Novel Methods for the Detection and Prediction of Changepoints

Jamie-Leigh Chapman, M.Math.(Hons.) M.Res

Submitted for the degree of Doctor of Philosophy

at Lancaster University.

December 2018

Abstract

This thesis focuses upon the detection and prediction of changepoints in time series. In particular, we develop a range of methods, both parametric and non-parametric, to detect, predict, and forecast in the presence of changepoints. We consider a range of data applications. These include economic, environmental and telematics data sets.

The first part of this thesis concentrates on forecasting. We propose two approaches to incorporate changepoints into the forecasting process. Each of these approaches are flexible. Additionally, we develop methodology to predict future changepoints in a time series. In particular, we can predict changepoints at both future time points, and changes near the end of the time series for which we do not yet have enough observations to detect. This also includes a new approach to pre-whitening time series that accounts for changes in the second order structure of the explanatory time series.

The second part of this thesis is concerned with changepoint detection. We introduce methodology for detecting changes in both the variance and the autocovariance of time series. To do this we consider a local measure of the variance and the autocovariance over time. The approach is non-parametric and resilient to the presence of outliers.

Acknowledgements

I would like to thank all the staff and students, past and present, of the STOR-i centre for doctoral training for providing such a supportive and rewarding research environment. I am most grateful to my two supervisors Idris Eckley and Rebecca Killick, both of whom has supported, encouraged and guided me over the years. I only hope we can continue working together in the future.

I am very grateful for the financial support provided by the EPSRC and the Office for National Statistics, and the academic support received from the BigInsight centre for research-based innovation.

Special thanks are to Catherine Ridout who inspired me to pursue Mathematics, from which I found my way to Statistics. I am very lucky to have had both her and Rebecca Killick as strong female role models. I hope that I am able to have such a positive influence on others in the future.

Lastly, I would like to express my utmost gratitude to my parents, my husband Aaron and my best friend Matthew for their support.

Declaration

I declare that the work in this thesis has been done by myself and has not been submitted elsewhere for the award of any other degree.

The word count of this thesis is approximately 47,000.

A version of Chapter 6 has been accepted for publication as Chapman, J-L., Eckley, I.A, Killick, R. (2019), *A nonparametric approach to detecting changes in variance in locally stationary time series*, Environmetrics.

Jamie-Leigh Chapman

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Data is becoming increasingly important to industries operationally and, with an uprise in the number of companies that provide data warehousing and other cloud based data management services, it is becoming faster and cheaper to perform large scale analytics. Consequently, time series are increasing in size.

Forecasting is one of the many important areas of time series analysis. When forecasting it is often assumed that the statistical properties of the time series remain constant throughout time. However, as a time series becomes longer, this assumption is less likely to hold. The focus of this thesis is the development of methods to detect such changes and incorporate them into forecasting.

Chapter 2 provides a literature review of changepoint detection with a focus upon time series. Throughout this review, it is apparent that change detection is of use in a multitude of application areas. This is reflected within this thesis. Chapter 3 considers microeconomic data, whereas Chapters 4 and 7 focus upon Telematics data. In contrast, Chapter 6 has an environmental focus.

Chapter 3 builds upon the changepoint methodology, reviewed in Chapter 2, to propose two methods for using changepoints to improve forecasts. The first considers

identifying changes during the data preprocessing stage before building our forecasting model. The second detects changes in the model we use to forecast the time series. This chapter allows forecasts to be produced in the presence of changepoints however, it does not have the facility to forecast changes explicitly. Hence, Chapter 4 introduces a framework in which future changepoints can be predicted. This methodology relies on constructing a relationship between two time series which both exhibit related changes. The location of changes in one series are then used to estimate the changes in the other. Chapter 4 also introduces an alternative approach to pre-whitening time series, which does not rely on the assumption of second order stationarity. This methodology exploits the changepoint detection methodology introduced in Chapter 3.

Chapters 3 and 4 highlight that many aspects of time series modelling rely on effectively capturing second order dependence structure. The methodology in Chapter 3 has the ability to capture changes in the second order structure of a time series, however it is limited to assuming that the time series follows an autoregressive (AR) or a moving average (MA) model. The remainder of this thesis seeks to remove this AR/MA assumption. These chapters use wavelets in their approach. Wavelets are suited to modelling the time-varying second order structure of time series due to their localisation properties. Chapter 5 provides a review of wavelets and outlines the methodology required as a basis for Chapters 6 and 7. Chapter 6 introduces a method to detect changes in the variance of a time series and Chapter 7 generalises this to the case of the autocovariance. These methods make no AR/MA assumptions on model form. It is demonstrated that each of these methods are robust to the presence of outliers.

Chapter 2

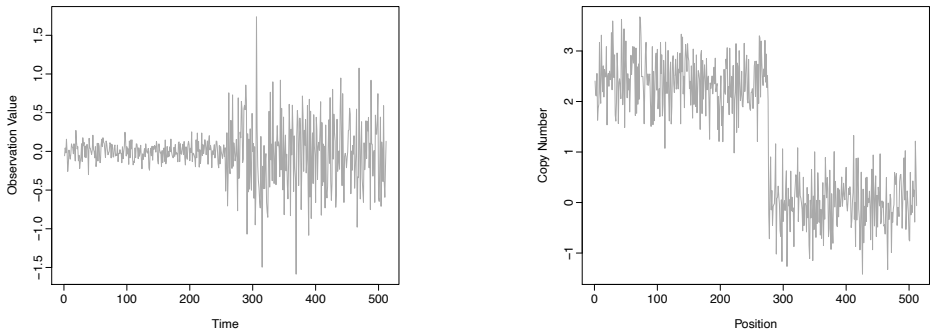
Literature Review

In this literature review we focus upon changepoint analysis. This provides a basis for the first part of this thesis. In Chapter 5, we provide a separate literature review on wavelets.

2.1 Introduction

A *changepoint* is a point, or position, in an ordered data sequence where the statistical properties change in some way. For example a changepoint could represent a point in time such that the variance of the observations prior to the change differ to those after the change, see for example Figure 2.1a. Alternatively for genomic data, in which observations are ordered by position on a chromosome, a changepoint could indicate a position where the mean level of the copy number of the DNA is smaller prior to the change, than afterwards, see for example Figure 2.1b.

Specifically in a time series setting, changepoints could occur in lower order structures, such as the mean, or they could occur in higher order structures such as the autocovariance. More than one statistical property could change at the same time. It



(a) (b)

Figure 2.1: Two examples of changepoints: (a) a point in time where the variance changes and (b) a position along a chromosome where the mean level changes.

is vital, in a modelling or forecasting setting, that we account for changepoints within our frameworks, failure to do so will result in flawed inference. As a consequence of the practical importance of changepoints, a vast literature surrounding the area has arisen over the last fifty years.

Since the early work in changepoints by Page (1954) in the context of quality control, changepoint detection methods have been extensively developed in a range of different application areas. Some classical applications of changepoint detection include: climatology (Reeves et al., 2007; Ruggieri et al., 2009); finance (Spokoiny, 2009; Andreou and Ghysels, 2009); model validation (Fryzlewicz and Subba Rao, 2010). More modern applications include network security (Lvy-Leduc and Roueff, 2009; Bodenham and Adams, 2014), neuroscience (Aston et al., 2012; Kirch et al., 2015) and linguistics (Kulkarni et al., 2015).

In this chapter we discuss a range of approaches to the problem of detecting changepoints. We restrict our attention to the problem of retrospectively detecting changepoints in an "off-line" setting. The contrasting sequential setting is described by Lai (1995) and Polunchenko and Tartakovsky (2012). For a general review of changepoint detection we refer the reader to (Carlstein et al., 1994; Chen and Gupta, 2013; Eckley et al., 2011; Jandhyala et al., 2013).

The aim of this chapter is to form the basis for the work presented in Chapters 3 and 4 of this thesis. To begin, in Section 2.2 we introduce the single changepoint model. Then in Section 2.3 we introduce binary segmentation and its variants. These can be used to extend any single changepoint model into the multiple changepoint setting. In Section 2.4 we describe the penalised cost function approach to multiple changepoint detection and in Section 2.5 we briefly review some other changepoint frameworks. Specifically, in Section 2.6 we review changepoint detection methods focused on second order structure.

2.2 Single Changepoint Detection

Consider an ordered data sequence of length n, say $y_{1:n} = (y_1, \ldots, y_n)$, and let $Y_{1:n}$ be the corresponding sequence of random variables. Then, following the notation of Eckley et al. (2011), a single changepoint occurs if there exists a location $\tau \in \{1, \ldots, n-1\}$ such that the statistical properties of $\{y_1, \ldots, y_\tau\}$ and $\{y_{\tau+1}, \ldots, y_n\}$ differ. It is natural to introduce the detection of a single changepoint as a likelihood-ratio test.

The likelihood approach to detect changepoints was first proposed by Hinkley (1970) for detecting changes in mean within a sequence of i.i.d. Normally distributed observations. This was later generalised to other distributions, for example Gamma (Hsu, 1979), Exponential (Haccou and Meelis, 1988), and Binomial (Hinkley and Hinkley, 1970). It has also been extended to detect changes in other properties of the data, for example the variance (Chen and Gupta, 1997).

In the likelihood approach, we present the detection of a single changepoint as a

hypothesis test with the null and alternative hypothesis given by

$$\mathrm{H}_0: \text{ No changepoint.} \qquad \mathrm{H}_1: \text{ A single changepoint.}$$

Under the null hypothesis, H_0, the maximum log-likelihood is given by $\ell(y_{1:n}|\hat{\theta})$, where $\ell(\cdot)$ is the log-likelihood of the probability density function associated with the entire data and $\hat{\theta}$ is the maximum likelihood estimator of the parameters.

Assuming independence across segments, under the alternative hypothesis, the maximum profile log-likelihood for a given changepoint $\tau \in \{1, 2, \ldots, n-1\}$ is given by

$$Pr\ell(\tau) = \ell(y_{1:\tau}|\hat{\theta}_1) + \ell(y_{\tau+1:n}|\hat{\theta}_2),$$

where $\hat{\theta}_i$, $i = 1, 2$, are the maximum likelihood estimators of the parameters for segment i. The location of the changepoint is discrete, therefore the maximum log-likelihood under H_1 is: $\max_\tau Pr\ell(\tau)$.

The log of the likelihood ratio test statistic is:

$$\lambda(y_{1:n}) = 2\left[\max_\tau Pr\ell(\tau) - \ell(y_{1:n}|\hat{\theta})\right].$$

We then choose a threshold β such that if $\lambda(y_{1:n}) > \beta$, we reject the null hypothesis. In this case the position of the changepoint, $\hat{\tau}$, is estimated by the profile log-likelihood for τ:

$$\hat{\tau} = \arg\max_\tau Pr\ell(\tau).$$

In order to select the appropriate threshold β for a required significance level, the asymptotic distribution for the likelihood ratio test must be attained. These distributions, and consequently the thresholds, for the case of Normal, Binomial and Poisson distributions are derived by Chen and Gupta (2013).

In the following section, we illustrate how the likelihood approach could be extended into a multiple changepoint setting.

2.3 Binary Segmentation

Binary segmentation (BS), first introduced by Scott and Knott (1974), is arguably the most widely used method for detecting multiple changepoints and can be used to extend any single changepoint method, for example the likelihood-ratio approach (Eckley et al., 2011).

To perform binary segmentation we first apply the chosen single changepoint detection method to the entire data set. If no changepoint is found then the algorithm has finished. If a changepoint is detected, call this τ, then the data is split into two segments, $y_{1:\tau}$ and $y_{\tau+1:n}$. We then apply the single changepoint method to the two segments and repeat iteratively. We stop when no further changepoints are detected.

Binary segmentation has since been implemented by Venkatraman (1992) and Chen and Gupta (1997) to detect changes in independent Normal observations. Cho and Fryzlewicz (2012) and Killick et al. (2013) have used it in conjunction with the wavelet spectrum to detect changes in the second order structure of time series. Venkatraman (1992) and Cho and Fryzlewicz (2012) prove the consistency of the algorithm in the case of an unknown number of changepoints with additive and multiplicative errors, respectively.

Despite being fast, $\mathcal{O}(n \log n)$, binary segmentation does have some disadvantages. A drawback to its computational efficiency is that it is only approximate. This is because the changepoint locations identified are conditional on previously identified changepoints. Another drawback is that binary segmentation may fail to identify small segments between larger ones, or 'epidemic' changepoints, when we have two

changepoints and the first and last segment follow the same distribution.

To overcome these drawbacks, Olshen et al. (2004) introduce a modification of BS, called circular binary segmentation (CBS). At each iteration, this algorithm can detect either a single changepoint or two changepoints. As the name suggests, it considers the data in a circular fashion, and at each iteration the data within which you are searching for a changepoint/s is joined at either end to form a circle.

Willenbrock and Fridlyand (2005) and Lai et al. (2005) both compare circular binary segmentation against other methods for detecting changepoints in comparative genomic hybridization (CGH) data and show that it performs well, however from the methods compared, Lai et al. (2005) conclude that CBS is one of the slowest. The loss in computationally efficiency of circular binary segmentation is attributed to the non-parametric methods used to calculate the p-value, and as such, the algorithm grows quadratically with the length of the data.

A faster CBS algorithm is later developed by Venkatraman and Olshen (2007) in which the p-value of the test statistic is calculated using a Gaussian random field. A stopping rule is also added which limits the number of iterations of the algorithm when there is strong evidence of a change. The changes implemented by Venkatraman and Olshen (2007) improve the efficiency of CBS with only a small loss in accuracy.

Another modification to binary segmentation (BS) is wild binary segmentation (WBS) (Fryzlewicz et al., 2014). This calculates the test statistic on random draws from the data thereby sacrificing computation time for an increase in accuracy. This modification also alleviates the small segment issue and can identify changes of smaller magnitude.

More specifically, WBS calculates the test statistic on multiple intervals with start and end points which are drawn uniformly, with replacement, from the set $\{s, \ldots, e\}$, where s and e are the start and end points of the current segment. Having done this,

the test statistics are weighted according to the length of the interval and the largest test statistic is tested against the threshold value.

In the WBS setting, in addition to choosing a threshold for detecting a changepoint, the number of intervals drawn at each iteration also needs to be chosen. This introduces a trade off between accuracy and computationally efficiency. An increased number of intervals will increase accuracy, but this comes at a loss of computational efficiency. Fryzlewicz et al. (2014) discuss the choice of penalty and number of interval in order to obtain good results. In addition to having more tuning parameters, WBS has increased computational time over BS, because the test statistic needs to be computed for multiple intervals.

In summary, BS is easy to understand and it can be used with any changepoint test thus providing a simple route from single changepoint detection to multiple changes. However, there are clear disadvantages in terms of approximation error, which are not wholly overcome by the new variants. The following section discusses a penalised cost function approach to changepoint detection which, in contrast to BS, can be guaranteed to give the optimal solution.

2.4 Penalised Cost Functions

In a multiple changepoint setting, one commonly used method is the penalised cost function approach. Following Eckley et al. (2011), consider m changepoints with positions $\tau = (\tau_1, \ldots, \tau_m)$. Each changepoint position τ_i, is an integer between 1 and $n-1$ and we define: $\tau_0 = 0$ and $\tau_{m+1} = n$. The changepoints are ordered such that: $\tau_i < \tau_j \iff i < j$. Thus the m changepoints split the data series into $m+1$ segments with the i^{th} segment containing $y_{(\tau_{i-1}+1):\tau_i}$. In practice we impose a minimum segment length, g, such that $\tau_{i+1} - \tau_i \geq g \geq 2$. Then, in order to determine the locations of

the changepoints, we aim to solve the *penalised minimisation problem*:

$$\min_{m,\tau_{1:m}} \sum_{i=1}^{m+1} \left[\mathcal{C}(y_{(\tau_{i-1}+1):\tau_i})\right] + \beta f(m), \tag{2.1}$$

where $\mathcal{C}$ is a cost function over a segment and $\beta f(m)$ is a penalty based on the number of changepoints m (Killick et al., 2012). The penalty term is introduced to prevent over-fitting. An example penalty is the Schwarz Information Criterion (SIC, (Schwarz et al., 1978)) ($\beta = p \log n$), where p is the number of additional parameters introduced by an additional changepoint. If this penalty is set too high, we run the risk of under-fitting. Generally, the value of the threshold can have substantial impact on the number of changepoints estimated, see Haynes et al. (2017a) for examples. The function $f(m)$ is often taken to be the number of changepoints m, resulting in a penalty that is linear in the number of changepoints.

Detecting multiple changepoints is more computationally challenging than the single changepoint case. Specifically, as the length of the data sequence increases, the number of possible changepoint positions increases rapidly. For this reason, much of the multiple changepoint literature is dedicated to developing efficient algorithms.

In the following, in Section 2.4.1, we first review dynamic programming approaches to solving the minimisation problem in equation (2.4). In Section 2.4.2 we discuss the choice of cost function in equation (2.4). Finally, in Section 2.4.3 we discuss the choice of penalty for equation (2.4).

2.4.1 Dynamic Programming

The first dynamic programming approach to changepoint detection was undertaken by Auger and Lawrence (1989) in their Segment Neighbourhood Search (SNS) algorithm. This assumes that there is some maximum number of changepoints, M, and for each

number of changes $1, \ldots, M$ it determines the best partition of the data. This solves a constrained version of equation (2.4). The computation is of order $\mathcal{O}(Mn^2)$. This method does not have a choice of penalty but as, in practice, the number of changes is often unknown, it can be equally difficult to return a single segmentation.

