "The wavelet transform is a tool that cuts up data or functions or operators into different frequency components, and then studies each component with a resolution matched to its scale" (Debauchies, 1992)

Wavelet coefficients are generated through a series of expansions and dilations using a family of functions called the mother (ψ) and father wavelets (ϕ). The mother and father wavelet differ based on the family of wavelets used. Using multi-resolution analysis and the mother and father wavelet, (1) wavelet coefficients are generated

from a signal. This process is called the discrete wavelet transform (DWT). The DWT "[decomposes] a time series into time frequency space one is able to determine both the dominant modes of variability and how those modes vary in time" (Torrence and Compo, 1979). This is depicted in equation (1).

$$f(t) = \sum_{k} c_{j_0,k}\phi_{j_0,k} + \sum_{j>j_0}\sum_{k} d_{j,k}\psi_{j,k} \tag{1}$$

The different modes of variability are represented by the coarse coefficients (c_{j0}) on the mother wavelet, and the detailed coefficients ($d_{j,k}$) on the father wavelet. The benefit of the wavelet transform is that individual frequency components can be localized over time. Time localization makes it possible to map wavelet coefficients to a specific data point in functional data.

2.4.1 Thresholding

Thresholding wavelet coefficients is a necessary step succeeding the DWT in order to remove noise from the sparse resolution levels of the DWT. Removing noise from the wavelet coefficients results in a smoother signal when the thresholded coefficients are reconstructed through the IDWT. Thresholding is applicable to every application, however, different types of thresholding are suitable to different applications. There are two types of wavelet coefficient thresholding: term-by-term and block thresholding. The popular thresholding procedures of each type and their applications will be discussed. Additionally, recent research has combined both types of thresholding with statistical techniques that improve their efficacy.

2.4.2 Soft and Hard Thresholding

The methods for calculating a threshold value based on a set of data are vast, however there are two main thresholding schemes in practice. The first scheme is

hard thresholding. Under the hard thresholding scheme the group or individual data are either kept or killed depending on the threshold value. Coefficients or groups of coefficients exceeding the hard threshold value are kept and the remainder are "killed" being set to zero. The soft threshold scheme shrinks coefficients by the value of the threshold.

2.4.3 Term by Term Threshold Methods

Term-by-term thresholding occurs when each term is thresholded individually. A wavelet coefficient is either kept, shrunk or killed based on the threshold value and thresholding scheme. Donoho and Johnstone (1994) VisuShrink, and SUREShrink methods were developed as term-by-term methods. These methods are examples of hard thresholding techniques, proposed by Donoho and Johnstone (1994). SUREShrink is an adaptation of VisuShrink. SUREShrink applies level dependent thresholds and applies the SURE threshold in the case of extremely spares levels.

VisuShrink is well suited for applications where noise is unacceptable. VisuShrink removes more noise from a signal than any other method, but has a byproduct of removing some signal. VisuShrink is a useful method for SPC such as the case of determining a difference between a model and a system. Atkinson analyzes such a case in Atkinson et al. (2018b). Removing maximum noise will decrease the WANVOA coefficients where noise is a contributing factor to the difference between a model and system, heightening the importance of coefficients where the actual signals are different. In contrast VisuShrink is a poor method for removing noise from an image as some of the coefficients representing important imagery in the photo may be removed.

2.4.4 Block Threshold Methods

Block threshold techniques refer to thresholding of wavelet coefficients in groups rather than individually. Most of the block thresholding techniques use the soft thresholding scheme. This means wavelets coefficient groups are shrunk by a specific threshold value rather than kept or killed. Block thresholding is presented in Cai (1999) and McGinnity et al. (2017). A popular method known as Bayeshrink uses block thresholding.

> "By looking at coefficients in blocks rather than individually, more precise comparisons between the coefficients and the threshold is allowed, resulting in improved rates" (Chicken, 2003)

Using information on neighboring wavelet coefficients improves the precision of the technique in some applications. The block threshold technique used in this research is presented by Cai (1999). The first step subsequent to the DWT is determining the block width for the groups of wavelet coefficient. Cai (1999) presents, L = log(n), as a proven block length, n is the number of data points in the wavelet domain. Cai (1999) shows that $L \geq 3$ dominates the mini-max error rates of term-by-term thresholding techniques for some applications (Cai, 1999). The final step is simultaneously calculating the different threshold values and applying the soft thresholding scheme to the data. The threshold for each block is calculated according to equation (2). The value of λ is a constant determined in Cai (1999) which is set to 4.50524.

$$\hat{\theta}_{B_j}(L,\lambda) = (1 - \frac{(L\lambda\sigma^2)}{S^2})_+ * \theta(B_j) \tag{2}$$

Block thresholding is an improvement on term-by-term thresholding in some applications as it uses information from neighboring coefficients to improve the efficiency of the threshold value. Additionally the log penalty is removed and the operations

required to threshold are n rather than log n (McGinnity et al., 2017). The technique presented by Cai is well suited for denoising images since it preserves signal and removes slightly less noise based on the more accurate threshold as determined by James and Stein (1992). This preserves more of the original image while denoising effectively.

2.4.5 Notable Wavelet Thresholding Techniques

There has been significant research seeking a more robust thresholding estimate for wavelet techniques. Donoho and Johnstone (1995), Neumann and Von Sachs (1995), Nason (1996), and McGinnity McGinnity et al. (2017) discuss non-parametric methods to threshold wavelet coefficients that are effective for functions with noise and varying degrees of smoothness. These methods are also not reliant on a strong independence of error assumption that are normally associated with classic function approximation. Donoho and Johnstone (1995) present the SUREshrink method which is based on Stein's unbiased risk estimate (SURE). This method treats each resolution level as it's own multi-variate normal threshold estimation problem. Additionally, SUREshrink uses a fixed value for resolution levels that mostly consist of noise. This means the noise is removed from the more important coefficients and the noisy coefficients using separate functions. Neumann and Von Sachs (1995) explore the non-Gaussian situation for functional analysis and still derives asymptotic normality for functions with non-homogeneous smoothing and functions without strong independence. Traditional wavelet techniques from Donoho and Johnstone (1994) were acceptable and still achieved optimal mini-max risk rates for these spaces. McGinnity et al. (2017) uses Nason's even-odd cross validation technique, but also uses block thresholding and resolution dependent thresholding. The even-odd cross validation technique validates the thresholds and DWT created from the even data and odd

data and corrects based on the size of the samples. McGinnity's method establishes a wavelet transform technique that performs similarly to traditional techniques in the homogeneous case, achieving optimal mini-max rates. This method performs as well or better in the non-homogeneous case in terms of Mean Square Error (MSE).

2.4.6 Wavelet Error Distribution and Thresholding

A nice property of wavelets is that there are limited assumptions. There is not a specific error term distribution required for a function undergoing wavelet analysis. In fact the error term distribution could be completely unknown. Further, the error terms can be slightly dependent, noisy, and non homogeneous. However, the classical threshold techniques may not yield optimal results.

> "Surprisingly enough wavelet thresholding even works for situations where we have little knowledge on the actual underlying distribution and the correlation structure of the empirical coefficients." (Antoniadis and Oppenheim, 2012)

Neumann and Von Sachs (1995) considered the "Non-Gaussian Situation". Neuman presented a thresholding method where error distributions may not be known immediately, specifically non-parametric regression and multi-dimensional density estimation. Neuman determined that functions with non-Gaussian error terms could be proven to have asymptotically normal error terms. Given this, classical thresholding methods could be used. Donoho and Johnstone (1995) published a foundational wavelet shrinkage paper which presented SUREshrink based on Stein's unbiased risk estimator (SURE) (Donoho and Johnstone, 1995). Neumann and Von Sachs (1995) proved that theoretical threshold calculations could be used for functions with non-homogeneous smoothness since they maintained optimal minimax rates for Besov spaces, but Donoho and Jonstone improved MSE for the eight classic wavelet test

functions using SUREshrink. SUREshrink treats resolution levels as their own multivariate estimation problem. Resolution levels dominated by noisy coefficients are thresholded based on a set value rather than the SURE estimate. This method found similar optimal minimax rates for thresholding anisotropic functions such as the Sobolev class as smooth functions. Nason (1996) developed a non-parametric method for wavelet thresholding allowing some dependence and non-parametric error terms. Nason used even-odd cross validation to find a proper threshold value. This method takes the even and odd wavelet coefficients and derived separate threshold values for cross-validation. Gao (1997) extended the methods of Donoho and Johnstone (1995) for exceptionally unsmooth functions. Gao presents a resolution specific thresholding technique for wavelet estimation of a functions log-smoothed periodogram. The guiding principal behind this technique is smoothness. Gao states that functions estimations shouldn't have micro-spikes and estimating the periodogram smoothes these spikes. McGinnity et al. (2017) used the even-odd cross validation technique presented by Nason (1996), but also used block thresholding and resolution dependent thresholding. The even-odd cross validation technique validated the thresholds and DWT created from the even data and odd data and corrects based on the size of the samples. McGinnity's method establishes a wavelet transform thresholding technique that performs similarly to traditional techniques in the homogeneous case, achieving optimal mini-max rates. This method performs well or better in the non-homogeneous case in terms of Mean Square Error, MSE.

In addition to techniques for dealing with non-Gaussian error distributions, there is some research on functional data that is not equispaced. Cai (1999) presented an early technique for wavelet analysis of non-equispaced functions such as those built from uniform processes using term by term thresholding. Chicken (2003) furthered this work by using block thresholding directly on non-equispaced data from Poisson

and uniform processes. They found these techniques upheld optimal mini-max rates for Holder and some Besov classes of functions.

2.5 Functional domain

An advantage of modern data collection is that some collection can be controlled by autonomous sensors. The main benefit of autonomous data collection is that it is more precise and can sample the data faster than when humans do the tasks. A typical side effect of autonomous data collection is that data collected by sensors is raw performance data. Raw performance data is often measured by performance curves or a series of performance curves at the level the data is being recorded.

Functional data analysis (FDA) is a relatively new area of study. More recently, Ramsay and Silverman (2007) coined the term FDA which has led to the development and research of a unified set of methods and techniques. There are three key components of FDA: functional data are conceptually continuous, data are represented as functions or curves rather than individual data points, time is trivial, and there is a limited smoothness requirement to enable derivative methods (Ramsay and Silverman, 2007). Two representative examples of functional data from the Ramsay data are the non-durable product data set and a handwriting data set.

In the non-durable data set, each curve represents a year. The question at the functional level is how the years differ from one another. The functional data being compared are the years themselves rather than the samples that make up the curve for a year. Conceptually, the data is continuous as the non-durable product can be calculated at any point in time. However, the data are discretely sampled on a monthly basis. The non-durable product is a function of time, however time per se is not expected to effect the response. There is a distinction between time and factors associated with time such as quarters, or seasons. Figure 1 shows the non-

durable product index reported monthly from 1919-1998. Visual inspection indicates that the non-durable index is increasing, but more exogenous factors are difficult to pinpoint. In this case the authors were forced to transform the data to account for the exponential growth of the economy over the 100 years encompassed. Once, the data are pre-processed, nuanced FDA can identify years of interests where occurrences of factors known to impact the non-durable product such as war, illness famine, and economic downturns occur.

In the handwriting data set, each curve represents a handwriting sample, Figure 2. The question at the functional level is how each handwriting sample differs. Conceptually, the data are continuous since it represents the smooth continuous motion of pen on paper. The handwriting samples are not a function of time, but are X,Y coordinate system samples measured in meters. The set includes 20 registered samples of one individual writing "fda". Visually, the handwriting samples seem to be consistent.

There are many FDA methods, but most depend on preprocessing the functional data. The preprocessing techniques are similar to techniques used for most data. In (Ramsay and Silverman, 2007), the authors notice an exponential increase in the non-durable product from 1919 to 1998. To better compare the different curves the authors apply a logarithmic transform to the data to correct for the exponential growth in the economy. In the handwriting data set, the curves underwent a rigorous registration and transformation process so the handwriting samples were of equal position and magnitude for comparison (Ramsay and Silverman, 2007). The preprocessing methods for each data set are tailored to the context of the given data.

Analysts can use dynamic models to characterize some functional data based on differential equations to describe the behavior of the curves. Ramsay and Silverman break down a conceptual version of the dynamic models using phase plane plots.

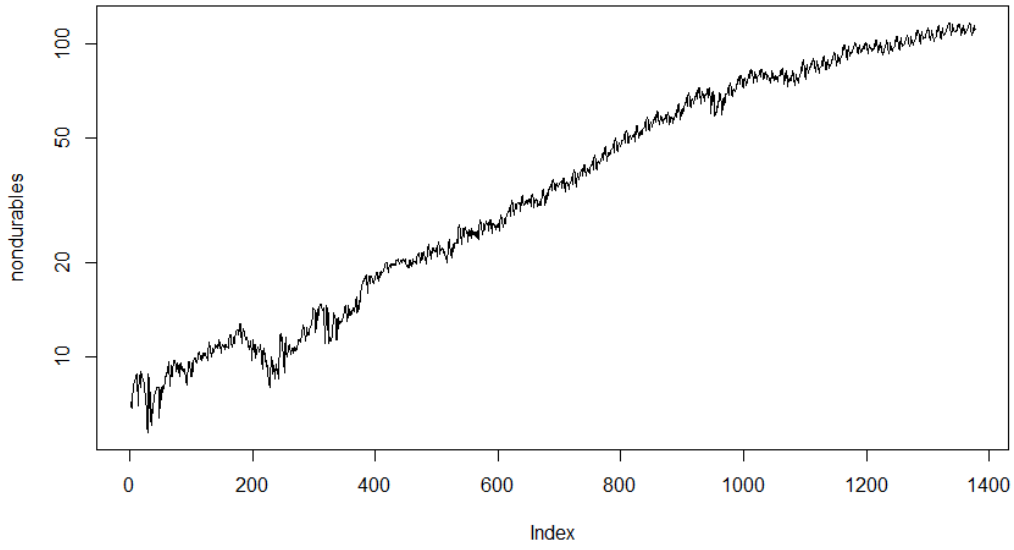

Figure 1. Illustration of Non Durable Products by Year (1919-1998)

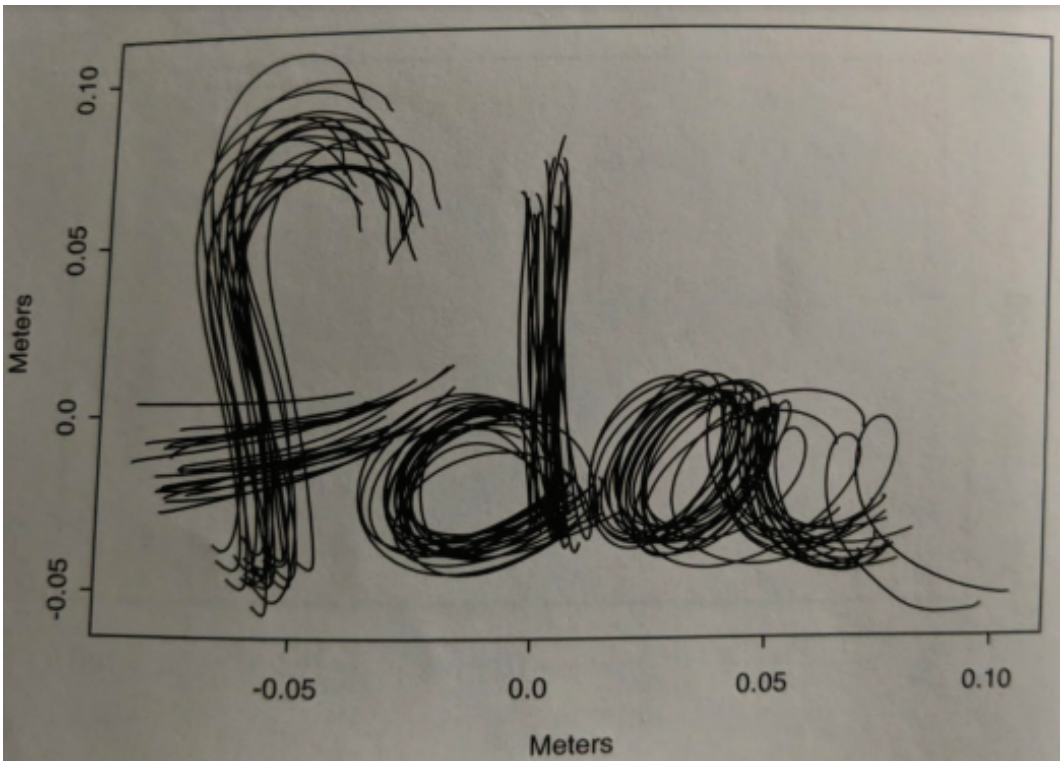

Figure 2. Illustration of Twenty Handwriting Curves as Functional Observations

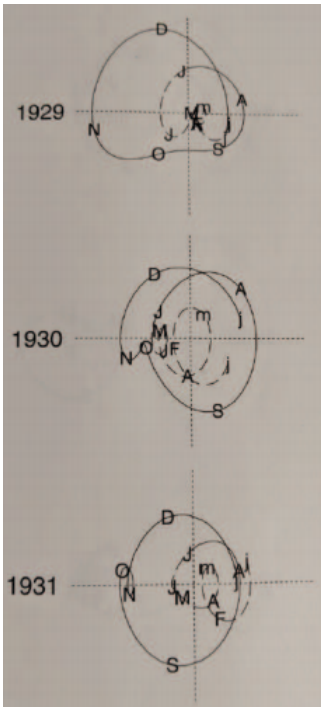

Figure 3. Illustration of phase plane plots for select years of non-durable product

Figure 3 shows the phase plane plots for a few years of the log smoothed, non-durable product data. The x-axis represents the first derivative (velocity) and the y-axis represents the second derivative (acceleration). The phase plane plot accurately characterizes the behavior of the non-durable product data. The plots show the magnitude of three seasonal oscillations which are apparent in the original yearly non-durable product plots. The phase planes can be described using a dynamic differential equation model. The smooth curve $h(t)$ is fit based on the criterion in Equation (3). This equation has two terms, one representing a traditional mean squared error (MSE) metric and the second measuring the smoothness of the curve. The second term, uses h^{iv} the fourth derivative of h, measures the curvature of the second derivative, acceleration. The second derivative represents curvature so the model penalizes the curvature, or low degrees of smoothness. A challenge with this technique is choosing a smoothing parameter λ. The smaller the value of λ the rougher $h(t)$ is and the larger value for λ the smoother $h(t)$ is. For this particular analysis the authors use $10^{-9.5}$.

The choice of λ has a significant effect of oversmoothing and undersmoothing in the model. Oversmoothing is typically indicated by a missing peaks, such as seasonal spikes or spikes caused by exogenous factors. Undersmoothing, includes noise that overshadows or swamps important spikes.

$$PENNSE_{\lambda}(h) = \sum_{i=1}^{973}([y_i - h(t_i)] + \lambda \int_{1999}^{2001} [h^{iv}(t)]^2 dt. \tag{3}$$

There are several techniques, most extended from multi-variate analysis, used in FDA. An example is the use of principle component analysis to characterize functional data sets. The basics of FDA require fitting a function that minimizes MSE and smoothness. Transformations using wavelets have proven exceptional at achieving near optimal mini-max rates in Besov and Holder spaces for signal approximation. Additionally they have natural smoothing properties which eliminate the need for selecting smoothness parameters.

2.6 Outlier Analysis

Cook references the Longley data set which is used to regress six economic factors on U.S. employment from 1947-1962 (Yamanishi and Tanaka, 2005). The economic data set appeared to be ordinary, but Longley struggled to estimate the six economic regression parameters, β_i, within a 10% confidence region. The results of the original regression indicated bias, but the explanation and location of the bias were not obvious. Longley suspected bias was occuring in some of the border years, 1951 and 1962, based on visual inspection of the data. Excluding the border years changed the confidence regions significantly for the regression parameter estimates, from 35% to 15%. Even though the presence of bias could be confirmed for the border years, the exact cause of the bias was not clear. Upon further research, the years 1951 and 1962 were influenced by exogenous factors, factors not considered in the regression.

The analysis in the above scenario was effected by biased data called outliers, and the study of the biased data or outliers is called outlier analysis. A popular definition of outliers are those data that seem to be generated by a separate or altered process. More formally, Hyndman and Shang (2010b) define outliers in terms of curves, similar to the curves for the comparative years of economic data that Longley researched. An important distinction in their definition is the classification of two types of functional outliers; shape and magnitude.

> "We consider that a curve is an outlier if it has been generated by a stochastic process with a different distribution from the large majority of the curves. For example, a functional outlier can have a different shape or magnitude from the rest of the data" (Hyndman and Shang, 2010b).

Once outliers are defined, the next logical path is to measure, or quantify their leverage. Clearly, some altered processes will have a more pronounced effect on data than unaltered processes. Just as it is important to detect an outlier, it is equally important to understand their effect, in magnitude and shape, on the analysis the data support. Leverage is defined as the product of outliers in the independent variable and influence in the dependent variable (Montgomery et al., 2012). In terms of a series of data occurring over time, data falling near the end or beginning of the time interval, in general, have more leverage on the regression. These data heavily influence the fit of data to a linear model since there is less certainty and data are more scarce further from the origin of the analysis. This is similar to physical leverage; the further force is applied from the center of mass, the greater the leverage will be. In the Longley study, two outlying years in this study, only slightly anomalous in the dependent variable, had significant leverage on the effectiveness of regression parameter estimation. The year with the greatest studentized residual, a measure of influence, was 1956. However, the leverage on the regression from this year is less

pronounced due to it's spatial orientation at the center of the data time interval. This occurs because there are many other observations near the center of the time interval which are also considered in regression fitting. In general, data points at the boundary of a data set have more leverage and make effect estimation more challenging. Still, influence in the dependent variable can substantially effect a regression, as would be expected for the effect of a major war, or other exogenous variable described in this analysis.

In practice, analysts are cautioned about the impact an outlier can have on an otherwise satisfactory analysis. Methods such as visual inspection, and various metrics are standardized techniques to indicate the presence of outliers as well as quantify their influence and leverage in regression. Montgomery discusses the potential impact outliers can have on a field of classical analysis, Design of Experiments (DOE).

> "Experimenters often worry about the impact of outliers in unreplicated designs, concerned that the outlier will invalidate the analysis or render the results of the experiment useless" (Montgomery, 2009)

Referring to classic categorical data, without a major time component, Montgomery mentions that outliers can often be visually detected in normal probability plots. Montgomery suggests replacing the outlier with an estimate or removing the data point altogether.

In addition to true outliers, erroneous data entries and data collection errors can present as outliers. The data collection in the famous Fisher Iris flower experiment was manually collected. The reason for the success of this experiment as a case study is due to its meticulous collection of data. The experimenter collected flowers "all from the same pasture, and picked on the same day and measured at the same time by the same person with the same apparatus" (Andrews and Herzberg, 2012). There are many statistical experiments that fail due to errors in data collection. Classical

methods exist that may mitigate the effect of some measurement error or erroneous data entries. Still, manual data collection is tedious causing data errors to persist. Even when data collection is by autonomous sensors, errors can occur.

2.7 Domain for Outlier Analysis

Unlike the longitudinal data in the Iris experiment, the data domains of interest in this research are time series and functional data. Time series data meets the definition for functional data, however, there are two approaches to time series analysis, a functional method and the Box Jenkins and econometric methods. Since there are two prominent methods for time series and functional data analysis they are presented separately in here. In addition, this research examines the simulation domain and the two-dimensional image domain.

2.8 Time Series Domain

The data described in the initial explanation of outliers and the measure of outliers has to this point been comparative of data over certain time intervals. The economy data are sampled monthly over the course of 99 years. Data collected over time, often equally spaced, intervals are considered a time series. The National Institute of Standards and Technology (NIST) defines time series analysis as the study of the underlying structure of the time series data.

> "Time series analysis accounts for the fact that data points taken over time may have an internal structure (such as autocorrelation, trend or seasonal variation) that should be accounted for."

A well known time series analysis tool, Auto-Regressive Integrated Moving Average (ARIMA), is a combination of previous time series approaches and the Box

Jenkins ARMA methodology. The origins of ARIMA time series analysis began in 1926 when Yule presented autoregressive models, or AR models (Makridakis and Hibon, 1997). These models exploit the structure of the time series data by analyzing the correlation between a data point with the previous point or points. AR models approximate time series data according to equation (4). The ϕ is the coefficient of autocorrelation value; the subscript represents the level of autocorrelation. AR2 models measure the autocorrelation between a data point and the two data points prior.

$$x_t = \phi_1 x_{t-1} - \phi_2 x_{t-2} - ...\phi_p x_{t-p} + e_t \tag{4}$$

In 1937, Slutsky introduced the moving average component models. These models exploited the structure of the time series data by analyzing the error between a single data point with the previous point or points (Makridakis and Hibon, 1997). The moving average models approximate time series data according to equation (5). The θ represents the coefficient of the moving average and the subscript represents the level of data points referenced. θ_2 represents the error between the current data point and two data points prior.

$$x_t = \theta_1 e_{t-1} - \theta_2 e_{t-2} - ...\theta_p e_{t-q} + e_t \tag{5}$$

The ARIMA method combines these two principles into a single model according to equation (6). δ is a constant and a_t is a white-noise process. The Integrated part of ARIMA reflects the differencing preprocessing step in order to make the time series data stationary.

$$x_t = \delta + \phi_1 x_{t-1} + \phi_2 x_{t-2} + ...\phi_p x_{t-p} + e_t + \theta_1 e_{t-1} - \theta_2 e_{t-2} - ...\theta_p e_{t-q} + a_t \tag{6}$$

The ARIMA model requires defining p and P, auto-regressive components at the non-seasonal and seasonal level, respectively, q and Q, the moving average components at the non-seasonal and seasonal level, respectively and d and D, the order of differencing at the non-seasonal and seasonal levels respectively, This is denoted ARIMA(p,d,q)(P,D,Q)[m] where m is the season length. The Box Jenkins process for ARIMA lays out the approach for differencing the time series data and selecting the appropriate levels in the model for p and q (Makridakis and Hibon, 1997).

> "Box and Jenkins (1976) provided both a theoretical framework and practical rules for determining appropriate values for p and q as well as their seasonal counterparts P and Q" (Makridakis and Hibon, 1997)

2.9 Functional Data Analysis of Time Series Data

Functional Time can be treated categorically as in the Longley data set, but as time becomes a larger and more trivial part of the data set, functional analysis becomes more appropriate. There are two types of functional analysis for time series data, time domain and frequency domain analysis (Gao, 1997). In the time domain, ARIMA models can be used to analyze patterns and relationships in the data. Instead of an effect-based model, lags of different magnitudes and trends in the response are used to approximate a time series function. Time series analysis in the frequency domain involves patterns in frequency and periodicity in frequencies.

Spectral analysis is especially well suited for time series data or functions with varying degrees of smoothness. The degree of specialization associated with estimating a certain function composed of functional data is typically determined by the smoothness of the function. A perfectly smooth function can be estimated using many tools, such as kernels, splines or orthogonal series estimates. However, functions that are not smooth, or have non-homogeneous smoothness, typically represented by

discontinuities in the function or non-linear functions such as those containing spikes or jumps, should be estimated with more specific methods.

> "Kernels, splines, and orthogonal series estimates even with optimal choices of the smoothing parameter, would be unable to perform in a near-minimax way over many spaces in the Besov scale" (Donoho and Johnstone, 1995).

In function estimation, it is important to specify the class of functions to which the function of interest belongs. In wavelet analysis many functions are classified into the Besov space. These are functions that have homogeneous smoothness almost everywhere. Functions with more irregular smoothness fall within other spaces such as Sobolev and Holder spaces (Donoho and Johnstone, 1995). Analogous to linear regression outliers, spectral analysis deals with the degree of smoothness at certain periods of time. The smoothest portions of the function are similar to outliers at the center of the regression and the non-smooth intervals are similar to the outliers further from the center of the regression.

The smooth portions of the function are estimated by the mother wavelet. The properties and smoothness of the underlying mother wavelet determine how well the smoothness of a function can be represented by the smooth wavelet coefficients. The coarse or less smooth portions of the function are estimated by the different resolution levels of the father wavelet.

In addition to smoothness, the noise inherent to time series and other functional data can be thought of as an outlier. The noise is not representative of the true system and can be unevenly dispersed, especially in the wavelet domain. Debauchies (1992) states that much of the wavelet domain is sparse and mostly noise, and most of the true signal is represented in few coefficients. These larger coefficients are more effected by the noise than the smaller coefficients that are mostly noise. Additionally, noise effects both the coarse and detail components of the wavelet coefficients differently

which makes it difficult to properly threshold the coefficients.

2.10 Simulation Domain

This research examines outlier analysis in the modeling and simulation (M & S) domain. Specifically, computer generated simulations digitally reproduce some process or processes. Simulations are often conducted in concert with the development of systems, in lieu of an overly costly, time consuming or impractical application or prototype. Simulations may also be used to measure the effect of a new policy on a given process. Often analysts are interested in performance statistics such as wait time, throughput or other queue statistics to inform system decisions. Simulations are a good tool to provide insight into a process, but they can not perfectly imitate the system of interest. Hill and Miller state that "A model is a simplified representation of the entity it imitates or simulates." (Hill and Miller, 2017). As such, there are over-simplifications in simulations that may bias subsequent simulation analysis. One such area is the warm-up period in a simulation prior to normal operations. This biased region of data is known as the transient. Transient data is not representative of normal system operations and including the data in analysis can bias the analysis. Transient data meets the traditional definition of outlying data as generated from a separate stochastic process. Instead of a single data point the entire curve representing the transient data are the outlying data, or the outlying curve, biasing the analysis. Therefore, since the transient meets the definition of a functional outlier it is a domain of interest in this research.

2.10.1 Functional Data

Since the inception of classical signal processing, signal processing has evolved into multiple domains. In this dissertation we examine the classical, functional, and

graph domain. Ramsay and Silverman (2007) discuss some functional data analysis applications for signal processing. Functional data differs from traditional signal processing data in that each observation is a curve rather than a single data point. Data such as handwriting, children's heights, bone density, lifetime crime data, sea surface temperature data and the non-durable goods index are considered functional data. Comparing criminal records for 413 men from ages 11-35 is a functional data set since we are interested in comparing the patterns in the curves such as shape and behavior. A figure of these 413 observations are show in figure 4.

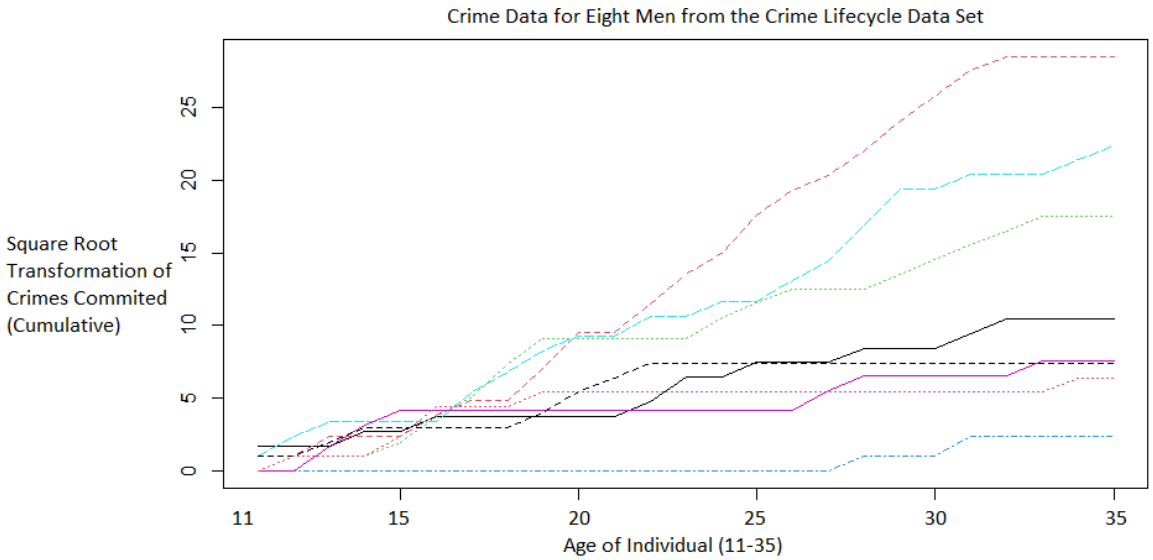

Figure 4. This graph depicts a subset of functional data from the study conducted by Ramsay and Silverman (2007) which shows the frequency of crimes committed by the 413 criminals in the data set. This figure shows the eight available functional observations for men between the ages of 11-35. As suggested by Ramsay and Silverman (2007) each curve has a square root transformation to correct for skewness. Each curve represents cumulative crimes committed for each of the eight individuals over their lives.

The pattern shown shows that criminal behavior increases in the early adolescent years, stabilizes in the 20s and decreases for men in their 30s. Some of the focus of our work is on functional outliers. Those are observations or curves that vary significantly in shape or magnitude from the other curves. In this data set, there are some potential outliers. In terms of shape, there is a subject that does not commit

crimes until later in their life. The curve is flat and spikes rapidly as the individual approaches 35. In terms of magnitude, one subject continuously commits crimes from age 11-35. This is interesting as most criminals are incarcerated or take a break from crime for a short period before relapsing. There are many applications of functional data as well as techniques such as functional outlier analysis. In this paper, we are interested in using wavelet analysis for functional outlier analysis.

2.11 Outlier Analysis for Functional Domains

Despite the wealth of research in functional analytics, there is not a well established method for determining the presence of outliers for the domains specified in this paper. Outlier detection methods and outlier mitigation methods used for classical regression are not effective for some functional data. Analysts have less standardized tools to determine the presence of outliers or measure their impact on the functional analysis. Yet, functional data are equally susceptible to anomalous data entries. There are a few tools that analysts have at their disposal for some of the domains listed in this paper. Tukey's depth method are modified by Fraiman and Muniz (2001) for functional data outlier detection. (Febrero et al., 2007) bootstraps the depth methods of (Fraiman and Muniz, 2001) to create outlier cutoff regions. Hyndman and Ullah (2007) and Sun and Genton (2011) use data visualization using box plots for functional data. The functional box plots are capable of classifying outliers.

2.11.1 Depths

A principal challenge of functional data is finding a measure of centrality for the data set. Taking the mean or median of the data composing the curves may achieve some centrality, but may not reflect the characteristics or behavior of the performance

curve. An obvious example of this is with the handwriting data set. Taking the mean at each point in the twenty samples of handwriting data may bias the letters "f", "d", "a" to the point they are not legible. The mean of the functional data is unlikely to resemble the natural smooth strokes of handwriting shown in the original samples. In 1974, Tukey presented a method known as depth which is calculated at the functional level. Depth for a single data point, x_i, in a univariate dataset, where $X_i = X_1, ..., X_n$, is written as $min(\#(X_i <= x), \#(X_i >= x)$ (Tukey, 1977). The depth of a data point measures the centrality of the data point compared to the rest of the data. This method, when applied at the functional level gives a distribution for the centrality of a curve. Fraiman and Muniz (2001) extend Tukey's depth method for functional data. The empirically derived cumulative distribution function is given by (7).

$$Fn(x_i(t)) = \frac{1}{n}\sum_{k=1}^{n} I[\# : x_k t <= x_i t] \tag{7}$$

This is simply an indicator function that ranks a curve from 1 to n at each point t based on magnitude. The curves are compared based on centrality as each curve has an associated depth for each point. There are several other functional depth methods which are surveyed in (Febrero et al., 2007).

2.11.2 Functional Box Plots using Band Depths

The Box Plot was first introduced by Tukey in 1977 for data visualization listing five major components; median, first and third quartiles, non-outlier minimum and maximum (Tukey, 1977). Box plots for longitudinal univariate data help visualize the data (Tukey, 1977). The first step in creating a box plot is ordering the plot based on magnitude of the individual data points. In the functional case or multivariate case ordering data is not straight forward. Researchers have used the concept of depth

to order functional data and more specifically López-Pintado and Romo (2009) use band depth for functional box plots.

Band Depths may use any particular depth method and order curves from the center out. The most central curve $x_c(t)$ is the median of the box plot. $x_i(t)$ are added to the left and right of the most central curve in order of decreasing depth (López-Pintado and Romo, 2009). The region of centrality, α, presented by Liu et al. (1999) is the depth percentile selected to represent the curves falling most centrally in a set of curves. α can be any value from 0 to 1, and is set to 0.50 as a default value to emulate the univariate box plot. Curves falling outside of this region are trimmed. The central region is presented as the box; this is the middle 50% of data for the uni-variate case. The whiskers in the uni-variate box plot represent the data falling within 1.5 times the inter-quartile range (IQR). The whisker section of the plot are curves that fall outside of the inter-quartile range (IQR) to include non outlying minimums and maximums. Outliers are any curves that fall outside of the whisker region in terms of band depth. The Functional Box Plot is modeled after the uni-variate case using the depths of the curves rather than the value of an individual data point. This technique clearly presents functional data and gives an indication of a curves outlier status. Figure 5 shows functional box plots for the heights of boys and girls at different years (Sun and Genton, 2011).

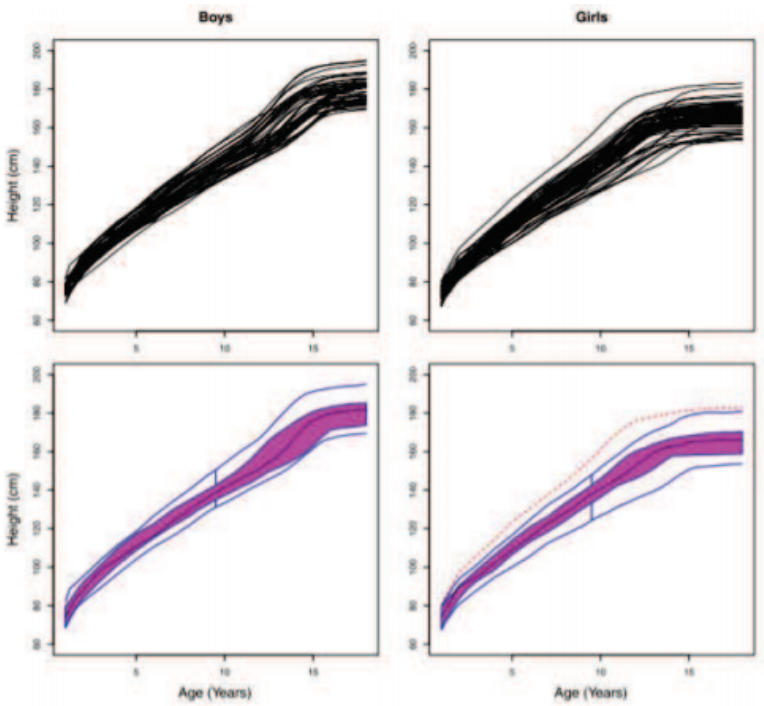

Figure 5. Illustration of Functional Box Plots for the Heights of School Age Children.

2.12 Functional Bag Plots

Hyndman and Ullah (2007) present the idea of functional bag plots, similar to the Functional Box Plots presented in Sun and Genton (2011). The researchers set $\alpha = 0.5$ for the functional bag plot and inflate this region of centrality by a factor of 2.58 to detect outliers. Instead of using depths the functional bag plot ranks curves based on the first two principle components of the curve set.

2.12.1 Bootstrap and Depths for outliers

Febrero uses the concept of depth and variance stabilized bootstrapping to create a statistical calculation for outlier detection in functional data (Febrero et al., 2008). The bootstrap method was developed in 1979 to estimate the standard error of some parameter $\hat{\theta}$ using computer processing (Efron and Tibshirani, 1994). In terms of single data observations $x = (x_1, x_2, \ldots, x_n)$ with an unknown distribution, F, a boot-

strap sample can be written as a random selection with replacement from x of length m such that $\hat{F} \rightarrow x_1^*, x_2^*, ..., x_m^*$ (Efron and Tibshirani, 1994). In terms of functional data with an unknown distribution f and curve observations $Y = (Y_1, Y_2, \ldots, Y_n)$, a bootstrap sample can be written as a random selection with replacement from Y with length m, $\hat{f} \rightarrow Y_1^*, Y_2^*, \ldots, Y_m^*$. In the functional case, entire curves are selected as the single observations in the method described in (Efron and Tibshirani, 1994). Statistics such as, $\hat{theta}$, and are calculated for the bootstrapped curves sample. Febrero et al. (2007) introduce a concept called smoothing in conjunction with the bootstrapping method. The distribution of the smoothed bootstrapped subset,$\hat{f}_\lambda$, includes a penalty for roughness λ. Efron produces curves from the bootstrapped data points using quadratic splines. These curves are based on the standard error, θ, of the bootstrapped data points, x, and are subject to a smoothness penalty parameter. Quadratic splines with multiple curves are less desirable than flatter curves. Febrero uses a similar technique, at the functional level, adding a penalty to the confidence region generated by the bootstrapped curves of, $\gamma = 0.05$ by the sum of the covariance of the curve. This penalty increases the standard error in the confidence region of the curves proportionally to the variance among the data points that build the curve. Febrero combines these methods to present a method for detection of functional outliers. The final step is to use depths to provide a measure of centrality of the curve set. Further, Febrero trims a percentage of the curves, α, to create a trimmed set of curves.

The trim set are then bootstrapped and smoothed according to the methods in (Efron and Tibshirani, 1994). The depths of each smoothed and bootstrapped curve are then calculated compared to the original data set. The median of the lowest one percentile of depths, for the smoothed bootstrapped curves, are then chosen as a cutoff value, C. Data falling below the cutoff value C are considered outliers (Febrero

et al., 2007).

2.12.2 Graph Data

Another extension beyond classical signal processing is signals on graphs. Ortega et al. (2018) presents a summary of graph signal processing (GSP) which is the processing of signals on graphs. GSP is useful when the signal of interest has an underlying structure that should be included in the analysis. Traffic and network data are among many of the data types where it benefits the analysis to include the structure of the data. Transportation systems across the world are becoming more technologically advanced, such as in Singapore where their intelligent transport systems (ITS) contain a network of sensor nodes which monitor traffic flow Mohan et al. (2014). Similarly to SP, the interest in this Singapore traffic study is the detection of traffic events such as congestion, and slow downs at different areas of the highway network. Mohan et al. (2014) find that traditional traffic analytical tools such as PCA, robust PCA, and classic wavelet analysis are less effective than graph wavelets.

Figure 6 shows the ability of graph wavelets to detect traffic events based on the shape and magnitude of graph signal disturbances and subsequent choke points in the Singapore traffic network. In this work we include additional wavelet analysis techniques, such as thresholding, from classical signal processing to improve graph wavelet analysis for problems such as the Singapore traffic network.

and designs spectral graph wavelets, one such spectral graph wavelet is the Meyer graph wavelet. These methods are developed based on the EVD of the graph Laplacian, graph frequencies. The spectral graph wavelet transform using the Meyer wavelet, for example, are a family of filters fitted to the graph frequencies

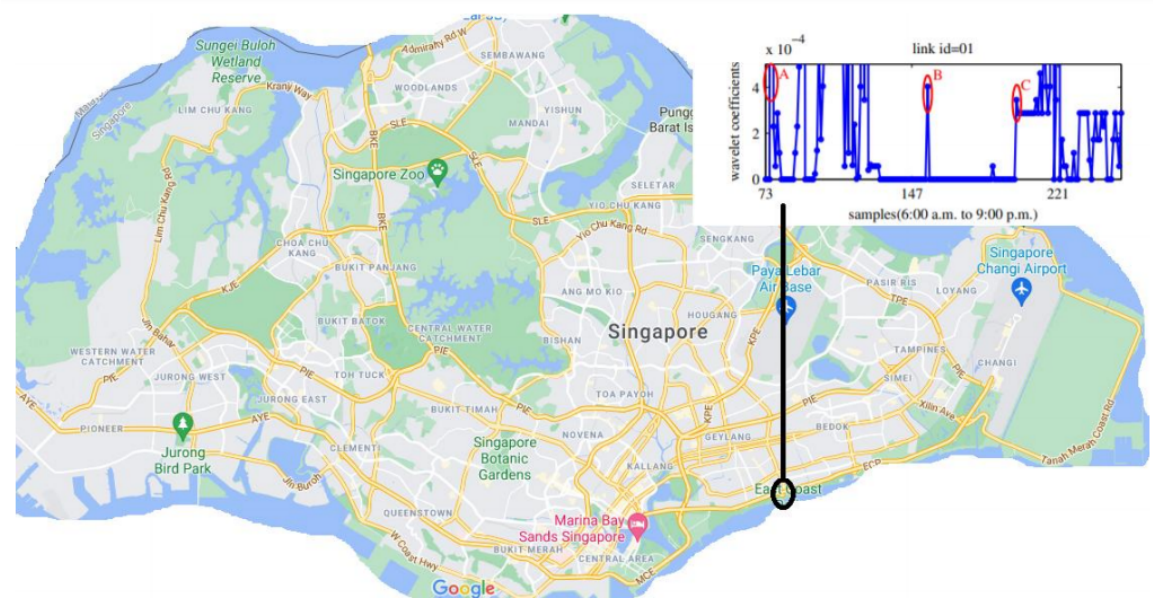

Figure 6. The goal of the study by Mohan et al. (2014) is to detect traffic flow events in Singapore. This figure shows the Singapore road network with a graph wavelet coefficient overlay. The disruptions in the graph wavelet coefficient represent changes in the speed of traffic flow for a particular area of the road network. Decreasing traffic flow is detected in the morning, afternoon, and evening for a particular choke point, reflecting rush hour in Singapore. The wavelet coefficient graph is from (Mohan et al., 2014)

2.12.3 Summary

Wavelet analysis is applied successfully in many new areas of signal processing, such as in functional analysis and graph theory. However, many of the techniques that leverage the beneficial properties of wavelet analysis have been left behind, especially thresholding. The purpose of this work is to include and improve state of the art wavelet thresholding in different areas of statistics.

III. Wavelet Analysis of Variance box plot

The functional box plot is a non parametric method which uses a depth method to visualize and rank smooth functional curves in terms of a mean, box, whiskers, and outliers. The functional box plot improves upon other classic functional data analysis tools such as functional principal components and discriminant analysis for outlier detection.

This research adds wavelet analysis as a generating mechanism along with depth for functional box plots to visualize functional data and calculate relevant statistics. The decomposition properties of the wavelet analysis of variance box plot tool gives competitive error rates in Gaussian test cases with magnitude outliers, and outperforms the functional box plot, in terms of error rates, for Gaussian test cases with shape outliers. Further, we show wavelet analysis is well suited at approximating irregular and noisy functional data and show the enhanced capability of WANOVA box plots to classify shape outliers which follow a different pattern than other functional data for both simulated and real data instances.

3.1 Introduction

The functional box plot provides statistical and visual context for rich data processes that traditional techniques are unable to provide. This is needed due to the capacity of many industrial processes to produce large quantities of functional data. These data can be challenging conceptually and are not always suited for traditional analysis techniques. Subtle differences in behavior, shape or magnitude have been shown to have substantial process implications.

The functional bag plot method of Hyndman and Shang (2010a) uses a dimension reduction technique, known as robust principal component analysis, to classify the

curves in a given set. Curves are then ranked based on the bivariate plot of the first two robust principal components using half space location depth and then determined to be a part of the central bag region, the outer fence region, or the outlying region. The visualization of the bag plot provides analysts with a baseline perspective of a given function using this classification system. In this current work we are interested in the outlier classification functionality of the bag plot. Sun and Genton (2011) compare the effectiveness of the functional bag plot with their functional box plot method in terms of error rates for functional outlier classification. The functional box plot is similar to the functional bag plot in that it uses an underlying functional statistic to categorize and visualize curves. The functional box plot categorizes curves into a box region, a whisker region, and an outlying region. The functional box plot differs from previous methods as the underlying statistical method is a functional depth method. Febrero et al. (2007) survey several depth methods which extend Tukey (1977) half space location depth to the functional domain. These include median band depth (MBD) Fraiman and Muniz (2001) and random projectile depth (RPD) Cuevas et al. (2007). These methods are proven to be robust nonparametric estimators for curves in the context of the Hilbert space.

The WANOVA box plot for functional data in this research is similar to the bag plot and functional box plot in terms of visualizations; it contains a central region, an outer region, and an outlying region. However, the WANOVA box plot uses a wavelet based analysis of variance (WANOVA) technique as an additional mechanism for determining statistical differences in functional data. Girimurugan et al. (2013) have shown the WANOVA method can detect small differences between curves, and Atkinson et al. (2017, 2018a) adapt this method to detect local and global anomalies such as change points in noisy automotive manufacturing data. In a Gaussian comparison between functional box plots, the WANOVA box plot and

traditional functional data analysis methods, the WANOVA box plot significantly outperforms the other functional methods for correct detection of outliers.

This work presents the WANOVA box plot method for curve categorization, and visualization. Section two describes the background of wavelet analysis, thresholding and the WANOVA. Section three presents the methodology and tests the sensitivity of the WANOVA box plot for outlier detection. Section five applies the WANOVA box plot method to Gaussian functions and compares the results to other functional box plot techniques. Section six applies the proposed methodology to real world urban Nitrogen Oxide (NOx) contamination functional data. Section 7 applies the proposed methodology to real world COVID-19 data Section six applies the proposed methodology to real world COVID-19 data.

3.2 Background

3.2.1 Wavelets

Wavelet transforms are adept for functional analysis and removing noise from signals.

> "The wavelet transform is a tool that cuts up data or functions or operators into different frequency components, and then studies each component with a resolution matched to its scale" Debauchies (1992)

Wavelet coefficients are generated through a series of translations and dilations using a family of orthonormal functions consisting of a father wavelet, ϕ, and a mother wavelet, ψ. There are many wavelet families which are designed for different intents. In this research, we use the Debauchies family of wavelets which are often used for change-point detection and statistical process control (SPC). Equations (8) and (9) show the translation and dilation of the father, ϕ, and mother, ψ, wavelet used to

represent a function

$$\phi_{jk}(x) = 2^{j/2}\phi(2^j x - k), \tag{8}$$

$$\psi_{jk}(x) = 2^{j/2}\psi(2^j x - k). \tag{9}$$

The subscript j represents, the j^{th} wavelet resolution level and k represents the k^{th} wavelet coefficient within a given level j. Typically, the inner product of the original function $f(t)$ and the wavelet translation and dilation functions generate the coarse and detailed wavelet coefficient matrices,

$$\hat{\theta}_{jk}(x) =< f(x), \psi_{jk}(x) > . \tag{10}$$

This is known as the discrete wavelet transform (DWT). Conceptually, the DWT decomposes the signal $f(t)$ into a coarse approximation of $f(t)$ and a more detailed approximation of $f(t)$. The detailed approximation contains coefficients at multiple resolution levels through multi-resolution analysis (MRA). The DWT "[decomposes] a time series into time frequency space one is able to determine both the dominant modes of variability and how those modes vary in time" (Torrence and Compo, 1979). Coarse coefficients represent the smoothest aspects of the signal of interest and the detailed coefficients represent the detail in the signal such as spikes, peaks, or edges.

Equation (11) is the representation of the reconstruction of the the original function, known as the Inverse DWT (IDWT)

$$f(t) = \sum_k c_{j_0,k}\phi_{j_0,k} + \sum_{j>j_0}\sum_k d_{j,k}\psi_{j,k}. \tag{11}$$

In equation (11), $\phi_{j_0,k}$ is the father wavelet translation function and $c_{j_o,k}$ represents the coarse wavelet coefficients. The term $\psi_{j_0,k}$ represents the mother wavelet

translation function and d_{jk} represents the k^{th} detailed wavelet coefficients at the different detailed wavelet levels, denoted by the subscript j. Applying the IDWT to the wavelet coefficients produced by the DWT will result in a reconstruction of $f(t)$.

3.2.2 Wavelet Thresholding

Wavelet coefficient matrices are naturally sparse as most of the signal is represented by a few large coefficients so that many of the coefficients represent noise (Debauchies, 1992). Wavelet thresholding is the removal of the noisy wavelet coefficients, which fall below a certain noise threshold. The benefit of removing noisy coefficients extend to many applications such as "lossless" compression, smoothing, and signal and image processing (Debauchies, 1992). In the case of the DWT as used in the WANOVA box plot for determining the difference among signals, wavelet thresholding removes coefficients representing negligible differences among signals. This highlights larger coefficients which represent those differences which are large in magnitude or caused by abrupt differences in the signals such as bumps, spikes and jumps.

There are two main techniques for developing thresholding criteria and two main rules for thresholding. The two main techniques are term-by-term thresholding and block thresholding. Term-by-term thresholding keeps, shrinks, or removes each coefficient term one-by-one based on a global threshold value. Block thresholding groups coefficients together and keeps, shrinks, or removes the group of coefficients together. The two main thresholding rules are soft and hard thresholding. Soft thresholding shrinks coefficients towards zero based on a generated threshold value whereas hard thresholding keeps or removes coefficients based on the generated threshold value. The universal threshold,

$$\lambda = \sigma\sqrt{2 \cdot log(n)}, \tag{12}$$

$$hard: \quad \nu_{\tau}^{h}(y) = y \cdot I(|y| > \tau) \tag{13}$$

$$soft: \quad \nu_{\tau}^{s}(y) = sgn(y) \cdot (|y| > \tau)_{+} \tag{14}$$

is used as the threshold technique in VisuShrink. VisuShrink is a well known term-by-term soft threshold technique which estimates the true coefficients by shrinking the wavelet coefficients which fall below the universal threshold Donoho and Johnstone (1995). Equation (13) shows the mechanism for shrinking wavelet coefficients based on the global universal threshold provided in equation (12) where n represents the number of observations in a signal $f(t)$, and σ represents the standard deviation of the highest level of wavelet coefficients. In equation (13), τ represents the threshold value, y represents the wavelet coefficients, and ν represents the thresholded wavelet.

VisuShrink removes more noise from a signal than any other technique; however, it also has the potential to remove more signal than other techniques. Visushrink is well suited for cases where the removal of noise is more important than the preservation of the signal.

Sureshrink determines a value for each resolution level, j, of the detailed wavelet coefficients based on the minimization of Stein's unbiased risk estimate (SURE) which is explained in further detail in (Donoho and Johnstone, 1994).

Cai (1999) presents a block thresholding technique which groups wavelet coefficients to be thresholded as a group rather than individually. Block thresholding uses a soft thresholding rule, equation 15. This thresholding value is less than that of the VisuShrink method, therefore, less wavelet coefficients are thresholded. Block thresholding groups wavelet coefficients into blocks of size $w = log(n)$ where n is the length of the signal of interest. Cai (1999) proves, based on a maximum likelihood

estimator, that block thresholding is effective when the block size $w \geq 3$.

The threshold value generated for each group of coefficients is calculated such that,

$$\hat{\theta}_{B_j}(L, \lambda) = (1 - \frac{(L\lambda\sigma^2)}{S^2})_+ \theta(B_j). \tag{15}$$

The coefficient λ is set constant to 4.50524. The σ^2 term represents the within block wavelet coefficient variance. The S^2 term represents the wavelet coefficient variance for the given signal. The term θ represent the wavelet coefficients. The term B_j represents the block of wavelet coefficients, j, of length L.

In equation (15), each block of coefficients are scaled by Cai's block specific threshold value in the interval [0,1]. This shrinks each block of wavelet coefficients toward zero to remove noise from the observed function. There are additional thresholding methods based on specific sets of assumptions and application which improve function estimation. A detailed list of these methods are listed in McGinnity et al. (2017).

3.2.3 Cross-validation Thresholding

Methods proposed by Donoho and Johnstone (1994, 1995) such as VisuShrink, the global term by term soft thresholding technique which removes nearly all noise in a signal, and SureShrink, which sets resolution dependent threshold values based on optimizing SURE, rely on the assumption that wavelet coefficient error terms are normally distributed. Recent research has considered the non-Gaussian case in regards to wavelet coefficient error distributions. Nason (1996) presents a method which uses even-odd cross validation, a standard method for choosing smoothing parameters for other smoothing functions. The function is split into two sets containing the even and odd data, respectively. The division of the original data set into the two even-odd

sets occurs according to equation (16) where

$$y^e = y_0, y_2, ...y_n{}', and \quad y^o = y_1, y_3, ...y_{n-1}{}' \tag{16}$$

where $y(t)$ represents the original function where e, and o represent the even and odd time increments, t, respectively. In even-odd cross validation a threshold parameter is selected such that the mean integrated square error (MISE) between the reconstruction of the two data sets is minimized. Cross validation ensures that models are not over-or under-fit based on threshold selection. McGinnity extended Nason's method by including block thresholding (Cai, 1999) and level dependent thresholding (Johnstone and Silverman, 2005b).

Block thresholding uses the information from neighboring coefficients to evaluate thresholding criteria. Nason noted level dependent thresholding would improve threshold estimation efforts for functions that are correlated and have non-Gaussian error covariance structures (Nason, 1996). The different levels of wavelet decomposition have varying levels of magnitude and sparsity which may require a different threshold value.

McGinnity et al. (2017) include these improvements and an additional threshold optimization term, the even and odd reconstruction. Equation (17) shows the optimization equation selecting the cross validated threshold,

$$\hat{\lambda} = argmin_\lambda \left\{ \frac{1}{2}\{||\hat{f}^o_\lambda - y^e|| + ||\hat{f}^o_\lambda - y^o||\} + \frac{1}{2}\{||\hat{f}^o_\lambda - \hat{f}^e_\lambda||\} \right\} \tag{17}$$

where f^e_λ and f^o_λ represent the reconstructed even and odd function for a given λ. y^e and y^o represent the even and odd data sets. This method does not assume an error distribution for the wavelet coefficients and improves MISE for noisy and irregular non-parametric function approximation. We choose this method to threshold wavelet

coefficients representing the difference in functional data sets. This permits non-Gaussian wavelet approximation of functional data and wavelet statistical testing which we prove through this paper is a useful addition to non-parametric functional depth for outlier classification

3.2.4 WANOVA

The WANOVA statistic presented by Girimurugan et al. (2013) is the functional version of the difference of means hypothesis test, known as the analysis of variance (ANOVA). The WANOVA is used to determine whether functions or groups of functions are statistically different. The functions are differenced and transformed using a DWT. The wavelet statistic for the difference or pairwise differences in groups of functions is calculated by taking the sum of squared wavelet coefficients for each pair of curves and treatment level, if applicable, according to

$$\kappa_\nu = \sum_{i=1}^{t} \sum_{k=1}^{T_i} \tilde{\theta}_{i,k}^2. \tag{18}$$

The $\tilde{\theta}$ are the thresholded wavelet coefficients for each coefficient representing a signal $f(t)$ at each resolution level k used to calculate the WANOVA statistic. Theshold-ing is not required for the WANOVA, but relevant thresholding will increase the effectiveness of the test. Similar to the ANOVA the resultant WANOVA values are compared to a critical value of an appropriate distribution for significance; instead of the F-distribution the WANOVA critical statistic is compared to the χ distribution (Girimurugan et al., 2013). Girimurugan et al. (2013) find that this test, when applied to test data and real world application, has competitive control of Type I error and strong power.

An example of the WANOVA hypothesis test used by Atkinson et al. (2017) to

determine whether model and system functions were statistically different is,

$$H_0 \quad : \quad S = M \tag{19}$$

$$H_A \quad : \quad S \neq M \tag{20}$$

where S represents a specific system and M represents a specific model. If the difference between a system and a model, WANOVA statistic, is not more extreme than the WANOVA critical value in magnitude, we fail to reject that S and M are statistically different. If the difference is more extreme than we conclude that there is a statistically significant difference between S and M.

Atkinson et al. (2017, 2018a) used the hypothesis test from (19) to test whether a noisy function was in-control (IC) or out of control (OOC) based on the WANOVA test statistic. Further, they were able to locate and characterize the magnitude of these OOC differences. Atkinson et al. (2017, 2018a) extend the WANOVA method to measure agreement or disagreement between a system and a model. In addition, to finding an OOC difference between a model and system, they also determined the location and magnitude of the difference. This idea of local analysis of signals is extended in this paper to compare the magnitude of similarity or difference among multiple signals through the development of the WANOVA box plot.

3.2.5 WANOVA Mechanisms

The two mechanisms selected for the WANOVA box plot are wavelet analysis and functional depth. Wavelet analysis is conducted using the WANOVA test which is based on the discrete wavelet transform (DWT). Functional depth is used as a classification method. In functional data analysis, there are many classification methods, some of which can be found in (Ramsay and Silverman, 2007) and (Ferraty and Vieu, 2006). Like many functional data techniques some of the classification methods are

elevated from the finite dimension to infinite dimensional functional data. (James and Hastie, 2001) presents functional linear discriminant analysis (FLDA) an extension of Fisher's discriminant analysis, the staple supervised classification method in the finite dimension. FLDA paramaterizes splines based on a selected basis to model and classify individual observations. Spline models, though, are designed for regular functions yet functional data have no strict regularity requirement and in practice are often irregular (Antoniadis and Fryzlewicz, 2006). Cuevas et al. (2007) consider FLDA, but use an adaption of k-nearest neighbor (KNN) classification as a non-parametric benchmark to their non-parametric depth methods. Cérou and Guyader (2006) present a functional adaption of KNNs which classify curves based on a majority vote among the k-nearest curves, where k is a tuning parameter determined in advance. More information about functional KNNs can be found in (Ferraty and Vieu, 2006). Cuevas et al. (2007) conduct a study which concludes that functional depth methods are similarly capable for regular curves and more capable for irregular curves than previous methods, such as functional KNNs and FLDA, since they are non-parametric and easily extend to functional data. This is a theme we highlight throughout this work; functional data have no strict regularity requirement and are often irregular. The functional depth methods in our research meet prior capabilities for regular functions and exceeds prior capabilities for irregular functions.

In addition to functional depth methods outperforming other functional classification techniques for irregular data, the wavelet basis is well suited for irregular data because of spatial adaptivity and time localization properties. Spatial adaptivity is the clustering of important wavelet coefficients in only a few areas, whereas the other areas are barren. Time and frequency localization means the wavelet coefficients can be mapped back to specific segments of the original signal. In figure 16, we show the benefit of these properties for peak contaminate classification.

Many FDA signal approximation methods such as functional principal components analysis (FPCA) and Fourier analysis are reliant on the eigenbasis which is not spatially adaptive, information is evenly distributed among the basis. Further, the eigenbasis is not time localized, certain segments of the basis do not map directly to segments of the signal Debauchies (1992).

Hyndman and Ullah (2007) present functional bag plots as a functional extension of the Tukey (1977) finite dimensional box plots. The functional bag plot is based on robust principle component analysis (RPCA) using the first two robust principle components. Outliers and other box plot components are classified by ranking the half space location depth of the bivariate plot of the first two robust principle components.

The functional box plot by Sun and Genton (2011) uses functional depth methods of López-Pintado and Romo (2009) which outperform the functional bag plot for a variety of different tests in outlier classification. These results are available in this section 3.5.

Extending the ideas of Hyndman and Ullah (2007) we test whether functional principal components improve upon robust principal component based functional bag plots. Similarly to the functional bag plot, we classify outliers based on the bivariate plot of the first two functional principal components. These results, are included in section 3.5. FPCA is a good classifier for outliers for simple test cases, but falters for bumps and spikes in irregular test cases presented in Sun and Genton (2011).

In this work, we use the wavelet basis to represent the difference among signals since wavelet estimators are near mini-max optimal for linear and non-linear functional estimation, and spatially adaptive even for irregularities (Donoho and Johnstone, 1995). Donoho and Johnstone (1995) show several instances of irregular and regular signals where the wavelet basis outperforms, spline models and the Fourier basis in both cases.

3.3 Methodology

3.3.1 Detailed Methodology

This section describes the novel WANOVA box plot method. The WANOVA box plot is a wavelet based functional analysis tool which can be used to classify functional data sets based on functional agreement and disagreement relative to the functional set. Similarly to the other functional box plot methods, the WANOVA box plot method adopts the longitudinal box plot method, to describe functional data in terms of centrality, quarterlies, and outliers. Centrality is measured in terms of agreement, using WANOVA statistics and depth methods, with the functional set in terms of smoothness and magnitude.

There are two main steps which result in the WANOVA box plot statistics. These statistics are the calculation of the central curve region, the WANOVA centrality statistics, and a functional box plot. The first step determines the central curve and central region. The second step conducts recursive WANOVA tests to determine whether any non-central curves are outliers. Non-central curves are those curves not determined to be in the central region, calculated in the first step. The key statistics are displayed visually in a WANOVA box plot.

In the first step, the central region is determined according to a pairwise WANOVA statistic comparison between curves and a functional depth method to rank the curves. Atkinson et al. (2018a) reveal the capability of the WANOVA statistic to determine agreement within a system in addition to regions of disagreement. To this end, the discrete wavelet transform generates wavelet coefficients, according to equation (10), for all of the curves.Wavelet coefficients are generated from a family of wavelet functions generated from a discrete wavelet transform (DWT). The wavelet functions used in the DWT are the Daubechies family of wavelets which are selected independently of the data. The resultant wavelet coefficients for each curve are differenced in a

pairwise manner with the wavelet coefficients from the other curves such that,

$$\kappa_\nu = \sum_{i=1}^{k} \sum_{k=1}^{T} \tilde{\theta}_{i,k}^2, \quad \forall i = 1, \ldots, k \quad \forall k = 1, \ldots, T. \tag{21}$$

These differenced wavelet coefficients are then thresholded according to equation (17) and used to calculate the WANOVA statistic, κ_ν. In equation (21), $\tilde{\theta}^2$ represents the thresholded and differenced WANOVA coefficients for a given resolution level , k. T is the number of curves in the set, and i is the wavelet coefficients or wavelet coefficients for multiple resolutions which represents location i for a given signal. Each curve, y_i, will have n treatments or pairwise comparisons for κ_ν. The result of this procedure is $N-1$ WANOVA statistics for each curve, where N is the number of curves in the set. The $N-1$ statistics, κ_ν, describe the difference, in terms of wavelet coefficients, between a given curve and every other curve in the set. These N WANOVA statistic curves are ranked using functional depth to determine the central box region.