Jackson et al. (2005) introduce Optimal Partitioning (OP) which improves upon Segment Neighbourhood Search. Optimal Partitioning requires no such assumption on the number of changes in the data and is instead of order $\mathcal{O}(n^2)$. It aims to solve the penalised minimisation problem in equation (2.4). In contrast to creating a dynamic program across the number of changes, Jackson et al. (2005) create a dynamic program across time. This requires no upper bound on the number of changes but a penalty must be chosen.

Exploring the structure of this dynamic program further, OP first conditions on the last point of change and then calculates the optimal segmentation of the data up until that point. As the segments are independent, if we know the position of the last changepoint, then we can use this to calculate those prior to it. Thus if for every time point we know when the last change was prior to that, we can reconstruct the entire segmentation.

Formally, let $F(n)$ be a minimisation from equation (2.4) with $f(m) = m$. Then we can write

$$F(n) = \min_{\boldsymbol{\tau}} \left\{ \sum_{i=1}^{m+1} \left[\mathcal{C}(y_{(\tau_{i-1}+1):\tau_i)} + \beta \right] \right\}.$$

Then, denote the last changepoint τ_m as τ^*. If we condition on the location of the last change, then we can obtain

$$F(n) = \min_{\tau^*} \left\{ \min_{\boldsymbol{\tau}|\tau^*} \sum_{i=1}^{m} \left[\mathcal{C}(y_{(\tau_{i-1}+1):\tau_i)}) + \beta \right] + \mathcal{C}(y_{(\tau^*+1):n}) + \beta \right\}. \tag{2.2}$$

This procedure can then be repeated for subsequent changepoints. To illustrate the

iterative nature of this procedure, we can re-write equation (2.4.1) as

$$F(n) = \min_{\tau^*} \left\{ F(\tau^*) + \mathcal{C}(y_{(\tau^*+1):n}) + \beta \right\}.$$

Optimal Partitioning is of order $\mathcal{O}(n^2)$, so in order to make this approach faster, Killick et al. (2012) introduces the Pruned Exact Linear Time (PELT) algorithm. PELT is based on the Optimal Partitioning method of Jackson et al. (2005), but involves an inequality based pruning step within the dynamic program. PELT reduces the computational cost of OP whilst maintaining the exactness of the method.

PELT considers the data sequentially and the optimal segmentation up to that time point. At each time point, Killick et al. (2012) demonstrate that the number of changepoint configurations is restricted. For all times $t < s < n$, it is assumed there exists a constant K such that,

$$\mathcal{C}(y_{(t+1):s}) + \mathcal{C}(y_{(s+1):n}) + K \leq \mathcal{C}(y_{(t+1):n}).$$

Then, defining $F(\cdot)$ as in equation (2.4.1), if

$$F(t) + \mathcal{C}(y_{(t+1):s}) + K \geq F(s)$$

holds, at a future time $n > s$, t can never be the optimal last changepoint prior to n. This means that time t does not need to be considered in the calculations for future times greater than n for the rest of the dynamic program. Most cost functions satisfy this condition and Killick et al. (2012) provide details on the selection of K. If the cost function is the negative log-likelihood, then $K = 0$. However, in order to obtain the bound K, it must be assumed that the number of changepoints in the data increases linearly with the length of the data. In such as case, implementing this restriction,

or pruning step, means that the number of changepoint configurations is bounded by a constant number, K, at each time step. Thus PELT is of order $\mathcal{O}(Kn)$. In the case that the number of changepoint does not increase linearly with the length of the data, PELT can not achieve $\mathcal{O}(Kn)$, and so PELT is best in applications where the number of changepoints is large.

Maidstone et al. (2017) introduce a similar method to PELT which also uses inequality based pruning, but instead they apply it to SNS and call it Segment Neighbourhood with Inequality Pruning (SNIP), however this performs poorly in comparison to pruning SNS using *functional pruning*.

Rigaill (2015) introduce an algorithm called pDPA, and this is a pruned version of Segment Neighbourhood Search (Auger and Lawrence, 1989). Instead of performing inequality based pruning, they use functional pruning. Rigaill (2015) show empirically that the time complexity of pDPA is $\mathcal{O}(Kn \log n)$. A drawback of pDPA is that it is necessary to calculate and store the values of multiple cost functions. Additionally, as it implements functional pruning, it can only be used to detect changes in a single parameter.

Similarly, Maidstone et al. (2017) introduce Functional Pruning Optimal Partitioning (FPOP) which uses functional pruning on OP and they show that this always prunes more than PELT. They also perform an empirical study which suggests that FPOP is computationally efficient for large datasets regardless of the number of changepoints. However, once again, as this implements functional pruning, it can only be used to detect changes in a single parameter.

2.4.2 Cost Function

Cost functions for changepoint detection can be categorised as those which are parametric and based upon the likelihood of the data, and those which are non-parametric and so make no assumptions on the distributional form of the data.

When using the likelihood as the basis for a cost function of a segment, we use a scaled maximum log-likelihood: $-\log \ell(y_{(\tau_{i-1}+1):\tau_i}|\hat{\theta})$, where θ is the vector of parameters in which we want to find changes in. For example, changes in mean in i.i.d. Gaussian data can be detected by replacing the cost function $\mathcal{C}(\cdot)$ in equation (2.4) with twice the negative log-likelihood for a Gaussian distribution with common variance and segment specific mean. For the data in a segment $y_{(\tau_{i-1}+1):\tau_i}$, the segment cost of twice the negative log-likelihood will be

$$\mathcal{C}(y_{(\tau_{i-1}+1):\tau_i}) = \frac{1}{\sigma^2} \sum_{j=\tau_{i-1}+1}^{\tau_i} (y_j - \hat{\mu})^2,$$

where $\hat{\mu}$ is the maximum likelihood estimator for the segment mean.

A likelihood based cost function is effective if the distributional assumptions are realistic. However, as the data becomes increasingly different from the chosen distribution, the power to detect a changepoint will decrease. Therefore, if we model the data using the wrong distribution, changepoint locations are less reliable. Consequently, a non-parametric cost function may be preferable.

A commonly used non-parametric cost function for a segment is the quadratic loss function, defined as

$$\sum_{t=\tau_{i-1}+1}^{\tau_i} (y_t - \theta_i)^2 \tag{2.3}$$

where θ_i is the mean of the segment containing data $y_{\tau_{i-1}+1:\tau_i}$. The use of the quadratic loss function can be seen in Inclan and Tiao (1994) and Rigaill (2015). The quadratic

loss function (2.4.2) approach is susceptible to outliers and Fearnhead and Rigaill (2018) suggest the use of a cost function that increases at a slower rate in $|y - \theta|$, these include the Huber loss and the biweight loss (Huber, 2011).

Alternatively, Zou et al. (2007) introduce a non-parametric equivalent to the scaled log-likelihood. They propose a non-parametric log-likelihood function based upon the empirical cumulative distribution function (CDF) and use this in a likelihood ratio test to detect a single changepoint. This is later extended by Zou et al. (2014) into the multiple changepoint setting using Segment Neighbourhood Search (SNS). Zou et al. (2014)'s approach performs well however it is computationally slow, $\mathcal{O}(mn^2+n^3)$. This complexity is attributed to the pre-computation of the segment costs and running the SNS algorithm.

Later, Haynes et al. (2017b) build upon Zou et al. (2014)'s approach in order to improve the computationally efficiency; they simplify the segment cost using an approximation and use PELT instead of SNS. The resulting algorithm, which they call ED-PELT, runs with expected computational cost of $\mathcal{O}(n + n^2 \log n)$.

Other approaches which use a non-parametric cost function include the "E-Divisive" method of Matteson and James (2014). This uses a cost function which aims to maximise a Euclidean distance between two sub-segments at each iteration of BS. It is later used within a dynamic programming setting (James and Matteson, 2015).

Non-parametric approaches to changepoint detection can often be more robust as no distributional assumptions are made, however if the distribution is known, a parametric approach will be more powerful.

Having discussed the choice of cost function for use in equation (2.4), we now turn our attention to dynamic programming, an algorithm which can be used to solve this minimisation problem.

2.4.3 Penalty

In a penalized cost setting, the final model determined will be dependent upon the penalty used in equation (2.4). This penalty, consists of two components. The first, is the constant β and the second is the function $f(m)$. Usually, we set $f(m) = m$ such that it is linear in the number of changepoints (Killick et al., 2012). Picard et al. (2005) and Birgé and Massart (2007) offer some discussion on alternative penalty choices. Choices for the constant β are more varied in the literature.

Examples of penalties which are commonly used include Akaike's information criterion (AIC, (Akaike, 1974)), Schwarz information criterion (SIC, (Schwarz et al., 1978)) and the Hannan-Quinn information criterion (Hannan and Quinn, 1979), defined as

$$\begin{aligned} \text{AIC} &: \beta = 2p \\ \text{SIC} &: \beta = p \log n \\ \text{Hannan-Quin} &: \beta = 2p \log\log n, \end{aligned}$$

respectively. Asymptotically, the SIC and the Hannan-Quin penalties result the correct number of changepoints, see Yao et al. (1988) for details. Despite this, the Hannan-Quin penalty is less popular. The AIC is still popular despite it asymptotically over estimating the number of changepoints (Birgé and Massart, 2001). This has also been observed in practice by authors such as Haynes et al. (2017a), Kim et al. (2009) and Lavielle (2005). Alternatively, Lavielle (2005) propose an adaptive choice of penalty.

In many applications it may not be appropriate to choose only one penalty and segmentation. It may be better to have multiple segmentations of the data and then choose the most suitable according to a practitioner or the task at hand. Additionally, if the assumed distributional form of the data is incorrect, then the assumptions

that ensure these penalties provide consistent estimates may not be valid. For this reason, more recently, Haynes et al. (2017a) propose a method "Changepoints for a Range Of Penalties" (CROPS) which returns all possible segmentations for some penalty range in a computationally efficient manner.

In the following, we now turn our attention to an alternative approaches to detecting multiple changepoints.

2.5 Other Approaches

In Section 2.3 and 2.4 we described two approaches to detecting multiple changepoints. Here we briefly review some alternative approaches to the problem. Specifically, in Section 2.5.1 we describe a genetic algorithm approach, in Section 2.5.2 we describe a hidden Markov model approach and finally in Section 2.5.3 we describe a Bayesian approach to changepoint detection.

2.5.1 Genetic Algorithm

A genetic algorithm is like natural selection taking place in species evolution. Suppose we have a set of solutions which have weights according to some optimization criterion, then according to these weights, we select two 'parent' solutions. These two solutions form a new 'child' solution whose genes consist of the best genes from the parents. The procedure allows mutation to take place such that the algorithm does not get stuck in local optima.

The use of a genetic algorithm for changepoint detection has been implemented by a selection of authors. For example, Liang and Wong (2000) develop an evolutionary Markov Chain Monte Carlo (MCMC) routine for changepoint detection, Davis et al.

(2006) use a genetic algorithm to detect changes in autocovariance in a time series and Li and Lund (2012) follow by example to detect changes in the mean of climatic time series.

The genetic algorithm approach has the advantage that it will produce high quality segmentations very quickly. However the search is approximate and repeated runs on the same data may not produce the same results.

2.5.2 Hidden Markov Models

Hidden Markov models (HMMs) are an extension of Markov models first developed by Baum and Eagon (1967) at the Institute for Defense Analyses. They are used in applications such as pattern recognition (Rabiner, 1989) and clustering (Knab et al., 2003). A HMM can be characterised by an underlying process generating an observable sequence. This latent process is a Markov process and generates observations. Luong et al. (2012) provide an introduction the use of HMMs for changepoint analysis.

A HMM can be fitted using either a classical frequentist or a Bayesian framework and the hidden states (segmentations) can be inferred using, for example, Viterbi (Viterbi, 1967) and Posterior Decoding (Juang and Rabiner, 1991) algorithms, or the Forwards-Backward equations (Baum et al., 1970). For a recent contribution to changepoint detection using HMMs, please see the work of Ko et al. (2015), who propose an extension to the HMM of Chib (1998).

2.5.3 Bayesian Methods

A Bayesian framework for changepoint analysis was first introduced by Chernoff and Zacks (1964) for detecting a change in the mean of a sequence of independent normal random variables. In a Bayesian setting, we must specify a prior on the number of

changepoints, the location of changepoints and also upon the parameters for each segment. There are two ways to do this. The first is to put a prior on the number of changepoints and then another prior for their position given the number of changepoints (Barry and Hartigan, 1992). The second formulation is to specify a prior for both the number of changepoints and their positions indirectly through a distribution for the length of each segment (Pievatolo and Green, 1998).

In the first case, if the number of changepoints is known, then Markov Chain Monte Carlo (MCMC) is often used to estimate the changepoint locations and the associated segment parameters (Stephens, 1994; Chib, 1996, 1998). When the number of changepoints is unknown, a common approach is reversible jump MCMC (Green, 1995). Alternatively, Lavielle and Lebarbier (2001) propose a hybrid approach using the Metropolis-Hastings algorithms with a Gibbs-sampler.

More recently, Schwaller and Robin (2017) extend the product partition model of Barry and Hartigan (1992) by adding a graphical structure which could capture the dependencies between multivariate observations.

In the second case, where a prior is placed on the duration of each segment, the posterior can be sampled directly (Barry and Hartigan, 1993). This approach has been taken by Liu and Lawrence (1999) for DNA sequencing and has been used more generally by Fearnhead (2005) and Fearnhead (2006). This approach assumes independence across segments. Consequently, Fearnhead and Liu (2011) extend their approach to include dependence across segments.

More recent contributions include the work of Rigaill et al. (2012). They derive the exact posterior distribution of changepoint locations for exponential random variables with conjugate priors. This approach is later adapted by Cleynen and Robin (2016) in order to compare multiple series. Most recently Hinoveanu et al. (2019) propose a loss-based approach to Bayesian changepoint analysis.

We refer the reader to Eckley et al. (2011) for a detailed outline of the Bayesian changepoint framework and additional references can be found in Section 4.1.

2.6 Changes in Second Order Structure

Having focussed on changes in i.i.d. data sequences in the previous sections, in this section we consider a different setting. Specifically, we review the literature on detecting changes in the second order structure of a time series. We review contributions made using the following three approaches: a classical likelihood approach, an approximate likelihood approach and finally a nonparametric approach. Davis et al. (2006), Gombay (2008), Killick et al. (2013) and Fryzlewicz and Subba Rao (2014) all take a likelihood approach to detecting changes in second order structure. Below we briefly summarise each of these contributions.

The Auto-PARM approach of Davis et al. (2006) calculates the likelihood-based minimum description length (MDL) (Jorma, 1998) of an autoregressive process of order p. The basic idea of MDL is that the best-fitting model is the one than enables maximum compression of the data. The best fitting model, as decided by the MDL, is determined by optimizing some criterion. Davis et al. (2006) use a genetic algorithm to explore the search space of this optimization problem. This allows them to determine the number and location of the changes in the AR model efficiently.

Gombay (2008) also consider detecting changes in an autoregressive process. To do this they perform a hypothesis test for which the test statistics are based on the likelihood of the data. Gombay (2008)'s approach enables the identification of which parameters of the AR model have changed: the p AR coefficients, the mean and/or the variance of the white noise process. Davis et al. (2006)'s approach does not allow this, however Gombay (2008) can only detect a single change in the AR process.

Fryzlewicz and Subba Rao (2014) also use a likelihood approach to detecting changes in second order structure. They, however, consider multiple changepoints occurring in ARCH and GARCH processes. They use the binary segmentation algorithm to detect changes. Killick et al. (2013) also use binary segmentation in a likelihood framework. Their approach consists of modelling the likelihood of the wavelet spectrum of a *locally stationary wavelet process*.

An alternative approach to detecting changes in second order structure is to approximate the likelihood of the time series using the Whittle Likelihood (Whitle, 1951). In contrast to the classical likelihood approaches, analysis takes place in the frequency domain. This is because Whittle's likelihood approximates the likelihood of a time series in terms of its spectral density. Lavielle et al. (2000), Hsu and Kuan (2001), Yamaguchi (2011) and Yau and Davis (2012) all use Whittle's likelihood to detect changes in second order structure.

Lavielle et al. (2000) uses Whittle's pseudo-likelihood in a penalised cost function framework in order to detect changes in the spectral density of a time series. They test their approach on electroencephalogram (EEG) data. Alternatively, Hsu and Kuan (2001) consider macroeconomic time series. They propose a two step procedure in order to distinguish between the presence of long memory and changes in second order structure. It is only applicable when there is a single change. Yau and Davis (2012) are also interested in distinguishing between the presence of long memory and changes in second order structure. Yamaguchi (2011) is too interested in long memory, however they detect changes in the long memory parameter of an Autoregressive Fractionally Integrated moving Average (ARFIMA) process (Hosking, 1981).