The WANOVA method is used to determine statistical differences between two or more curves, yet it is not typically used to classify which curves belong to a particular functional distributions. In this work we use the (Fraiman and Muniz, 2001) functional depth method,it was proven to sufficiently order functional data in terms of centrality, such as the functional data developed using the pairwise WANOVA statistic curves. A depth method proposed by (Fraiman and Muniz, 2001) is then applied to the n curves, comprised of n pairwise WANOVA statistics from equation (21) such that,

$$Fn(x_i(t)) = \frac{1}{n} \sum_{k=1}^{n} I[\# : x_k t \leq x_i t]. \tag{22}$$

Equation (22) contains an indicator operator that ranks a curve from 1 to n at each point t based on magnitude relative to the other curves in the set. $I[\# : x_k(t) \leq x_i(t)]$

represents the number of data points, x_k at time t that are less than or equal to x_i at time t for a given curve. This value is summed over t and divided by the number of curves n to give the functional depth F_n for a curve $x_i(t)$. The lower 50^{th} percentile of curves are selected as the central region according to functional depth. This generalization is confirmed by Liu et al. (1999) who proved that the inner quartile range (IQR) can be set to the 50^{th} percentile for functional data in the same manner as a classical box plot for ordinal data. The wavelet used is the Debauchies wavelet with two vanishing moments.

The curve y_i associated with the smallest pairwise WANOVA depth curve is considered the central curve y_c. The smallest pairwise WANOVA depth curve is the curve with the smallest magnitude for WANOVA pairwise differencing and is selected as the central curve.

Once the central curve and the depth region are determined, the whisker region, non-outlying curves y_i which fall outside of the central box region, is determined.

The criteria to add a curve to the whisker region is based on the traditional WANOVA statistic, equation (18) between an outlying curve y_i and curves y_j falling within the central box region and the whisker region. When we say whisker region, we mean those curves that are not in the 50^{th} percentile of depth, but are determined to not be outliers. Each remaining curve, outside of the box and whisker region, y_i are considered for addition to the whisker region if they do not exceed the WANOVA critical value threshold with any single curve, y_j, falling within the box and whisker region. The WANOVA critical value is calculated based on a convolved χ^2 distribution. Girimurugan et al. (2013) develop this critical value and present helpful WANOVA critical value charts. Outliers curves remain outside of the box and whisker regions as they exceed the critical value with all resident curves in the box and whisker region. Further information on this process is available in section 3.9.

3.3.2 Creating the WANOVA box plot

We present the wavelet analysis of variance (WANOVA) box plot through use of a real data example. Meteorological and climate data are often the subject of functional data analysis. Scientists are interested in the change in environmental metrics across years to monitor the effect of different environmental factors. We use regional ocean temperatures to demonstrate the applicability of the WANOVA box plot to characterize several decades of functional data. Each year of data is a functional observation with 12 data points, sampled monthly. Hyndman and Ullah (2007) and Sun and Genton (2011) used functional methods to analyze this National Oceanic and Atmospheric Administration (NOAA) sea surface temperature data set covering January 1951 to December 2007 for a selected el Níno region. "These temperatures are measured by moored buoys in the 'Níno region', defined by the coordinates $0-10$ degrees South and $90-80$ degrees West" (Sun and Genton, 2011). They use this data to demonstrate the functional bag plot and the functional box plot. The data used to demonstrate the WANOVA box plot is the data for the same el Níno region, but for the years 1982-2020, according to NOAA.

The central region or central years are calculated from the different yearly sea surface temperature curves using the WANOVA test statistic in conjunction with a depth method as described in section 3.3.1.

For the sea surface temperature data, each curve, y_i, is compared to all other years based on the stated pairwise WANOVA process. Figure 7 shows that the year, 2019, is selected as the most central year according to the WANOVA depth process from 3.3.1. The central year y_c is indicated by the thick black line. The central or box region is shown in dark gray as selected by the stated depth method.

The second step, after determining y_c and the curves y_i which fall in the central region, is to conduct a traditional WANOVA test between each curve y_i outside of the

central region with the curves, y_i, in the central region. This is conducted recursively, such that the curves with WANOVA test statistics less than the critical WANOVA statistics are introduced into the whisker section of the WANOVA box plot. Those curves that do not pass the WANOVA test with a member of the central box or whisker region are considered outliers.

In Figure 7 the dark gray dashed region represents the whiskers. The outliers are represented in light gray dashed lines. The outliers in this data set are 1983 and 1997 which occur on or near el Níno years, known to cause deviations in sea surface temperature, for the specific region.

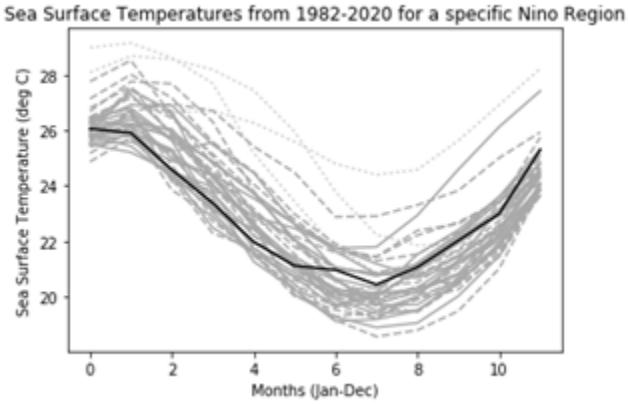

Figure 7. WANOVA box plot for sea surface temperatures in an El Níno region from 1982-2020. The black curve is the central curve. The solid gray lines are the central region (box). The dashed gray lines are the outer region (whiskers). The light gray dashed lines are the outliers which occur in the years 1983 and 1997.

3.4 SNR and Sensitivity of the WANOVA box plot

A simulation experiment tests the sensitivity of the WANOVA box plot to two types of differences found in profile monitoring, local and global. Global differences persist throughout a signal and local differences occur over some interval. The simulation study of Girimurugan et al. (2013) is replicated to determine the magnitude of difference that can be detected globally and locally between profiles using the

WANOVA box plot. A difference is detected if the WANOVA box plot determines a specific curve is an outlier in comparison to the other curves. The sensitivity of the WANOVA box plot is measured in terms of the signal to noise ratio (SNR), δ. The signal is the difference to detect and the noise is the added Gaussian error. In this experiment, differences are tested in the domain $x = [0.5, 0.6]$ for the function $f(x) = 7x^2 + sin(x) + 6cos(2x) + 3x^2 + \epsilon$. The error kernel is the identity function where $\sigma = 0.10$ such that $\epsilon = \sigma I$.

The global difference case includes two treatment groups. The first group of signals is the control group with equation $f_0(x) = 7x^2 + sin(x) + 6cos(2x) + 3x^2 + \epsilon$. The second group of signals has a global difference as a constant c such that $f_1(x) = 7x^2 + sin(x) + 6cos(2x) + 3x^2 + \epsilon + c$. There are ten functions of length $n = 128$ with nine functions drawn from the control group f_0 and one function drawn from the biased group f_1. At a power of 0.80, the WANOVA box plot can detect a global difference of $\delta = 0.40$. These results are shown in the top section of Figure 8. The graph in the top left is the power curve where δ is the SNR. The graph in the upper right of Figure 8 shows the minimum detectable global difference, at power = 0.80, between two curves drawn from f_0 and f_1.

The local difference case includes two treatment groups. The first group is the control group where $f_0(x) = 7x^2 + sin(x) + 6cos(2x) + 3x^2 + \epsilon$. The second group has a local difference as a constant g with duration c such that $f_1(x) = 7x^2 + sin(x) + 6cos(2x) + 3x^2 + \epsilon + g(c)$ where $c \in \mathbb{Z}[0, 128]$. The function $g(c)$ is the product of the magnitude of the error and the length of the error, n_c, of c. The additive noise used for calculating SNR is $\epsilon * n$. There are ten functions of length $n = 128$ with nine functions from f_0 and one function from f_1. At a power of 0.80, the WANOVA box plot can detect a local difference of $\delta = 1$. Figure 8, lower left, shows the minimum detectable difference in terms of SNR and associated power. Figure 8 lower right

shows the minimum detectable local difference at a power of 0.8 for the local error g(c).

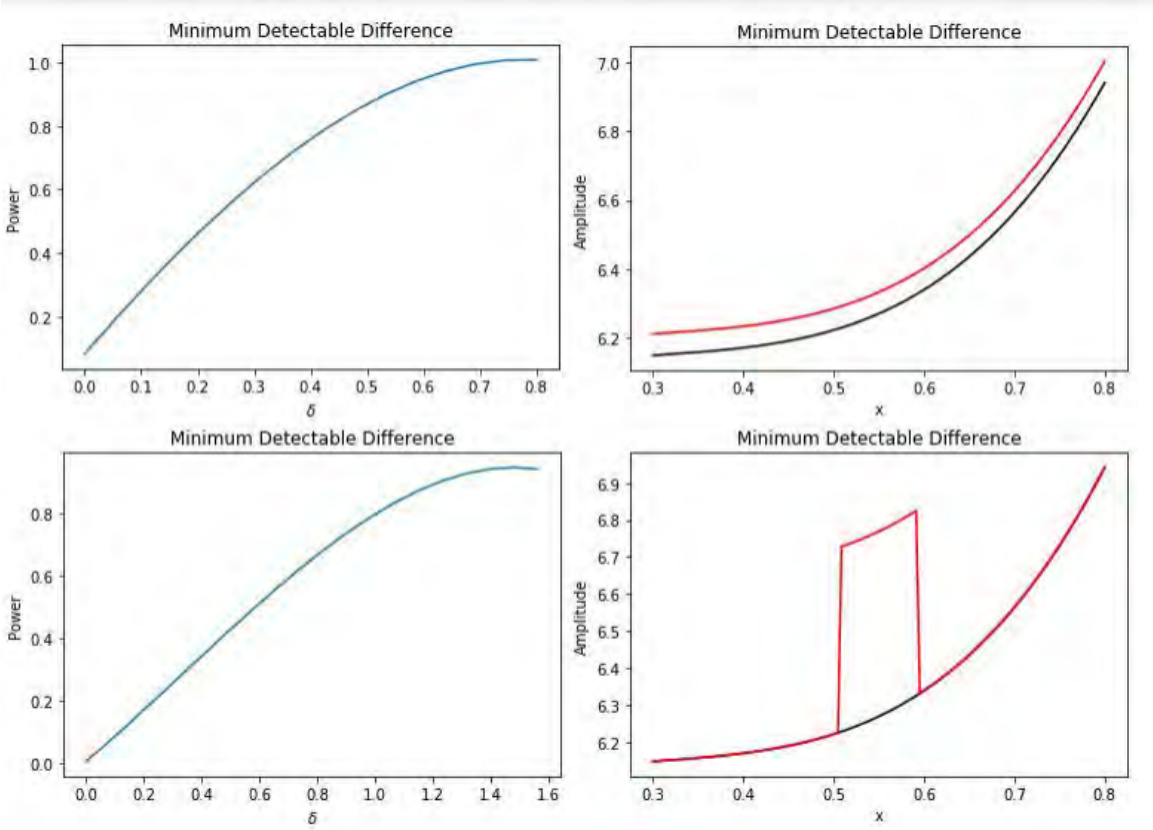

Figure 8. Sensitivity δ for local(top) and global(bottom) profile changes where $\sigma = 0.10$. The charts on the left are power curves which show the δ, SNR, for local and global profile changes. The charts on the right are graphical representations of the minimum detectable difference for the curves f_0 and f_1 and demonstrate the δ that is detectable by the WANOVA box plot at a power of 0.80.

This section depicts the theoretical limits of the WANOVA box plot method, in a controlled environment, to detect local and global anomalies. The WANOVA box plot is capable of determining small global profile changes when the anomaly or SNR, δ, is 0.4. It is shown that this SNR relates to a minimal difference between a control and anomalous curve. The WANOVA box plot is sensitive to local anomalies with local SNR of 1.0. This is shown to be a small to moderate difference between a control and an anomalous curve. The results of this study show the WANOVA Box plot is a useful tool for real world anomaly detection in local and global profile monitoring

problems. This capability of the WANOVA box plot is demonstrated in a real world application discussed below.

3.5 Gaussian Simulation Study from Sun and Genton

Sun and Genton (2011) use five Gaussian simulation test cases to compare the functional bag plot with the functional box plot. This section replicates their study by comparing these methods with the WANOVA box plot. Additionally we create and consider a FPCA extension of the functional bag plot in the simulation study. The actual simulation statistics for the functional box plot and the functional bag plot are from Sun and Genton (2011), but the figures are generated outside of that work based on the simulation study conducted in this work.

The description of the five models from (Sun and Genton, 2011) follows.

1. Model 1 is $X_i(t) = g(t) + e_i(t), \qquad 1 \leq i \leq n$, with mean $g(t) = 4t, t \in [0, 1]$, and where $e_i(t)$ is a stochastic Gaussian process with zero mean and covariance function $\gamma(s, t) = exp\{|ts|\}$.

2. Model 2 includes a symmetric contamination: $Y_i(t) = X_i(t) + c_i\sigma_i K$, where c_i is 1 with probability q and 0 with probability $1 - q$, K is a contamination size constant, and σ_i is a sequence of random variables independent of c_i taking values 1 and $\neg 1$ with probability $\frac{1}{2}$.

3. Model 3 is partially contaminated: $Y_i(t) = X_i(t) + c_i\sigma_i K$, if $t \geq T_i$, and $Y_i(t) = X_i(t)$, if $t > T_i$, where T_i is a random number generated from a uniform distribution on [0, 1].

4. Model 4 is contaminated by peaks: $Y_i(t) = X_i(t) + c_i\sigma_i K$, if $T_i \leq t \leq T_i + l$, and $Y_i(t) = X_i(t)$ otherwise, where T_i is a random number from a uniform distribution in $[0, 1\neg l]$.

5. Model 5 considers shape contamination with different parameters in the co-

variance function $\gamma(s,t) = kexp\{\neg c|t\neg s|^{\mu}\}$. The basic model 1, $X_i(t) = g(t) + e_{1i}(t)$, has parameter values $k = 1, c = 1$, and $\mu = 1$ for the covariance function of $e_{1i}(t)$. To generate irregular curves, let $Y_i(t) = g(t) + e_{2i}(t)$, where $e_{2i}(t)$ is a Gaussian process with zero mean and covariance function parameters $k = 8, c = 1$, and $\mu = 0.2$. The contaminated model is given by $Z_i(t) = (1\neg c_i)X_i(t) + c_iY_i(t), \qquad 1 \leq i \leq n$, where c_i is 1 with probability q and 0 with probability $1\neg q$. In the simulation studies, we generate $n = 100$ curves with parameters $q = 0.1$, $K = 8$, $l = 3/49$.

The first model includes 100 uncontaminated curves that are generated based on the methods in Sun and Genton (2011). The basic model is $y = 4t + e(t)$ where $e(t)$ is a Gaussian error kernel. The first model is a control set since it contains no contaminated curves. Figure 9 shows a sample of 100 uncontaminated curves, and the associated box plots for the data. Each functional method is tested for 1000 replications of 100 uncontaminated curves. 10 shows the rainbow plot from Hyndman and Ullah (2007) and separately the FPCA extension of the functional bag plot which includes the first two functional principal components and the FPCA bivariate plot for the uncontaminated curves. The rainbow plot shows the degree of centrality for each curve based on the color of the rainbow, which represents the 100 uncontaminated curves well. Further, the functional principal components bivariate plot shows a close spread for the 100 uncontaminated curves where the plot of the first two functional principal components show the basic trends in the uncontaminated functions.

Table 1 lists, $\hat{p}_0$, which is the probability, for a given replication, that there are no contaminated curves detected for a set of 100 curves. Table 1 also lists $\hat{p_f}$ and associated standard deviation. $\hat{p_f}$ is the probability of classifying an uncontaminated curve as contaminated. In this study, $\hat{p_f}$ can be thought of as the Type I error rate.

Table 1 shows the WANOVA box plot has a probability of 0.760, $\hat{p_0}$, that there are no contaminated curves detected in a set of 100 uncontaminated curves. The

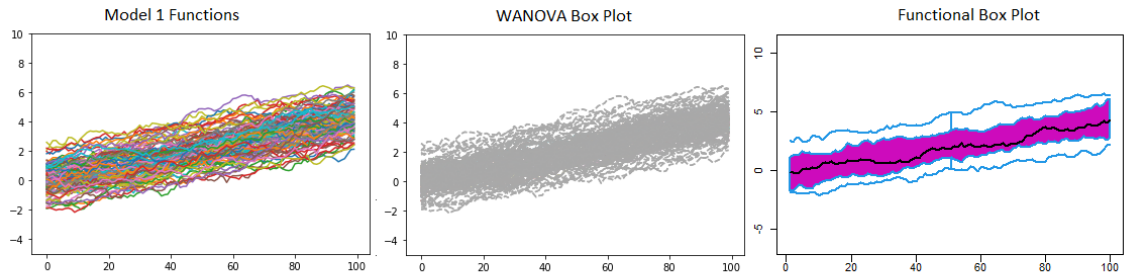

Figure 9. 100 uncontaminated Gaussian functions (left), WANOVA box plot of the 100 Functions (middle), Functional box plot of the 100 Functions (right). The two functional box plot methods accurately classify the 100 uncontaminated Gaussian Functions.

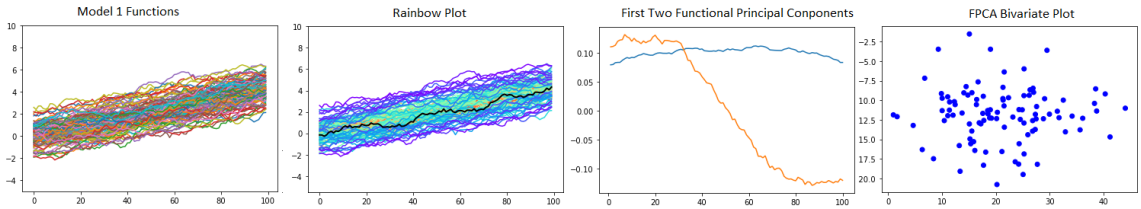

Figure 10. 100 uncontaminated Gaussian functions (left), Rainbow plot of the 100 Functions (middle left), First two functional principal components of the 100 functions (middle right). The FPCA bivariate plot of the 100 functions.

functional box plot and functional bag plot have probabilities of 0.932 and 0.244, respectively, of detecting zero contaminated curves in a sample set of 100 uncontaminated curves. The WANOVA box plot falls within the range of current methods for $\hat{p_0}$ rates for this study. The probability of declaring an outlier for any given uncontaminated curve, type I error, is 0.0034 for the WANOVA box plot. The functional box plot and functional bag plot have Type I error rate probabilities of 0.0007 and 0.024, respectively. The WANOVA box plot falls within the range of current type I error rates for functional outlier detection. While type I error rates are very low, $\hat{p_0}$ is not as low. It is not uncommon that there will be a type I error in a set of 100 curves. The slightly higher type I error rate is acceptable in many applications where type II error rate is more important and can be seen as more of a nuisance factor. For a given uncontaminated curve there is less than a one percent probability that it will be classified as a contaminated curve by the WANOVA box plot method. This

is an acceptable type I error rate for many industrial processes.

The thresholding method from equation (17) removes, kills, 54.1% of the wavelet coefficient, listed in table 3. This indicates, for the DWT with three levels, that many of the detailed wavelet coefficients are retained and included in the WANOVA statistic calculation. This is likely because of a small threshold value since there were only small differences, in shape or magnitude, represented by the detailed wavelet coefficients such that most of the detailed coefficients were retained.

Sun and Genton (2011) use models two through five to test the given functional outlier methods against different types of model contamination. Models two through four introduce a few magnitude contaminated curves with decreasing degrees of bias intervals; this is intended to represent a magnitude outlier. The smaller bias intervals make the detection of magnitude bias increasingly challenging.

Figure 11 shows the original data as well as the WANOVA box plot and the functional box plot. In the single simulation two event, both box plots clearly indicate the outlying functions which are larger in magnitude than the more central curves. Over many simulation, the WANOVA box plot slightly outperforms the functional bag plot, and the functional box plot based on $\hat{p_f}$ and $\hat{p_f}$. The WANOVA box plot is 0.005 and 0.090 more probable, respectively, to correctly detect a contaminated curve and 0.020 and 0.003 less probable, respectively, to incorrectly detect an uncontaminated curve as contaminated for this simulation, table 2.

The rainbow plot effectively presents the centrality of each curve in the rainbow gradient, but does not clearly distinguish the outlier region from the central region in terms of color. Some of the outlying region curves are similar to the curves on the fringe of the central region. The first functional principal components shows the general pattern in the curve and the second principal component shows the additional pattern of the magnitude outliers. The subsequent FPCA bivariate plot shows a clear

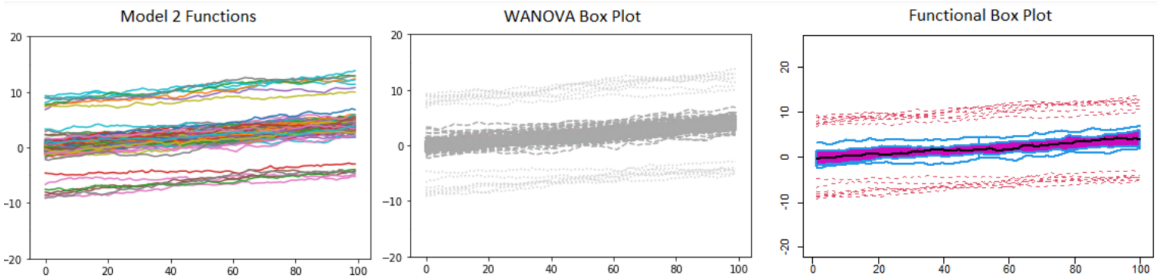

Figure 11. Illustration of simulation case study two which shows mostly uncontaminated curves from model two and several curves contaminated with a magnitude shift. Column one shows the original data. Column two shows the WANOVA box plot. Column three shows the functional box plot. The two functional box plot methods accurately categorize the curves as contaminated and uncontaminated. Note that the WANOVA box plot can detect global profile changes with a SNR of 0.4 at a power of 0.80 as shown in section 3.4.

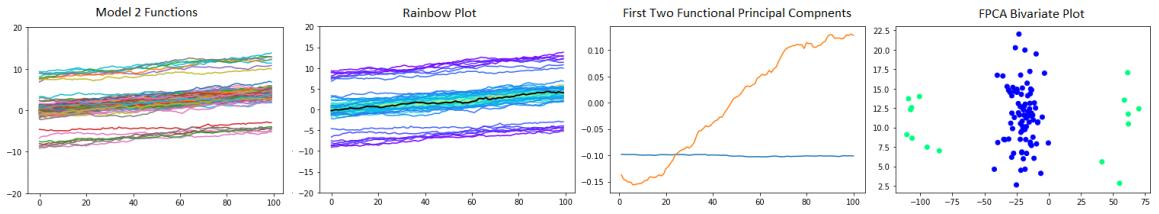

Figure 12. Illustration of test case two with (left) and two functional outlier methods, the rainbow plot (middle left)and the FPCA plots (middle right and right). The rainbow plot clearly shows the curves in terms of centrality based on a color gradient where the central curves are lighter and the outliers are progressively darker. The first two functional principal components, used to create the bivariate plot, are shown (middle right). The FPCA bivariate plot (right) of the 100 functions shows the non-outlying colors clustered together in blue and the outliers outside of the central region in green.

separation between the outlying curves and the uncontaminated curves, as shown in figure 12

The thresholding method from equation (17) retains 51.8 % of the wavelet coefficient, listed in table 3. This indicates, for the DWT with three levels, that some of the magnitude outlier is contained in larger wavelet coefficients which increases the threshold value and kills more wavelet coefficients. Most of the magnitude outlier information, though, is maintained in the smooth wavelet coefficients which are not thresholded.

Model 3 is an extension of model 2 with the change that the contaminated functions jump to magnitude outliers at a random time step. In the single instance of simulation three, shown in figure 13, the WANOVA box plot correctly classifies the contaminated and uncontaminated curves. The functional box plot fails to detect one of the contaminated curves which spikes near the end of time, such that it is a central curve for nearly the entire curve. As discussed in section 3.2.5, wavelet analysis is sensitive to abrupt changes such as the jumps in model 3, and represent this signal as a large detailed wavelet coefficient generated from the DWT. The WANOVA box plot is 0.132 and 0.075 more probable to detect a simulation two contaminate and 0.0003 and 0.0150 less probable to detect an uncontaminated curve as a contaminated curve than the functional bag plot and functional box plot, table 2.

The rainbow plot performs similarly to the the functional box plot; correct classification with a few missed detections of curves which jump at the end of the signal. The last contaminated curve in the rainbow plot in figure 14 can be seen to have one of the most central colors, since it is one of the most central curves for almost the entirety of it's length. The first and second functional principal components are similar to those of the uncontaminated curves while also including micro jump patterns. The FPCA bivariate plot shows good separation between the contaminated

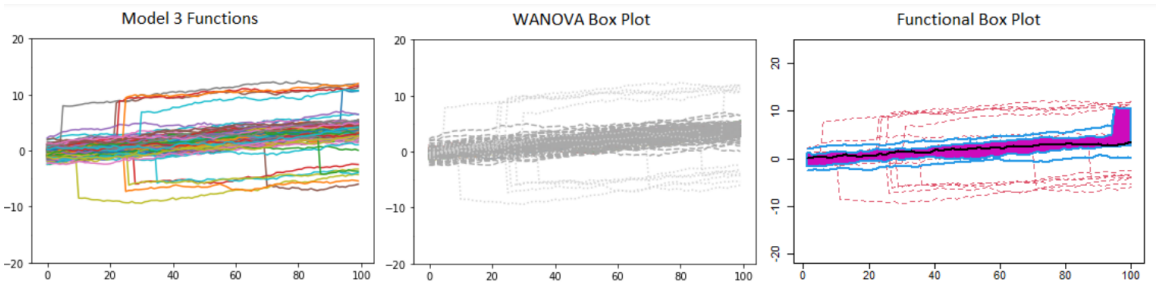

Figure 13. Illustration of simulation model 3 with partial contamination. Column one shows the original data. Column two shows the WANOVA box plot which accurately detects the contamination and uncontaminated curves. Column three shows the functional box plot which fails to detect one of the contaminated curves. This can be seen near the end of time where a peak is included in the blue central region.

and uncontaminated curves and appears competitive with the WANOVA box plot in this simulation instance

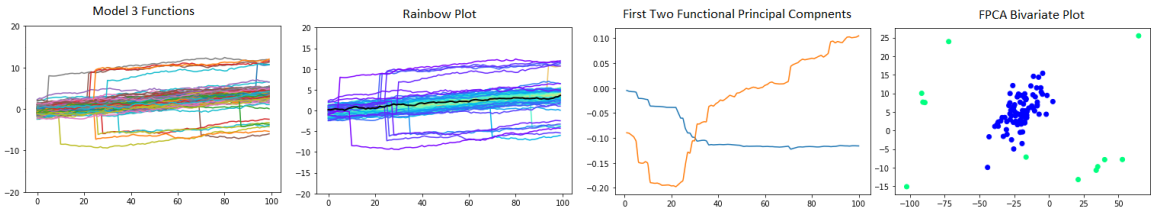

Figure 14. Illustration of test case three with (left) and two functional outlier methods, the rainbow plot (middle left)and the FPCA plots (middle right and right). The rainbow plot shows the uncontaminated curves in terms of centrality based on a color gradient, but includes a few of the contaminated curves in the lighter central region. The first two functional principal components, used to create the bivariate plot, show some of the jump pattern present in the jump contaminates (middle right). The FPCA bivariate plot (right) of the 100 functions shows the non-outlying colors clustered together in blue and the outliers outside of the central region in green. Note that one of the curves is very close to the central region, but still possible to delineate from the uncontaminated curves.

The threshold method retains 48.3% of the original wavelet coefficients for this simulation model. The wavelet threshold value increases for the detailed wavelet coefficients representing the large single spike in the contaminated curves. This sets more wavelet coefficients to zero.

In simulation four, the contaminated curves form peaks; they quickly jump to magnitude outliers, remain for a short period then return to center. While the WANOVA box plot does not degrade in performance for this test case, $\hat{p_c} = 0.997$, the

functional bag plot and the functional box plot degrade in contamination detection performance. The functional bag plot has a probability of detecting contaminated curves, $\hat{p_c} = 0.186$. Slightly better, the functional box plot has a $\hat{p_c}$ of 0.550, see table 2. Figure 15 shows an instance of this test case where the WANOVA box plot accurately classifies curves based on contamination; however, the functional box plot miss classifies three of the contaminated curves as uncontaminated.

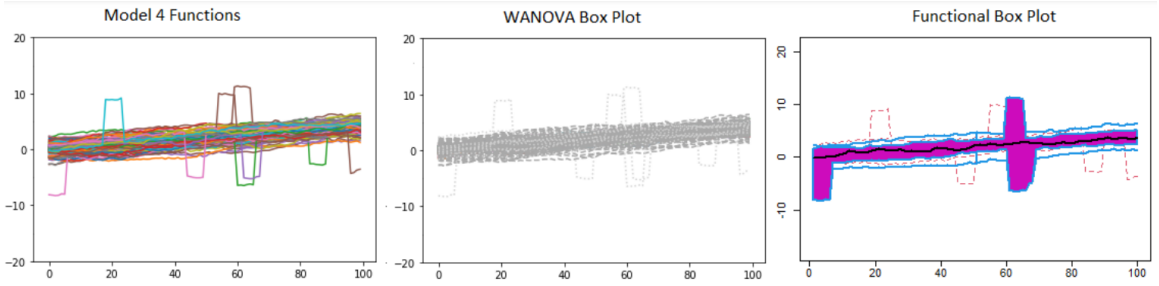

Figure 15. Illustration of simulation model four with partial peak contamination. Column one shows the original data. Column two shows the WANOVA box plot which accurately detects the contaminated and uncontaminated curves. Column three shows the functional box plot which fails to detect four of the contaminated curves with peaks. This can be seen where the peaks are included in the inner central region.

In section 3.4, we show the WANOVA box plot can detect local differences with SNR, $\delta = 1$, at a power of 0.8. These peaks from Sun and Genton (2011) have much larger SNRs, such that the WANOVA box plot could detect much smaller peaks. The very small missed detections in this simulation are likely from contaminated curves that contain only a small portion of the peak. We justify the selection of the wavelet basis for this application in section 3.2.5 due to time localization and spatial adaptivity properties as a linear and non-linear estimator of functional data. Figures 16 and 17 show how these properties capture irregularities represented as DWT generated wavelet coefficients. 16 presents three curves, two contaminated with a peak and one uncontaminated curve. Following the results of the simulation, the WANOVA box plot detects the contaminate and the functional box plot does not. In the first two rows of figure 16, the approximate (smooth) wavelet coefficients, the

first level of detailed wavelet coefficients, and the WANOVA test are shown. In terms of spatial adaptivity, the majority of the signal is stored in a few wavelet coefficients. These large wavelet coefficients are time localized at the peak where $t = 80$. Note that this is $t = 40$ in the wavelet domain. The third column shows the WANOVA statistic across t, which remains less in magnitude than the WANOVA statistic until peak, where both the approximate and detailed coefficients sharply increase. In the final row, the difference in the uncontaminated signals, in terms of the smooth wavelet coefficients, the detailed wavelet coefficients, and the WANOVA statistic is shown. These two signals are are actually moderately different, but we fail to reject based on the WANOVA test. The smooth coefficients show the moderate approximate difference between the two signals, but the detailed wavelet coefficients are mostly noisy and very small.

The rainbow plot and the FPCA bivariate plot perform poorly for the model four simulation. The first two functional principal components show the general pattern of the uncontaminated curves and some of the peak patterns, but overall the FPCA bivariate plot only classifies a few of the peaks as outliers.

In this model 0.475, see table 3, of the wavelet coefficients are retained. The large peaks, created by two abrupt jumps, significantly increases the threshold value such that many of the wavelet coefficients not representing the jumps difference between signals are shrunk to zero.

Model five contains curves contaminated such that the shape of the curve is generated differently than the original curve; this is intended to represent a shape outlier. Model five is shown in Figure 19 in the fourth row. The contaminated curves oscillate near the values of the uncontaminated curves which makes outlier detection challenging. The proportion of contaminated curves correctly detected, $\hat{p_c}$, by the different methods over 1000 replications for model five is shown in Table 2.

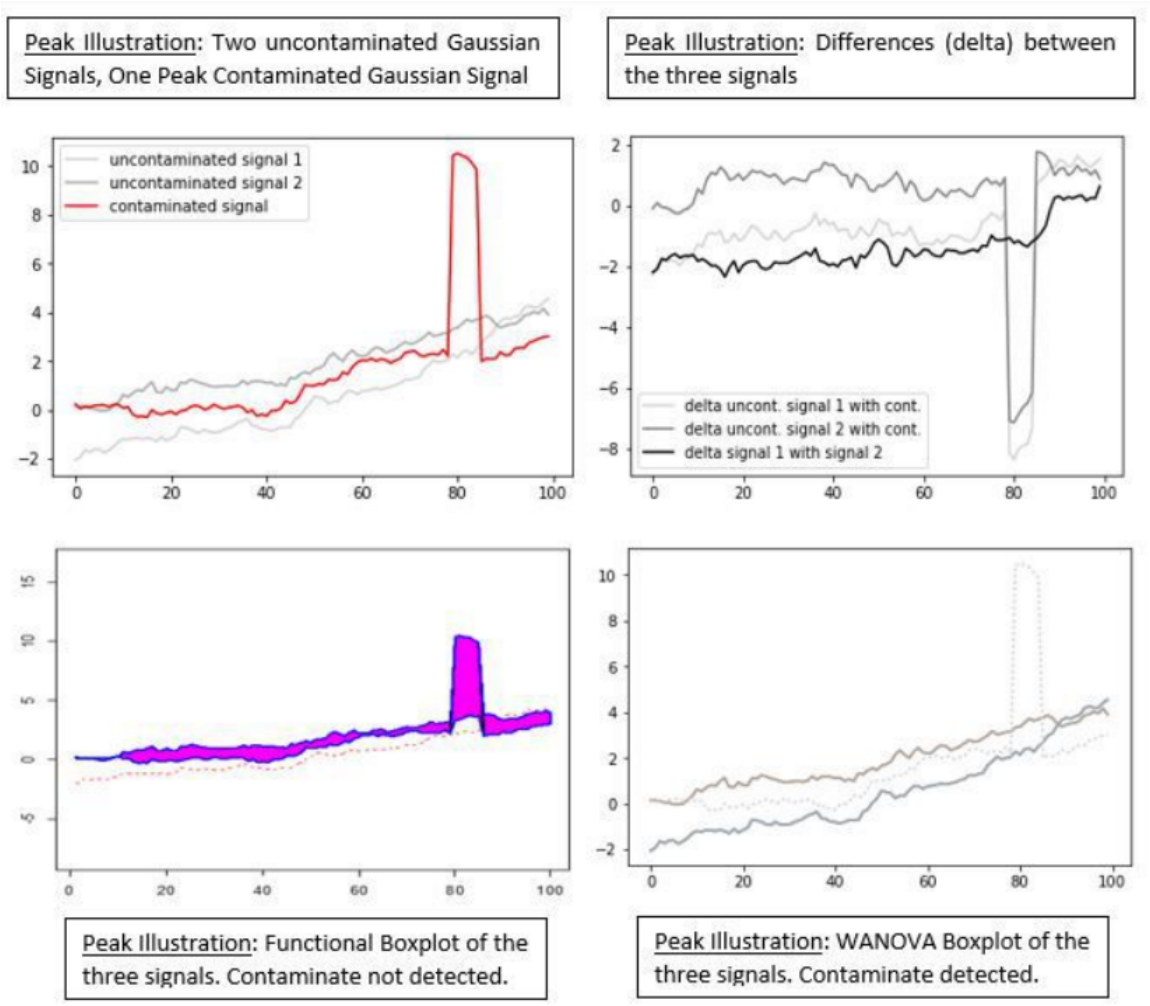

Figure 16. Illustration of the WANOVA box plot method DWT method. The first plot (upper left) shows three of the test case four curves plotted with one peak contaminate and two uncontaminated curves. The second plot shows the pairwise differences among the three curves. The third plot (bottom left) shows the functional box plot of the three curves which includes an uncontaminated and contaminated curve as the central region and an uncontaminated curve as an outlier. The fourth plot (bottom right) shows the WANOVA box plot which correctly classifies the three curves.

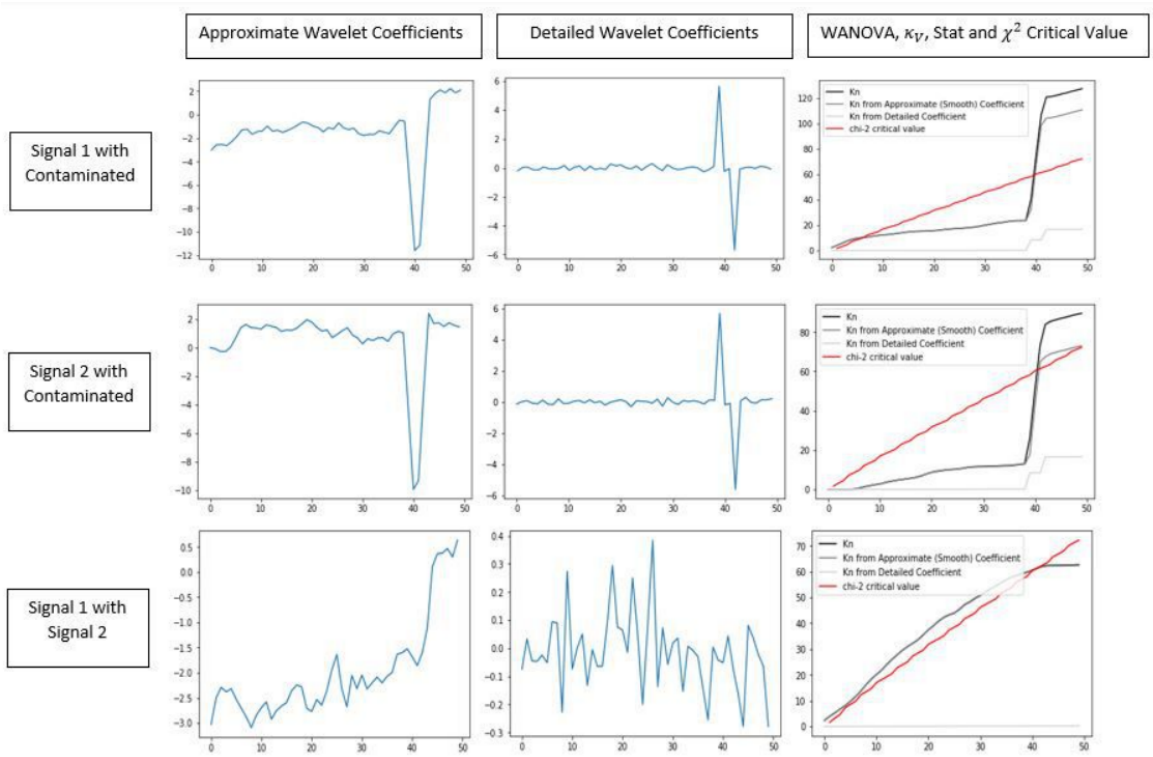

Figure 17. These figures show how the pairwise WANOVA scores are calculated and highlights the ability of the DWT to capture significant artifacts from a signal. In the first column the smooth wavelet coefficients are shown for the difference among signals which are labeled. The main takeaway from the first column is that it is an approximation of of the difference among the signals. In the second column the detailed coefficients are shown for the DWT of the difference between signals. Note that in the first and second row there are significant detailed wavelet coefficients representing the spike localized at the correct part of the curve. In the last row of the second column, the detailed coefficients represent small noisy changes in the signal. The last column shows the pairwise WANOVA scores and how the smooth and detailed coefficients contribute to the WANOVA statistic. Note that by the end of the signal the first two rows exceed the WANOVA critical value and the third row does not. Note also that the detailed coefficients have little effect on the WANOVA statistic until the peak is represented in the signal

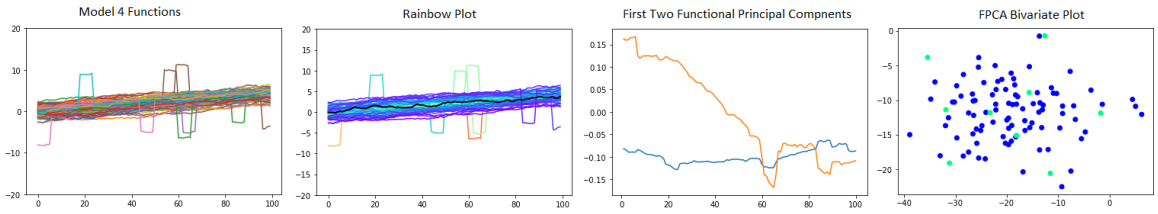

Figure 18. Illustration of test case four with (left) two functional outlier methods, the rainbow plot (middle left)and the FPCA plots (middle right and right). The rainbow plot shows the uncontaminated curves in terms of centrality based on a color gradient, but includes many of the the contaminated curves in the lighter central region. The first two functional principal components, used to create the bivariate plot, show some of the peak pattern present in the peak contaminates (middle right). The FPCA bivariate plot (right) of the 100 functions shows the non-outlying colors clustered together in blue, but also shows the outliers, in green, clustered in the same region. Note that FPCA bivariate plot does not do well distinguishing peak contaminates with uncontaminated curves for this test case.

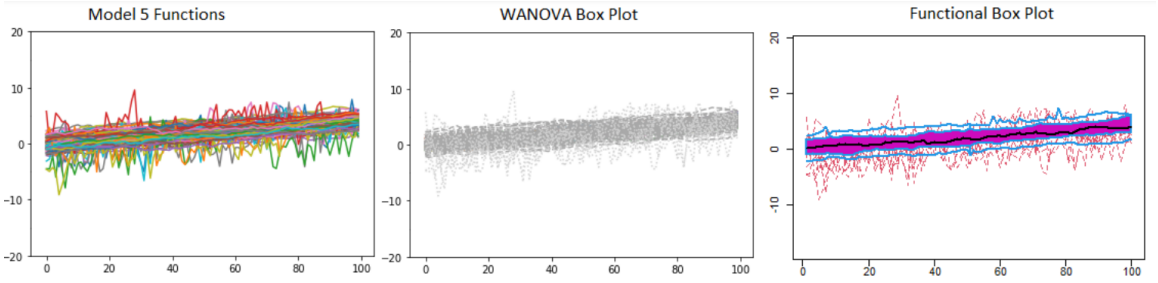

Figure 19. Illustration of test cash five which includes shape contaminates, curves that follow a different pattern than the uncontaminated Gaussian curves. The WANOVA box plot (middle) accurately classifies the curves. The functional box plot (right) misclassifies some of the contaminates as uncontaminated. The contaminate pattern can be seen in the blue curves that define the whisker region.

The functional box plot correctly detects a given curve from the model five set with probability 0.786. The functional bag plot detects contaminated curves with probability 0.327. The WANOVA box plot correctly detects contaminated curves with probability 0.835. The WANOVA box plot outperforms the other methods in terms of p_c and p_f error for shape outliers in this study. Table 2 also lists the probability of false detection, $\hat{p_f}$, of contaminated curves and associated standard deviation. This is incorrectly detecting that an uncontaminated curve is contaminated. The false detection of contaminated curves rate for the three methods are 0.0003, 0.0166 and 0.000, respectively. The rainbow plot and the bivariate plot in figure 20 do not classify the shape contamination's properly.

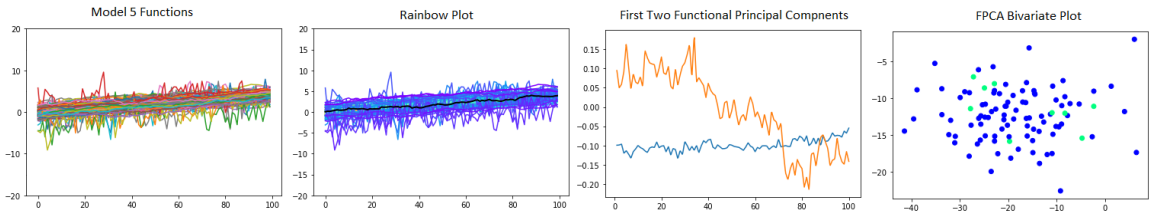

Figure 20. Illustration of test case five with (left) two functional outlier methods, the rainbow plot (middle left)and the FPCA plots (middle right and right). The rainbow plot shows the uncontaminated curves in terms of centrality based on a color gradient, but includes many of the the contaminated curves in the lighter central region. The first two functional principal components, used to create the bivariate plot, show some of the peak pattern present in the shape contaminates which abruptly shift direction up and down (middle right). The FPCA bivariate plot (right) of the 100 functions shows the non-outlying colors clustered together in blue, but also shows the outliers, in green, clustered in the same region. Note that FPCA bivariate plot does not do well distinguishing shape contaminates with uncontaminated curves for this test case.

The WANOVA box plot outperforms the other methods for the Sun and Genton (2011) simulation case studies. The WANOVA box plot is effective at classifying local and global difference for magnitude and shape outliers because of the spatial adaptivity and localization of the wavelet basis and classification capabilities of functional depth.

Outlier detection for the uncontaminated model 1			
Method	$\hat{p_0}$	$\hat{p_f}$	$SD(\hat{p_f})$
Functional box plot	0.9320	0.0007	0.0027
Functional bag plot	0.2440	0.0240	0.0062
WANOVA box plot	0.7600	0.0034	0.0068

Table 1. Outlier detection for the uncontaminated curves from mode1 1 across the various functional methods. $\hat{p_0}$ is the probability that none of the 100 uncontaminated curves for each particular instance of the 1000 runs is detected as a contaminate. $\hat{p_f}$ is the probability of classifying an uncontaminated curve as contaminated. SD is the standard deviation for the 100 curves over 1000 replications

Outlier detection for contaminated models 2-5				
Method	Model 2	Model 3	Model 4	Model 5
$\hat{p_c}(StandardDeviation)$				
Functional box plot	.991 (.031)	.837(.139)	.550(.184)	.786(.153)
Functional bag plot	.995(.055)	.884(.108)	.186(.157)	.327(.170)
WANOVA box plot	1.00 (.000)	.969(.053)	.997(.017)	.835(.153)
$\hat{p_f}(StandardDeviation)$				
Functional box plot	.0003(.0019)	.0003(.0020)	.0005(.0027)	.0003(.0018)
Functional bag plot	.0180(.0580)	.0150(.0460)	.0180(.0550)	.0170(.0510)
WANOVA box plot	.0000 (.0000)	.0000 (.0000)	.0014(0.0049)	.0000 (.0000)

Table 2. Outlier detection for well behaved Gaussian models with contamination for three outlier detection methods. $\hat{p_c}$ is the probability of detecting a contaminate for a given contaminate curve. Note that the WANOVA box plot outperforms the other box plot methods for contaminate detection across the different simulation models with the exception of false detection of uncontaminated curves, $\hat{p_f}$, for model four where the functional box plot is marginally better.

Thresholding for Gaussian Simulation	
Model	Remaining Coefficients
Model 1	0.541
Model 2	0.518
Model 3	0.483
Model 4	0.475
Model 5	0.465

Table 3. Remaining wavelet coefficients from the DWT after thresholding using cross validated block thresholding. The threshold values are calculated based on equation 17. The threshold values change based on the specific curves being compared. Model 1 has the fewest irregularities in the curves such that the detailed wavelet coefficients are relatively smaller and the resultant thresholding value is smaller so more wavelet coefficients are retained.

3.6 WANOVA box plot Application for Real World NOx data

The NOx data set analyzed by Febrero et al. (2007) presents NOx levels in an industrial neighborhood in Spain, Poblenou. NOx is a contaminant produced by vehicles and industrial plants. The NOx levels are measured hourly for several years. The authors are interested in the effect driving has on the NOx levels; they consider data for working days and non working days and holidays. Table 4 shows a list of Spanish holidays. These dates are important for establishing the working day set, the non-working day set, and to justify outliers in both of the functional sets.

The data sets for the working and non-working curves used in the original study are compared to the data set received from the Catalonia government in Figure 21. The data is very slightly different from the NOx data presented in the original study. There are some disparities, such as the presence of peaks at hour 15 for working and nonworking days in the re-engineered data. Additionally, the magnitude of the NOx readings are larger in magnitude in the original study than in the re-engineered data. Some of the large entries in the original data approach 300 units of NOx near peak commuting hours, 0800. Still, the data are close enough for comparative purposes.

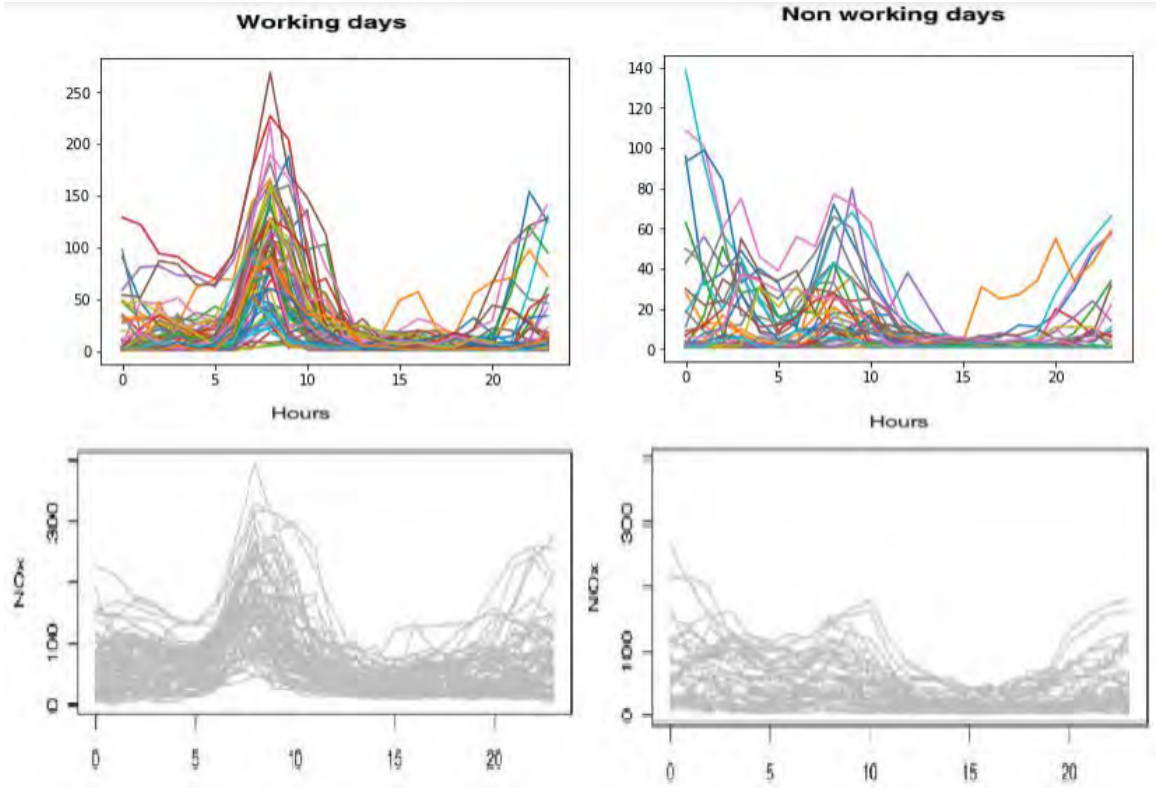

Figure 21. Comparison of NOx sorted working and non-working data; re-engineered working day NOx curves (top left) and re-engineered non-working days NOx curves (top right), original data is shown in the bottom column. The data are exact with the exception that the re-engineered working data is omits a single curve and the re-engineered data includes two additional curves. This is taken into consideration during the analysis.

Breakdown of Spain Weekends, Holidays and Missing Data	
Dates	Holiday
25 Mar, 28 Mar	Good Friday, Easter Monday
16 May, 02 May	Whit Monday, Labor Day
26 Feb, 27 Feb	Feb Weekends
05, 06, 12, 13, 19, 20, 26, 27 Mar; 02, 03, 09, 10, 16, 17, 23, 24, 30 Apr	Mar, Apr Weekends
01 07 08 14 15 21 22 28 29 May; 04, 05, 11, 12, 18, 19 Jun	May, June Weekends
27 Mar	Missing Data Weekends and Festive Days
10, 14 March, 05-08, 11, 12, 14 April,	
10 May, 17 June	Missing Data Working Days
72 Complete Working Days	
40 Complete Non-Working Days	Total Work, Non-Working Days

Table 4. The different dates for holidays in Spain during 2005. These are important in the subsequent analysis since traffic flow, which influences NOx emissions, changes significantly on holidays.

Even with some data discrepancy the WANOVA box plot tests compared well to the bootstrap methods of Febrero et al. (2007). The WANOVA box plot method found two extra outliers in the working data sets, a total of four. The functional box plot detects the 24 hour curves for the dates 11, 15, 18 March and 29 April as outliers. The depth methods find that the dates 18 March and 29 April are outliers. 11 and 15 March are additional findings from the WANOVA box plot. These dates may be justified as outliers since they fall near the Eastern holiday in Spain in 2005. Febrero et al. (2007) use the Eastern vacation to justify the 18 March outlier.

The WANOVA box plot detects 19 March, and 30 April as outliers in the non-working data set. 19 March and 30 April intersect with the dates detected by the depth method. The authors justify 30 April as an outlier because it coincides with the beginning of a long weekend. They justify 19 March as an outlier since it fell on the beginning of the Eastern vacation in Spain.

3.7 COVID 19 Case Study

The New York Times COVID-19 repository, "https://github.com/nytimes/covid-19-data", contains data sets which measure different aspects of the pandemic across the world from January 2020 through March 2021. We use the cases per county data which measure the total number of COVID-19 cases reported by each county in the United States. We narrow our scope to the Middle Western (Midwest) United States during the Summer period of June 21, 2020 to September 20, 2020.

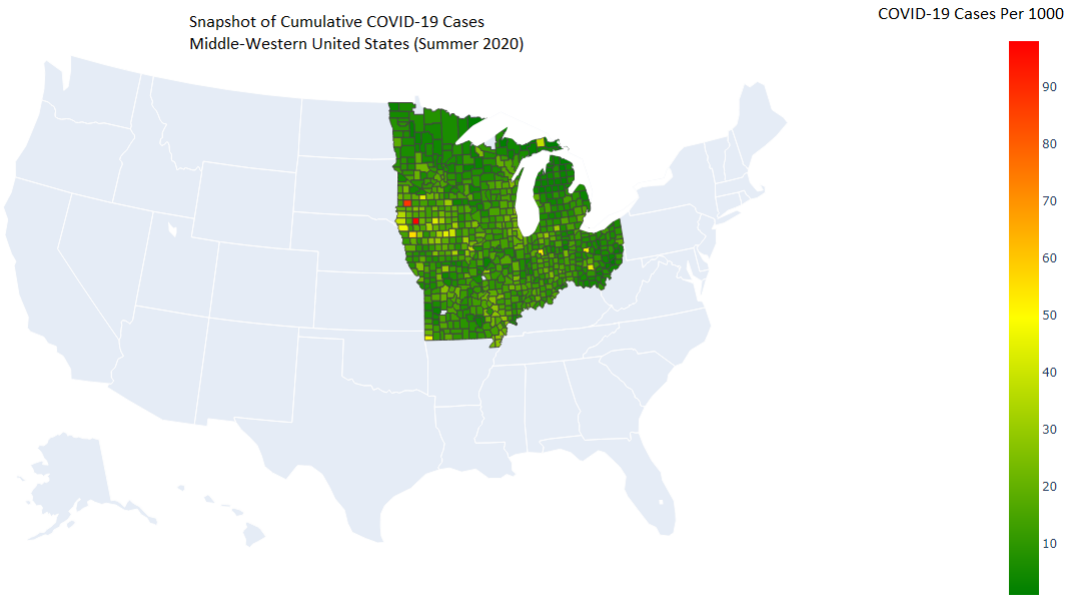

Figure 22. Cumulative COVID-19 Cases per 1000 people by county September, 9^{th} 2020. The color grid indicates the number of people, per capita, in each Midwest county that tested positive for COVID 19 on or before county September, 9^{th}. Most of the counties are in the 10 cases/1000 range, but several counties across the center of the region have elevates cases closer to 90 cases/1000. These counties are primarily in Iowa, Indiana, Minnesota, and Ohio

The data show a high rate of infection uniformly spread across the Midwest United States with a county high of 94 cases of COVID-19 per 1000 people by September, 9^{th} 2020. We extend this data analysis to functional data analysis by considering the patterns and shape of the curves representing cumulative COVID-19 cases across

Summer 2020. In particular, we apply the functional box plot and WANOVA box plot to determine whether there are any Midwest counties that are functional outliers.

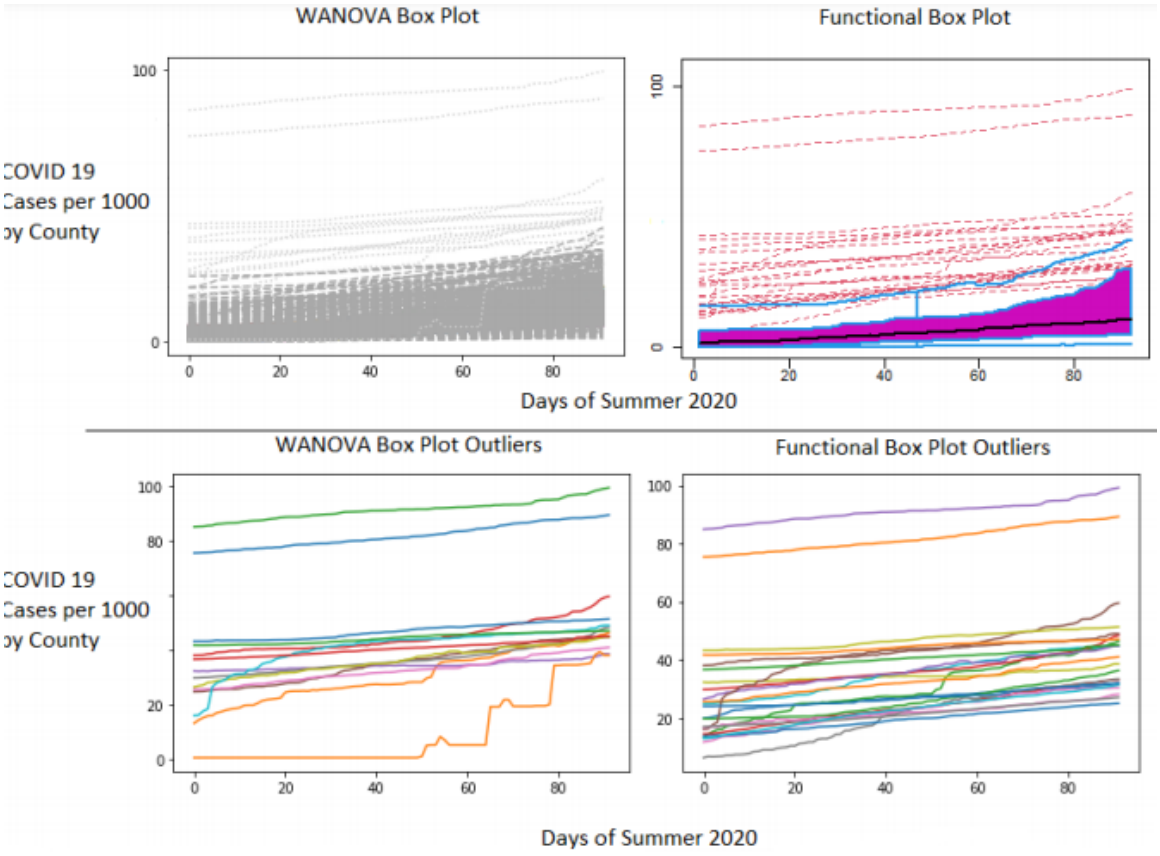

Figure 23. The WANOVA box plot for cumulative COVID-19 cases in Midwest counties for the Summer of 2020 (top left) and the functional box plot for the same data (top Right). The COVID-19 cumulative case curves for the Summer of 2020 generated by the WANOVA box plot (bottom left) and the functional box plot (bottom right) are also shown. The WANOVA box plot detects a shape outlier that falls within the range of both box plots and the functional box plot detects several additional magnitude outliers on the fringe of the whisker region for both box plots.

Figure 24 shows the difference between the outliers detected by the WANOVA box plot and the functional box plot. An interesting difference between the two methods is that one of the counties, which appears to be a shape outlier, is detected by the WANOVA box plot and not detected by the functional box plot. The particular county looks similar to a step function rather than a relatively smooth curve, like the rest of the data. This county starts with a relatively low number of cases with a

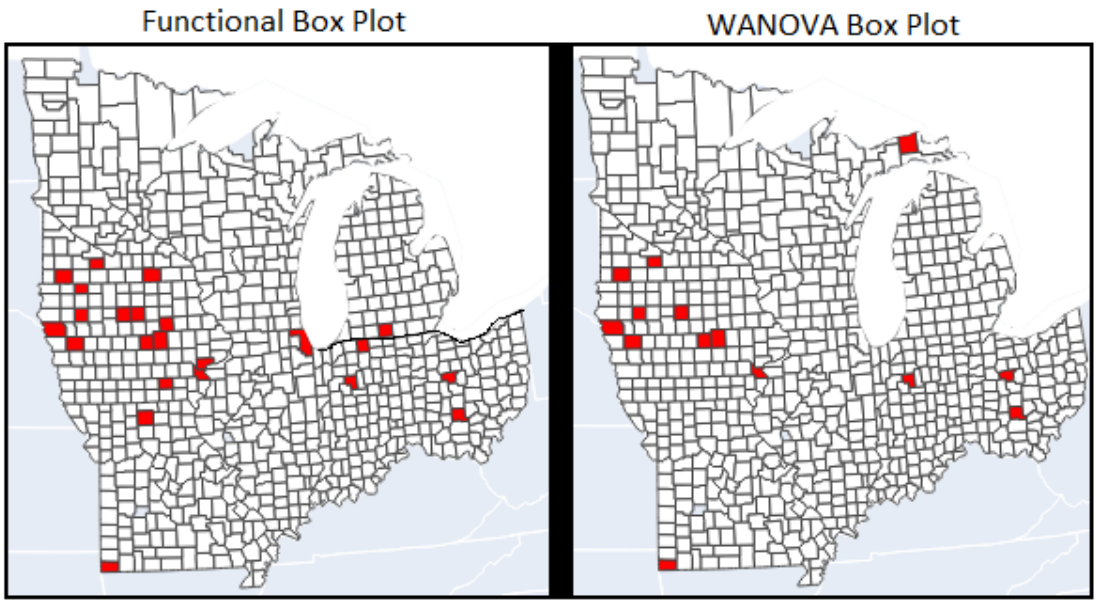

Figure 24. Outliers are listed in red for Midwest U.S. counties for Summer 2020 cumulative COVID-19 cases per 1000 residents based on the Functional box plot(left) and the WANOVA box plot (right). The outliers may reflect counties with high volume of Summer traffic as they roughly form a band that follow U.S. highways 70, 80 and 90. Additionally, the WANOVA Box Plot detects Luce county as a shape outlier which is a popular Summer recreational destination in Michigan's Upper Peninsula.

low rate of growth. By the middle of Summer the reported COVID-19 cases in this county then grow at a faster rate than the rest of the Midwest counties. Eventually, the number of cases in this county exceed the magnitude of the box region for both of the box plot methods yet the WANOVA box plot is the only method to capture this observation. Luce County, Michigan is a smaller county, in terms of population, and likely had a mid-Summer increase in COVID-19 cases due to Summer recreational tourism in the Upper Peninsula. Further their reporting, may look like a step function as they attempted to tabulate new cases. A second difference between the two techniques is that the functional box plot is more sensitive to magnitude outliers. While the WANOVA box plot includes all of the borderline magnitude outlying counties in the whisker region, the functional box plot includes them as outliers. Neither approach is wrong, however the sensitivity of the data set is an important consideration when choosing a box plot for smooth functional data. The additional outliers added by the functional box plot adds to the narrative that per capita cases were higher in the center of the Midwest region shown in figure 24.

This real data study shows ability of the WANOVA box plot and underlying wavelet basis to properly classify functional data that have a different pattern than comparable functional data, but similar magnitude such as in simulation five in section 3.5.

3.8 Computational Time

The computational time associated with a wavelet transform requires $(O)(n)$ operations which is efficient compared to other transforms Mallat and Wen Liang (1992). For example, the fast Fourier transform requires $O(nlog(n))$ computations. While the wavelet transform is a computational efficient baseline, we add a few additional operations which effect the number of computations required. In general, term-by-term

thresholding with a global threshold value, such as VisuShrink, adds a $O(nlog(n))$ penalty. Block thresholding according to Cai (1999) removes the log penalty such that only $O(n)$ computations are required. Additionally, the two-fold cross validation step from Nason (1996) has a computational cost of $O(n)$. In total the transform and thresholding method used in this paper have a computational cost of $O(n^3)$. For code and further reference see "https://github.com/JeffreyDWilliams/WaveletBoxPlot".

3.9 Conclusion

This research extends the use of wavelet based techniques for SPC to descriptive statistics for functional data sets with Gaussian error distributions. The capabilities of the WANOVA method to determine agreement and disagreement between functions are used to create WANOVA box plots. The WANOVA box plot method is sensitive to local and global changes in profiles with low signal to noise ratios. Additionally this method is shown to effectively measure the agreement for Gaussian curves with varying degrees of local and global differences including both magnitude and shape outliers. In this pursuit, the WANOVA box plot is competitive with other functional box plot methods for Gaussian functions. The WANOVA box plot outperforms existing functional methods for outlier detection and is shown to be sensitive to local and global anomalies with small anomaly signal to noise ratios. In the test cases and real world data, the WANOVA box plot significantly improves upon previous methods for outlier detection.

The WANOVA box plot provides previously unavailable statistics in the context of wavelet analysis; the central curve, the IQR of curves, non-outlying extrema, and direct outlier detection. The statistical components of the WANOVA box plot are equally important at the onset of a wavelet analysis as they are for any analysis. Managing outlying data and an obtaining an idea of centrality will improve the ef-

fectiveness of a wavelet analysis in applications such as signal approximation, profile monitoring, and potentially threshold selection.

Appendix

While the WANOVA is a useful tool for finding differences among functional data, it does not tell us which curves are from the true functional population. In order to determine the curves central to the true population from a functional set we find the 50^{th} percentile of functional depth from the comparative WANOVA curves, outlined in figure 25. The comparative WANOVA process finds the WANOVA value between each curve and the rest of the set. That is the original data is temporarily replaced with curves of WANOVA values between a given original curve and the rest of the original observations. These WANOVA curves are then ranked by a depth method.

All curves inside of the central 50^{th} percentile in depth become the sample representation of the true functional population. This is also the box region of the WANOVA Box Plot. Through testing we have found this method to be stricter than using functional depth methods as in Sun and Genton (2011) and this method decreases the likelihood that outliers are included in the box region, particularly shape outliers.