A third approach to detecting changes in second order structure is a non-parametric one. For example, Giraitis et al. (1996) take an approach based upon Kolmogorov-Smirnov type tests. They perform a hypothesis test which can detect changes in

dependence in both short term and long term memory processes. In a very different approach Ombao et al. (2001) introduce a new basis called smooth localized complex exponential (SLEX) transforms to decompose a time series. Using this representation, they detect changes in second order structure using a non-parametric test statistic. They can however, only detect changes at dyadic points in time. Conversely, Cho and Fryzlewicz (2012) use the locally stationary wavelet (LSW) representation of a time series. In contrast to Killick et al. (2013), they model the the wavelet coefficients using a non-parametric test statistic.

Chapter 3

Changepoint Identification to Improve Forecasts

3.1 Introduction

Many economic and financial time series are subject to changepoints, see for example the systematic study performed by Stock and Watson (1996) and additional works such as Alogoskoufis and Smith (1991); Garcia et al. (1991); Bai and Perron (1998); Hendry and Clements (2000); Timmermann (2001); Pesaran and Timmermann (2002).

The causes for these changes in economic or financial time series could be attributed to things such as:

- changes in market sentiments or mechanisms;
- national or global recessions.

Consider, for example, Figure 3.1 which displays the United Kingdom's Gross Domestic Product (GDP) growth quarter on quarter. GDP is important as it enables policy makers and central banks to determine if the economy is contracting or expanding

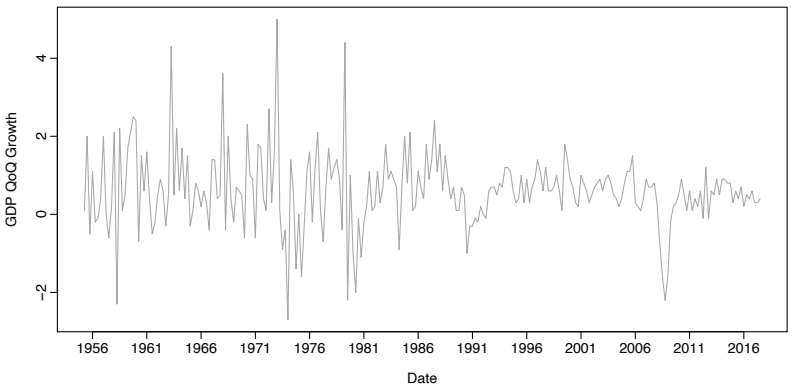


Figure 3.1: United Kingdom's Gross Domestic Product quarter on quarter growth

and if it needs a boost or restraint. In Figure 3.1 there are noticeable periods of time for which the data are behaving differently to one another. This raises questions such as, how much historical data should be used to build forecasting models, and should the model for GDP prior to a recession be different to the one used afterwards? Questions such as these are considered by Pesaran and Timmermann (2002); Clark and McCracken (2005); Elliott (2005) and in particular, Pesaran and Timmermann (2004) discuss and quantify the costs associated with ignoring changepoints when forecasting in an macroeconomic and financial setting.

In Pesaran and Timmermann (2002) the authors only use post-break data to estimate the forecasting model, and they estimate the location of the break to be the most recent changepoint which is obtained using a reversed CUSUM procedure (Brown et al., 1975). In further work, Pesaran and Timmermann (2007) propose that if the goal is to minimise the mean squared forecast error, then some pre-break data may be useful for model fitting. This so called "trade off window" approach of Pesaran and Timmermann (2007), which uses both pre- and post-break data, is motivated by the trade off between bias and forecast error variance. Providing that the structural break is not too large, by introducing more observations, they are reducing variance at the cost of possible bias which may overall result in improved forecasts.

In this chapter we propose an approach to forecasting using changepoints which uses only post-break data to estimate the time series model we use to produce forecasts. In order to detect the changepoints, we use a penalised cost function approach which solves a constrained minimisation problem exactly. This approach allows us to control the trade off between bias and forecast error variance from within the changepoint framework.

The methodology we propose also takes into account the forecasting process as a whole. Often when practitioners construct a model to produce forecasts, they do so in multiple stages. The first of these involves a preprocessing step, this may consist of identifying outliers or anomalies within the data, so in Section 3.2 we show how the changepoint methodology can be implemented in the preprocessing stages of forecasting. In particular, we illustrate how the changepoint approach can be used to identify level shifts and incorporate these into the model. A second stage of the forecasting process is identifying the best model for the data. Hence, in Section 3.3 we describe a competing approach to using changepoints to improve forecasts which incorporates changepoint detection into the model fitting stage of forecasting.

The structure of this chapter is as follows. In Sections 3.2 and 3.3 we describe our two approaches to using changepoints to improve forecasts. In Section 3.4 we then compare each of these methods with a stationary forecasting model and finally in Section 3.5 we test our methods on the UK GDP data in Figure 3.1.

3.2 Preprocessing

Typically, when constructing a model to use for forecasting, it is common to perform some sort of preprocessing. We propose here that testing for changepoints in the historical data should form a part of this preprocessing step. This is our first approach

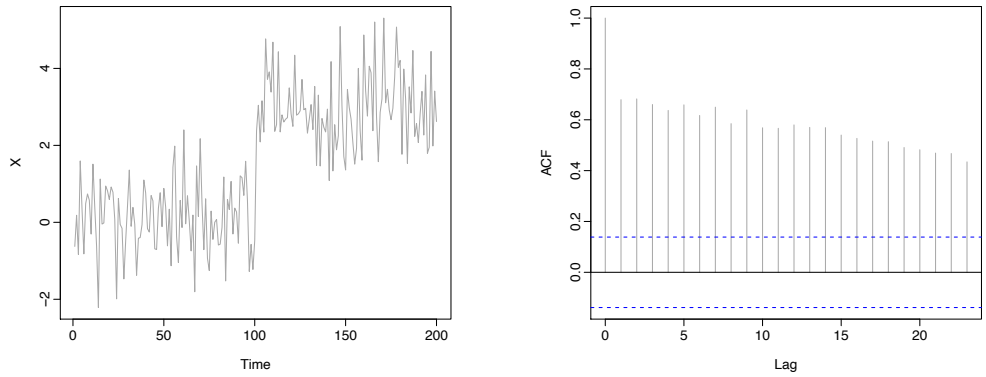


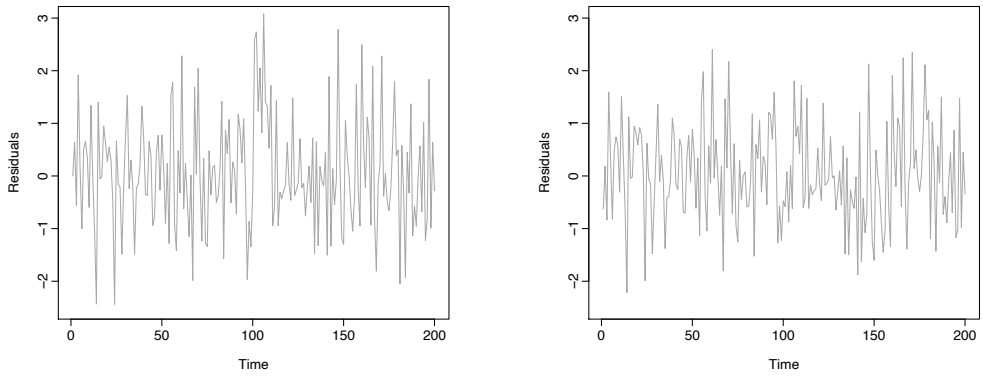
Figure 3.2: (a) An i.i.d. Gaussian time series X_t with change in mean and (b) its autocovariance function (ACF).

to using changepoints in forecasting.

By way of example, consider a time series of independent identically distributed Gaussian observations of length 200 exhibiting a change in mean from zero to three at time 100. Figure 3.2 shows one realisation of this process together with the autocorrelation of the time series. It is evident that despite the observations being i.i.d., autocorrelation is present. This is an example of a lower order structure change affecting the estimates of higher order structures in a time series, and has also been observed by Norwood and Killick (2018).

If we naively used the `forecast` package (Hyndman et al., 2007) to fit a model to this data, we would typically first difference the data and then fit a time series model to it. By analysing the autocorrelation function (ACF) and partial autocorrelation function (PACF) of the differenced data, the appropriate model is an ARIMA(0,1,1) model. Figure 3.3a shows the residual errors given by an ARIMA(0,1,1) model fit to the data. We can see generally larger residuals around the location of the change in mean.

Instead of differencing the data, one approach we can take is to detect the change in mean during a preprocessing step, and then incorporate it explicitly into our forecasting model as a dummy variable. When we do so, we correctly identify that there is



(a) (b)

Figure 3.3: The residuals for Y_t when fitted with (a) an ARIMA(0,1,1) and (b) an ARIMA(0,0,0) model with a regressor.

no autocorrelation in the process, and the most appropriate ARIMA model is white noise. Figure 3.3b shows the residual errors for this model fit.

If we want to be robust to the presence of changes in mean, whilst correctly modelling the autocorrelation structure of the data, it is important to consider changes in mean as a part of the preprocessing step of building a model used to forecast. As such, here we outline a changepoint preprocessing method which detects changes in mean and incorporates these into the time series model.

3.2.1 The Model

As an introduction to building our model for forecasting, we first test for any changes in mean. To do this, we take a penalized likelihood approach to changepoint detection, as described in Section 2.4. In this setting, we replace the cost function $\mathcal{C}(\cdot)$, in equation (2.4), with twice the negative log-likelihood for a Gaussian distribution with common variance and segment specific mean.

The second component of equation (2.4) is the penalty used to prevent over-fitting to the mean of the data. We are assuming independent Gaussian observations. However,

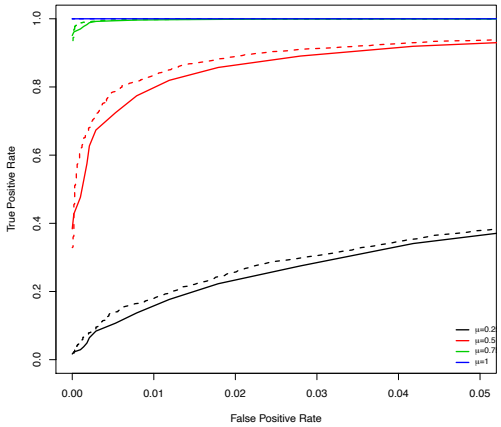


Figure 3.4: Receiver operating curves for (solid line) i.i.d. normal data and (dashed line) correlated normal data (Autoregressive data with parameter 0.8) which exhibits a change in mean from zero to a new mean level μ.

in a forecasting setting our data will most likely contain autocorrelation structure. Despite this, the algorithm is still effective at locating changes in mean (Lavielle and Moulines, 2000).

Figure 3.4 shows the receiver operating curves (ROCs) for detecting a change in mean both with and without autocorrelation. In this simple example we can see that when there is autocorrelation present, we have increased power to detect changes. However, this results in an increased false positive rate. This inflation of the type I error rate can also be seen in Lund et al. (2007). To remedy this, practically we inflate the standard penalty chosen as suggested by Lavielle and Moulines (2000).

3.2.2 Forecasting

Once we have detected changes in mean, we can incorporate them into our time series model using external regressors. Algorithm 1 provides pseudo code for forming a matrix of regressors based upon the changepoint locations. Having attained a matrix of regressors representing the level changes, we can fit a multiple linear regression

between these and the data in order to remove the effect of the level shifts. This can be done independently of the time series model, in which case we would fit the time series model to the residuals of the linear regression, or it can be done as part of the modelling process.

Algorithm 1: Incorporating mean changes into forecasts

Data: Time series $\mathrm{Y} = (y_1, \ldots, y_n)$, $x_i \in \mathbb{R}$
Result: Matrix of external regressors representing mean changes to be used for both model fitting and forecasting.

Let $\tau_0 = 0$ and $\tau_{m+1} = m$ and detect changes in mean τ_j for $j = 1, \ldots, m$.;
if $m = 0$ **then**
  $V = NULL$;
  $V^{out} = NULL$;
else
  $V \in \mathbb{R}^{m \times n}$;
  for $j \in [1, \ldots, m]$ **do**
    $v_{j,i} = \begin{cases} 1 & i \in (\tau_{j-1}, \tau_j], \\ 0 & \text{otherwise.} \end{cases}$
  end
  $V^{out} = 0_{m \times 1}$;
end
return V, V^{out}

In Section 3.4 we test this approach in a simulation study. In the next section, we turn our attention to an alternative approach to using changepoints to improve forecasts.

3.3 Modelling

For the purpose of forecasting, we wish to detect statistically significant changes in the model we are using to produce forecasts. Thus, in order to improve forecasts, we propose to use a cost function, $\mathcal{C}(\cdot)$, based upon the log-likelihood of our time series model. In the following, we describe this for the case of using an autoregressive moving average (ARMA) model for forecasting our time series. For an overview of the use of ARMA models in time series, we refer the reader to Shumway and Stoffer

(2000).

Suppose the time series we are trying to forecast, $\{y_t\}_{t=1,\ldots,n}$ is not stationary. To model this non-stationarity, we can segment the data into stationary autoregressive moving average (ARMA) processes. Let the i^{th} segment of the series, $y_{\tau_{i-1}+1:\tau_i}$, be modelled by the ARMA($\mathrm{p_i}, \mathrm{q_i}$) process

$$y_{\tau_{i-1}:\tau_i} = \mu_i + \frac{\theta_i(B)}{\phi_i(B)}\epsilon_{t,i},$$

where $\phi_i(B)$ is the autoregressive operator and $\theta_i(B)$ is the moving average operator, each represented as a polynomial in the backwards shift operator given as

$$\phi_i(B) = 1 - \phi_{i,1}B - \ldots - \phi_{i,p_i}B^{p_i},$$
$$\theta_i(B) = 1 + \theta_{i,1}B + \ldots + \theta_{i,q_i}B^{q_i},$$

and the noise process $\epsilon_{t,i}$ is i.i.d. with mean zero and variance σ_i^2. Note that as well as allowing both the order of the ARMA model to change, and the coefficients of the fitted model, we are also allowing for a change in mean level to occur by the inclusion of μ_i.

It is often the case that our time series will also have some seasonality structure with seasonal cycle of length f. For example, the GDP data in Figure 3.1 is quarterly data and we may wish to model this cyclic variation and allow for changes in the seasonality structure. In this instance, we can model $y_{\tau_{i-1}+1:\tau_i}$ as a multiplicative seasonal autoregressive moving average process, denoted ARMA($\mathrm{p_i}, \mathrm{q_i}$) $\times$ ($\mathrm{P_i}, \mathrm{Q_i}$)$_\mathrm{f}$, and write

$$y_{\tau_{i-1}:\tau_i} = \mu_i + \frac{\theta_i(B)\Theta_i(B^f)}{\phi_i(B)\Phi_i(B^f)}\epsilon_{t,i}, \tag{3.1}$$

where $\Theta_i(B^f)$ and $\Phi_i(B^f)$ are the seasonal moving average and autoregressive opera-

tors, respectively, given by

$$\Phi_i(B) = 1 - \Phi_{i,1}B^f - \ldots - \Phi_{i,p_i}B^{fP_i},$$
$$\Theta_i(B) = 1 + \Theta_{i,1}B^f + \ldots + \Theta_{i,q_i}B^{fQ_i}.$$

In addition to exclusively modelling the response time series, it may also be necessary to include external regressors into the model. In this situation we model the i^{th} segment of the series as linear regression model with seasonal ARMA errors. In this case we have

$$y_t = \beta_{0,i} + \beta_{1,i}x_{1,t} + \ldots + \beta_{k,i}x_{k,t} + r_{t,i}, \quad \tau_{i-1} < t \leq \tau_j,$$

where the linear regression residuals follow a seasonal ARMA process as in equation (3.3) and $x = (x_1, \ldots, x_k)$ are the explanatory variables. When the model is estimated, it is important to remember that we minimize the sum of squared values $\epsilon_{t,i}$, and not the $r_{t,i}$.

Having specified the model for each of the segments of our data, we can detect the locations of changes in the regression model for the time series by incorporating twice the negative log-likelihood of the model into the optimisation problem in equation (2.4). Appendix 3.A outlines a procedure for doing this in practice.

3.3.1 Forecasting

Once we have detected changes in the model we are using to forecast, we then forecast the time series based on the most recent segment of data using the model for that segment. As a consequence of this approach, once a changepoint has occurred, we are deeming pre-change data uninformative. Recall, from Chapter 2, that when detecting

multiple changepoints we impose a minimum segment length, g, such that $\tau_{i+1} - \tau_i \geq g \geq 2$. It important that our minimum segment length is not set so small such we are producing out-of-sample forecasts based only on a small amount of data. In particular, if the data has seasonality, then we must allow enough observations in a segment to estimate this seasonality. Also, the longer the minimum segment length, the more time we have to wait to detect a change. Consequently, we could be fitting an incorrect model to the last segment of the data therefore introducing bias into our model. In addition to this, penalty choice is important because it allow us to control the sensitivity of the changepoint algorithm, if we set it high, then we are only concerned with macro changes that occur in the data, if we set it low, then we wish to detect more changes.

The combination of penalty and minimum segment length can have a large influence on the detected changepoint locations and hence the window we are using to estimate our forecasting model. The combination of these two allows us to control the trade off between the bias and variance of our forecasts. As such, in practice, one could compare, or combine, multiple forecasting models based upon the different segmentations obtained when changing the combination of minimum segment length and penalty. In the next section, we test the performance of the methodology described here in a simulation study.

3.4 Simulation Study

In this simulation study we test the performance of using changepoints to improve forecasts. We compare the following three models:

- **M1:** A stationary S/ARIMA model;
- **M2:** The preprocessing approach described in Section 3.2;

- **M3:** A piecewise S/ARMA model.

In order to only access the relative gain from detecting changepoints, a S/AR(I)MA model is used in all three models. To estimate this model, we use the `forecast::auto.arima` function (Hyndman et al., 2007). This could be replaced with an alternative time series model, for example an exponential smoothing model.

To detect changes in mean for model M2 we use the `changepoint::cpt.mean` function (Killick and Eckley, 2014). This function implements the PELT algorithm for a change in mean under the assumption of Gaussian data. Note that the `changepoint::cpt.mean` function assumes a variance of one. This means that the data should be pre-scaled to variance one prior to detecting changes in mean.

In order to fit model M3, we adopt the approach outlined in Appendix 3.A.

In each instance we simulate 500 realisations of the models and report a selection of commonly used in-sample and out-of-sample performance metrics:

- Mean Error (ME);
- Root Mean Squared Error (RMSE);
- Mean Absolute Error (MAE);
- Mean Percentage Error (MPE);
- Mean Absolute Scaled Error [1];
- The autocorrelation at lag 1 of the residual errors of the model (ACF1).

Each of these metrics for a model can be attained using the `forecast::accuracy` function in R, providing convenient model evaluation for the user. These are reported in Table 3.1 for the training (in-sample) set and in Table 3.2 for the test (out-of-sample) set.

[1]MASE calculation is scaled using MAE of training set naive forecasts for non-seasonal time series, training set seasonal naive forecasts for seasonal time series and training set mean forecasts for non-time series data (Hyndman et al., 2007).

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Model A							
M1	0.0016	**1.0060**	0.8037	**47.0690**	346.8433	0.9177	0.0023
M2	**-0.0000**	0.9914	**0.7916**	49.0590	**336.7375**	**0.9040**	**0.0022**
M3	0.0016	**1.0060**	0.8037	**47.0690**	346.8433	0.9177	0.0023
Model B							
M1	0.0333	1.0255	0.8175	**-1.3018**	264.2667	0.9309	0.0029
M2	-0.0029	**1.0002**	**0.7984**	-13.5072	250.8929	**0.9089**	**0.0016**
M3	**-0.0008**	1.0017	0.7985	-12.6762	**203.3246**	0.9101	0.0048
Model C							
M1	0.0003	1.0106	0.8055	52.8808	296.6232	0.8728	**0.0012**
M2	0.0017	0.9979	0.7951	**39.7553**	301.7013	0.8617	0.0016
M3	**-0.0000**	**0.9966**	**0.7938**	50.4126	**263.5910**	**0.8392**	0.0039
Model D							
M1	0.0013	1.0400	0.8285	89.0696	**293.5366**	0.7906	0.0046
M2	-0.0008	1.0274	0.8186	**86.7520**	299.5224	**0.7812**	0.0029
M3	**0.0001**	**1.0046**	**0.8034**	70.5495	305.7549	0.8053	**0.0020**
Model E							
M1	**-0.0001**	1.0206	0.8124	**-10.0230**	665.4235	0.5624	-0.0005
M2	-0.0007	**1.0113**	**0.8056**	-31.3873	679.9682	**0.5569**	**-0.0004**
M3	0.0037	1.0374	0.8261	-191.9323	**604.2020**	0.5932	-0.0005
Model F							
M1	-0.0019	1.7558	1.3751	6461.9811	7115.0305	0.5908	**0.0013**
M2	**-0.0011**	1.7444	1.3670	6464.0948	7118.3103	**0.5855**	**0.0013**
M3	0.0167	**1.2740**	**1.0198**	**41.7718**	**239.5152**	0.8775	0.0226
Model G							
M1	0.0030	2.3865	1.8896	34.7642	265.1804	0.5545	**0.0029**
M2	**-0.0003**	2.2755	1.7986	**33.3716**	255.8116	0.5286	0.0033
M3	-0.0087	**1.8575**	**1.4690**	34.4501	**191.2260**	**0.4271**	-0.0039

Table 3.1: Mean Error, Root Mean Square Error, Mean Absolute Error, Mean Percentage Error, Mean Absolute Square Error and the autocorrelation at lag 1, to four decimal places, for the in-sample forecasts for 500 realisations of Models (A)-(G) using methods M1 (stationary model), M2 (stationary model with level changes) and M3 (piecewise stationary model).

	ME	RMSE	MAE	MPE	MAPE	MASE
Model A						
M1	**-0.1563**	**1.1671**	**1.0282**	113.3594	**169.0906**	**1.1740**
M2	-0.2580	1.3524	1.1853	**130.0618**	184.9175	1.3521
M3	**-0.1563**	**1.1671**	**1.0282**	113.3594	**169.0906**	**1.1740**
Model B						
M1	-0.1769	**1.1521**	**1.0119**	**-17.0005**	221.3590	**1.1517**
M2	-0.2434	1.3917	1.2360	-202.3013	456.4171	1.4062
M3	**-0.1411**	1.1583	1.0183	-17.8884	**206.2740**	1.1602
Model C						
M1	0.1661	1.0751	0.9265	68.4680	188.4346	**1.0055**
M2	**0.1211**	1.1043	0.9442	110.8610	196.6434	1.0249
M3	0.1711	**1.0643**	**0.9143**	**68.0920**	**156.3316**	0.9695
Model D						
M1	**0.0313**	0.9381	0.7881	102.6425	160.8289	**0.7535**
M2	0.0325	0.9767	0.8260	129.5255	207.3461	0.7888
M3	0.0354	**0.9100**	**0.7644**	**76.8555**	**133.8241**	0.7679
Model E						
M1	**-0.0711**	1.1381	0.9713	90.1890	216.4954	**0.6714**
M2	-0.1162	1.1723	1.0029	100.3486	244.7225	0.6941
M3	-0.0752	**1.1093**	**0.9436**	**86.3775**	**201.4708**	0.6774
Model F						
M1	-0.0669	1.8831	1.6579	117.3606	278.6023	**0.7133**
M2	-0.1936	2.1786	1.9157	178.7814	263.3264	0.8164
M3	**-0.0197**	**1.7353**	**1.5182**	**90.9365**	**194.1363**	1.3165
Model G						
M1	-0.1240	2.7002	2.3515	81.9197	**142.3822**	0.6852
M2	**0.0237**	6.0104	5.3974	**-54.1455**	460.6030	1.5817
M3	-0.2269	**2.1403**	**1.8835**	61.0397	185.1860	**0.5466**

Table 3.2: Mean Error, Root Mean Square Error, Mean Absolute Error, Mean Percentage Error, Mean Absolute Square Error and the autocorrelation at lag 1, to four decimal places, for the out-of-sample forecasts for 500 realisations of Models (A)-(G) using methods M1 (stationary model), M2 (stationary model with level changes) and M3 (piecewise stationary model).

We simulate 500 realisations from the following scenarios, in which the residual process is given by $\epsilon_t \sim \mathcal{N}(0, 1)$.

(a) Stationary AR(2) model with no seasonal components. This scenario is designed to asses the method when there are no changepoints. Specifically, for this model, we simulate from

$$Y_t = 0.8Y_{t-1} - 0.2Y_{t-2} + \epsilon_t, \qquad 1 \leq t \leq 512.$$

For scenario (a) the stationary model with mean level changes, M2, produces an overall better in-sample fit than the other two models. For the test set, the stationary model (M1) and the piecewise stationary model (M2) produce better out-of-sample forecasts. Model M2 in this scenario is most likely to over-fit the data, producing a better in-sample fit but consequently producing worse forecasts. This is because the presence of autocorrelation can induce features which resemble changes in mean, a feature previously noted in the literature by Beaulieu et al. (2012). Figure 3.5 shows a single realisation from scenario (a) along with detected changes in mean. Despite inflating the penalty to account for the presence of autocorrelation, changepoints are still detected. Consequently, model M1 over-fits to the level of the time series, and as a result, will misspecify the autoregressive parameters of the model.

Tables 3.2 and 3.1 for models M1 and M2 are the same to four decimal places, this suggests a low false positive rate for the change in ARMA model.

(b) Stationary AR(2) model with no seasonal components and a change in level. This scenario is designed to asses the method when there are no changes in second order structure but there is a change in level. Specifically, for this model, we

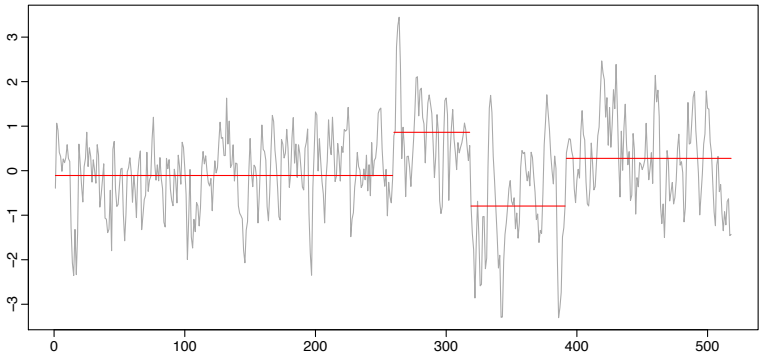


Figure 3.5: A realisation Y_t from scenario (a) with detected changes in mean. We can see that although there are no 'true' changes in mean, the autocorrelation causes them to be detected.

simulate from

$$Y_t = \begin{cases} 0.8Y_{t-1} - 0.2Y_{t-2} + \epsilon_t & 1 \leq t \leq 256 \\ 2 + 0.8Y_{t-1} - 0.2Y_{t-2} + \epsilon_t & 256 \leq t \leq 512 \end{cases}$$

Overall for scenario (b) model M2 produces a better fit to the training set, this is expected as it is the most appropriate method to use for the scenario. Out-of-sample however, model M1 produces the best forecasts. In this case we expect M3 to perform poorest because it should detect a change in the level of the AR model and then deem pre-break information uninformative. As a result, the autoregressive coefficients will be estimated using only a portion of the data.

(c) A piecewise stationary AR(2) model with changing coefficients. Specifically, for this model, we simulate from

$$Y_t = \begin{cases} 0.8Y_{t-1} - 0.2Y_{t-2} + \epsilon_t & 1 \leq t \leq 256 \\ 0.5Y_{t-1} - 0.1Y_{t-2} + \epsilon_t & 256 \leq t \leq 512 \end{cases}$$

For the piecewise stationary model in scenario (c), method M3 produces a better in-

sample fit to the data, again we expect this because this model is most in line with the nature of the behaviour of the time series. In this instance, the results for both the training and test set support the use of model M3.

(d) A piecewise stationary AR model which changes from a third order to a first order process with a short segment at the beginning of the time series. Specifically, for this model, we simulate from

$$Y_t = \begin{cases} 0.1Y_{t-1} - 0.6Y_{t-2} - 0.3Y_{t-3} + \epsilon_t & 1 \leq t \leq 50 \\ 0.3Y_{t-1} + \epsilon_t & 51 \leq t \leq 512 \end{cases}$$

In scenario (d) both the order and the coefficients of the AR model change and model M3, the piecewise stationary model, can capture this the best producing better in-sample results, it also achieves better out-of-sample forecasts.

(e) A piecewise stationary AR model which changes from a third order to a first order process with a short segment at the end of the time series. Specifically, for this model, we simulate from

$$Y_t = \begin{cases} 0.1Y_{t-1} - 0.6Y_{t-2} - 0.3Y_{t-3} + \epsilon_t & 1 \leq t \leq 462 \\ 0.3Y_{t-1} + \epsilon_t & 462 \leq t \leq 512 \end{cases}$$

In scenario (e) we again have a change in both the order and coefficients of the AR model, however in contrast to scenario (d), the change occurs at the end of the time series. Although the piecewise model M3 produces better out-of-sample forecasts than the stationary model M1, we can see in Table 3.2 that the results differ less than in scenario (d). This is expected because model (c) has a longer segment which will produce a better model fit with less variability and thus improved forecasts. For the in-sample errors in Table 3.1 we can see that for the training set, model M2 is actually

providing a better fit to the data, which may be a consequence of over-fitting.

(f) A piecewise stationary SAR model, frequency 4, whose AR seasonality component has a change in coefficients. Specifically, for this model, we simulate from

$$Y_t = \begin{cases} 0.9Y_{t-1} - 0.2Y_{t-2} - 0.9Y_{t-4} + \epsilon_t & 1 \leq t \leq 256 \\ 0.9Y_{t-1} - 0.2Y_{t-2} - 0.2Y_{t-4} + \epsilon_t & 256 \leq t \leq 512 \end{cases}$$

Here we extend the scenarios to include seasonality, in this case, we have a seasonal frequency of four corresponding to one quarter in practice. The GDP data in Figure 3.1 is quarterly. In this case model M3 produces the best in-sample results and model M1 produces the poorest.

(g) A piecewise stationary SAR model whose AR seasonality component has a change in order. Specifically, for this model, we simulate from

$$Y_t = \begin{cases} 0.9Y_{t-1} - 0.2Y_{t-2} - 0.9Y_{t-4} - 0.8Y_{t-5} + \epsilon_t & 1 \leq t \leq 256 \\ 0.9Y_{t-1} - 0.2Y_{t-2} - 0.9Y_{t-4} + \epsilon_t & 256 \leq t \leq 512 \end{cases}$$

In scenario (g) the seasonality component of the model exhibits a change in order. Method M3 captures this the best in-sample and out-of-sample, with the stationary model M1 producing the poorest results.

Overall we can conclude than the inclusion of changepoints in the modelling stages of forecasting produces better results. In particular, when the time series exhibits changes in its seasonal structure, or changes in order, then the piecewise stationary approach to forecasting out-performs a stationary approach.

At times, the stationary approach to forecasting including changes in mean level can over fit the data, however as those changes begin to occur in higher order structures of the time series, for example in scenario (g), this approach produces better out-of-

sample forecasts than the stationary approach alone.

In the following, we consider forecasting the UK's GDP using each of the methods.

3.5 Application to the United Kingdom's Gross Domestic Product

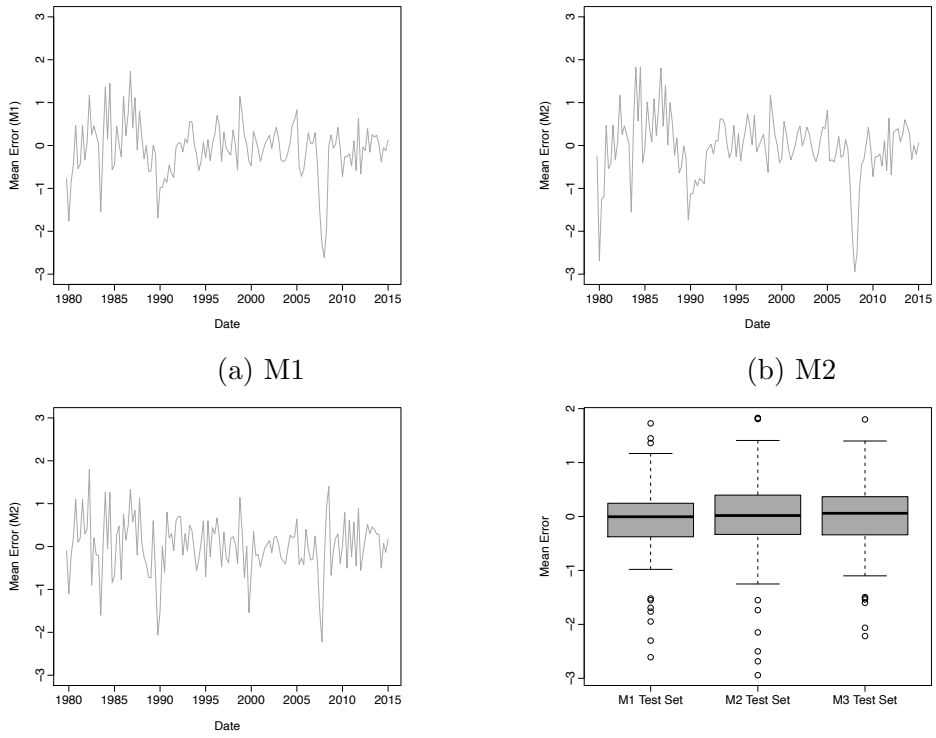
Figure 3.1 shows the UK's Gross Domestic Product quarter on quarter growth for the period from Q2 1955 to Q3 2017. We want to test the performance of models M1, M2 and M3. In order to do this, we set the following parameters:

- For model M2, we set a minimum segment length of $g = 2$. This allows for changes in mean which are of at least length two. The penalty we use is a scaled BIC ($6 \log n$).
- For model M3, we set a minimum segment length of $g = 8$, i.e. two years. This allows enough observation to fit a seasonal model. The penalty we use is the Modified Bayes Information Criteria (MBIC) (Zhang and Siegmund, 2007).

The MBIC penalty accounts for the lengths of the segments and encourages changes to be distributed evenly across the dataset. This is useful for forecasting as we want to discourage small segment lengths in order to reduce the error variance.

We fit each of the S/AR(I)MA models using the same approaches as in Section 3.4, i.e. using the `forecast::auto.arima` function (Hyndman et al., 2007). We fit a model of the form ARMA$(p, q) \times (P, Q)_f$. We do not allow p, q, P or Q to exceed three. In addition to this, we set the seasonality, f, to be four.