Now that there is a baseline sample of curves which represent the true population, the WANOVA test can classify a curve as an outlier. If we fail to reject that there is a significant difference between a suspected outlying curve and any single curve representing the true population, the curve is no longer a suspected outlier. Note that we are no longer using the comparative WANOVA curves, but the original data.

Figure 25 shows in more detail that this is an iterative process. Clearly, there are no curves in the whisker region during the first iteration so outlier candidates are only compared to the curves in the box region. In successive iterations, curves that

are no longer suspected as outliers are added to the whisker region. These whisker region curves are then added to the basis for comparison, or part of the estimated true functional population, for subsequent iterations.

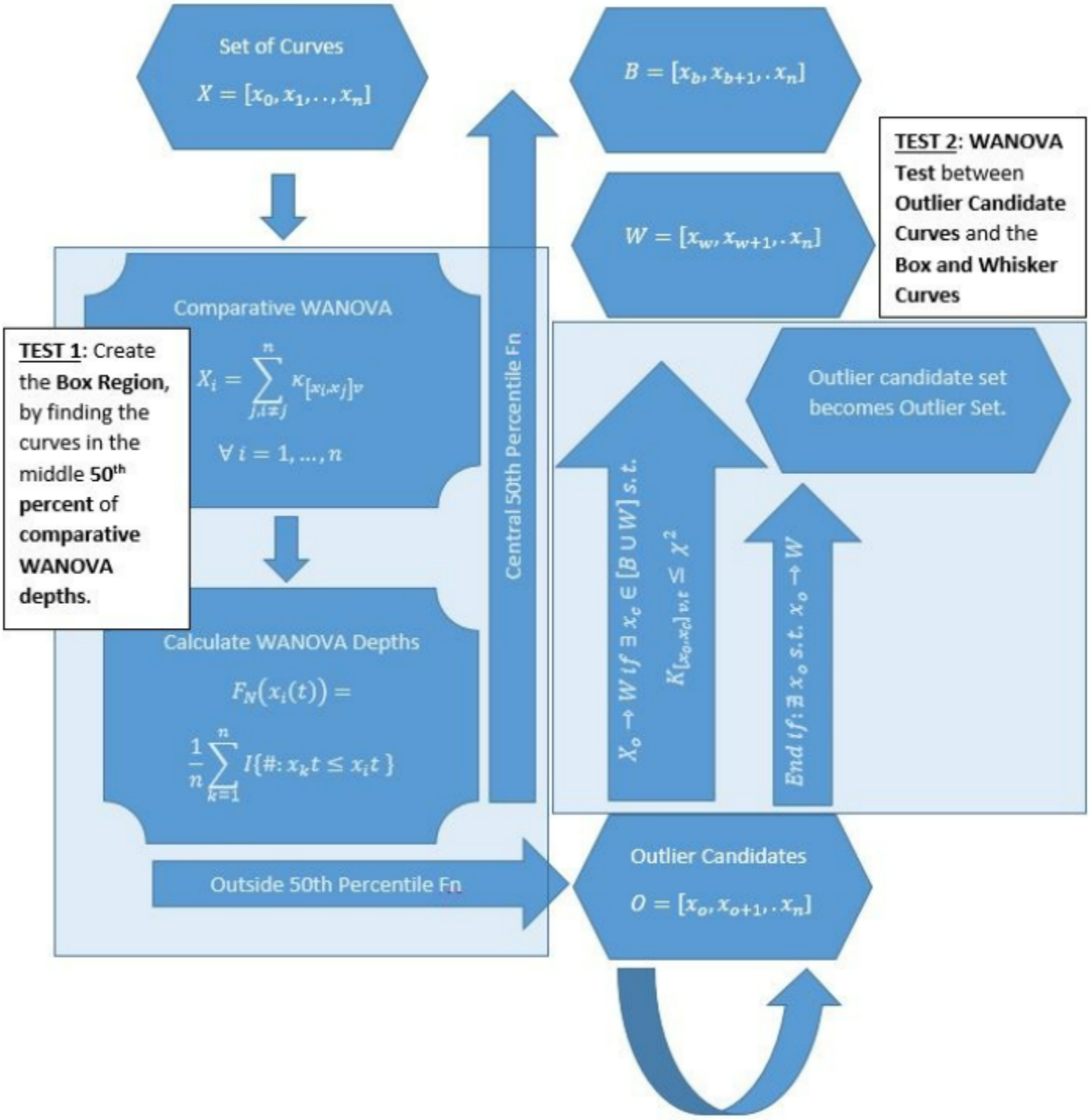

Figure 25. Flow Chart for the Construction of the WANOVA Box Plot. Note the two tests. The first stage is the comparative WANOVA Depth which finds the middle 50th percentile of curve depths in terms of WANOVA scores with each of the other curves. The second stage is the WANOVA test between the outlier candidate curve and the curves already assigned to the box and whisker regions. Note test two iterates until none of the outlier candidates pass the WANOVA test with a curve in the box or whisker region.

IV. Wavelet Thresholding

4.1 Introduction

Donoho and Johnstone (1995, 1994) present VisuShrink and SureShrink to remove noisy wavelet coefficients from a signal. These methods set the precedent for wavelet thresholding techniques for the removal of independent and Gaussian noise. Neumann and Von Sachs (1995) derive asymptotic normality for some non-Gaussian error situations proving that wavelets are intrinsically robust to some non-Gaussian errors. Antoniadis and Fryzlewicz (2006) leverage the multi-resolution analysis property of wavelets to apply wavelet thresholding to signals with some known non-Gaussian error distributions. Cai (1999) shows that thresholding wavelets as a block of coefficients rather than individually, or term-by-term, leverages additional information from neighboring wavelets. This additional information in some cases makes block thresholding more efficient at removing several types of noise. Further, block thresholding removes the $log(n)$ computational penalty associated with term-by-term thresholding. Since blocks of coefficients are considered independently for thresholding, Cai (1999) proves the assumption of independent error terms may be relaxed. While block thresholding is robust to some non-Gaussian error terms, the accuracy of block thresholding alone is sensitive to irregular or unknown error terms. Still, wavelet analysis and the stated block thresholding methods do not always result in robust estimation of some common irregular and noisy data.

Nason (1996) presents a thresholding method which makes no parameter assumptions in selecting thresholding values. Thresholding decisions are based on the cross validated wavelet thresholding reconstruction of the even data from the odd data and *vice versa*. McGinnity et al. (2017) combines the ideas presented in Cai (1999) and Nason (1996) to efficiently threshold wavelet coefficients from signals with varying

error distributions and levels of dependence. This synthesis method is extended by leveraging the multi-resolution analysis property of wavelets by making thresholding decisions resolution dependent. Kim and Oh (2006) present a similar cross validated resolution dependent wavelet threshold technique in R.

We present a thresholding technique subject to resolution dependence and even-odd cross validation where similar wavelet coefficients are carefully grouped using hidden Markov models by magnitude. The hidden Markov cross validation (HMCV) block threshold technique increases the accuracy and efficiency of thresholding estimates by leveraging information from similar wavelet coefficients for Gaussian and non-Gaussian signals.

4.2 Background

4.2.1 Introduction

Wavelet analysis is a set of statistical tools which have some useful properties; these include orthogonality, spatial adaptivity, time localization, and efficiency. These properties make wavelet analysis effective in many areas of statistics, such as statistical process control, image analysis, anomaly detection, and signal processing (Debauchies, 1992). This paper's focus is signal processing. In this section the wavelet reconstruction process for signal processing is discussed. Specifically, thresholding, related thresholding techniques, and the addition of hidden Markov models into the thresholding process are presented.

4.2.2 Wavelets

Wavelet analysis involves the manipulation of small wavelet functions, molded by wavelet coefficients into the shape of a specific signal. The coarse coefficients are associated with the father wavelets and are the inner product of these wavelets with

f, ϕ, and the detailed coefficients are formed by dilations and shifts in the mother wavelet ψ. Debauchies (1992) presents several variations of wavelets, known as the Debauchies family. Each family of wavelets comes with two functions, the mother, ψ, and father wavelet, ϕ. These functions are manipulated according to (23) and (24). The coefficients are the inner product of f and a wavelet as estimated by the DWT.

$$\phi_{jk}(x) = 2^{j/2}\phi(2^j x - k) \tag{23}$$

$$\psi_{jk}(x) = 2^{j/2}\psi(2^j x - k) \tag{24}$$

The wavelet coefficients can be transformed from the wavelet domain to the domain of the signal to create an approximation or reconstruction of the original signal. This inverse discrete wavelet transformation (IDWT) is conducted using the family of wavelet functions manipulated with shifts or dilations in the DWT. Equation 25 shows the reconstruction process where resolution level j represents shifts and k represents dilations. The DWT is used to estimate wavelet coefficients. There are $j = J - 1$ resolution levels where each resolution level j has 2^{j-1} coefficients.

$$f(t) = \sum_k c_{j_0,k}\phi_{j_0,k} + \sum_{j>j_0}\sum_k d_{j,k}\psi_{j,k} \tag{25}$$

4.2.3 Thresholding

Donoho and Johnstone (1994) proved that wavelet coefficient fields are naturally sparse; in many applications there are few meaningful wavelet coefficients which are large in magnitude and there are many noisy wavelet coefficients which are small in magnitude. To accurately approximate a noisy signal, the noisy wavelet coefficients are thresholded; they are decreased or set to zero. Thresholding magnifies the

effectiveness of the wavelet spatial adaptivity property. Shapiro (1993) showed that wavelet thresholding effectively compresses data using wavelet tree encoding. Donoho and Johnstone (1994) presented the Visushrink wavelet thresholding technique where coefficients in the wavelet domain are kept, shrunk, or set to zero if they are less in magnitude than a global threshold. Visushrink shrinks coefficients by decreasing the magnitude of coefficients by the universal threshold. The global thresholding value, known as the universal threshold, λ is calculated as $\lambda = \sigma \cdot \sqrt{2 \cdot log(n)}$. The parameter σ is often chosen from the variation in the highest resolution level where $j = J - 1$. The wavelet coefficients in this level are often assumed to be composed of mostly noise.

Visushrink is an accurate thresholding method when the error terms associated with the signal are normally and independently distributed with mean zero and constant variance. Since the DWT is an orthogonal transformation, the error distribution associated with the wavelet coefficients remain normal for a signal of interest. Antoniadis and Fryzlewicz (2006) show that the DWT is inherently robust to slight deviations in the error distribution and independence due to the multiple resolutions and asymptotic normality derivations.

Noisy and irregular signals are common in many industrial processes and other processes requiring statistical analysis, thus the need for thresholding methods with fewer restrictions. McGinnity et al. (2017) presents a resolution-dependent, cross validated block thresholding method which relaxes many assumptions. Cross validation examined by Nason (1996), uses the reconstruction of even and odd data to determine thresholding. This removes the normality requirement since a parametric statistic, such as standard deviation, is not used globally for threshold determination. In this paper we use the cross validated, resolution-dependent threshold methods as presented in McGinnity et al. (2017), but we assign blocks of wavelet coefficients to

hidden Markov states to improve signal approximation.

4.2.4 Hidden Markov Models in the Wavelet Domain

The wavelet transform, using the DWT, creates multiple levels, j, of descriptive wavelet coefficients. Each coefficient and its associate wavelet basis function represent features in the function, f, such as a spike, bump, jump and many typically represent noise (Donoho and Johnstone (1994)). Understanding whether a wavelet coefficient represents an import feature of the signal or whether it represents noise is a problem that can be solved using Hidden Markov Models (HMM).

Using HMM means wavelet coefficients are viewed as stochastic process. Stochastic processes are mathematical models, based on observable outcomes, which are defined in a probabilistic manner (Rabiner and Juang (1986)). A Markov model is a stochastic process with the memoryless property where the state of the system at time t, S_t, is solely determined by the previous state, S_{t-1}. The memoryless property makes modeling stochastic processes more tractable, and many stochastic processes occur as Markovian processes. HMMs model doubly stochastic processes, one hidden and one observable stochastic process. Rabiner and Juang (1986) presents a coin flip example, shown in figure 26, which can be modeled by HMMs. The observable outcome is a heads or a tail, and the hidden process is whether the coin flipped was biased towards head or biased towards tails. In this research, wavelet coefficients are modeled by HMM in the same way. The observed process is the magnitude and pattern of the wavelet coefficient blocks. The hidden process is the distribution of the wavelet coefficient blocks, whether they represent noise or signal.

The expectation-maximization (EM) algorithm or Baum-Welch algorithm, can be used to predict the hidden state of HMMs based on the observable states using maximum likelihood. In this research, the EM algorithm maps blocks of wavelet

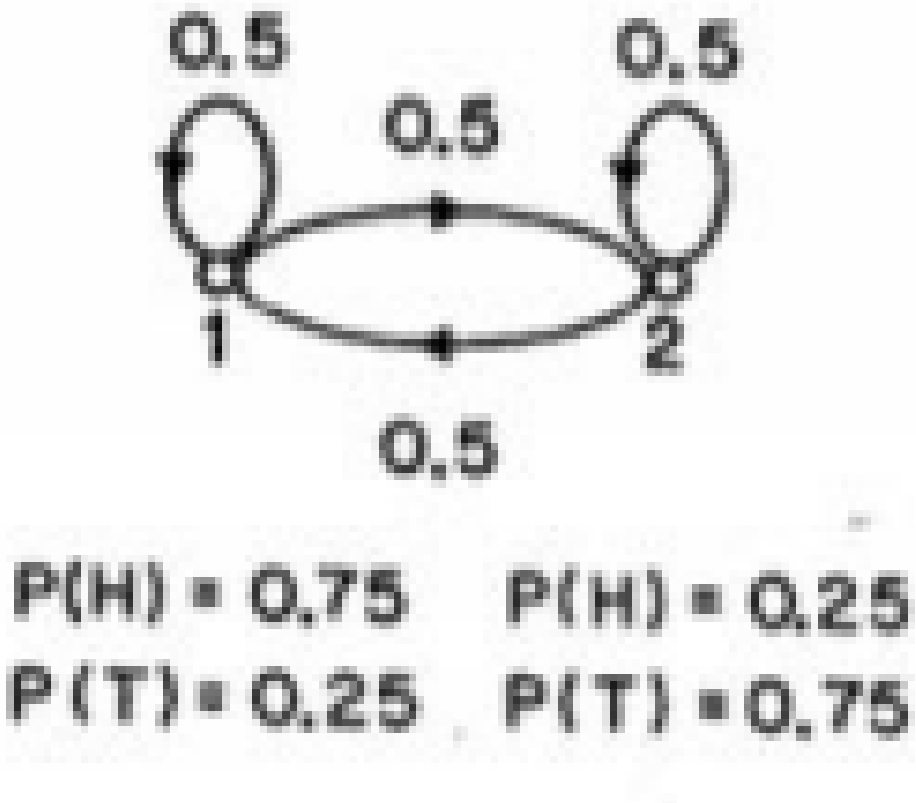

Figure 26. The two biased coin example from Rabiner and Juang (1986). The Markov chain models the hidden stochastic process for selecting coin one or two to flip, determined by the flip of an unbiased coin. The observed state is the result of the flip of coin one or two, unknown, where $P(H \mid Biased(H)) = 0.75$ and $P(T \mid Biased(T)) = 0.75$. The EM algorithm is used to determine which state, 1 or 2, the hidden process occured at based on the "H" or "T" outcome.

coefficients to hidden states, that represent noise and signal. This information permits a more targeted thresholding approach such that resolution dependent, cross validated block thresholding can effectively remove noise from wavelet coefficients. The EM algorithim is described in more detail in Bilmes (1998).

The HMCV presented outperforms all other thresholding techniques for non-Gaussian signal approximation and is competitive with state of the art methods in the Gaussian case.

4.3 Methodology

In wavelet analysis a signal is transformed by a family of orthogonal wavelet functions into wavelet coefficients. Many of these coefficients model noise while, in practice, few of the wavelet coefficients represent signal. However, since the DWT process does not explicitly label each wavelet coefficient as representing noise or signal, extensive research in wavelet analysis has focused on determining appropriate thresholding techniques to remove noise from wavelet coefficients. In this research, we assume an imaginary doubly stochastic process occurs simultaneously with the DWT that, at the hidden level, labels the wavelet coefficients as signal or noise and outputs observed blocks of wavelet coefficient. This process is modeled as an HMM and the EM algorithm is used to determine the hidden state from the observed blocks of wavelet coefficients. With this information, classic thresholding methods, such as the Nason (1996) even and odd (E-O) cross validation method, may be used to threshold blocks of wavelet coefficients based on their hidden states. The specific details of the HMCV block thresholding method are outlined below.

1. Split a signal, y, into two vectors by their index value. A vector containing the even data, y^e and a vector containing the odd data y^o.

2. Take the DWT of the signal y^e and the signal y^o; This work uses Debauchies least asymmetric (LA-8) wavelet.

3. For the detailed resolution levels $j = 1, .., J-1$, group wavelets into blocks according to the method of Cai (1999) of size $b_j = log(n_j)$, n_j is the number of coefficients in resolution level j.

4. Use the EM Baum-Welsh algorithm to fit the blocks of wavelet coefficients to an n hidden Markov State. In this paper, $n = 2$, is used with the hmmlearn package, found at `https://github.com/hmmlearn/hmmlearn`. The exact meaning for the two states are unknown, but similar blocks of wavelet coefficients are grouped together in the same hidden state. The optimal definition is that blocks of wavelet coefficients containing mostly noise are grouped together in one state and blocks of wavelet coefficients containing mostly signal are grouped together in the other state.

5. Use Nason (1996) E-O cross validation for each resolution level to optimize equation (26). The $\hat{f}^e_\lambda$ represents the reconstruction of the even data from the odd data and y^e are the original even data. $\hat{\lambda}$ is the estimated threshold value to threshold blocks of wavelet coefficients; $\hat{lambda}$ is determined by varying λ such that the norm of the E-O cross validation is minimized.

   Each block of wavelet coefficients have been mapped to a specific HMM state in the previous step. Traditionally λ is varied by removing blocks that increase the response in equation (26). In this work, rather than removing a block of coefficients at a time and determining how that removal impacted the accuracy of the reconstruction, we remove all blocks assigned to specific states assigned by the EM algorithm which increase $\hat{\lambda}$ equation (26).

$$\hat{\lambda} = argmin_{\lambda}\left\{\{||\hat{f}^{o}_{\lambda} - y^{e}|| + ||\hat{f}^{e}_{\lambda} - y^{o}||\}\right\} \tag{26}$$

6. IDWT of the thresholded coefficients. Note the HMM state associations no longer have meaning post thresholding.

4.4 Comparison of Threshold Methods for wavelet test cases

The HMCV method is compared to several other methods in a replication of a study presented in McGinnity et al. (2017) which adds different noise configurations to known wavelet test cases from Donoho and Johnstone (1994, 1995) to measure the ability of wavelet transforms, using the given thresholding methods, to remove noise and approximate the signal of interest. McGinnity et al. (2017) present level dependent thresholding with cross validation (LD) with a cross validation method from Barber and Nason (2004), shortened to (Nason), a complex multi-wavelet style shrinkage method (CMWS) from Downie and Silverman (1996), an unbalanced Haar method (UH) from Fryzlewicz (2007), and an empirical Bayes method from Johnstone and Silverman (2005a). Additionally, we conduct the simulation for a cross validation method (M*) available in R from Kim and Oh (2006) and include the results in the analysis. These thresholding methods are applied to the test cases in Figure 27 of length 2^9, 2^{10}, and 2^{11} for T3, lognormal and normal noise distributions with signal to noise ratios (SNR) of three.

McGinnity et al. (2017) provide a complete description of the standardization steps for the different methods used in comparison with the LD method. The inclusion of the HMCV method in this study is used through a simulation, implemented in Python, according to the standardization methods in McGinnity et al. (2017). This study uses the Debauchies least assymetric - 8 (LA-8) wavelet basis in the pywt.py, wavedec.lib, and waverec.lib libraries (Debauchies, 1992).

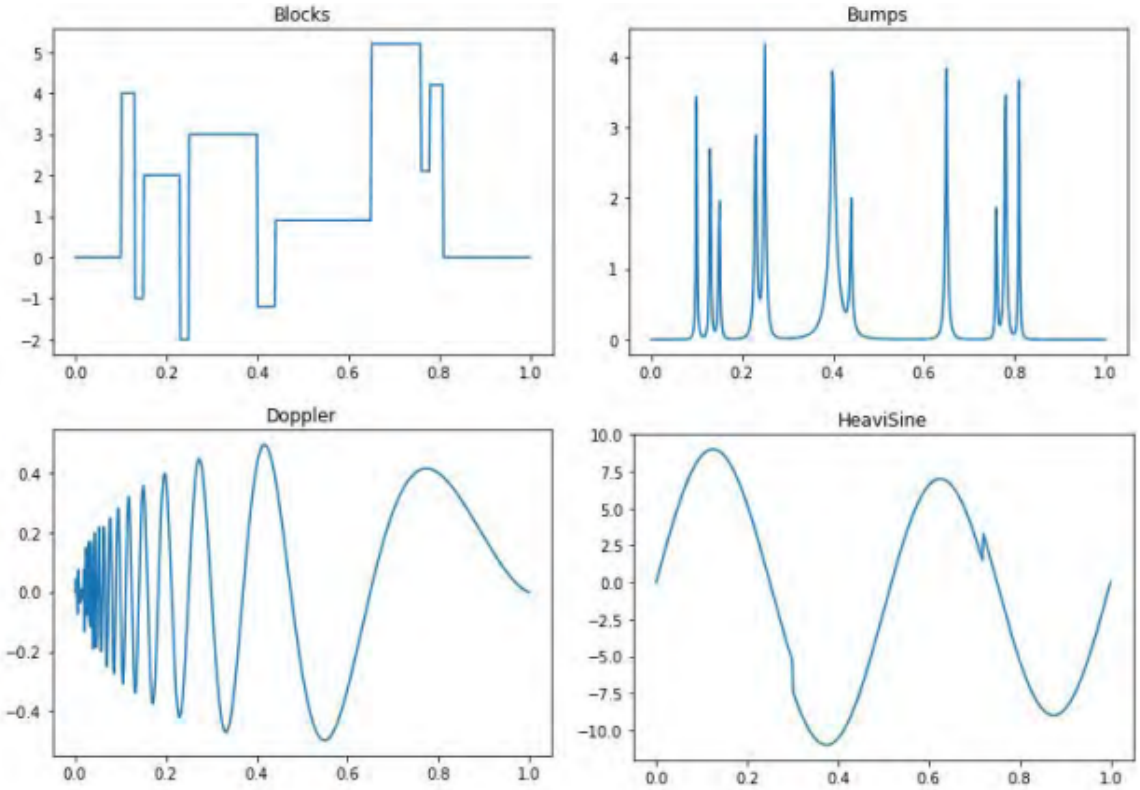

Figure 27. The four wavelet test cases in this paper with 512 samples (Donoho and Johnstone, 1994, 1995). The first graph is the blocks signal, the second graph is the bumps signal, the third signal is the doppler signal, and the fourth is the heavisine function

The first comparisons were conducted for lognormal error distributions for the different wavelet test cases from Donoho and Johnstone (1994, 1995). This error distribution tests the ability of the non-parametric HMM wavelet threshold method to handle non-Gaussian error. Table 5 shows the performance of the wavelet reconstruction for various thresholding techniques as a ratio of the MSE for signal approximation with the MSE of VisuShrink. The methods are considered a win, a tie, or a loss depending on a paired t test for the MSE ratios for a given test case, $\alpha = 0.05$. The paired t test is relative to the HMCV method and the thresholding method, of those considered, with the lowest MSE ratio. For the lognormal case with a SNR of 3, the HMCV method has 9 wins, 2 ties, and 1 loss. In the case of the nine wins, the next best method has an MSE ratio ranging from 68% to over 300% larger than the HMCV method. For the one loss the HMM threshold method is the second best method, and is only 28% larger than the M* method, in terms of MSE ratios. The nine wins are evenly distributed across the different wavelet test cases. This suggests that the HMCV method is a strong method since it outperforms the other thresholding methods for a non-Gaussian error distribution, lognormal.

Table 6 shows that the HMCV thresholding method performs well for T3 error from McGinnity et al. (2017). For T3 error the HMCV threshold method has 7 wins, 4 ties and 1 loss. In the case of the loss the HMCV threshold method is the third best performing method in terms of MSE ratio and is 41% larger than the best method's MSE ratio. Also, in many of the cases where the HMCV method wins, it significantly outperforms the other thresholding methods. In the case of the bumps and blocks functions the MSE ratio of the next best method is 200% to 400% larger than that of the HMCV method.

In the case of normal error, the HMCV method is also competitive. Table 7 shows that the HMCV method outperforms the other methods in some of the twelve

Table 5. Lognormal noise with SNR = 3: Ratio of average MSE for listed method compared to the average MSE for the VisuShrink method with the associated standard deviation

		Wavelet Test Cases (LogNormal error SNR 3)					
Function	n	HMCV	M*	LD	Nason	CMWS	UH
Blocks	512	**0.53**(0.23)	0.78(19.29)	0.76(19.78)	0.90(36.89)	0.98(37.80)	1.19(38.85)
	1024	**0.14** (0.08)	0.39(3.92)	0.44(13.64)	0.92(35.87)	1.04(36.24)	1.25(36.77)
	2048	**0.10** (0.07)	0.28(2.98)	0.32(8.18)	0.93(24.69)	1.09(25.39)	1.32(26.57)
Bumps	512	**0.30**(0.04)	1.17(3.17)	1.44(4.22)	0.83(3.59)	1.03(3.75)	1.13(3.79)
	1024	**0.23** (0.01)	0.76(2.64)	1.21(4.72)	0.91(3.64)	1.14(3.63)	1.25(3.72)
	2048	**0.25**(0.02)	0.43(0.76)	0.67(2.11)	0.93(2.47)	1.20(2.52)	1.34(2.65)
Doppler	512	**0.29**(0.029)	0.54(0.20)	0.53(0.26)	0.90(0.70)	1.27(0.73)	1.25(0.75)
	1024	0.32(0.06)	**0.22**(0.16)	0.40(0.27)	0.94(0.68)	1.26(0.69)	1.32(0.70)
	2048	0.23(0.04)	**0.20**(0.06)	0.23(0.15)	0.94(0.47)	1.26(0.48)	1.37(0.51)
HeaviSine	512	**0.09**(0.11)	0.12(9.90)	0.17(19.91)	0.97(74.50)	1.28(76.21)	1.33(79.83)
	1024	**0.027**(0.03)	0.11(7.23)	0.16(26.95)	0.97(72.00)	1.24(72.32)	1.35(74.13)
	2048	**0.08**(0.04)	0.11(4.79)	0.15(16.26)	0.96(49.30)	1.22(50.74)	1.39(53.60)

Bold font represents lowest value on the row

Table 6. T3 noise with SNR = 3: Ratio of average MSE for listed method compared to the average MSE for the VISUshrink method with the associated standard deviation

		Wavelet Test Cases (T-3 error SNR 3)					
Function	n	HMCV	M*	LD	Nason	CMWS	UH
Blocks	512	**0.53** (0.23)	0.98(18.83)	0.88(19.81)	0.81(39.75)	0.66(41.32)	1.12(42.93)
	1024	**0.32**(0.16)	0.62(4.28)	0.63(7.50)	0.84(27.36)	0.76(28.71)	1.24(31.25)
	2048	**0.26**(0.09)	0.44(2.55)	0.49(15.57)	0.85(27.04)	0.82(27.80)	1.32(29.10)
Bumps	512	**0.29**(0.01)	1.31(3.25)	1.58(4.55)	0.68(3.92)	0.94(4.05)	1.01(4.25)
	1024	**0.22**(0.01)	1.05(1.99)	1.61(4.26)	0.77(2.74)	1.13(2.91)	1.21(3.08)
	2048	**0.33**(0.02)	0.65(0.73)	0.95(2.58)	0.84(2.73)	1.20(2.76)	1.33(2.89)
Doppler	512	**0.40**(0.20)	0.74(0.23)	0.70(0.24)	0.78(0.77)	1.41(0.80)	1.23(0.82)
	1024	0.42(0.10)	**0.38**(0.14)	0.59(0.15)	0.84(0.52)	1.43(0.54)	1.34(0.60
	2048	0.48(0.16)	**0.34**(0.05)	0.39(0.30)	0.88(0.52)	1.35(0.52)	1.42(0.55)
HeaviSine	512	0.53(0.38)	**0.20**(7.14)	0.25(15.53)	0.93(80.76)	1.37(84.62)	1.35(89.06)
	1024	0.23 (0.29)	**0.21**(4.70)	0.25(14.34)	0.92(54.36)	1.29(56.92)	1.40(63.30)
	2048	0.53(0.20)	**0.20**(4.62)	0.26(31.10)	0.92(54.45)	1.18(55.46)	1.44(58.46)

Bold font represents lowest value on the row

Table 7. Normal noise with SNR = 3: Ratio of average MSE for listed method compared to the average MSE for the VisuShrink method with the associated standard deviation

		Wavelet Test Cases (Normal error SNR 3)					
Function	n	HMCV	M*	LD	Nason	CMWS	UH
Blocks	512	0.83 (0.17)	0.89(0.28)	1.44(18.82)	0.67(4.00)	0.96(5.86)	**0.22**(5.56)
	1024	0.66(0.21)	0.62(0.26)	1.15(2.22)	0.99(3.47)	0.75(2.34)	**0.17**(2.71)
	2048	0.49 (0.09)	0.47(0.19)	1.02(1.50)	0.99(1.60)	0.73(1.42)	**0.12**(1.52)
Bumps	512	**0.29**(0.01)	0.18(0.18)	1.54(2.79)	1.48(2.30)	0.51(0.54)	0.96(1.05)
	1024	0.22 (0.01)	**0.16**(0.03)	1.43(1.62)	1.43(1.55)	0.52(0.34)	1.09(0.57)
	2048	0.34(0.02)	**0.28(0.06)**	1.28(0.55)	0.94(0.31)	0.58(0.15)	1.32(0.26)
Doppler	512	0.51(0.43)	**0.48**(0.09)	1.50(0.14)	0.99(0.08)	0.55(0.07)	2.03(0.14)
	1024	0.46(0.11)	**0.29**(0.11)	0.83(0.09)	0.97(0.06)	0.61(0.04)	2.09(0.08)
	2048	0.39(0.08)	**0.25**(0.11)	1.12(0.03)	1.00(0.03)	0.70(0.02)	2.08(0.04)
HeaviSine	512	1.18(1.00)	0.97(1.27)	0.93(5.31)	1.07(7.25)	**0.90**(5.20)	2.48(9.00)
	1024	**0.45**(0.94)	0.87(1.14)	0.96(3.32)	1.06(4.75)	0.88(3.29)	1.95(4.91)
	2048	**0.39** (0.07)	0.74(0.90)	0.97(2.43)	1.03(2.96)	0.87 (2.28)	1.43(3.35)

Bold font represents lowest value on the row

different signal approximation configurations. The HMCV method has 3 wins, 3 ties, and 6 losses. But, there is no clear winning method for the normal error simulation. For the blocks signal reconstruction configurations, the HMCV method is beat by the UH method in each case, however, in two of the three cases the HMCV method is the second best method. The UH method outperforms the HMCV method by 400%, but the HMCV method outperforms the UH method by 400% in eight of the other nine configurations and 200% in one of the nine configurations. The M* method outperforms the HMCV method for two normal signal configurations by a very narrow margin, but the M* method has a nearly 100% larger MSE ratio for two of the heavisine configurations.

The HMCV thresholding method is a competitive method for the graphic approximation of noisy signals. In one of the few areas where the HMCV method performs worse or as well than other thresholding methods in terms of MSE, the Doppler test case with 1024 and 2048 data points, we show that it is a comparable method graph-

ically. Figure 28 shows the wavelet reconstruction for the HMCV, and M* method for the Doppler function with additive normal noise, N(0,1), zooming into the most detailed portions of the Doppler function, where signal approximation is challenging. The HMCV method generally outperforms the M* method in terms of MSE for the most detailed section of the signal as highlighted graphically in Figure 29.