In order to assess the performance of each of the methods, we perform an extending window estimation. To begin, we fix an initial estimation period from the start of the GDP data up until Q1 1980. Then we forecast 4 steps ahead (one year) and calculate



(a) M1 (b) M2

(c) M3 (d) Box Plot Comparison of the ME

Figure 3.6: The Mean Error for a four step ahead forecast with model estimation period starting at Q2 1955 and ending as indicated by the x-axis of the plots. Figures (a) - (c) show the expanding window Mean Error's of the forecast for models M1, M2 and M3 respectively, and figure (d) compares the Mean Error's for each of the models.

the mean error of the forecast. Having done this, we extend the estimation period by one time step and again forecast a year ahead and calculate the mean error. We iterate this procedure up until Q3 2016 to produce an expanding window forecast for GDP.

Figure 3.6 shows the results for the expanding window forecasts. Each of the models have a similar average mean error for the forecasts. However we can clearly see that model M3, the piecewise model, is capturing the behaviour of GDP better as the mean errors of the forecast look most like white noise. If we look at the box plot in Figure 3.6d we can see that model M3 has less extreme forecast errors. This is a consequence of the model's improved performance around the recession.

3.6 Discussion and Conclusion

In this chapter we have described two methods for using changepoints to improve forecasts. We have shown that often the incorporation of changepoints into forecasting produces improved results. In addition to this, we have shown that forecasts can be based on less historical data, whilst still producing reasonable results. As data is becomingly large scale, the need for reducing the amount of data used to fit models is becoming increasingly important, and questions such as "how much of my data is relevant for forecasting" can be answered using changepoint methodology.

Our modelling framework is flexible. We can produce variants on our model by altering the minimum segment length and penalty choice, and we can also adapt our methodology for any time series model provided we can define the cost function for a segment. The choice of minimum segment length and penalty together, give us control over the trade off between bias and forecast error variance.

It may be the case, that in practice, the cost function for a segment is hard to define. In such a case, the preprocessing methodology we present could instead be used in a post-processing step by applying the methodology to the residual errors of the forecast, such an approach can be seen in Beaulieu and Killick (2018). Finally, we applied our methodology to forecasting GDP and saw improved performance around the time of the recession. In order to improve this forecasting model, we could use explanatory variables for GDP, and also test for changes in their relationship to GDP.

3.A Appendix

In Section 3.3 we described modelling each segment of our data as an regression model with seasonal ARMA errors. In order to build this into the penalised cost framework

for changepoint detection, we need to estimate the log-likelihood of the model for each segment. In the following, we outline one practical approach to doing this.

To attain the cost function for each segment of our piecewise regression model, we can use the `forecast::auto.arima` function (Hyndman et al., 2007). This function is a wrapper for the `stats::arima` function which fits autoregressive integrated moving average (ARIMA) models by computing an exact likelihood using a state-space representation of the ARIMA process. It returns the best ARIMA model according to either Akaike's information criterion (AIC, (Akaike, 1974)), the corrected AIC (AICc, (Kletting and Glatting, 2009)) or the Schwarz information criterion (SIC, (Schwarz et al., 1978)) value. It implements a stepwise model selection algorithm as outlined in (Hyndman et al., 2007) where the default method for selecting seasonal differences is based on an estimate of the seasonal strength (Wang et al., 2006). We refer the reader to Hyndman et al. (2007) for further details.

The use of the `forecast::auto.arima` function allows us the flexibility to have changing model orders and coefficients, seasonal components, mean and regressors. These components can be incorporated in order to attain the cost function for each segment of our piecewise regression model. Having performed model selection using the `forecast::auto.arima` function, we can use the `stats::logLik` function to extract the log-likelihood for use in equation (2.4).

In practice, the fitting of the regression model can be done using any method/program available. As long as the log-likelihood or the Bayesian MAP can be determined for a segment, it can be easily incorporated into the penalised cost framework for changepoint detection. In addition, the `forecast::auto.arima` function could be replaced with a different time series modelling function, for example the `forecast::ets` function could be used to instead model the time series using an exponential state space model (Hyndman et al., 2002).

Chapter 4

Predicting Future Changepoints

4.1 Introduction

In Chapter 3 we considered the problem of forecasting in the presence of changepoints. In particular, we highlighted that many time series, especially economic data, are subject to changepoints and it is important to account for these during the forecasting process.

Consider again the UK's Gross Domestic Product (GDP) quarter on quarter growth in Figure 4.1. It is important that we can detect changepoints and produce forecasts

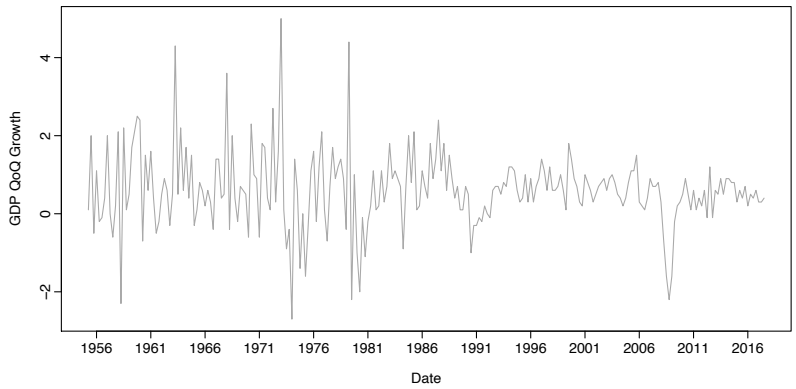


Figure 4.1: United Kingdom's Gross Domestic Product quarter on quarter growth.

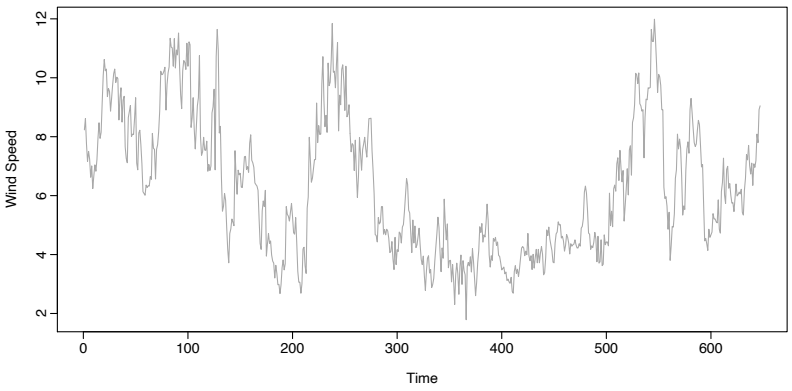


Figure 4.2: Wind speed in a region.

based on these for GDP. Methods of detecting and forecasting in the presence of changepoints are well established, however there exists little active research into the prediction of changepoints. However, as Hirade and Yoshizumi (2012) and Jiang et al. (2013) have identified, often from an applied perspective there is a need to predict the existence of changepoints. Some examples include:

- Microeconomics - predicting events such as recessions, or sudden increases in unemployment, see for example Figure 4.1;
- Technology - predicting, for example, a change from acceleration to deceleration in an hybrid car, enabling proactive control of the vehicle;
- Environmental - predicting changes in wind speed to more efficiently control wind turbines, see for example Figure 4.2.

Consequently, given the well established forecasting and changepoint literature, and worthy applications, the prediction of changepoints is potentially a very fruitful area of research.

The literature concerning forecasting of changepoints is limited. Current contributions tend to allow changepoints to occur out of sample, but future changepoint positions are not explicitly located. The models adopted are primarily Bayesian and in general the changepoint process is modelled as either a Geometric (Hashem Pesaran et al.,

2006), Bernoulli (McCulloch and Tsay, 1993; Maheu and Gordon, 2008; Jochmann et al., 2010; Geweke and Jiang, 2011) or Poisson process (Koop and Potter, 2007; Maheu and Song, 2014).

Each of the Bayesian approaches assume that the changepoints are a part of the data generating process, and so information concerning future changepoints is contained in the time series itself. However, in a model where we only predict changepoints, there would not, in general, be sufficient data to take this approach. Consequently, a very large number of changepoints would have had to have to occurred. This perhaps advocates the consideration of explanatory variables as an early warning that a change is likely to occur. Below we briefly review recent contributions to the literature in this area.

We start by considering the switching indicator model of McCulloch and Tsay (1993). Here a Bernoulli probability of an out of sample changepoint is allowed to depend on explanatory variables using a probit model. In a similar fashion, the non-Bayesian model of Giacomini and Rossi (2009) regresses what they define to be the "surprise loss function" on a set of explanatory variables. Neither of these models predict the changepoint locations out of sample, however each are aware that external variables will impact the probability of a future change. Hirade and Yoshizumi (2012) use machine learning techniques to predict future changepoints. They assume that the causes for changepoints can be characterized by the time interval between a changepoint and its symptom.

Adopting the use of explanatory variables, we develop a model which uses the time delay between a changepoint in an explanatory variable (input) and a future change in the variable of interest (response) in order to predict changepoints. In Section 4.2 we introduce our changepoint prediction methodology. Section 4.3 explains how to predict future changepoints given the model in Section 4.2. In Section 4.4 we

then conduct a simulation study to test our proposed method and in Section 4.5 we introduce an extension to our original methodology which considers the presence of changes in second order structure in the explanatory variable. Finally, in Section 4.6 we present an application of our method in predicting changepoints in vehicle speed data.

4.2 Changepoint Prediction Methodology

Suppose that the time series we wish to predict changes in, $y_{1:n}$, exhibits m_Y historical changes in mean or variance with positions $\tau^Y = (\tau_1^Y, \ldots, \tau_{m_Y}^Y)$. Each changepoint position, τ_i^Y, is an integer between 1 and $n-1$ and we define: $\tau_0^Y = 0$.

In order to predict future changes in y_t, at times $t > n$, we propose to use the relationship between y_t, and some explanatory time series x_t. One way we can relate an explanatory series x_t to a response series y_t is using a *transfer function* model.

Definition 4.2.1. *A transfer function model relates an explanatory series x_t, to the response series y_t using the following*

$$y_t = \nu(B)x_{t-d} + \epsilon_t = \sum_{i=0}^{s} \delta_i B^i x_{t-d} + \epsilon_t. \tag{4.1}$$

Here $\nu(B)$ is a polynomial in the backward shift operator, B, $\{\epsilon_t\}$ are the set of correlated observation errors, and $d \in \mathbb{Z}$ is the time delay. It is assumed that the explanatory time series x_t and the noise process ϵ_t are both stationary and mutually independent.

We can generalise the transfer function model in equation (4.2.1) to be a lagged regression model with correlated observation errors.

Definition 4.2.2. *A lagged regression model relates an explanatory series x_t, to the*

response series y_t using the following expression

$$y_t = \nu(B)x_t + \epsilon_t = \sum_{i=0}^{D} \delta_i B^i x_t + \epsilon_t. \tag{4.2}$$

Here $\nu(B)$ is a polynomial in the backward shift operator, B, and $\{n_t\}$ are the set of correlated observation errors. It is assumed that the explanatory time series x_t and the noise process ϵ_t are both stationary and mutually independent.

For notational convenience, let us define the following ordered sequence

$$S = \{i|\delta_i \neq 0\}, i = 0, \ldots, D, \tag{4.3}$$

to be the indices of the non zero coefficients in (4.2.2). Then the first element in this sequence is the delay, d, in equation (4.2.1), and the last is the parameter D in equation (4.2.2). The parameter s in equation (4.2.1) can be expressed as: $s = D - d$.

Given the above, our changepoint prediction setting is the following. Suppose that a response series y_t is related to an explanatory series x_t by equation (4.2.2). Further, suppose that x_t exhibits m_X changes in mean with positions $\tau^X = (\tau_1^X, \ldots, \tau_{m_X}^X)$. Then, our goal is to predict the changes, τ^Y, that will occur in y_t. Our challenge therefore is to be able to estimate the elements of the set S (4.2) and also the delay d in equation (4.2.1). In doing so, we will be able to estimate how much time we must wait until a change occurring in the explanatory series, will induce a change in the response series. We outline our approach to achieving this below.

4.2.1 Estimating d and S

The changepoint prediction problem described above can be posed as a time delay estimation problem, and also relies upon estimating the elements of the set S. The

estimation of these require us to accurately describe the cross-correlation structure between the impulse and response series. To achieve this we must first filter the two time series. This ensures that we can identify the relationship between the two series. In the literature, this is often called a pre-whitening procedure. Here we adopt the method of Box et al. (2013) which is most commonly used in time series analysis.

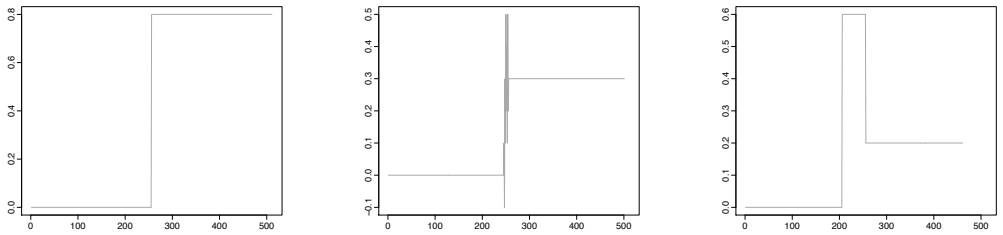
The first step of this method is remove any auto-correlation from the impulse series x_t. This can be achieved by fitting an autoregressive moving average (ARMA) model to the time series and taking the residuals of the model. The second step is to filter the response series, y_t, by the same transformation we applied to x_t. This is required in order to preserve the relationship in equation (4.2.1). Once the impulse series has been pre-whitened and the response series has been filtered, the cross-correlation between these two will reveal the form of the polynomial $\nu(B)$ in equation (4.2.1) (Shumway and Stoffer, 2000, Chapter 5).

Once we have pre-whitened our input series x_t, and filtered the response series y_t, we can examine the correlation between them in order to determine the delay d, in equation (4.2.1), and the set S (4.2.1).

Let w_t be the pre-whitened impulse series and $\tilde{y}_t$ be the filtered response series, both of length n. Denote $\hat{\gamma}_{\tilde{y},w}(\kappa)$ to be the sample cross-correlation function between $\tilde{y}_t$ and w_t at lag κ. Also, let Φ be the CDF for the Normal distribution and α the chosen significance level. Then the elements of the sequence S (4.2) are given by all lags for which the sample cross-correlation function between $\tilde{y}_t$ and lagged values of w_t are significant:

$$\hat{S} := \left\{ \kappa \;\middle|\; \hat{\gamma}_{\tilde{y},w}(\kappa) > \frac{\Phi^{-1}(1-\alpha)}{n} \right\}. \tag{4.4}$$

The time delay d is the first element of the ordered set S (4.2). In other words, it can



(a) Case 1 (b) Case 2 (c) Case 3

Figure 4.3: The response time series y_t for the three changepoint cases.

be estimated as the minimum lag for which this cross-correlation is significant:

$$\hat{d} := \min_{\kappa} \left\{ \kappa \;\middle|\; \hat{\gamma}_{\tilde{y},w}(\kappa) > \frac{\Phi^{-1}(1-\alpha)}{n} \right\}. \tag{4.5}$$

The parameter D, in equation (4.2.2), is the the maximum lag for which this cross-correlation is significant:

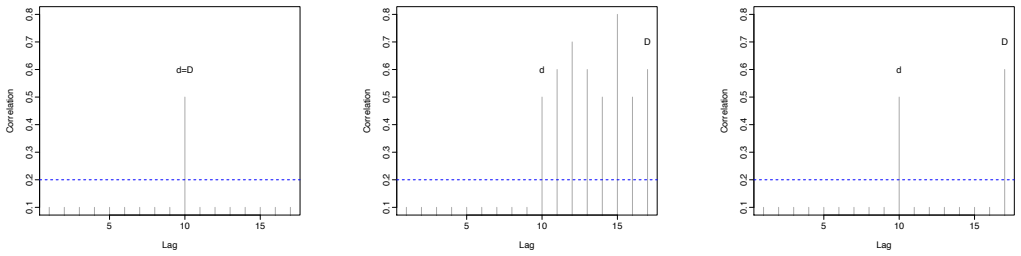
$$\hat{D} := \max_{\kappa} \left\{ \kappa \;\middle|\; \hat{\gamma}_{\tilde{y},w}(\kappa) > \frac{\Phi^{-1}(1-\alpha)}{n} \right\}. \tag{4.6}$$

Figure 4.4 shows illustrative plots of visual identification of d and D from a cross-correlation plot.

In the next section, we formulate the methodology for estimating the location of changepoints in the response series using estimates of the delay, $\hat{d}$, and of the set, $\hat{S}$. We do this first in the single changepoint case in Section 4.2.2, and then in Section 4.2.3 we extend to the multiple changepoint case.

4.2.2 Single Changepoint Case

Without loss of generality, assume that x_t begins as an i.i.d Gaussian process with zero mean. At time τ, $x_{t\geq\tau}$ exhibits an increase in mean to level $\mu > 0$. The variance of x_t remains constant throughout time.



(a) Case 1 (b) Case 2 (c) Case 3

Figure 4.4: Cross-correlation patterns for the three changepoint cases.