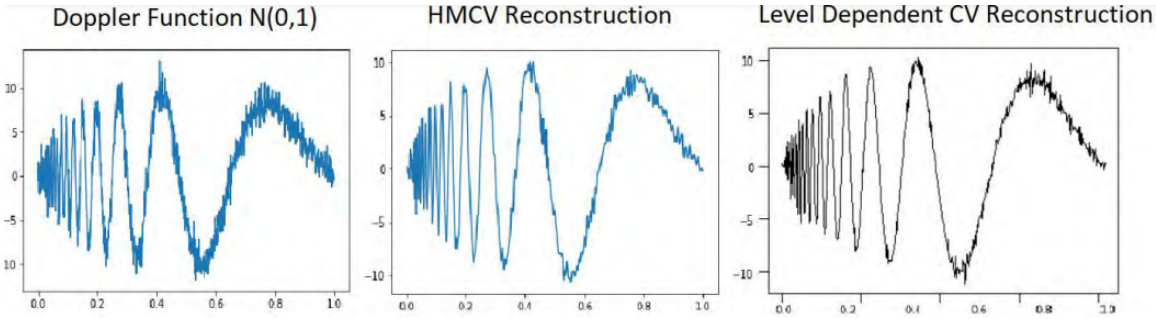

Figure 28. Doppler Function with additive Gaussian noise with $\sigma = 1$. The graphs represent the original data (left), the HMCV reconstruction (middle) and the M* reconstruction (right)

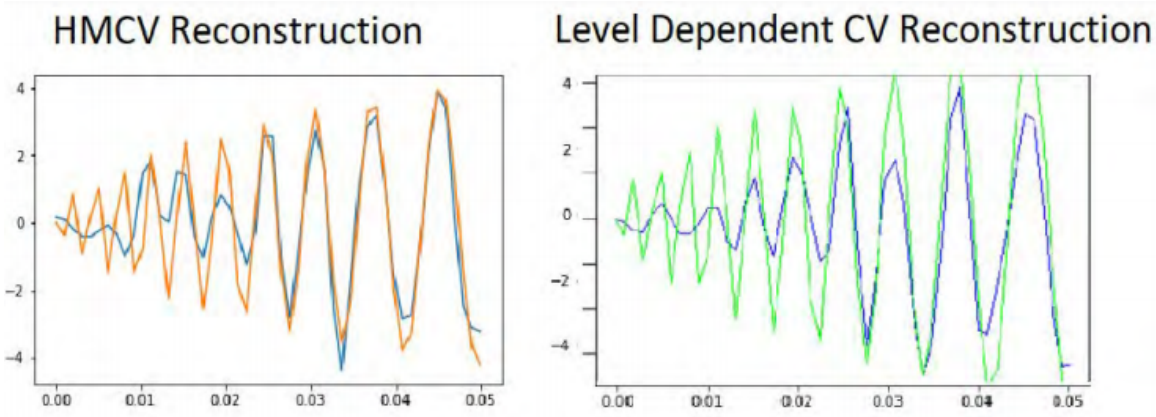

Figure 29. Doppler Function with Zoom for the wavelet approximation of the Doppler function with additive noise N(0,1). The resultant wavelet approximation using M* (MSE = 1.22) and HMCV (MSE = .92) threshold techniques are compared.

An area that the HMCV threshold method outperforms the other methods is the lognormal block function. The wavelet approximation of the test case in graphical form is in Figure 30. HMCV removes some of the large spikes created by the lognormal noise and presents a smoother reconstruction in general than the M* method.

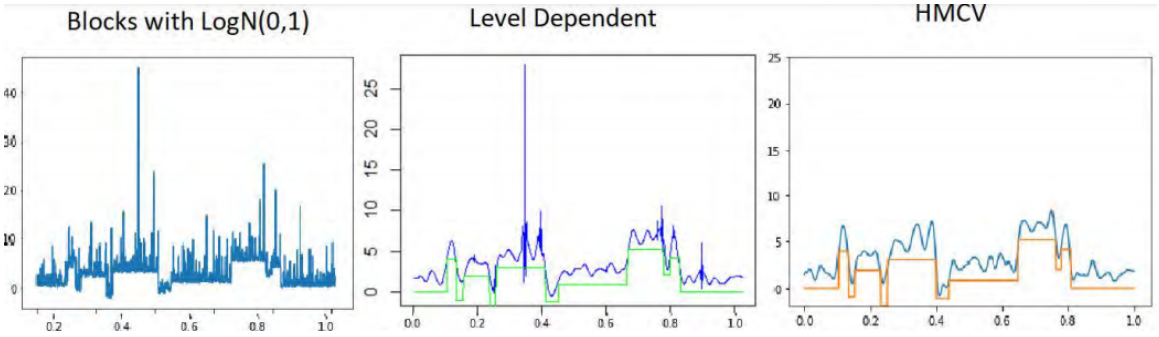

Figure 30. Comparison of wavelet reconstruction using M* (MSE=4.31) and HMCV (MSE=3.67) for the Blocks test case with additive lognormal noise. Noise standard deviation is 1 for the signal of length 1024.

4.5 Model and System Agreement example

This HMCV thresholding method is resistant to many error types. This section demonstrates the need for a non-Gaussian thresholding method for signal reconstruction in the context of statistical process control. Atkinson et al. (2018a) present a simulated system process control problem to demonstrate a unique capability of wavelets to localize regions of agreement and disagreement among a model and system. Following the wavelet analysis of variance (WANOVA) test, analysts can use the top percentile of non-zero wavelet coefficients to reconstruct the location and magnitude of anomalies.

Atkinson et al. (2018a) use the sine function with period four containing additive Gaussian noise with $\sigma = 1$ to represent the simulated system of interest. The system data does not contain biased regions. The hypothetical model in question, which is tested using the WANOVA hypothesis test for accuracy, is data biased with anomalous data over certain intervals. Following the original study, a bias of one over the interval [128,192] is used. This biased data is known as the model in this simulated system. The model and system data are shown in Figure 31.

In the case of Gaussian error, the localization of differences between the model and signal is effective using the method of Atkinson et al. (2018a). Figure 32 shows the

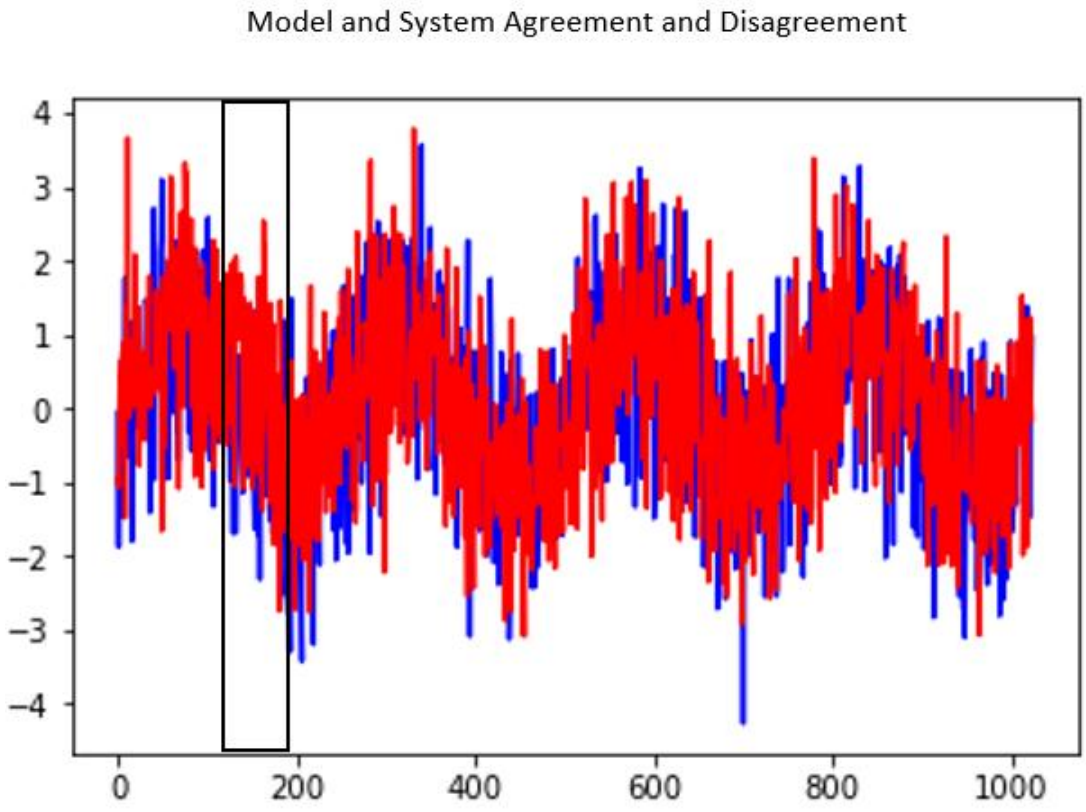

Figure 31. Comparison of a system, blue, and a model, red. Both signals contain Gaussian noise, however, the model data contain additive bias over the segment [128,192]. Noise standard deviation is 1 for the signals of length 1024.

wavelet reconstruction of the difference between the model and signal using only the greatest one percent of wavelet coefficients. Note that thresholding is not required for Gaussian error since we are only interested in the largest wavelet coefficients. The magnitude of the difference is roughly shown in the region of the additive bias [128,192] with a magnitude close to 1.

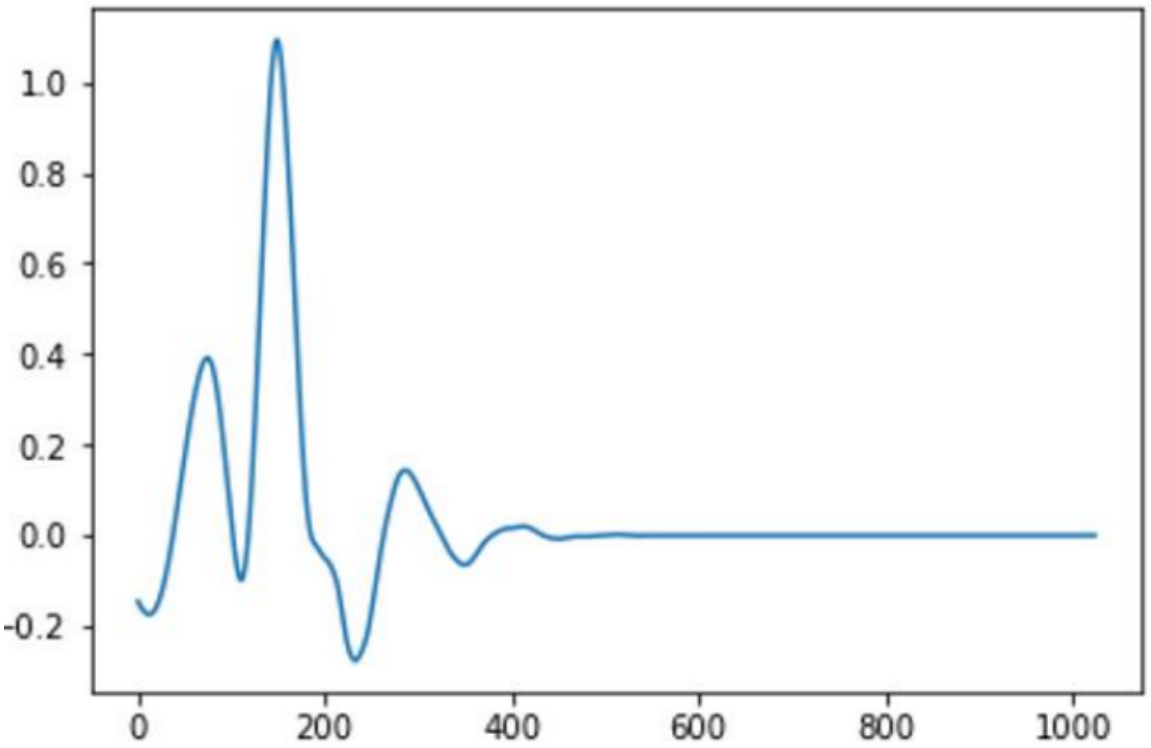

Figure 32. Comparison of a system and a model portrayed through a wavelet reconstruction using the top one percent of wavelet coefficients. The additive bias is shown over the segment [128,192] with magnitude 1.

In the case of non-Gaussian error terms, wavelet coefficient thresholding is required prior to the implementation of Atkinson et al. (2018a) method. The model and signal for lognormal error are shown in figure 33. Figure 34 shows the signal and the effect that thresholding has on the methods from Atkinson et al. (2018a). The small difference between the model and system is more challenging to detect in that it is overshadowed by the noisy signal. The plot of the wavelet reconstruction without thresholding of the top one percent of coefficients does not detect the interval of

additive bias as error nor does it suggest its magnitude, first row of 34.

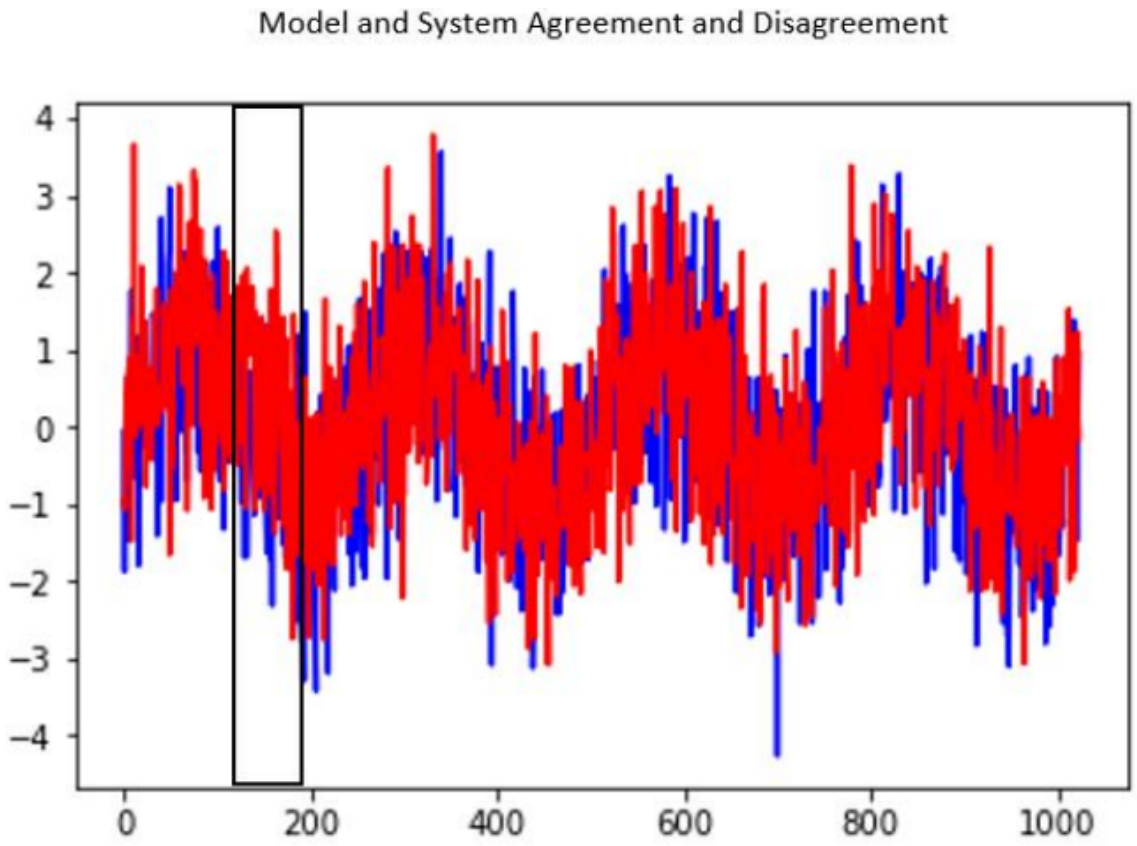

Figure 33. Comparison of a system, blue, and a model, red. Both signals contain lognormal noise, however, the model contains additive bias over the segment [128,192]. Noise standard deviation is 1 for the signals of length 1024

Since the Atkinson et al. (2018a) method does not detect the anomalous system data without thresholding, it is useful to remove the lognormal noise from the signal without removing the bias. The M* method removes some of the lognormal noise but the WANOVA reconstruction does not indicate the location of the error region, seen in the second row of figure 34. This means that for this data, and subsequent simulations, the HMCV threshold is the only method that removes the lognormal error effectively enough that the method from Atkinson et al. (2018a) may be applied, as seen in the third row figure 34.

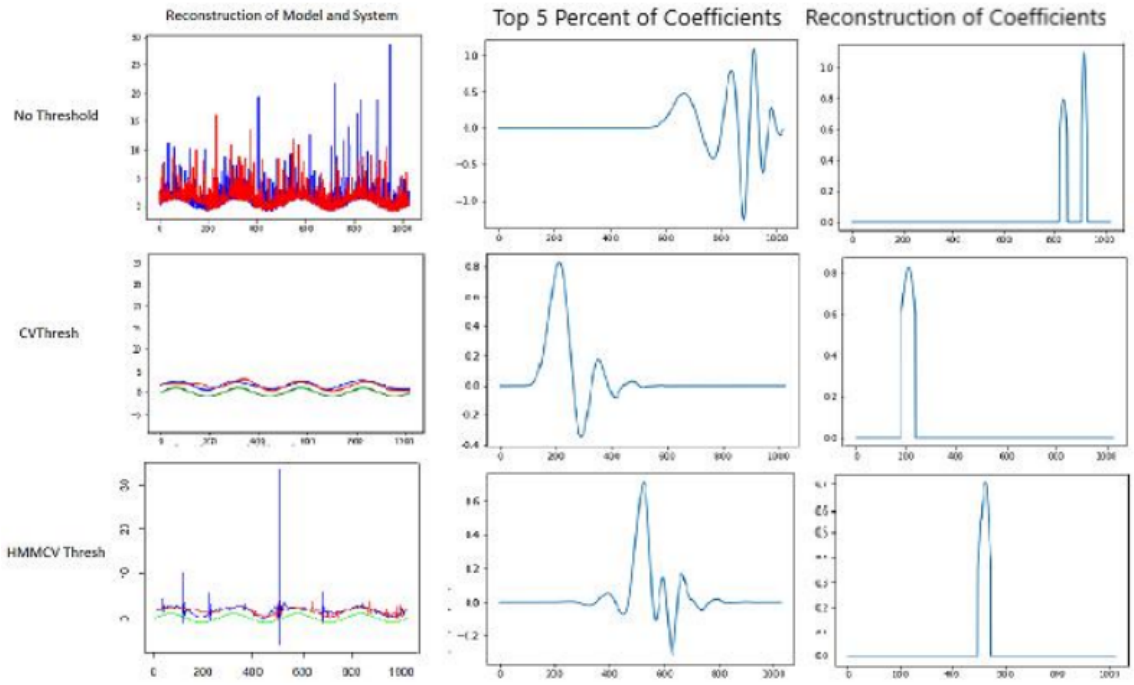

Figure 34. Comparison of a system, blue, and a biased model, red, both containing lognormal noise. The first row shows the percentile method without thresholding, the second row shows the method with M* thresholding and the third row shows the method with HMCV thresholding.

4.6 Real World Data Example

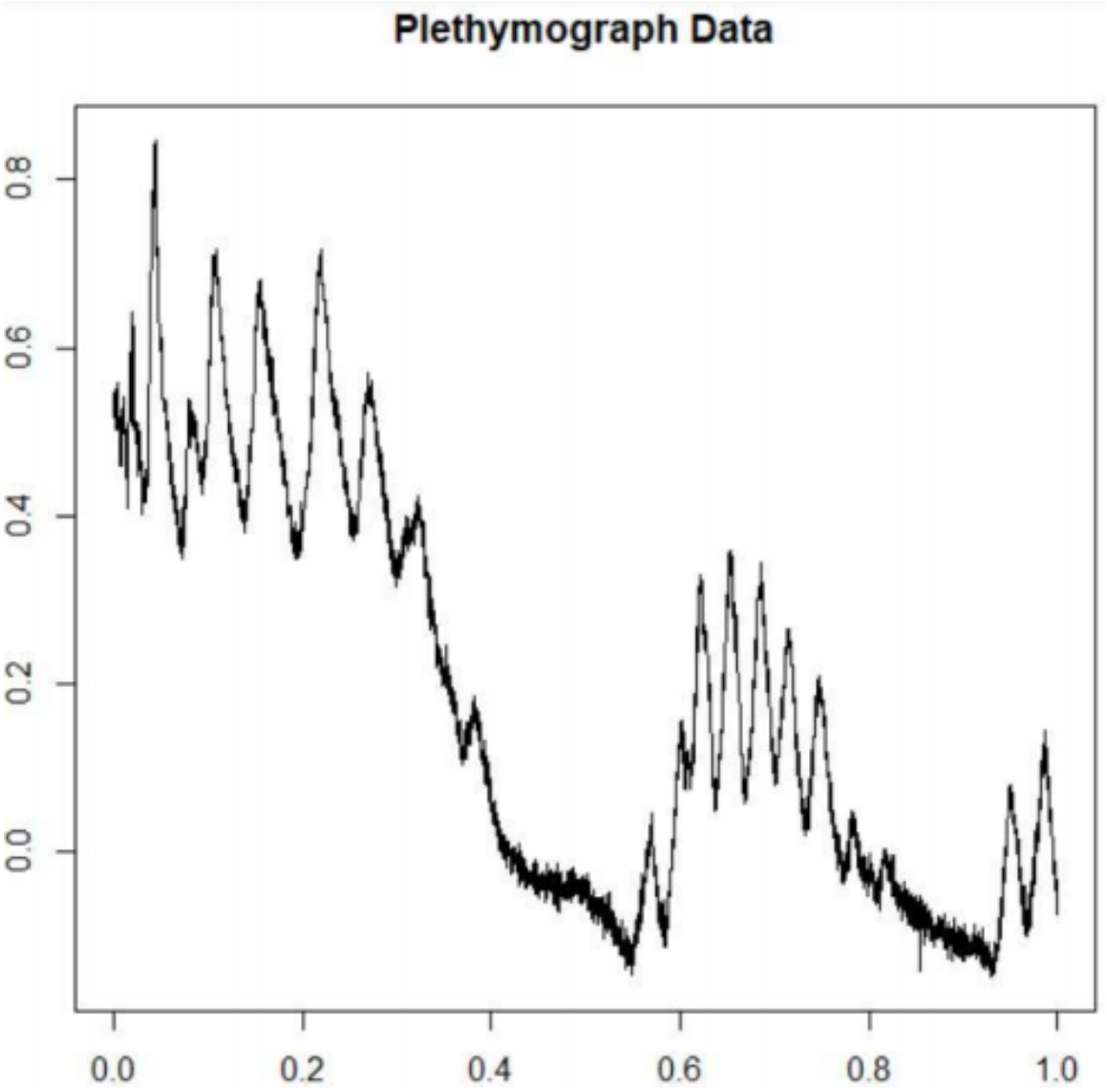

Figure 35. Plethymography data from Nason (1996) which shows the lung volume of a patient experiencing vomiting, 4096 observations. The data is very noisy, using wavelet analysis smooths the signal and makes the disruption event more noticeable

The cvthresh.lib library in R from Kim and Oh (2006) contains a data set called the Respiratory Inductance Plethysmography (IP) which Nason (1996) uses to present a visual reconstruction of a real world noisy signal using cross validation thresholding for noise removal. Plethysmography is a tool used for measuring the capacity of a

patients lungs over some time, t. Figure 35 shows the 4096 samples of the naturally noisy Plethysmography data for one patient over an 80 second period. The peaky oscillations represent normal breathing and the disturbance in oscillations which show low lung capacity occurs during vomiting.

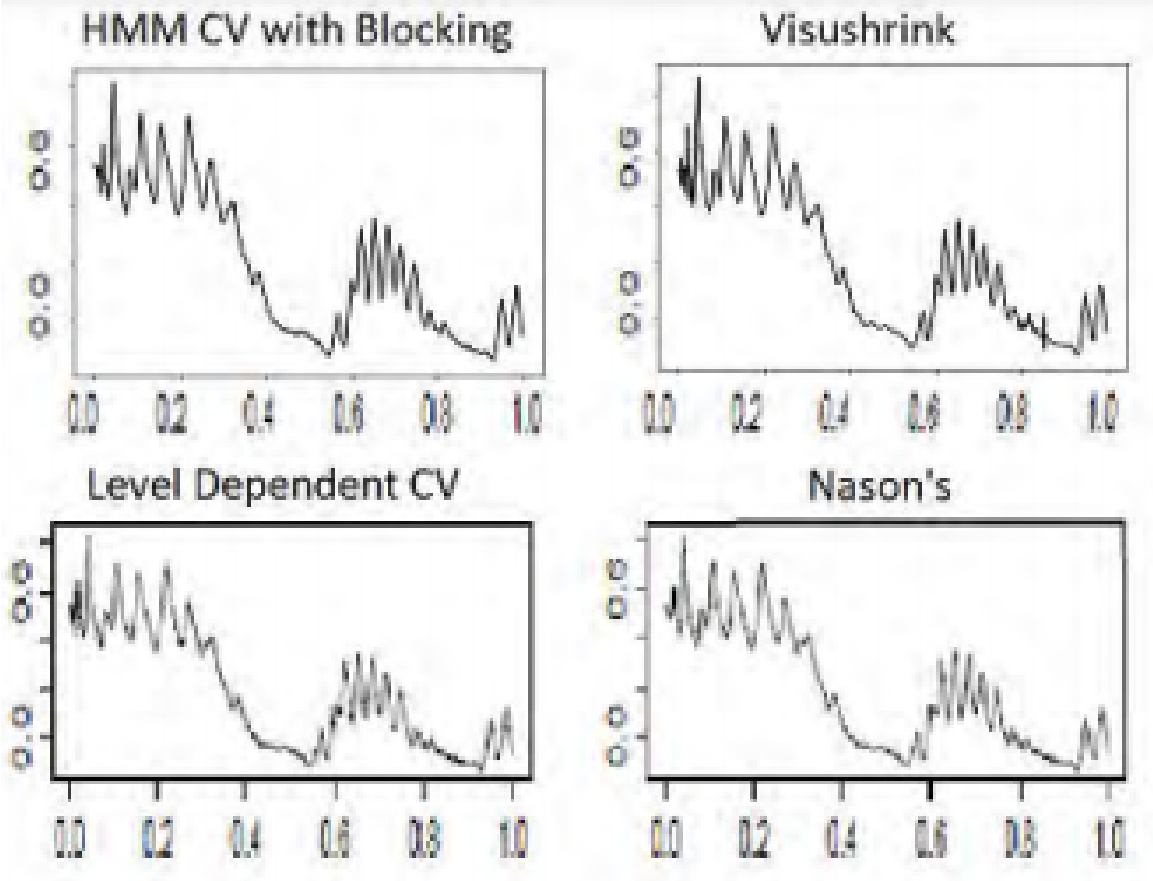

Figure 36. Wavelet reconstruction of the Plethymography graph using four different wavelet thresholding techniques. The HMCV method (top left) and the M* method (bottom left) were applied in this reseach and the VisuShrink and Nason method were included for reference from (McGinnity et al., 2017)

McGinnity et al. (2017) compares the thresholding methods presented in this paper for accuracy in visual reconstruction of this real world data. M* and LD perform as well or better than the other methods presented. Figure 36 shows the different methods to include the M*. The Visushrink method is included as a reference. We rank the reconstructions as $HMCV > CV > LD > Visushrink$ based on the visual

reconstructions of the signal. Since, we do not have a denoised baseline for this data, we do not have a mathematical metric to compare the methods and the resultant ranking is subjective. The Visushrink method leaves some noise in the disturbance period, shown as a small downward blip, which results in the worst reconstruction. The LD method leaves some noisy elements at the beginning of the reconstructed function which results in the third best reconstruction visually. M* is slightly noisier than the HMCV method during the disturbance period which results in the second best reconstruction. The HMCV method is smooth throughout the signal and appears to retain all of the key characteristics of the original signal.

4.7 Image Processing

The HMCV technique effectively thresholds Gaussian and non-Gaussian simulation and real world signal data. In this section, the HMCV method extends easily to two dimensional signal processing. Imagery data considered in this simulation consist of two image processing benchmarks, the Lena gray-scale image, figure 37, and the Barbara gray-scale image, 38.

Figure 37. 512 bit gray-scale Lena Original

Cross validating the 2D image involves a checkerboard approach to create an even

Figure 38. 512 bit gray-scale Barbara Original

and odd data set, as presented in Nason (1996). Additionally, a Roberts' non-linear edge detection filter is used to enhance details in the wavelet coefficient matrices such as edges and lines. This is done using the Roberts' filter from `scikit-image.org`, image processing tools in Python. This mathematical operation improves the efficiency of the algorithms which assign blocks of wavelet coefficients to hidden Markov states. Further, as discovered through testing, the number of states are set to nine. While more parsimonious models often perform well, the additional Markov states help dissect the complexity of the image's pixel distributions represented by the wavelet coefficients. As in the one dimensional technique, the blocks of wavelet coefficients are thresholded if the inclusion of their assigned hidden states worsens the cross validation. This indicates that most of the pixels in a given HMM state are noisy; smoothing these pixels through thresholding improves the reconstruction.

Figure 39 depicts the flow of the the HMCV wavelet denoising technique. The image is divided into even and odd blocks where CA and CD represent the coarse and detailed wavelet coefficients for the even and odd images. The figure shows for each level, j, the Roberts' filter is applied to the detailed wavelet coefficients, CD, and blocks of noisy filtered coefficients are thresholded through the HMCV technique.

Figure 40 shows the HMCV thresholding technique for a two level and two dimensional wavelet transform for a noisy Lena image. The Roberts' filter detects the edges in the noisy horizontal, vertical and diagonal wavelet components of the images at the two levels, $j = 1, 2$. HMCV thresholding smooths similar blocks of noisy filtered detailed wavelet coefficients simultaneous which increases the accuracy of noise removal. The reconstruction of the original wavelet coefficients with HMCV thresholding results in a clearer image with distinct edges, lines, and texture.

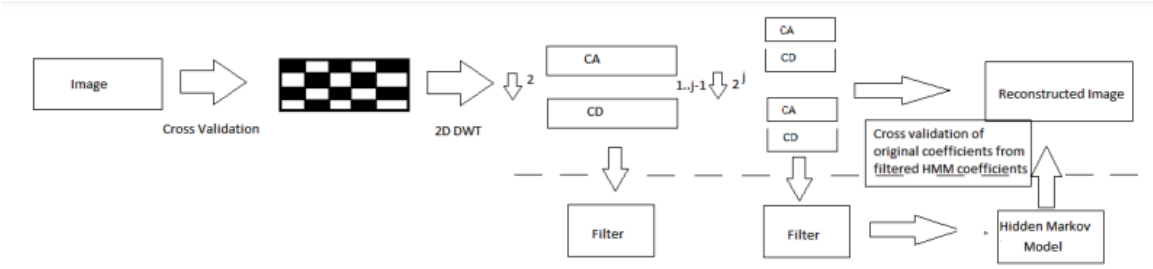

Figure 39. HMCV two dimensional thresholding flow chart. An image is partitioned into even and odd components and the 2D DWT is performed on the even and odd components. An edge filter (Roberts') is applied to the detailed wavelet coefficients and the original even and odd wavelet coefficients are grouped into several states based on the EM algorithm. The original detailed wavelet coefficients are then thresholded based on the cross validated level dependent block thresholding technique for each state. The image is then reconstructed based on the 2D IDWT.

The HMCV method is used to denoise the two images from Chang et al. (2000) where Gaussian noise with mean zero and a standard deviation range of $\sigma = [10, 35]$ was added to each pixel. The performance of the various thresholding methods from the study are reported in terms of mean square error (MSE), where MSE is relative to the differences between the original image with no simulated noise and the reconstructed and denoised image. The different methods are the HMCV 2D application, the SureShrink method presented in Donoho and Johnstone (1995), the BayesShrink method and the BayesShrink plus Compression presented in Chang G.S. and Vattereli (2000), the OracleShrink and OracleThresh values calculated in Chang G.S. and Vattereli (2000) and the Weiner filter for image processing presented in Portilla

Figure 40. Original Lena image Reconstructed using the HMCV technique. The (upper left) is the noisy Lena with Gaussian noise, $\sigma = 0.35$. The second image (upper right) are the coefficients from the 2D wavelet transform. The third image is the HMCV thresholded coefficients (bottom left). The fourth image (bottom right) is the reconstruction of Lena using the HMCV wavelet thresholding with two levels.

et al. (2001).

Table 8. Comparison of the HMCV technique and other thresholding methods for noise removal in the Lena image at varying levels of noise σ. Results reported in terms of MSE between the original image and the thresholded image.

Comparison of Lena Image Denoising

σ	HMCV	OracleShrink	SureShrink	BayesShrink	BayesShrinkC	OracleThresh	Wiener
10	**21.14**	28.31	29.21	29.65	40.72	34.93	28.52
20	**47.66**	59.56	63.96	61.73	81.66	72.57	82.11
30	**84.47**	89.74	94.13	92.06	119.37	110.16	175.36
35	105.47	**104.27**	107.19	106.45	138.62	127.68	236.63

For the denoising of the Lena image the HMCV technique generally outperforms the baseline image thresholding techniques tested in Chang et al. (2000). HMCV outperforms the image thresholding techniques for the first three test settings of $\sigma = 10, 20, 30$, as shown in figure 8. HMCV is narrowly outperformed by the OracleShrink method for $\sigma = 35$ which has a MSE of 104.27. HMCV obtains a MSE of 105.47 which is very near the MSE of the top method and lower than the other five thresholding methods. Figure 41 shows the reconstruction of Lena where $\sigma = 35$ for the HMCV method.

For the denoising of the Barbara image the HMCV technique generally outperforms the baseline image thresholding techniques tested in Chang et al. (2000). HMCV outperforms the image thresholding techniques for the last three test settings of $\sigma = 20, 30, 35$, as shown in 9. HMCV is narrowly outperformed by the OracleShrink method and the BayesShrink method for $\sigma = 35$ which have respective attained MSE of 45.96 and 51.27. HMCV attains a MSE of 54.61 which is very near the MSE of the top method and lower than four other thresholding methods. Figure 42 shows the reconstruction of $\sigma = 10$ for the HMCV method.

Table 9. Comparison of the HMCV technique and other thresholding methods for noise removal in the Barbara image at varying levels of noise σ. Results reported in terms of MSE between the original image and the thresholded image.