Figure 4.3 pictorially shows how a single change in mean in x_t would manifest in y_t for a specific case of the polynomial $\nu(B)$ in (4.2.1). In general, we have three cases:

- **Case 1:** A single change in mean in x_t manifests as a single change in mean in y_t. This occurs when $|S| = 1$.
- **Case 2:** A single change in mean in x_t manifests as a single change in mean in y_t however prior to this change there is a transition period with segment length given by $|S|$. This occurs when $|S| = D - d + 1$.
- **Case 3:** A single change in mean manifests itself as multiple changes in mean in y_t. This happens whenever $|S| < D - d + 1$.

Figure 4.4 illustrates the patterns which would be seen in the cross-correlation plot for each of these cases. Theoretically, we could also have a sequential combination of the cases 1 through 3. In case 2, this transition period could take two forms. Either, there will be a slope from the first mean level to the second, or there will be a change in mean and variance. In either case, the form of the transition period is entirely dependent upon the coefficients in (4.2.1). We formalise Cases 1-3 in the following proposition.

Proposition 4.2.3. *Let x_t exhibit a change in mean at time point τ_X and suppose*

that y_t is related to x_t via the relationship

$$y_t = \sum_{i=0}^{D} \delta_i B^i \; x_t + \epsilon_t, \tag{4.7}$$

where $\mathbb{E}[\epsilon_t] = 0$. Then, define the following ordered sequence

$$S = \{i | \delta_i \neq 0\}, i = 0, \ldots, D,$$

to be the indices of the non-zero coefficients in (4.2.3)*. Then locations of changes in mean in y_t, are given by*

$$\tau_j^Y = S_j + \tau^X, \text{ for } S_j \in S, \; j = 1, \ldots m_Y,$$

where $m^Y = |S|$ and we define $\tau_0^Y = 1$.

Proof. The proof is given in Appendix 4.A. □

Note that in Proposition 4.2.3, we are allowing for a change in mean of segment length one. A sequence of changes in mean of segment length one is more accurately described as a change in variance. To this end, define the following ordered sequence

$$S^* = \{S_1\} \cup \{S_k | S_k - S_{k-1} \neq 1\} \cup \{S_{|S|}\}, \text{ for } k = 2, \ldots, |S| - 1.$$

Then, the locations changes in mean and/or variance in y_t are given by

$$\tau_j^Y = S_j^* + \tau^X, \text{ for } S_j^* \in S^*, \; j = 1 \ldots m_Y^*,$$

where $m_Y^* = |S^*|$ and we define $\tau_0^Y = 1$.

In addition to estimating the locations of the changepoints in the response series y_t,

we can also estimate the mean levels of the time series.

Corollary 4.2.4. *The expectation of each segment of y_t is given by*

$$\mathbb{E}[y_{\tau_j+1:\tau_{j+1}}] = \sum_{i=j+1}^{|S|} \delta_{S_i}\mu_1 + \sum_{i=1}^{j} \delta_{S_i}\mu_2, \quad \textit{for } j = 0, \ldots, |S|.$$

Proposition 4.2.3 and Corollary 4.2.4 extend to the multiple changepoint case provided the minimum segment length for changepoint locations in x_t exceeds D.

4.2.3 Multiple Changepoint Case

Previously, we described the case of a single changepoint in the impulse series resulting in single or multiple changes in the response series. Now, we consider the case where the impulse series, x_t, exhibits multiple changes in mean, extending the propositions from Section 4.2.2 into the multiple changepoint setting.

Proposition 4.2.5. *Let x_t exhibit m_X changes in mean at time points $\tau^X = \{\tau_1^X, \ldots, \tau_{m_X}^X\}$ and suppose that y_t is related to x_t via the relationship*

$$y_t = \sum_{i=0}^{D} \delta_i B^i \; x_t + \epsilon_t, \tag{4.8}$$

where $\mathbb{E}[\epsilon_t] = 0$. Finally, define the following ordered sequence

$$S = \{i | \delta_i \neq 0\}, i = 0, \ldots, D,$$

to be the indices of the non-zero coefficients in (4.2.5)*. Then, provided $\tau_{j+1}^X - \tau_j^X < D \; \forall \; j$, changes in mean in y_t are given at times*

$$\tau^Y = S + \tau^X,$$

where $|\tau^Y| = m_Y = |S| \times m_X$.

Note, again, that in Proposition 4.2.5 we are allowing for a change in mean of segment length one. To this end, once again, define the following ordered sequence

$$S^* = \{S_1\} \cup \{S | S_k - S_{k-1} \neq 1\} \cup \{S_{|S|}\}.$$

Then, provided $\tau^X_{j+1} - \tau^X_j < D \; \forall \; j$, changes in mean and/or variance in y_t are given by

$$\tau^Y = S^* + \tau^X,$$

where $|\tau^Y| = m_Y = |S^*| \times m_X$.

Proof. The proof is given in Appendix 4.A. □

Similarly, we can extend Corollary 4.2.6 to the multiple changepoint case.

Corollary 4.2.6. *The expectation of each segment of* y_t *is given by*

$$\mathbb{E}[y_{\tau_{j+k|S|}+1:\tau_{j+k|S|+1}}] = \sum_{i=j+1}^{|S|} \delta_{S_i}\mu_{k+1} + \sum_{i=1}^{j} \delta_{S_i}\mu_{k+2}, \quad \textit{for } j = 0, \ldots, |S|, \; k = 0, \ldots, m-1.$$

Having outlined our model to estimate changes in mean in the response series y_t, based upon the locations of changes in mean in the impulse series x_t, in the next section we predict future changepoints in y_t.

4.3 Predicting Future changepoints

In Sections 4.2.2 and 4.2.3 we considered estimating the locations of changes in mean in y_t using the changes in mean in the explanatory series x_t for times $t = 1, \ldots, n$. These historical changepoints could, theoretically, be detected directly in the response

series y_t. We now turn our attention to predicting changepoints, i.e. we wish to predict changes in y_t for times $t > n$.

The forecast horizon will depend upon the relationship between the impulse and response series (4.2.2). At any point in time, $t \leq n$, we can only predict a changepoint in y_t a maximum of D (4.2.1) observations ahead from the most recent changepoint in x_t. Consequently, we may not be able to predict any *future* changepoints in y_t for times $t > n$.

In general, there are three scenarios that could occur. The first, is that we cannot predict any future changepoints in y_t, for times $t > n$. The second, is that we *predict* a changepoint, but it has occurred at some time $t \in [n-g, n)$, where g is the minimum length of a segment. This would happen in the case where the changepoint in y_t has occurred too close to the end of the series for us to have the power to detect it. Lastly, we can predict a future change to occur in y_t at some time $t > n$. We formalise these cases in the following proposition.

Proposition 4.3.1. *Suppose we have detected m_X changes in mean in x_t with locations τ^X for which we have imposed a minimum segment length of $g > D$. Further, suppose that x_t is related to y_t via the relationship in equation* (4.2.2) *and we have estimated the set S using equation* (4.2.1)*. Recall that d is the smallest value in this set, and D is the largest. Then at time n, we can predict future changes in y_t for times $t = n+1, \ldots, n+h$ where the horizon for changepoint prediction h, is estimated as*

$$\hat{h} := \min\{0, \tau^X_{m_X} + \hat{D} - n\},$$

where $\hat{\tau}^X_{m_X}$ is the largest detected changepoint location in the impulse series x_t. Then

- *If $\tau^X_m + \hat{S}_i < n - g \ \forall \ i$, our forecast horizon is zero and there are no predicted*

future changepoints;

- *If* $\exists\ i$ *s.t.* $n-g < \tau^X_{m_X} + \hat{S}_i \leq n$*, we have in-sample predicted changepoints given by*

$$\hat{\tau}^Y_{\hat{m}_Y+i} = \tau^X_{m_X} + \{\hat{S}_i | n - g < \tau^X_{m_X} + \hat{S}_i < n\},$$

where $\hat{m}_Y = m_X \times |\hat{S}|$.

- *If* $\exists\ i$ *s.t.* $n < \tau^X_{m_X} + \hat{S}_i$*, we have future (out-of-sample) predicted changepoints given by*

$$\hat{\tau}^Y_{\hat{m}_Y+i} = \tau^X_{m_X} + \{\hat{S}_i | n < \tau^X_{m_X} + \hat{S}_i\},$$

where $\hat{m}_Y = m_X \times |\hat{S}|$.

Proof. A proof is provided in Appendix 4.A. □

Proposition 4.3.1 illustrates that the quality of our changepoint predictions relies upon how well we estimate the relationship between the response series y_t and the explanatory series x_t, and how accurate the changepoint locations are in x_t. In Section 4.4, we perform a simulation study in order to access the quality of our methodology for varying forms of the relationship between the two time series.

4.4 Simulation Study

In this simulation study we wish to compare how the model form of the response series y_t, and in particular the structure of the innovations in equation (4.2.1), affects the performance of changepoint prediction.

In order to only consider the form of the response series, we keep the model of our impulse series x_t fixed. Explicitly, x_t is drawn from an i.i.d. Gaussian random variable,

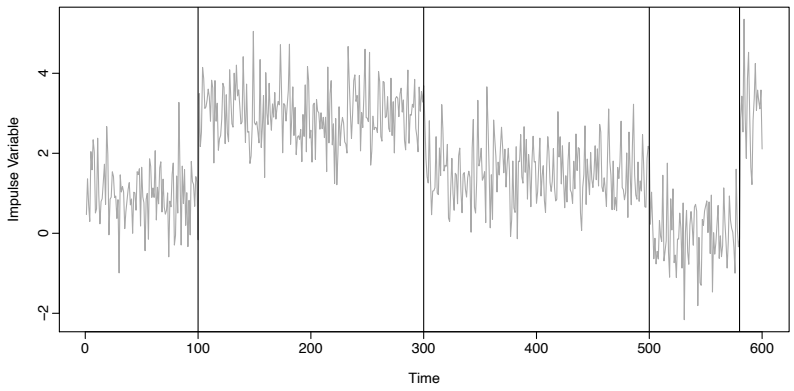


Figure 4.5: A realisation of the impulse time series x_t.

$\mathcal{N}(\mu_t, 1)$, with mean vector μ_t given by

$$\mu_t = \begin{cases} 1 & 1 \leq t < 100, \\ 3 & 100 \leq t < 300, \\ 1.5 & 300 \leq t < 500, \\ 0 & 500 \leq t < 580, \\ 3 & 580 \leq t < 600. \end{cases}$$

Figure 4.5 shows a realisation of the process x_t.

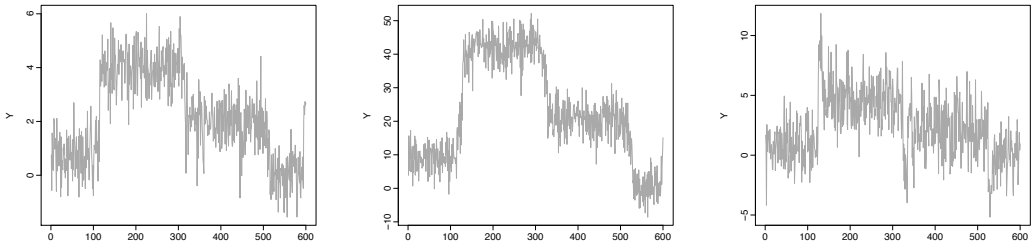
In each instance we estimate and/or predict changes in the response series y_t, using the methodology described in Section 4.2, and then validate the quality of these by detecting changes in mean and/or variance in y_t directly. That is, we are comparing the following:

1. **Detection (CPdet)**

   Detecting changes in mean and variance in y_t directly using the `cpt.meanvar` function from the `changepoint` R package (Killick and Eckley, 2014);

2. **Estimation or Prediction (CPpred)**

   First detecting changes in mean in x_t, using the `cpt.mean` function from the `changepoint` R package (Killick and Eckley, 2014) and then estimating or pre-



(a) Case 1 (b) Case 2 (c) Case 3

Figure 4.6: A single realisation from the model described by (a) Case 1 (b) Case 2 and (c) Case 3 with i.i.d innovations.

dicting changes in mean or variance in y_t by estimating the form of the transfer function model between the two time series, as described in Section 4.2.

We investigate each of the changepoint scenarios described in Section 4.2.2 by simulating the response series y_t from the following models.

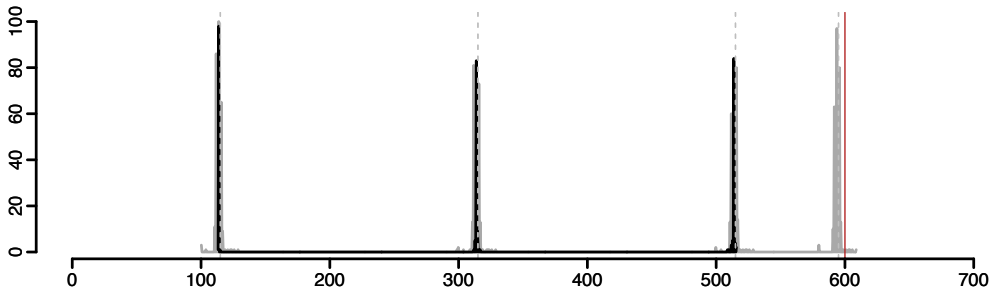
Case 1 - Each change in mean in x_t causes a single change in mean in y_t. Specifically, we simulate our response series y_t from the following model

$$y_{t,j} = 0.8x_{t-15} + \epsilon_{t,j},$$

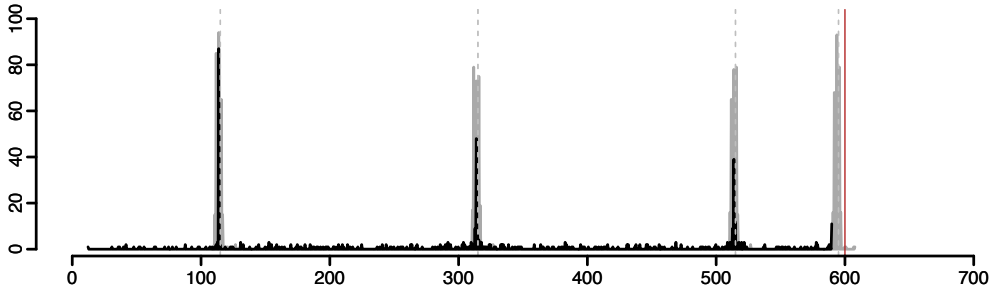
where the innovations are given by

$$\epsilon_{t,j} \sim \begin{cases} \mathcal{N}(0,1) & \text{if } j = 1; \\ AR(1) & \text{if } j = 2; \\ MA(1) & \text{if } j = 3. \end{cases} \tag{4.9}$$

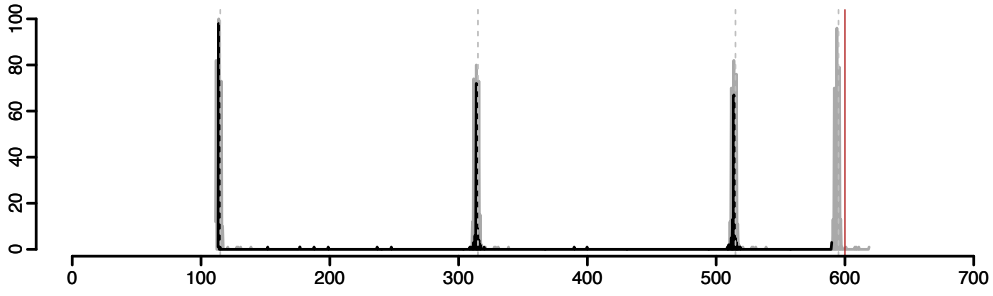
Figure 4.6a shows a single realisation for $y_{t,1}$ for Case 1. For this case, the delay d is 15 and the length of the last segment of the impulse series x_t is 20. As such, the 'true' changepoints in y_t should occur at times $t < n = 600$. We only present the results for the different types of innovation for this Case 1. However, for Cases 2 and 3, the AR and MA innovations had a similar impact on results.



(a) i.i.d innovations



(b) AR innovations



(c) MA innovations

Figure 4.7: Histogram to show the number of detected changes in mean and variance (CPdet) in y_t (black bars) along with the estimated/predicted changes in mean and variance (CPpred) (grey bars) for 500 realisations of Case 1.

Figure 4.7 shows a histogram of the detected (CPdet) and the predicted or estimated (CPpred) changepoints in the response series y_t in the presence of each of the three types of innovation (4.4). Overall, we can see that generally the type of innovation does not affect the detection of changepoints we estimate in y_t, however it is more difficult to detect the changes in mean directly. This is expected because CPdet assumes that the data we are detecting changepoints in are independently Gaussian distributed. For the AR innovations, more changepoints are detected than estimated. In general, there is larger uncertainty surrounding the locations of the estimated changepoints than the detected changepoints. As expected, we cannot detect the change in mean at the end of the time series however we can correctly predict it. This illustrates an example of CPpred predicting changepoints in-sample.

Case 2 - Each change in mean in x_t causes a single change in mean in y_t which is preceded by a period of disturbance. Specifically, we simulate from the following model

$$y_t = 1.6x_{t-15} - 1.2x_{t-16} + \ldots + 1.5x_{t-29} + 0.8x_{t-30} + \epsilon_t, \tag{4.10}$$

where the innovations $\{\epsilon_t\}_{t=1,\ldots,n}$ are a white noise process. For this scenario, the disturbance is simulated to occur at time $t < n = 600$ and the change in mean at $t > n = 600$. This is because the delay d is less than the length of the final segment of x_t (20), and the maximum lag for which x_t and y_t are related, $D = 30$, is larger than 20.