Comparison of Barbara Image Denoising

MSE/σ	HMCV	OracleShrink	SureShrink	BayesShrink	BayesShrinkC	OracleThresh	Wiener
10	54.61	**45.96**	56.21	51.27	81.43	58.04	67.84
20	**79.79**	118.11	121.19	121.52	170.2	150.94	136.12
30	**132.57**	190.63	201.09	192.6	250.86	253.1	241.74
35	**144.84**	226.3	246.07	229.66	289.77	300.11	308.51

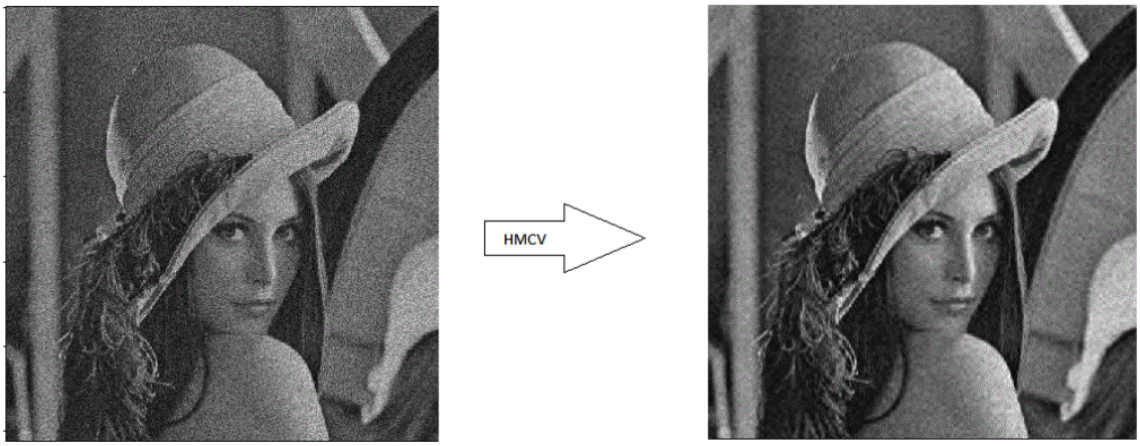

Figure 41. Reconstruction of Lena using HMCV

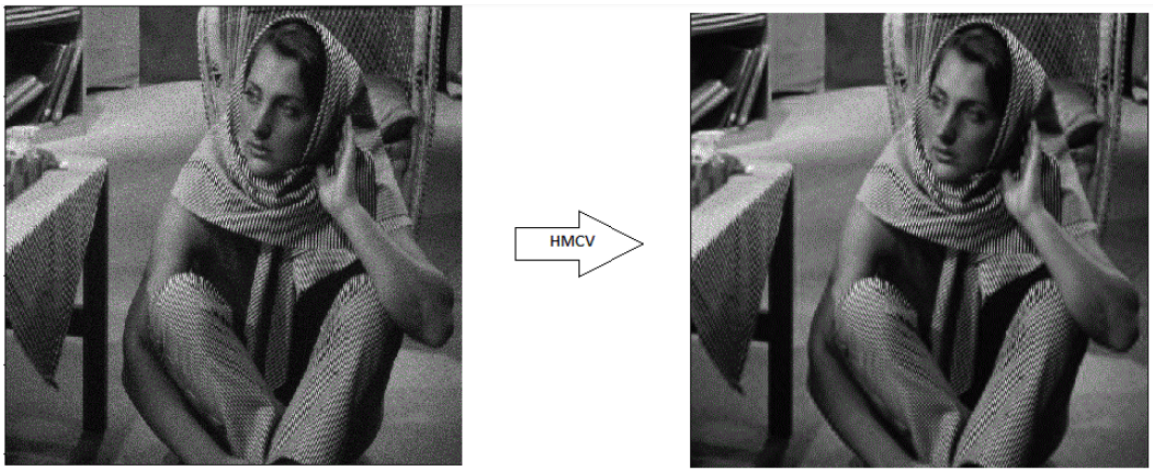

Figure 42. Reconstruction of Barbara using HMCV

Figure 43. Reconstruction of Lena for $\sigma = 10$ from Chang et al. (2000). Figure contains the original image (a) the noisy image (b) the HMCV reconstruction (c) OracleShrink (d) SureShrink (e) Bayeshrink and (f) Bayeshrink with compression.

In addition to a low MSE for reconstructing the two test images across multiple noise levels, figure 43 shows the visual reconstruction using the HMCV technique compared to the other techniques presented in the simulation. The reconstruction produced using the research in this paper is very clear and retains the key facial features as well as the textures included within the test images.

The HMCV method extends well to two dimensions for the image processing examples included in the simulation. The technique produces good visual reconstructions of the original images and attains low mean square error values. The two dimensional application is an added strength to the robust HMCV thresholding method.

4.8 Conclusion

The new HMCV wavelet block thresholding technique is presented. HMCV was developed to improve the accuracy and efficiency of signal processing, statistical process control, and image processing. This method performs well in the Gaussian situation as shown in a case study for several wavelet benchmark functions. Further, the method outperforms state of the art thresholding functions in the non-Gaussian situation for nearly all benchmark functions and configurations. The usefulness of the HMCV threshold method in the non-Gaussian case is demonstrated in a statistical process control application from Atkinson et al. (2018a). Some non-Gaussian noise scenarios bias the results of the wavelet based signal-model agreement method presented in Atkinson et al. (2018a). Using the HMCV thresholding method has the unique ability to remove this bias such that the statistical process control method may be successfully employed. Additionally, the HMCV method compete with state of the art image denoising which further demonstrates the maturity of this 1D thresholding method employed in 2D. Code associated with this research can be found at `https://github.com/JeffreyDWilliams/HMMThresholding`.

V. Graph Wavelet Signal Processing

5.1 Introduction

Graphs are useful for understanding network structure and information flow across both abstract networks like social media and physical structures like road systems. Social media networks such as Twitter and Facebook are intangible, but the networks created by users and their connections are well defined and govern the flow of information. Graphs can represent physical systems such as road networks, which show the structure and areas on which vehicle traffic flow. In both of these instances, graphs allow the analysis of the flow of information throughout the network. "Graphs offer the ability to model such data and complex interactions among them. For example, users on Twitter can be modeled as nodes while their friend connections can be modeled as edges" (Ortega et al., 2018).

A local news tweet disperses to those users that "follow" the news account, these followers can propagate the tweet by "Retweeting" the news bulletin to their followers. Eventually, using connections like virtual highways, tweets can flow through an entire virtual community. A graph analysis shows us which local twitter networks are most likely reached, for example, subscribers to a homeowners association group which happens to follow the local news account.

Analysis of these graphs are conducted with a group of tools called graph signal processing (GSP), which are often signal processing tools elevated to graphs. A good resource for GSP tools can be found in Ortega et al. (2018) which present several tools that exploit the frequency of the graph. One of these GSP tools, which exploits graph frequencies, is wavelet analysis. In signal processing, wavelet analysis has smoothing and sparseness properties making them an attractive tool for signal approximation and denoising. These properties are equally important for wavelet GSP

approximation and denoising. Hammond et al. (2011) bring wavelets to the GSP domain with the spectral graph wavelet transform (SGWT). The SGWT localizes wavelet filters applied to the spectral decompostion of a given graph to smooth the resultant approximation of a noisy graph signal. Alternatively, Irion and Saito (2014) build the Hierarchical Graph Laplacian Eigen Transform (HGLET), which transforms the graph signal through a process called graph partitioning. Coifman and Maggioni (2006) present the diffusion wavelet where the basis is chosen from the graph structure based on the propagation of local and global graphical relationships. Similar to the concept of diffusion wavelets and the best basis selections of Irion and Saito (2014), Defferrard et al. (2016) present a machine learning method to select a best basis using neural networks. The goal of these improvements and variations of the SGWT is to improve graph signal reconstructions for noisy graph signals.

One of the challenges associated with accomplishing this goal for wavelets on graphs is the limited work in graph wavelet thresholding for the various methods of generating wavelets on graphs. Irion and Saito (2014) employ the universal threshold effectively in conjunction with the HGLET partitioning method, proving this wavelet threshold is effective for HGLET graph transformations. de Loynes et al. (2021) present a GSP lifted SUREShrink method, a method originally proposed by Donoho and Johnstone (1995) for traditional signal processing. The GSP SUREShrink method substitutes a term Ξ for the covariance term in equation 30 to compensate for correlated noise in the graph domain. This method is effectively applied to the Minnesota Road Map network, the NYC taxi cab, and a Facebook social network example. This method improves the reconstruction of graph signals and improves the visual reconstruction of the graph signals. de Loynes et al. (2021) also employ a James-Stein (JS) type block thresholding estimator for GSP. This JS block thresholding technique was originally presented by Cai and Silverman (2001), but de Loynes et al. (2021)

note that it is not as effective as term-by-term thresholding such as used in the GSP SUREShrink threshold estimator.

The contribution of this paper is modifying the JS block thresholding technique from Cai and Silverman (2001) and de Loynes et al. (2021) such that block thresholding wavelet coefficients on graph methods are effective for graph signal approximation. Our modification uses the JS blocking method for wavelet coefficents based on spectral orientation rather than spatial orientation. Our method is intuitive in GSP since many GSP techniques are elevated signal processing tools which leverage graph frequencies, or graph spectral information. Additionally, this method follows the observation of Ortega et al. (2018), that spectral ordering appears to be more effective than spatial ordering for graph wavelets. Further, this modification improves upon SUREShrink and graph partitioning methods for several graph signal approximation scenarios presented in this paper.

5.2 Graph Preliminaries

A simple undirected graph consists of a set of vertices, also called nodes, interconnected via edges. In application, graphs are typically weighted such that each edge describes some relationship between the vertices that edges connect. Like graphs the definition of edge weights are versatile and adapted to the particular application. In some applications the edge weights represent distances between nodes or in the case of the work of Pang et al. (2015) the edge weights represent the difference in pixel intensities in a graph representation of an image. A simple undirected graph involves a finite number of vertices, and edges with no self loops. Further, at least one of the vertices does not have a predetermined flow direction. The graph theory applications in this paper are extended to complex and directed graphs, but most of the research presented in this paper focuses on the simple undirected graph. Notationally, graph

$G := \{V, E, W\}$ where V represents the set of vertices, E represents the set of edges and W represents the edge weights. W is defined such that only vertices that are connected have a non-zero weight. The relations between the vertices in a graph are provided in adjacency matrix form to facilitate analysis. The adjacency matrix, A, gets its name from the positive entries for a set, $v_m, v_n \in V$, indicating there is an edge, $e \in E$, between the two vertices and $A_{m,n} > 0$. In this case $a_{m,n} = 1$ or $a_{m,n} = w$ $\forall w \in W$ for a weighted graph. Note in a simple graph, $A_{m,n} = 0$ if $m = n$ since there are no self loops.

$$a_{m,n} : \begin{cases} w & e \in E \\ 0 & \text{otherwise} \end{cases} \tag{27}$$

The graph Laplacian, $\mathcal{L}$, is found by $\mathcal{L} = A - D$ where D is the degree matrix. The degree matrix involves the sum of all adjacent weights for each vertex m such that $D(m) = \sum_{n=1}^{N} a_{m,n}$ where N is the number of nodes in the adjacency matrix, A. D is a diagonal matrix with the degree of each vertex on the diagonal (Hammond et al., 2011). The normalized Laplacian is defined as $\mathcal{L}_N = D^{-1/2}\mathcal{L}D^{-1/2} = I - D^{-1/2}AD^{-1/2}$. This graph Laplacian is used as the matrix representation of the graph in this work.

The Laplacian matrices rely on some form of spectral decomposition or Eigenvalue Decomposition (EVD) of the graph Laplacian. The complete version of the EVD Equation, (28), shows the calculation for the EVD of $\mathcal{L}$.

$$\mathcal{L}X_{\mathcal{L}} = \lambda_{\mathcal{L}} X_{\mathcal{L}} \tag{28}$$

$X_{\mathcal{L}}$ represents eigen vectors and $\lambda_{\mathcal{L}}$ represents the associated eigenvalues where it is assumed that the eigenvalues are ordered such that $0 < \lambda_0 \leq \lambda_1 \leq ... \leq \lambda_{N-1}$.

5.3 Spectral Graph Wavelet Transforms

A well known area of GSP is wavelet analysis on graphs. Hammond et al. (2011) provide a comprehensive guide to developing and using the spectral graph wavelet transforms (SGWT). In this paper, our interest is thresholding the graph wavelet coefficients generated by the SGWT and the properties of the SGWT.

For graphs, a real-valued kernel function g is fixed and analogous to the mother wavelet, $\hat{\psi}^*$, (Hammond et al., 2011). The kernel function g must maintain the classical wavelet properties such that $g(0) = 0$ and $lim_{\lambda\to\infty} g(\lambda) = 0$ (Hammond et al., 2011). The first property is important for this research, since wavelet coefficients generated from g that are thresholded to zero are reconstructed as zero for the graph signal. This makes SGWT coefficient thresholding intuitive using JS type thresholding. The second property is analogous to the vanishing moment requirement of classical wavelets. This means a family of expanded and dilated kernels g can effectively represent a graph signal as SGWT coefficients.

The Meyer wavelet kernel is used in this work due to its tight frame properties. The frame bounds keep the reconstruction values within the bounds of the graph signal. Effective SGWT coefficient thresholding depends on tight frame kernels, g, as discussed in de Loynes et al. (2021). Other kernel functions are also discussed in Hammond et al. (2011). Equation 29 indicates how the SGWT coefficients are derived at the different scales, s, from the kernel g. The left side of equation 29, $W_f(s, n)$, represent the SGWT coefficients at scale s for node n where $f \in \mathbb{R}^N$ is an arbitrary signal sampled on the graph G, and N is the number of nodes in G. The second term in equation 29, is the SGWT operator T_g^s at scale s and the arbitrary graph signal f. The SGWT operator is defined as $T_g^s = g(s\lambda_l)$ where λ_l represent the eigenvalues of the graph laplacian scaled by s. The SGWT operator is defined as the wavelet kernel, analagous to the mother wavelet, as a function of the the graph frequencies at scale

s. The third term in equation 29 represents the modulation of the wavelet operator scaled by s for each mode of the Fourier transform of the arbitrary function f with the associated eigenvectors, $\mathcal{X}_l(n)$, of the graph Laplacian at node n.

$$W_f(s,n) = (T_g^s f)(n) = \sum_{l=0}^{N-1} g(s\lambda)\hat{f}(l)\mathcal{X}_l(n) \tag{29}$$

In this paper we work with the SGWT coefficients $W_f(s,n)$ using JS block thresholding with spectral blocking. The SGWT ceofficients are analagous with classical wavelet coefficients often represented as $\theta_{j,k}$ where j is the scale of the k^{th} wavelet coefficient. The packages provided by Hammond et al. (2011) in Python, PYGSP, and de Loynes et al. (2021) in R, gasper, consolidate these equations creating a graph wavelet transform tool. This work uses the Meyer wavelet in Python for the tight frame properties useful in signal reconstruction.

5.4 Graph Wavelet Thresholding

Graphs and graph data clarify the behavior of networks and physical systems. Analytical insights are improved when noisy graph signals are properly denoised. There are a few instances of thresholding wavelets in GSP research. Irion and Saito (2014) add a soft thresholding scheme to both of their graph partitioning methods, the hierarchical graph laplacian Eigentransform (HGLET) and the generalized Haar Walsh transform (GHWT). The universal threshold, $\lambda_{universal} = \sigma\sqrt{2log(n)}$, where n is the length of the signal and σ is the standard deviation of the graph wavelet coefficients, removes excess noisy GHWT and HGLET coefficients to improve the GHWT and HGLET approximations, respectively, of a graph signal. In the Minnesota road network simulation from the work of Irion and Saito (2014), only 11 percent of GHWT transformed best-basis coefficients were retained and 48 percent of HGLET coefficients were retained, yet the approximation of the graph signal improved beyond

the full set of coefficients.

de Loynes et al. (2021) adapt the JS SURE metric and JS block metrics to threshold SGWT coefficients and provide a justification for the use of the metric in the graph wavelet transform domain. They specifically include the work of Donoho and Johnstone (1995) and Cai (1999) among others as JS estimators which optimize MSE risk for graph wavelet estimators. They suggest additional terms in the estimators to include correlated noise in the graph wavelet domain. Since, in some cases, the SGWT are not completely orthogonall,equation (30) includes the covariance of the correlated noise in the divergence term for the JS type estimators with h is a weakly differentiable thresholding process for denoising the graph signal, $\tilde{F} = F + \Xi$. $Sure(h)$ are the thresholding values for the levels j determined by the thresholding process h.

$$SURE(h) = -n\sigma^2 + \|h(\tilde{F}) - \tilde{F})\|^2 + 2\sum_{i,j=1}^{n(J+1)} Cov(\Xi_i, \Xi_j)\partial_j h_i(\tilde{F}) \tag{30}$$

The SGWT SUREShrink threshold estimator is term-by-term and level dependent. Term-by-term means each SGWT coefficient is thresholded separately without considering other SGWT coefficients generated according to equation (29). Level dependent implies that a SURE threshold value is determined for each graph wavelet filter level, j, such that the variance term is only impacted by the SGWT level of interest. This improves the SGWT graph signal approximation. The SUREShrink method is shown effective for multiple applications in de Loynes et al. (2021) and outperforms methods which do not use graph wavelet thresholding techniques, such as graph trend filtering.

The de Loynes et al. (2021) approach lifts JS estimators to include block estimators to graph wavelets for the estimator of interest in this paper. The JS type estimator, neighcoeff, from Cai and Silverman (2001) leverages information from immediate neighbors for a particular wavelet coefficient to estimate the threshold value for that

particular coefficient. Due to the additional thresholding information, neighcoeff decreases the MSE risk in signal processing estimation. neighcoeff can be thought of as a moving window where, B_i^s, moves one coefficient for each new window, i for each coefficient n at each scale s . In equation (31), the window around each coefficient extends to include the neighboring $L = 3$ coefficients and $\lambda = \frac{2}{3}log(N)$. The σ^2 is the variance of the graph wavelet coefficients while $S_{s,i}^2 = \sum_{(s,n)\in B_i^s} \tilde{W}_{f_{s,n}^2}$, where $S_{s,i}^2$ is the norm of the SGWT coefficient block. The i represents the i^{th} block of SGWT coefficients centered at coefficient n with window size $L = 3$ at scale s. Note from equation 31 that the SGWT coefficient threshold value generated for block i is only used to threshold the SGWT coefficient at n, the center of the window, at scale s. Also note that we do not include a correlation term as de Loynes et al. (2021) prove that SGWT coefficient thresholding is robust to some level of correlation among SGWT coefficients.

$$\hat{W}_f(s,n) = B_i^s \tilde{W}_f(s,n) = (1 - \lambda L\sigma^2/S_{s,i}^2)_+ \tilde{W}_f(s,n) \tag{31}$$

In signal processing block thresholding leverages information from wavelet coefficients which are spatial neighbors. This is effective as wavelet coefficients have persistence and depth (Crouse et al., 1998). Therefore, spatial neighbors contain relevant and perhaps additional information to improve threshold criteria estimation since information from wavelet transforms persist through nearby wavelet coefficients. However, in graphs, spatial neighbors may not be similar. In the case of the Toronto road network, available publicly at `www.toronto.ca`, two spatially neighboring graph nodes could have a different status, such as urban and rural areas. Considering dissimilar graph nodes for thresholding decisions may decrease the accuracy of the threshold estimation. Figure 44 shows the difference between spatial and spectral neighbors. The spectral neighbors include similar nodes in terms of graph frequencies based on

the Euclidean distances between road intersections in Toronto.

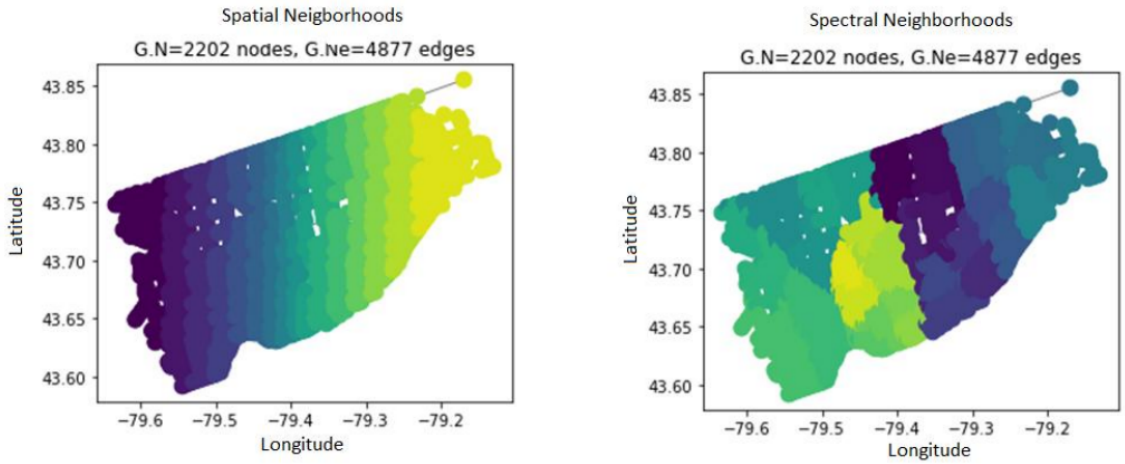

Figure 44. These graphs depict the Toronto road network where each node is an intersection and the edges are the roads that connect the intersections. The graph on the left shows the graph partitioned based solely on location. Notice the different blocks are just slices from South to North. The graph on the right shows the data neighborhoods based on spectral graph partitions. The neighborhoods are of different shapes and sized based on the frequency of each node.

In the context of the Minnesota road network, Irion and Saito (2014) argue that partitioning graphs in a spectral manner is more effective than spatial organizations. Further, de Loynes et al. (2021) find that block thresholding SGWT coefficients is less effective than term by term thresholding for graph signal estimation. In this paper, JS block estimators for SGWT coefficient thresholding are adapted to make wavelet block thresholding useful for graph signal approximation. Spectral neighbors based on graph frequencies are used for block thresholding.

5.5 WAVEBT

WaveBT is a new and effective SGWT coefficient block thresholding method presented in this paper. We use neighcoeff for graphs defined in equation 31 as developed in Cai and Silverman (2001) and brought to graphs by de Loynes et al. (2021). However, we order the graph based on graph frequency and then apply the SGWT and block thresholding such that windows of SGWT coefficients are spectral neighbors

with similar graph frequencies rather than spatial neighbors. This improves the value of information derived from neighboring coefficients for threshold estimation.

The method used to order the graph signals is graph partitioning. An example of graph signal partitioning is developed and presented in Irion and Saito (2014); below is an excerpt describing the process. We only modify the ending conditions $|V(G_k^{j+1})| = L = 3$ based on the suggested neighcoeff window length.

1. Set $G_0^0 = G$ and $N_0^0 = N = |V(G)|$; initialize $K^0 = 1$ and $K^1 = 0$; set $j = 0$ and $k = 0$.

2. Contruct the Laplacian matrix $L(G_k^j)$.

3. Compute its eigenvectors, $\{\phi_{k,l}^j\}_{l=0}^{N_k^j - 1}$

4. If $N_k^j > 3$ then partition G_k^j by the sign of the Fiedler vector $\phi_{k,1}^j$ into $G_{K^{j+1}}^{j+1}$ and $G_{K^{j+1}+1}^{j+1}$; set $N_{K^{j+1}}^{j+1} = |V(G_{K^{j+1}}^{j+1})|$, and $N_{K^{j+1}+1}^{j+1} = |V(G_{K^{j+1}+1}^{j+1})|$, and $K^{j+1} = K^{j+1} + 2$; else set $G_{K^{j+1}}^{j+1} = G_k^j$, $N_{K^{j+1}}^{j+1} = |V(G_K^j)|$ and $K^{j+1} = K^{j+1} + 1$

5. If $k + 1 < K^j$, then set $k = k + 1$ and go back to step 1; else go to step 5.

6. If $|V(G_k^{J+1})| = 3$ for $k = 0, ..., K^{j+1} - 1$; then finish; else set $j = j + 1, k = 0, K^{j+1} = 0$, and go back to Step 1.

We then apply the SGWT to the graph signal, f, associated with each graph partition G_k^j of length $|V(G_k^j)| = 3$. We adapt equation 29 to reflect that the SGWT is being applied to the graph partition such that $W_{f_{G_k^j}}(s, n) = (T_g^s f_{G_k^j})(n)$ where the SGWT operator T_g^s is the SGWT kernel g as a function of the graph frequencies of G_k^j. Then neighCoeff is applied to the SGWT coefficients W for each scale, at each node for a given graph partition according to equation 31.

5.6 Results

We next compare several GSP methods used to denoise simulated and real world signals on graphs. To compare the different GSP methods, we use signal to noise ratio (SNR) as the metric for assessing the reconstruction error of graph signals generated from the different denoising methods. SNR is used as the primary metric for comparison for several graph signal denoising research such as in Irion and Saito (2014), and de Loynes et al. (2021). The goal of denoising graph signals is to decrease the noisy coefficients to a relative error of zero and retain meaningful coefficients. SNR is an accurate reflection of relative error shown in (Irion and Saito, 2014). In the Minnesota road network and Toronto road network studies, SNR is calculated as $20log_{10}(||f||_2^2/||\hat{f}-f||_2^2)$ where f is the original graph signal and $\hat{f}$ is the graph signal reconstruction of interest. For this metric, a larger SNR indicates a better graph signal approximation. In the Facebook network we adopt the negative signal to noise ratio used in Wang et al. (2015) and used for comparison in Irion and Saito (2014). The -SNR equation is $10log_{10}(MSE/||f||_2^2)$ where $MSE = \sum((\hat{f}-f)^2)$ and the noisy signal is calculated by $10log_{10}(n\sigma^2/||f||_2^2)$. f is the original graph signal and $\hat{f}$ is the graph signal reconstruction of interest. A smaller -SNR reflects a better graph signal approximation based on this metric.

Visual comparisons are more subjective, but are useful since an attractive feature of GSP is the visualization of the graph signal and the analysis of the graph signal. This clearly and uniquely conceptualizes the data in a data native structure. Another useful technique is to look at the variance in graph signal intensity for different areas of the graph of interest. A useful approximation may have slightly different graph signal values, with respect to the original graph signal, but represent the variance within similar graph areas and between different graph areas similar to the original graph signal. The goals of GSP visualization are met since the way graph data is

effected by different areas of the graph improves graph signal conceptualization.

5.7 Minnesota Road Network

Two separate experiments involve the Minnesota road network. The first experiment is a synthetic signal with varying degrees of impulses from de Loynes et al. (2021). The intent of using different impulses is to simulate different data for the different areas of the graph. The signal is a Bernoulli random variable with $f_{\nu,k} = W^k x_\nu / \lambda_1^k$. The two signals used in this experiment are generated using the gasper library in R and shown in figure 45. W is the weighted adjacency matrix ν can be thought of as the probability a nodes signal is one of k outcomes, an impulse or zero. The parameters $\nu = 0.01$ and $k = 2$ form one test signal and the parameters $\nu = 0.001$ and $k = 4$ form the second test signal. The first signal tested used varying levels of white Gaussian noise where $\sigma = 0.005, 0.01, 0.02$. Table 10 shows that the block thresholding method outperforms SUREShrink for three separate noise values for the reconstruction of these graph signals with varying degrees of impulses.

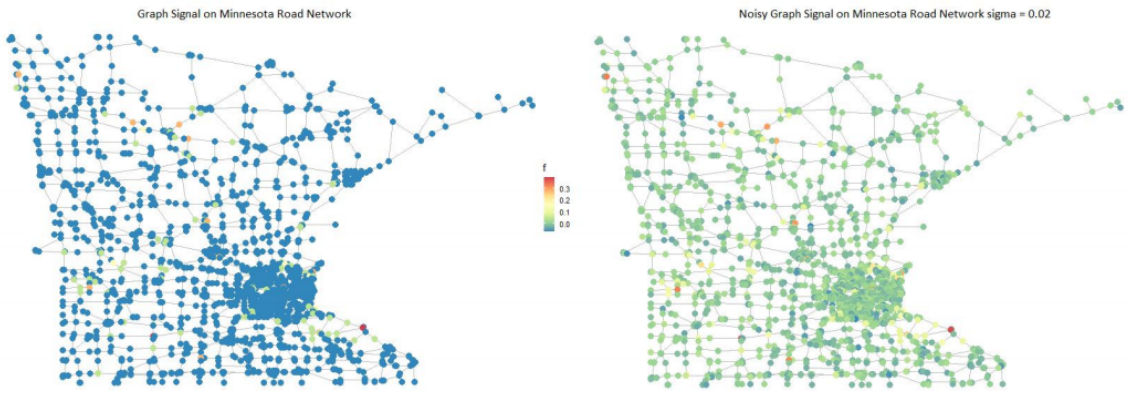

Figure 45. Minnesota Road Network with Bernoulli random variable signals from (de Loynes et al., 2021) overlayed. The graph on the left is the original function. The graph on the right is the original function with added noise. A majority of the graph signal is near zero with several impulses between 0.1 and 0.3. The goal of graph signal denoising is to detect the different the areas with impulses, determined by the original graph.

Table 10. Comparison of SureShrink and Block Thresholding for the Minnesota Road Network and the Bernoulli Random Variable graph signal. The WaveBT method outperforms the SUREShrink method for various SNR for the Bernoulli Random Variable. The values are reported in SNR (de Loynes et al., 2021)

SNR Comparison of SureShrink method with the WaveBT for various noise levels.

SNR/Method	SUREShrink	WaveBT
0.005	115.62	**125.50**
0.010	38.73	**40.11**
0.020	12.70	**13.95**

Irion and Saito (2014) present the HGLET and the HGWT methods which generate an orthonormal basis for a graph function based on graph partitioning. They employ these methods for several simulated and real world data sets. One simulation involves plotting a mutilated Gaussian signal on the Minnesota graph network. The mutilated Gaussian is a Gaussian distribution where some of the distribution is reorganized such that it appears there are two Gaussian distribution, a higher and a lower distribution. Figure 46 shows the mutilated Gaussian signal and the added noise. The authors of this work provide a multi scale transforms for signal on graphs (MTSG) toolbox as well as the data described in this simulation, `https://github.com/JeffLIrion/MTSG_Toolbox`. The noise added to the signal in this simulation is white Gaussian noise with a signal to noise ratio (SNR) of 5.00 dB.

Figure 47 shows the denoising results for the noisy mutilated Gaussian graph signal. The SnR is calculated using $20log_{10}(||f||_2/||\hat{f} - f||_2)$. The GHWT performs the best with a SNR of 11.56 dB, Block Thresholding second at 7.01 dB, HGLET third at 6.77 dB, and SURE at 4.99 dB. Visually, the differences in the graphs are slight, but important. The original graph signal, prior to the addition of noise, shows well definied regions which are relatively homogoneous, bottom right graph in figure 47. The WaveBT reconstruction, bottom left graph in figure 47, displays homogenous regions except for the central region where there is higher variance for the original graph signal. The noisy reconstruction and the SUREShrink reconstruction have

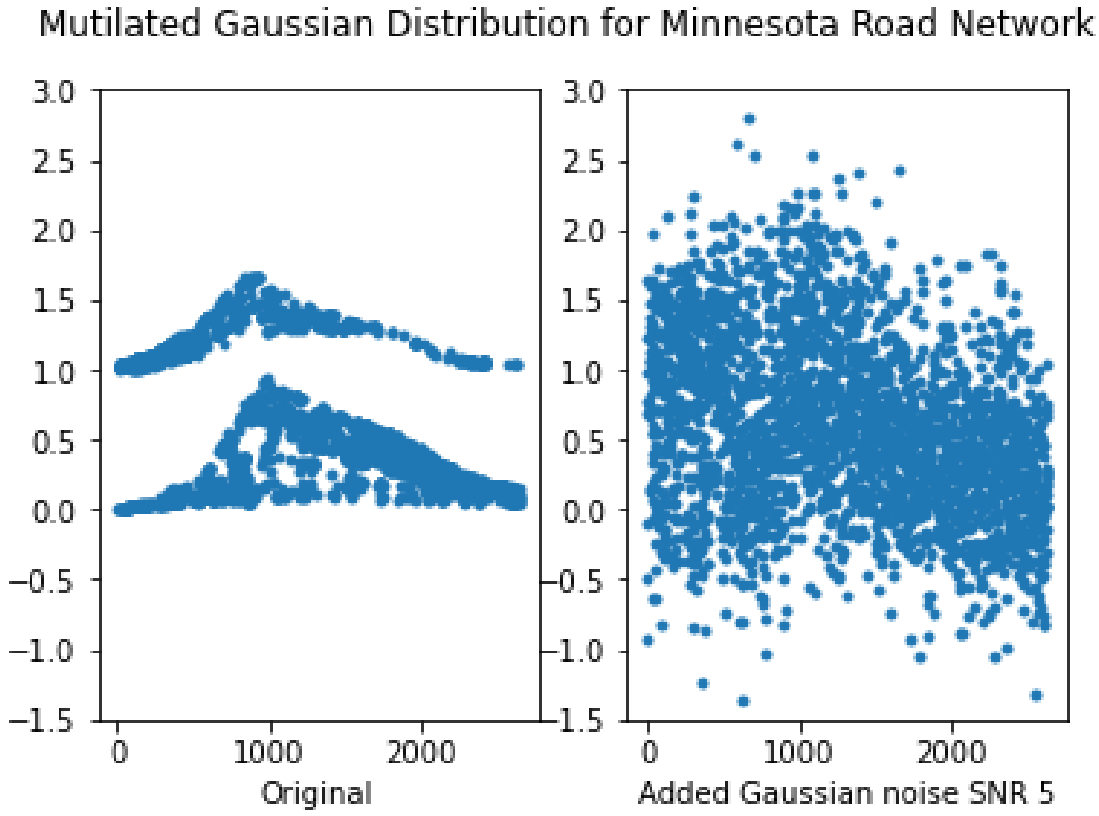

Figure 46. Graph of the mutilated Gaussian signal and the mutilated Gaussian signal with Gaussian noise of SNR 5 from (de Loynes et al., 2021). This is a challenging signal denoising problem as the two groups of data in the mutilated Gaussian are difficult to distiguish once the noise is added to the signal.