Figure 4.8a displays the results for Case 2, when a single change in x_t manifests as a change in y_t which is preceded by a disturbance period of length 15. When detecting the changepoints, CPdet detects the location of the change in mean and often misses the disturbance period prior to this. Figure 4.6b shows a single realisation from this

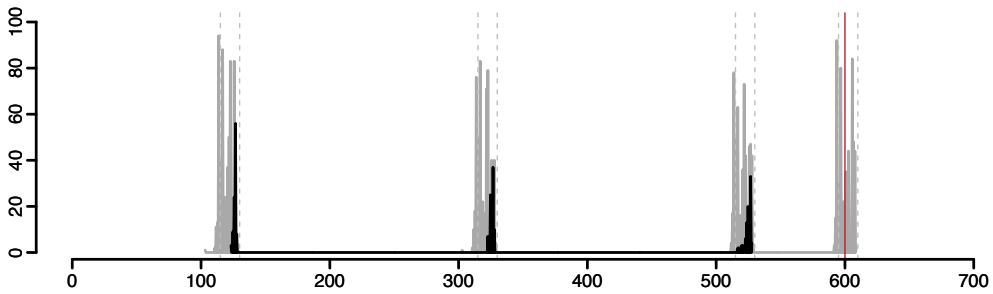
model. It shows that the disturbance period manifests as a slope to the next mean level in which the transition to the new mean level is the most abrupt towards the end of this transition period. This explains why the changepoint detection algorithm prefers this second change. The estimated changepoints (CPpred), on the other hand, detect the first change more frequently than the second. This is because the coefficients in the model (4.4) are larger for the smaller lags. Overall, detection rate (CPdet) is generally lower than estimation (CPpred) rate. For Case 2, we predict changes at times less than $n = 600$ and greater than $n = 600$. This illustrates an example of CPpred predicting changes both in-sample and out-of-sample.

Case 3 - Each change in mean in x_t causes two changes in mean in y_t. For this scenario, we consider the following model

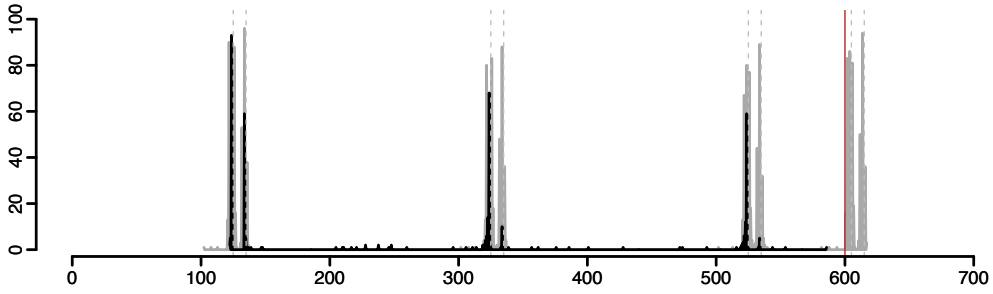
$$y_t = 2.1x_{t-25} - 1.2x_{t-35} + \epsilon_t, \tag{4.11}$$

where the innovations $\{\epsilon_t\}_{t=1,\ldots,n}$ are a white noise process. For this case, both of the changes in mean occur at times $t > n = 600$. This is because the delay d and and the maximum lag for which x_t and y_t are related (D), are both larger than the length of the final segment of x_t (20).

Figure 4.8b shows the results for Case 3, when a single change in x_t manifests as two changes in mean in y_t. In this case, the changepoint detection algorithm (CPdet) prefers the first induced change over the second. This is because the first coefficient in equation (4.4) is almost twice as large as the second, this means the first change in mean is of greater magnitude, this can be seen in the realisation of the model in Figure 4.6c. When estimating the locations of the changepoints (CPpred), the pairs of changes have greater separation than those detected in Case 2 (Figure 4.8a), this is because we have two distinct changes in mean instead of a disturbance prior to



(a) Case 2



(b) Case 3

Figure 4.8: Histogram to show the number of detected changes in mean and variance in y_t (black bars) along with the estimated/predicted changes in mean and variance (grey bars) for 500 realisations of (a) Case 2 and (b) Case 3.

the change. In this case, both the changepoints are predicted in the forecast horizon (out-of-sample) at times $t > n = 600$.

Overall, in each of the models, we estimate changes more frequently by first detecting changes in x_t and then estimating the changes in y_t using the transfer function model (CPpred). Despite this, there is greater uncertainty regarding the locations of the changes because the locations of the estimated changes rely on the covariance structure between the impulse and response time series.

For Cases 2 and 3, when detecting changes directly in y_t (CPdet), we often only detect one of the two changes induced. In practice, this would be dependent upon the application at hand, we may need to decide if we would want to detect one or two changes.

In all of the models, the AR distributed innovations made it most difficult to detect

changes in y_t accurately. This is because when autoregressive structure is present, we have more power to detect changes in mean, however these are often false positives. This feature was also encountered in Chapter 3 and has been noted by authors such as Lavielle and Moulines (2000) and Beaulieu et al. (2012).

This simulation study has only considered the form of the response series y_t and up until now, we have only considered the case where x_t is independently distributed or it is second order stationary. It is often the case that our explanatory series is not second order stationary. If this is the case, then our method of pre-whitening may not uncover the true delay between the two time series, and as a consequence, our predicted changepoint locations will be incorrect. The following section proposes a method of pre-whitening which allows our explanatory series to be piecewise second order stationary.

4.5 Piecewise Pre-whitening

The method of pre-whitening introduced by Box et al. (2013), and described in Section 4.2.1 assumes that the explanatory series x_t is second order stationary. However, it may be the case that it is piecewise second order stationary. Changes in mean are often accompanied by changes in second order structure (Yau and Davis, 2012; Sturludottir et al., 2017).

If x_t experiences changes in second order structure, and we do not take these into account in our pre-whitening process, then we may estimate the parameters in equation 4.2.1 incorrectly. In particular, the transformed input series w_t may not be a white noise process and so the form of the polynomial in equation (4.2.1) may be misleading.

Section 4.5.1 outlines our proposed method for pre-whitening a time series when there are changes in second order structure and in Section 4.5.2 we illustrate, using

a simulation study, how not accounting for changepoints can lead to an incorrect estimate of the delay d and the set S and how our approach rectifies this.

4.5.1 Method

Section 4.2.1 described the pre-whitening process of Box et al. (2013) used to remove auto-correlation before using the cross-correlation to determine the delay between the response and explanatory time series. This method, however, assumes second order stationarity of the innovations of the explanatory series. We propose to test a new method of pre-whitening which explicitly takes into account the presence of changes in second order structure in the explanatory time series. This method is outlined below.

During the usual pre-whitening of Box et al. (2013), an ARMA model is fit to x_t and the coefficients used to filter y_t. Our amended method, changepoint pre-whitening (CPPW), is to:

1. Fit a changepoint ARMA model to x_t. This identifies both the location(s) of any changepoints and also the ARMA models for each of the segments. We detect changes in the ARMA model of x_t using the methodology outlined in Chapter 3;

2. Filter y_t with the same segmented ARMA model.

If x_t exhibits changes in second order structure, the auto-correlation induced in y_t, from x_t, should now be removed accurately. The simulation study in Section 4.5.2 demonstrates this.

4.5.2 Simulation Study

In this section we investigate the method of piecewise pre-whitening. We simulate explanatory time series x_t which exhibit changes in autocovariance, and from these we simulate a response time series y_t via a transfer function relationship (4.2.1).

In each case we simulate 100 realisations of each model and for each realisation we estimate the delay d (4.2.1) and the elements of the set S (4.2.1) using standard stationary pre-whitening and piecewise pre-whitening. Figure 4.9 shows histograms for the estimated time delays using standard pre-whitening and changepoint pre-whitening. Figure 4.10 shows histograms for the estimated elements of the set S.

For the response time series y_t, we simulate from four types of transfer function model (4.2.1), given by:

$$y_t = 0.8x_{t-3} + \epsilon_t, \tag{TF1}$$

$$y_t = 0.8x_{t-5} + 0.6x_{t-12} + \epsilon_t, \tag{TF2}$$

$$y_t = 0.8x_{t-7} + 0.6x_{t-8} + \epsilon_t, \tag{TF3}$$

$$y_t = 0.8x_{t-10} + 0.6x_{t-11} + 0.4x_{t-12} + \epsilon_t, \tag{TF4}$$

where the noise process ϵ_t is given by the autoregressive process $\epsilon_t = 0.8\epsilon_{t-1} + \eta_t$, $\eta_t \sim \mathcal{N}(0, 1)$. This allows us to consider a range of delays, d, and sets, S. For models (TF1), (TF2), (TF3) and (TF4) there are 1, 2, 2 and 3 elements in the set S, respectively.

In order to evaluate the effects of different types of changes in second order structure in the impulse series x_t, we simulate data from a range of ARMA models. We simulate the impulse series x_t from the following models.

(1) A stationary AR(2) model. This simulation is designed to asses the accuracy

of the pre-whitening techniques when there are no changepoints. Specifically, for this model, we simulate from

$$x_t = 0.9x_{t-1} - 0.2x_{t-2} + \eta_t, \qquad 1 \leq t \leq 512.$$

Figures 4.9a and 4.10a show the results for the delay detected and the estimated set S, respectively, for model (1). The results for stationary pre-whitening and piecewise pre-whitening are almost the same - this indicates that there is a low false positive rate for the changes detected in the explanatory series. We can also see that for transfer function relationships (TF1), (TF2), (TF3) and (TF4), larger values of the true delay are harder to estimate. This is logical because the sample covariance at higher lags will be estimated using less observations than at the lower lags.

(2) A piecewise stationary AR model of order 3 with changing coefficients. Specifically we simulate from

$$x_t = \begin{cases} 0.9x_{t-1} - 0.2x_{t-2} - 0.4x_{t-3} + \eta_t & \text{for } 1 \leq t \leq 300, \\ 0.2x_{t-1} - 0.3x_{t-2} + 0.7x_{t-3} + \eta_t & \text{for } 300 < t \leq 512. \end{cases}$$

In model (2) the order of the AR process remains constant however the coefficients of the model change. From Figure 4.9b it can be seen that the delay is underestimated considerably using stationary pre-whitening. This indicates that the auto-correlation in x_t has not been sufficiently removed during the pre-whitening process. This is further supported by Figure 4.10b where we can see that, if changes in second order structure are not taken into account, too many significant lags in the covariance between x_t and y_t are detected. Using piecewise pre-whitening leads to improved estimation of both the delay d and the set S in all instances of model (2).

(3) A piecewise stationary AR model which changes from a third order to

a first order model. Specifically we simulate from

$$x_t = \begin{cases} 0.1x_{t-1} - 0.6x_{t-2} - 0.3x_{t-3} + \eta_t & \text{for } 1 \leq t \leq 200 \\ 0.3x_{t-1} + \eta_t & \text{for } 200 < t \leq 512. \end{cases}$$

Figures 4.9c and 4.10c show the results for the delay detected and the estimated set S, respectively, for model (3). In this case, the order of the AR model changes, and this seems to impact the stationary estimation of the parameters less than a change in coefficient. This suggests that during the stationary pre-whitening procedure, more auto-correlation can be effectively removed.

For model (3) the stationary pre-whitening procedure would most likely over-fit the model, fitting an AR(3) process to the entire time series. Despite being an incorrect model, it would still remove much of the auto-correlation. However for model (2), the fitted stationary model will tend to under-fit to the coefficients meaning much of the auto-correlation at lower lags remains - this can be seen in Figure 4.9b where the stationary approaches often selects the delay to be less than two.

(4) A piecewise stationary AR model which changes from a first order to a third order model with a short segment at the beginning of the time series. Specifically we simulate from

$$x_t = \begin{cases} 0.3x_{t-1} + \eta_t & \text{for } 1 \leq t \leq 50 \\ 0.1x_{t-1} - 0.6x_{t-2} - 0.3x_{t-3} + \eta_t & \text{for } 50 < t \leq 512. \end{cases}$$

Model (4) is similar to model (3) however the change in order of the autoregressive process is reversed and the duration of the AR(3) segment is longer. We can see that when there is a segment which is sustained for longer, the change in second order structure has less of an impact on the estimate of d. The same can be seen in Figure

4.10 for the estimation of the set S.

(5) A piecewise stationary AR model which changes from a third order to a fifth order. Specifically we simulate from

$$x_t = \begin{cases} 0.1x_{t-1} - 0.6x_{t-2} - 0.3x_{t-3} + \eta_t & \text{for } 1 \leq t \leq 256 \\ 0.9x_{t-1} - 0.2x_{t-2} - 0.4x_{t-4} + 0.3x_{t-5} + \eta_t & \text{for } 256 < t \leq 512. \end{cases}$$

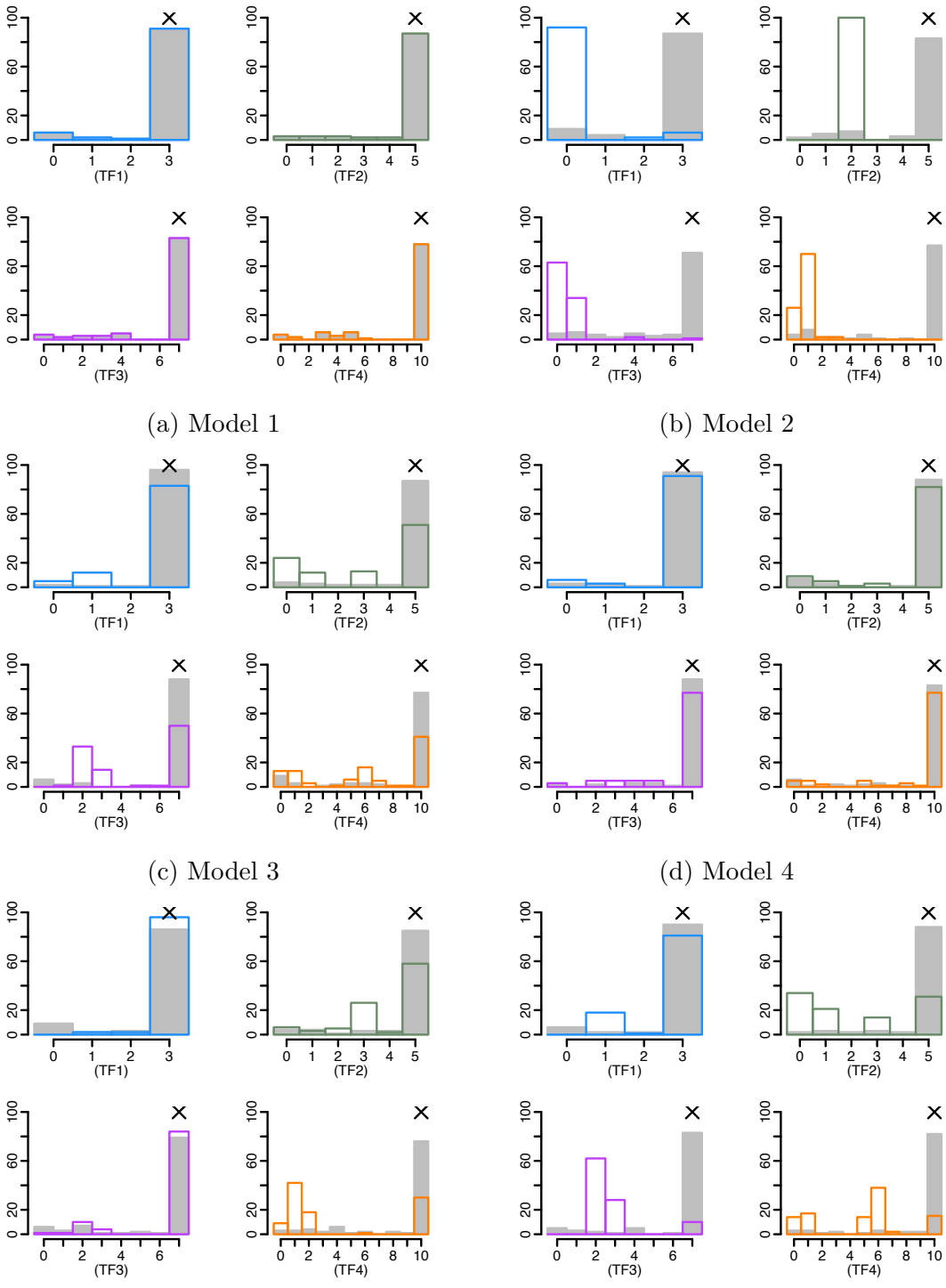
In model (5) we increase the order of the AR model. From Figure 4.9e we can see that for the transfer function models TF1 and TF3, the stationary pre-whitening estimates the correct delay more often than the piecewise pre-whitening. In these two cases piecewise pre-whitening is identifying an incorrect delay of zero more often than stationary pre-whitening.