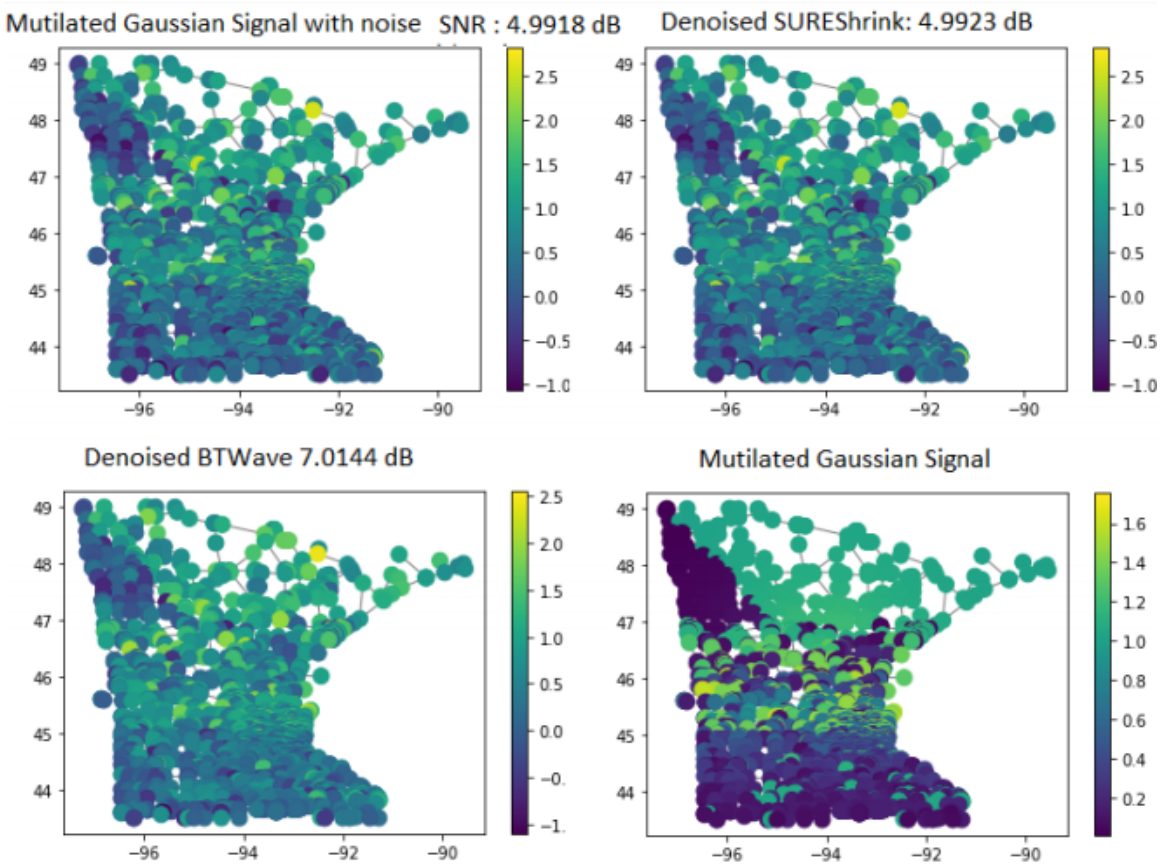

Figure 47. These figures represent the Minnesota road network with the mutilated Gaussian graph signal from (Irion and Saito, 2014). The values at each road intersection, or node, are the mutilated Gaussian values which were simulated to represent traffic data at a given time. The first graph (from left to right) is the noisy signal with SNR 4.9918. The second graph is an approximation of the signal using SUREShrink, SnR 4.9923. The third graph is the approximation of the signal using WaveBT, SnR 7.01. The fourth graph is the original signal. Visually, the WaveBT graph signal, in particular regions of the graph, are more homogeneous than the noisy graph signal and the SUREShrink approximation.

some extreme values in the lower region compared to the rest of the region. In the BTWave case these graph signal values are shrunk closer to zero based on graph wavelet coefficient thresholding.

5.8 Social Network

The Facebook graph from `Snap.Stanford.edu` is a nonplanar graph which represents Facebook users as nodes connected to other users or friends. This type of graph is used to analyze interconnected friend networks and analyze the flow of different types of data across these networks. This analysis is valuable to marketing, political, and media organizations since it provides insight into how their data and informational and marketing products will spread throughout these types of social media networks. Figure 48 shows the graph of the 4,039 Facebook users with 88,234 "friend" connections.

Wang et al. (2015) simulate several different types of graph data which are generated in R using the code provided by de Loynes et al. (2021) at `https://github.com/fabnavarro/SGWT-SURE`. Denoising these different types of signals gives a better approximation of how the signals behave on the graph. Figure 49 shows the different methods used for denoising the different signals. Block Thresholding, Sureshrink (de Loynes et al., 2021) and trend filtering (Wang et al., 2015) are used for denoising the different signals at -SNR of [-25,-20,-15,-10,5,0,-5,10]. Block wavelet thresholding has a larger negative SNR across the different noise levels for the dense Poisson signal. Trend Filtering and SureShrink perform better than the Block thresholding technique for the lowest noise level of the inhomogoneous random walk model. The block thresholding model has a larger -SNR for the larger noise levels.

WaveBT leverages signals on neighboring graph nodes, with similar frequencies, to make a thresholding decision for a graph signal at a particular node. In this case,

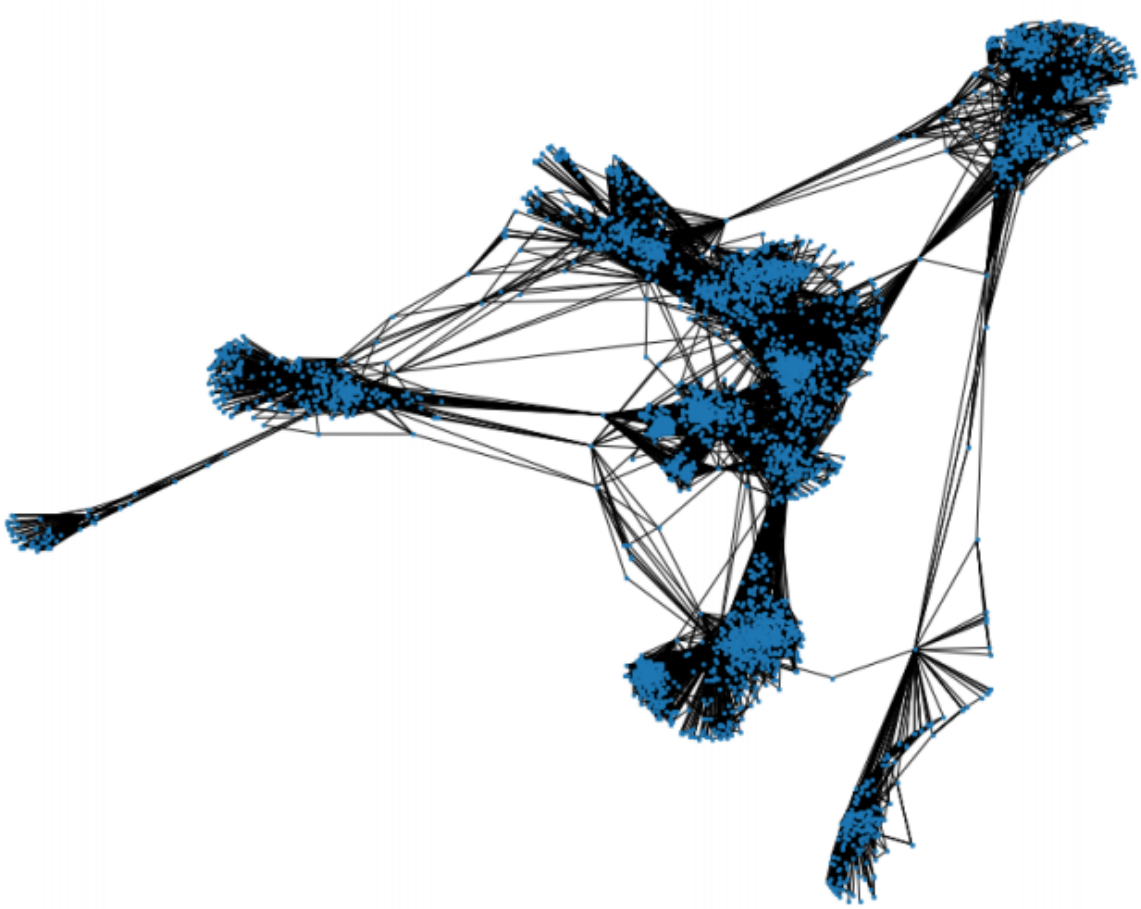

Figure 48. Graph from SNAP of 4039 social media users and their connections which provides an analytical structure representing the flow of information between friends on Facebook. The different clusters are ego centers, representing a single user with many connections. The graph shows an interesting degree of connection between these ego users and other ego users that are only connected through mutual friends

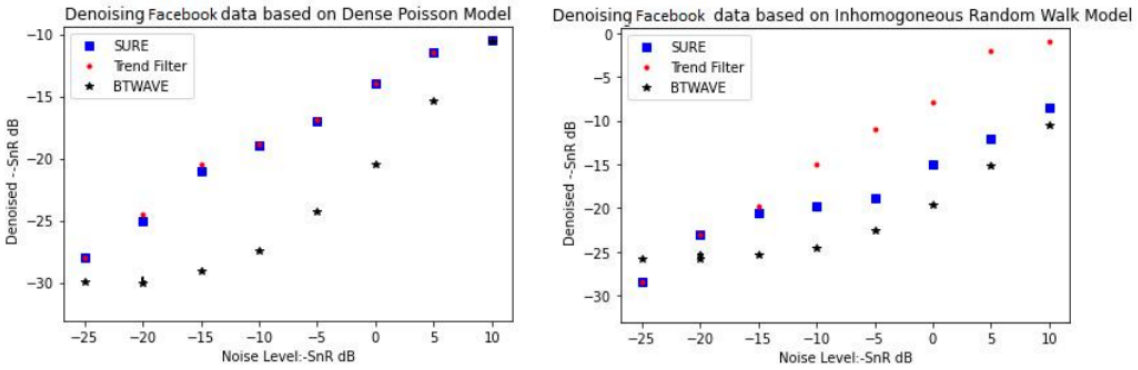

Figure 49. Graph of 4039 social media users denoised for two different test models applicable to social networks. The x axis values are the different simulation noise levels in terms of -SNR and the y axis values are the differences between the original signal and the denoised signal in terms of -SNR. The first graph shows the performance of three denoising methods for the Dense Poisson signal from (de Loynes et al., 2021). WaveBT outperforms SURE denoising and Trend Filtering for 8 of 9 noise levels, in terms of -SnR for the Dense Poisson graph signal. The graph on the right shows that WaveBT outperforms the other methods for 8 of 9 noise levels for denoising the Inhomogonous Random Walk graph signal.

graph signals at nodes with similar frequencies are Facebook users with similar status. Those Facebook users that connect large ego circles, or large networks of friends, to other large ego circles have similar status. Those Facebook users whose connections lie within a specific ego cluster also have similar status. Figure 50 shows the different graph frequencies and the respective wavelet operators at each scale s for each node n. The mother wavelet functions at each node, n, are formed by scaling and forming each scaled kernel functions, g_s, localized at each graph node n, such that $\psi_{s,n} = T_g^s \delta_n$ for each scale s. δ_n is a matrix where the selected node $n = 1$ and all other nodes are zero. The wavelet coefficients for each scale s and node n can also be expressed as $W_f(s,n) = \langle \psi_{s,n}, f \rangle$. Blocking graph wavelet coefficients to reflect similar user status is an intuitive and effective method for approximating and denoising signals on the SNAP social network graph.

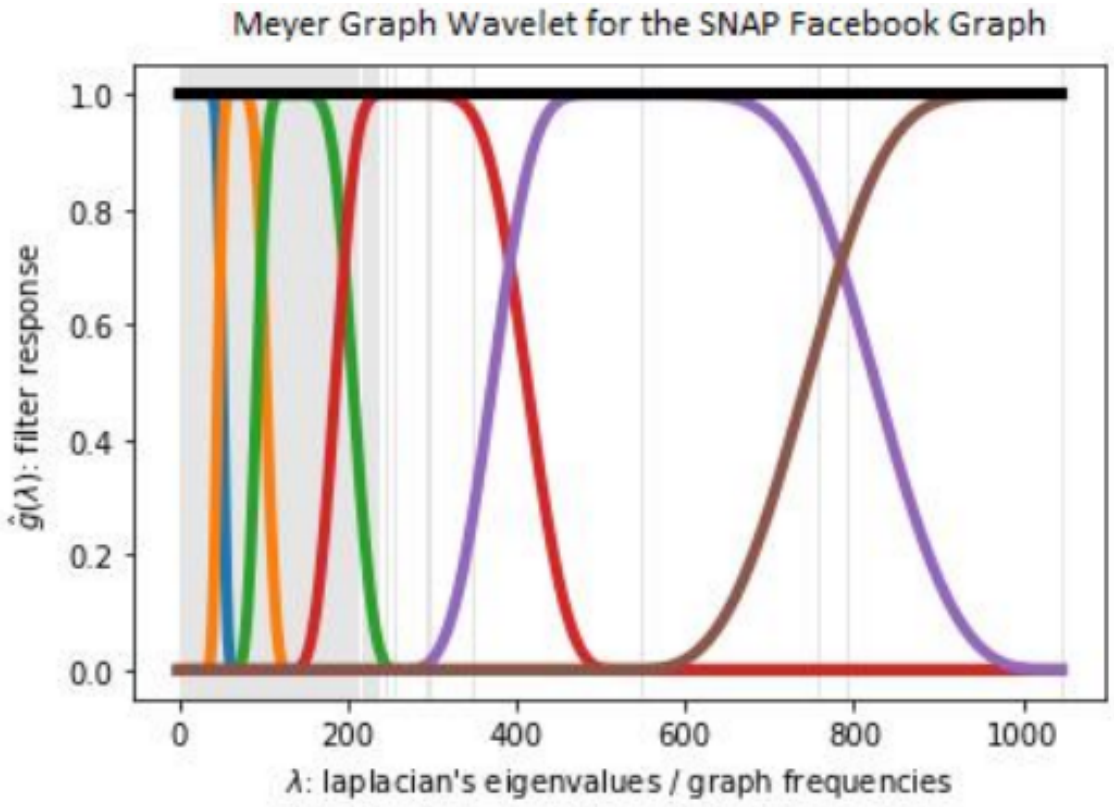

Figure 50. Graph Wavelet Frequencies: the graph frequencies are mostly lower frequencies with few higher frequencies. The lower frequencies represent the clusters of friends that are closely connected and the higher frequencies represent the few friends that connect clusters. The wavelet filters reflect this and the detailed wavelet filters from the SGWT cover the lower frequencies. The black lines at zero and one show the tight frame property of the Meyer graph wavelet

5.9 Real World Toronto Road Map Traffic Data

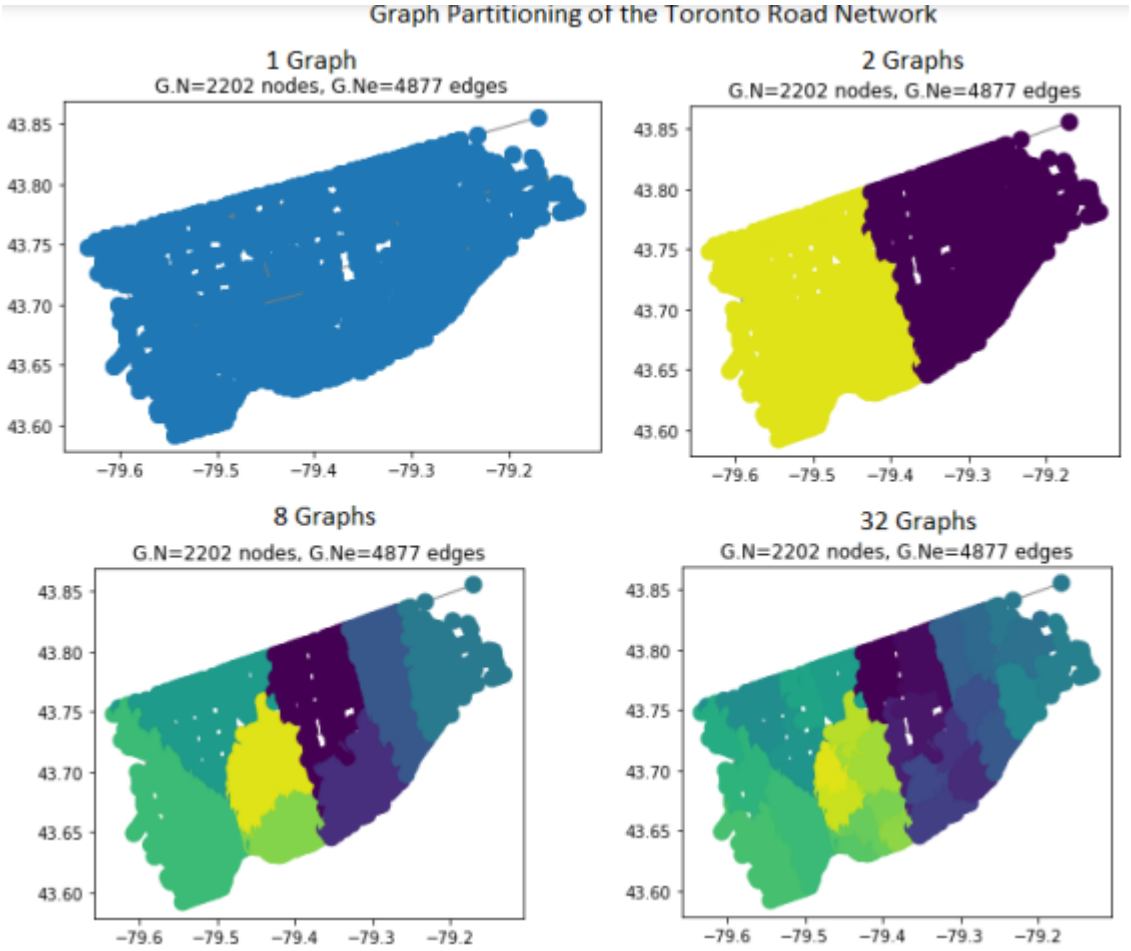

Figure 51. Graph partitioning of the Toronto road network for wavelet graph thresholding. Based on the inverse Euclidean distance between each node, the graph (top left) is partitioned into two graphs (top right), and progressively larger levels with finer graphs. A few examples of these are included such as 8 graphs (bottom left) and 32 graphs (bottom right). The puzzle piece shapes of the partition are a result of spectral partitioning rather than spatial. Spatial partitions are more block shaped, given the network has a rectangular shape, such as in road networks. In WaveBT, the partitions typically have different size, n, where $n = log(N)/2$

WaveBT is next applied to real world traffic data for the Toronto road map. The graph of the Toronto road map is a network of nodes and edges where the nodes represent intersections in the city of Toronto and the edges represent the roads which connect them. The traffic data from Irion and Saito (2014) were collected over a 24 hour period at each intersection in the graph. The weighted adjacency matrix is

based on the inverse Euclidean distance between intersecting nodes.

Partitioning graph wavelet coefficients based on the Toronto road network graph frequency is effective and intuitive. Coefficients thresholding decisions leverage information from similar graph signals at intersections with similar frequencies.

Figure 51 shows the original graph of the Toronto road network as well as the different levels of partitioning for the graph network. Different sectors of the city with similar road networks, based on distance between nodes and connectivity, are partitioned together. This is an effective method for blocking and subsequently thresholding graph wavelet signals for real world traffic data.

WaveBT improves upon other graph denoising methods such as the HGLET, GHWT and the SGWT with SUREshrink. Table 11, shows that graph wavelet block thresholding method outperforms the other methods in terms of SNR at 9.10 dB.

Table 11. Comparison of graph partition methods with WAVEBT for denoising the Toronto road network. The values represent the SnR of each method denoising the Toronto Road Network with real world data.

Comparison of SureShrink method with the Block Thresholding Method

Method	Toronto
WaveBT	**9.10 dB**
GHWT	8.23 dB
HGLET	8.96 dB

5.10 Conclusion

Using graph frequencies is an effective method for blocking wavelet coefficients. This work presents WaveBT, a new block thresholding scheme for graph signals to improve wavelet approximation and denoising. WaveBT performance is demonstrated using real world physical graphs with simulated and real world signals, achieving competitive and often superior results with block thresholding compared with stated of the are graph partitioning, trend filtering, and SGWT thresholding methods. The block thresholding method WaveBT, is shown as an effective technique for the analaysis of social network graphs. Code and examples from the work are stored at `https://github.com/JeffreyDWilliams/GSPWaves`.

VI. Conclusions and Recommendations

6.1 Conclusions

This research examined several areas where wavelet analysis has been applied. In general, the wavelet transform improves the quality of analysis for signal processing by smoothing noisy signals which improves the conceptualization of the area of interest. Yet, our three contributions show that the benefits of applying the wavelet transform in signal processing can be significantly improved with the addition of thresholding methods and other wavelet analysis tools. In paper one, we show that functional outliers can accurately be detected using wavelet thresholding methods in our WANOVA box plot. In paper two, we present the HMCV block thresholding method which improves smoothing and approximating signals with Gaussian and non-Gaussian error. In paper three, we present the WaveBT graph signal thresholding method which improves the approximation of graph signals. These three contributions propose new methods to improve signal processing applications.

Bibliography

Andrews, D. F. and A. M. Herzberg (2012). *Data: a collection of problems from many fields for the student and research worker*. Springer.

Antoniadis, A. and P. Fryzlewicz (2006). Parametric modelling of thresholds across scales in wavelet regression. *Biometrika 93*(2), 465–471.

Antoniadis, A. and G. Oppenheim (2012). *Wavelets and statistics*. Springer.

Atkinson, A. D., R. R. Hill, J. J. Pignatiello Jr., G. G. Vining, E. White, and E. Chicken (2017). Wavelet anova approach to model validation. *Simulation Modelling Practice and Theory 78*, 18–27.

Atkinson, A. D., R. R. Hill, J. J. Pignatiello Jr., G. G. Vining, E. D. White, and E. Chicken (2018a). Exposing system and model disparity and agreement using wavelets. *Journal of Verification, Validation and Uncertainty Quantification 3*(2), 1 – 6.

Atkinson, A. D., R. R. Hill, J. J. Pignatiello Jr., G. G. Vining, E. D. White, and E. Chicken (2018b). Wavelet anova bisection method for identifying simulation model bias. *Simulation Modelling Practice and Theory 80*, 66–74.

Barber, S. and G. Nason (2004). Real nonparametric regression using complex wavelets. *Journal of the Royal Statistic Society: Series B (Statistical Methodology) 66*, 927–939.

Bilmes, J. A. (1998). A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute 4*(510), 126.

Cai, T. T. (1999). Adaptive wavelet estimation: A block thresholding and oracle inequality approach. *The Annals of Statistics 27*(3), 898–924.

Cai, T. T. and B. W. Silverman (2001). Incorporating information on neighbouring coefficients into wavelet estimation. *The Indian Journal of Statistics B*, 127–148.

Cérou, F. and A. Guyader (2006). Nearest neighbor classification in infinite dimension. *ESAIM: Probability and Statistics 10*(1), 340–355.

Chang, S., B. Yu, and M. Vetterli (2000). Adaptive wavelet thresholding for image denoising and compression. *IEEE transactions on image processing 9*(9), 1532–1546.

Chang G.S., Yu, B. and M. Vattereli (2000). Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing 9*, 1532–1546.

Chicken, E. (2003). Block thresholding and wavelet estimation for nonequispaced samples. *Journal of Statistical Planning and Inference 116*(1), 113–129.

Coifman, R. and M. Maggioni (2006). Diffusion wavelets. *Applied and Computational Harmonic Analysis 21(1)*, 53–94.

Crouse, M. S., R. D. Nowak, and R. G. Baraniuk (1998). Wavelet-based statistical signal processing using hidden markov models. *Transactions on signal processing & Data Analysis 46*, 886–902.

Cuevas, A., M. Febrero, and R. Fraiman (2007). Robust estimation and classification for functional data via projection-based depth notions. *Test 22*(3), 481–496.

de Loynes, B., F. Navarro, and B. Olivier (2021). Data-driven thresholding in denoising with spectral graph wavelet transform. *Journal of Computational and Applied Mathematics 389*, 113319.

Debauchies, I. (1992). *Ten Lectures on Wavelets*. SIAM.

Defferrard, M., X. Bresson, and P. Vandergheynst (2016). Convolutional neural networks on graphs with fast localized spectral filtering. *ArXiv 1*, 1.

Donoho, D. L. and I. M. Johnstone (1995). Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association 90*(432), 1200–1224.

Donoho, D. L. and J. M. Johnstone (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika 81*(3), 425–455.

Downie, T. and B. Silverman (1996). The discrete multiple wavelet transform and thresholding methods. *IEEE Transactions in Signal Processing 46*, 2558–2561.

Efron, B. and R. J. Tibshirani (1994). *An introduction to the bootstrap*. CRC PRess.

Febrero, M., P. Galeano, and W. Gonzlez-Manteiga (2007). A functional analysis of nox levels: location and scale estimation and outlier detection. *Computational Statistics 22*(3), 411–427.

Febrero, M., P. Galeano, and W. GonzlezManteiga (2008). Outlier detection in functional data by depth measures, with application to identify abnormal nox levels. *The official journal of the International Environmetrics Society 19*(4), 331–345.

Ferraty, F. and P. Vieu (2006). *Nonparametric functional data analysis: theory and practice*. Springer Science & Business Media.

Fraiman, R. and G. Muniz (2001). Trimmed means for functional data. *Test 10*(2), 419–440.

Fryzlewicz, P. (2007). Unbalanced haar technique for nonparametric function estimation. *Journal of the American Statistic Association 102*, 1318–1327.

Gao, H. Y. (1997). Choice of thresholds for wavelet shrinkage estimate of the spectrum. *Journal of Time Series Analysis 18*(3), 231–251.

Girimurugan, S., E. Chicken, J. J. Pignatiello Jr., and M. S. Zeisset (2013). Wavelet anova for detection of local and global profile changes. In *Industrial and Systems Engineering Research Conference*. Institute of Industrial and Systems Engineering.

Hammond, D., P. Vandergheynst, and R. Gribonval (2011). Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis 30*(2), 129–150.

Hill, R. R. and J. O. Miller (2017). A history of united states military simulation. In *Proceedings of the 2017 Winter Simulation Conference*, Piscataway, New Jersey, pp. 346 – 364. Institute of Electrical and Electronics Engineers, Inc.

Hyndman, R. J. and H. L. Shang (2010a). Rainbow plots, bagplots, and boxplots for functional data. *Journal of Computational and Graphical Statistics 19*(1), 29 – 45.

Hyndman, R. J. and H. L. Shang (2010b). Rainbow plots, bagplots, and boxplots for functional data. *Journal of Computational and Graphical Statistics 19*, 29–45.

Hyndman, R. J. and M. S. Ullah (2007). Robust forecasting of mortality and fertility rates: a functional data approach. *Computational Statistics & Data Analysis 51*(10), 4942–4956.

Irion, J. and N. Saito (2014). Wavelets on graphs via spectral graph theory. *JSIAM Letters 6*, 21–24.

James, G. M. and T. J. Hastie (2001). Functional linear discriminant analysis for irregularly sampled curves. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 3*(63), 533–550.

James, W. and C. Stein (1992). Estimation with quadratic loss. *In Breakthroughs in statistics 1*, 443–460.

Johnstone, I. and B. Silverman (2005a). Empirical bayes selection of wavelet thresholds. *Annals of Statistics 33*, 1700–1752.

Johnstone, I. M. and B. W. Silverman (2005b). Empirical bayes selection of wavelet thresholds. *Annals of Statistics 33*(4), 1700–1752.

Kay, S. (1993). *Fundamentals of statistical signal processing: estimation theory*. Prentice-hall.

Kim, D. and K. Oh (2006). Cvthresh: R package for level dependent cross validation thresholding. *Journal of Statistical Software 15*(1), 1–13.

Liu, R. Y., J. M. Parelius, , and K. Singh (1999). Multivariate analysis by data depth: Descriptive statistics, graphics and inference. *The Annals of Statistics 27*, 783–858.

López-Pintado, S. and J. Romo (2009). On the concept of depth for functional data. *Journal of the American Statistical Association 104*(486), 718–734.

Makridakis, S. and M. Hibon (1997). Arma models and the boxjenkins methodology. *Journal of Forecasting 16(3)*, 147–163.

Mallat, S. and H. Wen Liang (1992). Singularity detection and processing with wavelets. *IEEE transactions on information theory 38*(2), 617–643.

McGinnity, K., R. Varbanov, and E. Chicken (2017). Cross-validated wavelet block thresholding for non-gaussian errors. *Computational Statistics & Data Analysis 106*, 127–137.

McGinnity, K., R. Warbanov, and E. Chicken (2017). Cross-validated wavelet block thresholding for non-gaussian errors. *Computational Statistics & Data Analysis 106*, 127–137.

Mohan, D., M. Asif, N. Mitrovic, J. Dauwels, and P. Jaillet (2014, October). Wavelets on graphs with application to transportation networks. In *In 17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1707–1712. IEEE.

Montgomery, D. C. (2009). *Design and Analysis of Experiments* (7 ed.). John Wiley & Sons.

Montgomery, D. C., E. Peck, and G. G. Vining (2012). *Introduction to linear regression analysis, 4th Edition*. Wiley.

Nason, G. P. (1996). Wavelet shrinkage using crossvalidation. *Royal Statistics Society 58*(2), 463–479.

Neumann, M. H. and R. Von Sachs (1995). Wavelet thresholding: Beyond the gaussian i.i.d. situation. *In Wavelets and statistics 1*, 301–329.

Ortega, A., P. Frossard, J. Kovaevi, J. Moura, and P. Vandergheynst (2018). Graph signal processing: Overview, challenges, and applications. *Proceedings of the IEEE 106(5)*, 808–828.

Pang, J., G. Cheung, A. Ortega, and O. Au (2015, April). Optimal graph laplacian regularization for natural image denoising. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2294–2298. IEEE.

Portilla, J., V. Strela, M. Wainwright, and E. Simoncelli (2001). Adaptive wiener denoising using a gaussian scale mixture model in the wavelet domain. In *Proceedings of the 8th International Conference of Image Processing*, Thessaloniki, Greece.

Rabiner, L. and B. Juang (1986). An introduction to hidden markov models. *IEEE ASSP magazine 3*(1), 4–16.

Ramsay, J. O. and B. W. Silverman (2007). *Applied functional data analysis: methods and case studies*. New York, New York: Springer.

Shapiro, J. M. (1993). Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on signal processing 41*(12), 3445–3462.

Sun, Y. and M. G. Genton (2011). Functional boxplots. *Journal of Computational and Graphical Statistics 20*(2), 316–334.

Torrence, C. and G. Compo (1979). A practical guide to wavelet analysis. *Bulletin of the American Meteorological Society 79*, 61–68.

Tukey, J. W. (1977). *Exploratory Data Analysis, Classics Edition*. Addison-Wesley.

Wang, Y., J. Sharpnack, A. Smola, and R. Tibshirani (2015, February). Trend filtering on graphs. In *In Artificial Intelligence and Statistics*, pp. 1042–1050. PMLR.

Yamanishi, Y. and Y. Tanaka (2005). Sensitivity analysis in functional principal component analysis. *Computational Statistics 20*(2), 311–326.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
17-09-2021	PhD Dissertation	Sept 2018 — Sept 2021

4. TITLE AND SUBTITLE

New Methods in Wavelet Analysis for Applications of the Wavelet Transform

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Williams Jeffrey D., Capt, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENS-DS-21-S-055

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Intentionally Left Blank

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution Statement A:
Approved for Public Release; Distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Common in signal processing (SP) is the detection of events. For instance, seismologists seek to detect abnormalities in an electromagnetic (EM) signal to detect the occurrence of an earthquake. Since many signals are noisy, such as those produced by a seismograph, it can be challenging to distinguish a significant EM pulse from incident noise. In SP, smoothing is used to remove the rough portions of a signal representing noise such that events are more obvious in a signal. This research applies and improves wavelet analysis methods across multiple domains and applications of signals since Wavelet analysis smooths signals while preserving important signal artifacts such as a large EM pulse representing an earthquake. Further, there are several useful properties of wavelet analysis such as time localization and sparsity which improve detection ability in SP. In this dissertation, we explore several applications, and domains of SP such as classical data, functional data, and graph data. We improve event detection such as outliers, and introduce new methods to detect and remove noise across these domains to improve SP analysis.

15. SUBJECT TERMS

signal processing, wavelets, image processing, thresholding

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Raymond R. Hill, AFIT/ENS
U	U	U	U	154	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636, x7469; rhill@afit.edu