(6) A piecewise stationary AR model which changes from a third, to a second and to a first order model. Specifically we simulate from

$$x_t = \begin{cases} 0.1x_{t-1} - 0.6x_{t-2} - 0.3x_{t-3} + \eta_t & \text{for } 1 \leq t \leq 200 \\ 0.9x_{t-1} - 0.2x_{t-2} + \eta_t & \text{for } 200 < t \leq 400 \\ 0.5x_{t-1} + \eta_t & \text{for } 400 < t \leq 512. \end{cases}$$

Finally, in model (6) we consider three changes in AR order. From all of the models that exhibit a change in AR order, we can see from Figure 4.9 and 4.10, that the gain from using a piecewise pre-whitening approach is the largest for model (6). This suggests that an increase in the number of changepoints has a larger effect on the estimation of the delay d and the set S.

In general, piecewise pre-whitening offers significant improvements to the estimation of the parameters used to predict changepoints. As the true delay increases, the benefit from using a piecewise pre-whitening approach becomes larger. In particular,



(a) Model 1

(b) Model 2

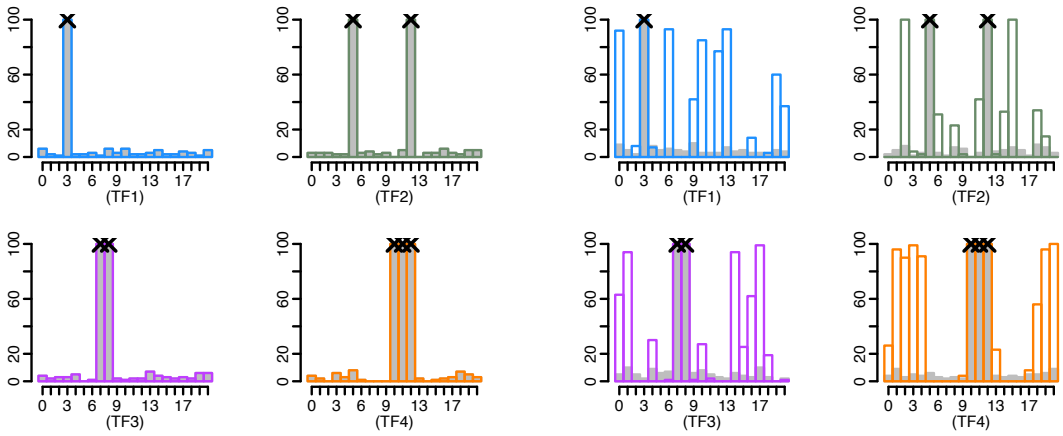
(c) Model 3

(d) Model 4

(e) Model 5

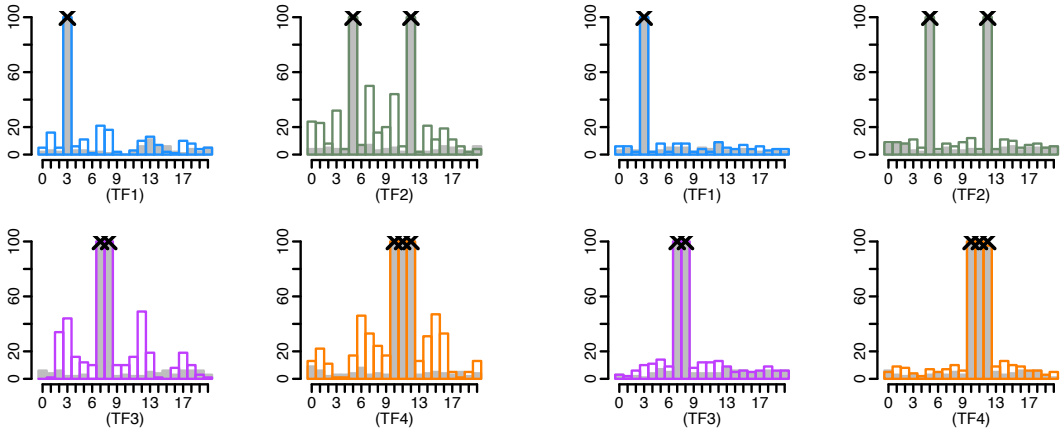
(f) Model 6

Figure 4.9: Histograms showing the estimated delay $\hat{d}$ for 100 realisations of models (1) - (6) for transfer function relationships (TF1), (TF2), (TF3) and (TF4). In each case, the grey solid bars represent the piecewise pre-whitening approach and the coloured unfilled bars represent a stationary pre-whitening approach. The solid cross is the true value of the delay.



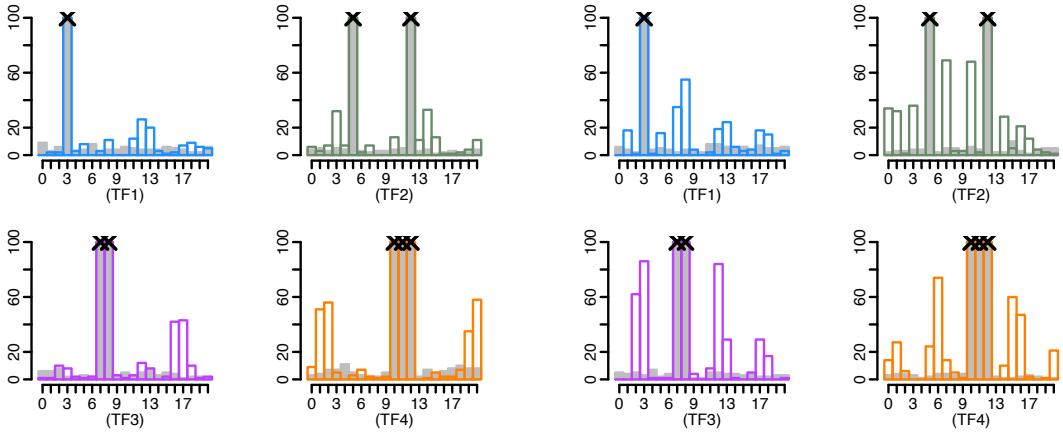
(a) Model 1

(b) Model 2



(c) Model 3

(d) Model 4



(e) Model 5

(f) Model 6

Figure 4.10: Histograms showing the estimated elements of the set S for 100 realisations of models (1) - (6) for transfer function relationships (TF1), (TF2), (TF3) and (TF4). In each case, the grey solid bars represent the piecewise pre-whitening approach and the coloured unfilled bars represent a stationary pre-whitening approach. The solid crosses are the true elements of the set S.

it helps to eliminate many of the incorrect lags of cross-covariance function between the response and explanatory series, as we can clearly see in Figure 4.10.

Stationary pre-whitening gave the poorest results when the order of the AR process remained the same, but the coefficients of the model changed, and also when there was more than one change in the explanatory series.

In practice it is likely that our explanatory series will exhibit changes in second order structure, and we have illustrated through simulations, that failing to consider these will interfere with the estimation of the parameters of the transfer function model (4.2.1) between the two time series. As a result, the estimated and predicted locations of the changepoints in the response series y_t will be incorrect.

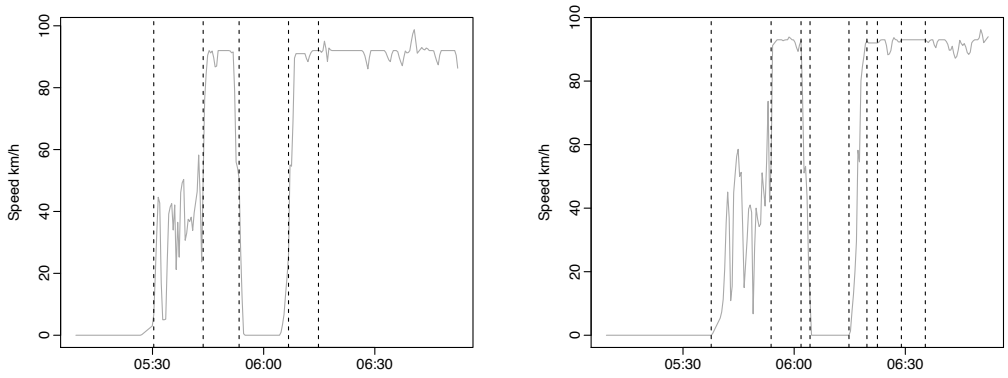
In the following section, we test our changepoint prediction methodology on an application to Telematics data.

4.6 Data Application

In this section we apply our methodology to an example relating to autonomous driving, and in particular, to the haulage industry.

In the situation where a company has a fleet of vehicles, there is often a leading vehicle and then other vehicles following behind them. The following vehicles could, for example, be autonomous. In such a situation, the leading vehicle could inform the vehicles that follow. So, if the leading vehicle experiences a change in mean speed, we would expect the following vehicles to exhibit a change in mean in their speed at a slightly later time, depending on how far the vehicles are from each other.

In Figure 4.11 we have two time series for the speed of two heavy goods vehicles (HGVs). The data has been interpolated to ensure the observations are equally



(a) Leading HGV (b) Following HGV

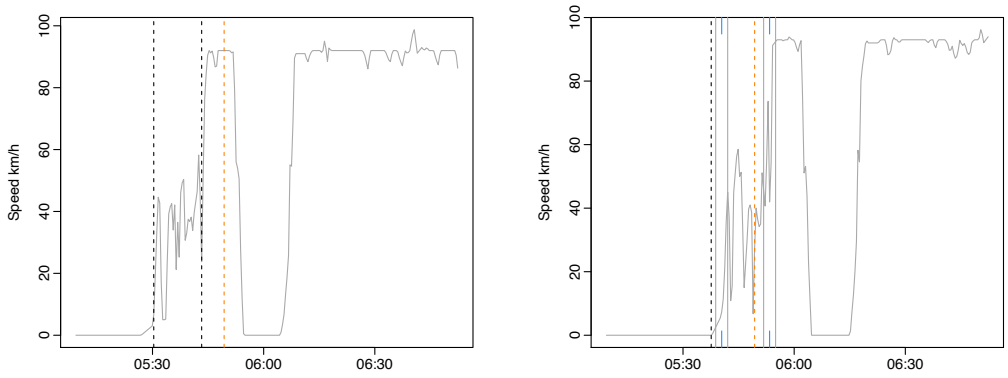
Figure 4.11: Speed over time of two HGVs performing the same journey one after another along with detected changes.

spaced.

The HGVs are performing the same journey one after another, Figure 4.11a is the leading HGV and Figure 4.11b is the one following. For the leading vehicle, Figure 4.11a also shows the detected changes in mean and variance (CPdet), and for the following vehicle, Figure 4.11b shows the detected changes in mean.

In order to test our changepoint prediction methodology, using both piecewise and stationary pre-whitening, we choose a selection of training periods for which we fit the changepoint prediction models, and then estimate or predict changes. The first training period we use is from the start of the journey up until time 05:50am, this is indicated in Figure 4.12 by the vertical dashed orange line. Figure 4.12a shows the detected changes in mean in the leading HGV. We have detected two changes in mean. Figure 4.12b shows the detected changes in mean and variance (CPdet) in the following HGV, along with changepoints that have been estimated or predicted (CPpred). In this case, we have changes estimated within the training period, and also changes we have predicted outside of the training period. If we compare these changes to those we detected in Figure 4.11, then they seem reasonable.

It is interesting to compare the results obtained from stationary and from piecewise

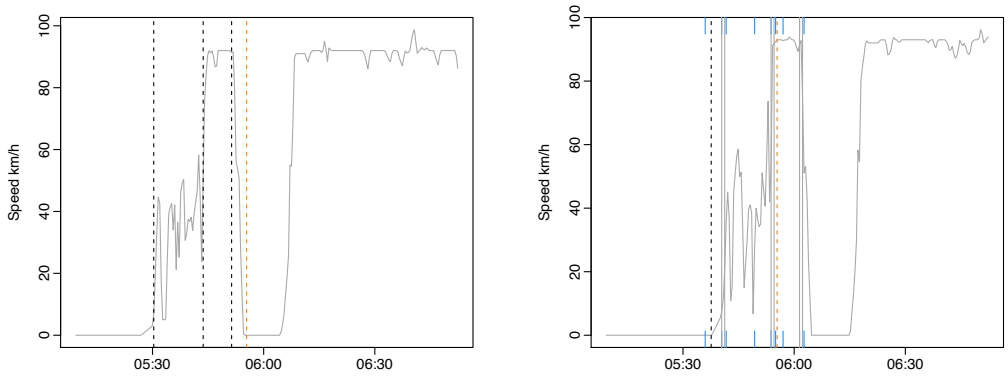


(a) Leading HGV (b) Following HGV

Figure 4.12: Speed over time of two HGVs performing the same journey one after another with detected changepoints (vertical dashed black lines), estimated or predicted changes using stationary pre-whitening (blue markers) and estimated or predicted changes using piecewise pre-whitening (grey vertical lines). The vertical dashed orange line is the end of the training period.

pre-whitening. If we consider the changes we have estimated within the training period (CPpred), in Figure 4.12b, piecewise pre-whitening has identified a feature that stationary pre-whitening has failed to capture. Specifically, we can see in Figure 4.12a that the leading HGV accelerates from being stationary very abruptly, however, the following HGV has a different driving behaviour - it has a more gradual increase in speed. The changepoints estimated using piecewise pre-whitening capture the start and end of this incline in speed. The same features can be seen in Figure 4.11. For the leading HGV, changes in mean level are captured using a single changepoint, however for the following HGV, there is often two changes which enclose a slope. When we asked an industry expert, they said that this is indicative that the following HGV had a heavier load than the leading HGV. This is an illustration of the case where a single change in the impulse series, manifests as a change in the response series with some disturbance period beforehand.

The second training period we use is up until time 05:55am, this is indicated by the vertical dashed orange line in Figure 4.13. For this time period we have detected 3



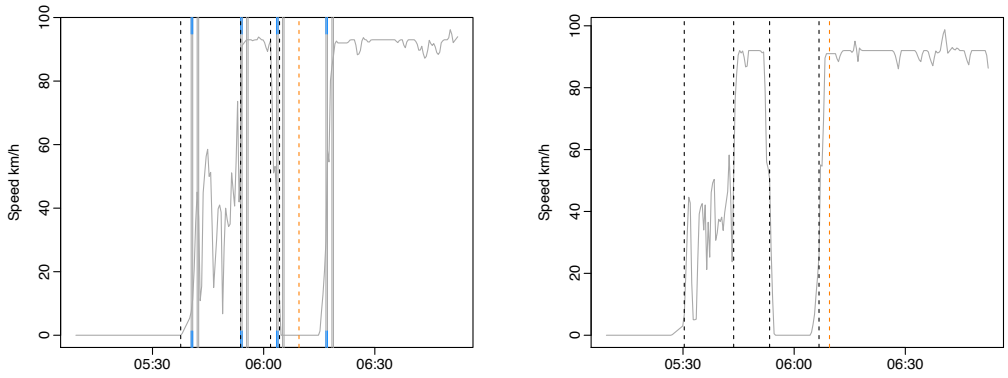
(a) Leading HGV (b) Following HGV

Figure 4.13: Speed over time of two HGVs performing the same journey one after another with detected changepoints (vertical dashed black lines), estimated or predicted changes using stationary pre-whitening (blue markers) and estimated or predicted changes using piecewise pre-whitening (grey vertical lines). The vertical dashed orange line is the end of the training period.

changes in mean in the speed of the lead HGV, see Figure 4.13a. For the following HGV, in Figure 4.13b, we estimate changes and predict changes (CPpred) both inside and outside of the training period. We still only detect a single change in the training period due to the change being close to 05:55am. Comparing the changes estimated or predicted using stationary or piecewise pre-whitening, it seems that stationary pre-whitening is detecting more spurious changepoints, implying that the pre-whitening process did not remove enough auto-correlation in the speed of the leading HGV.

The last training period we consider is up until 06:10am, indicated by the vertical dashed orange line in Figure 4.14. During this period of time, we detect 4 changes in mean in the speed of the leading HGV and for the following HGV, we estimate multiple changes within the training period and predict multiple changes outside of the training period.

We can see in Figure 4.14a that when piecewise pre-whitening is used, twice as many changes are estimated and predicted in the speed of the following vehicle. Overall, it is difficult to determine which segmentation seems most reasonable, however using



(a) Following HGV (b) Leading HGV

Figure 4.14: Speed over time of two HGVs performing the same journey one after another with detected changepoints (vertical dashed black lines), estimated or predicted changes using stationary pre-whitening (blue markers) and estimated or predicted changes using piecewise pre-whitening (grey vertical lines). The vertical dashed orange line is the end of the training period.

piecewise pre-whitening results in two very spurious changes prior to 06:00am. This change was not estimated using piecewise pre-whitening when the training period was smaller, however a similar change was identified using stationary pre-whitening in Figure 4.13b. This could suggest that the model relating each of the HGVs is changing over time.

4.7 Conclusions and Future Work

In this chapter we have developed a method to predict future changepoints based upon a transfer function model between an explanatory and response series. In addition to this, we have have developed a new approach to pre-whitening time series which considers changes in the second order structure of the explanatory series. This is useful in a wider time series and forecasting context, and does not need to be restricted to predicting changepoints. We tested our changepoint prediction methodology using a range of simulation studies and applied it to an example in the Haulage industry.

We predicted changepoints in the mean speed of the following HGV successfully, and were also able to identify interesting driving behaviours.

One potentially interesting avenue for future research would be to consider a model which allows the delay between the two time series to vary overtime. The delay between the two series relies on their cross-covariance, therefore to detect a change in the delay, we would detect a change in the cross-covariance between the series. This would be an important avenue for further research, especially in regards to the example of a fleet of vehicles. This is because, as time increases we may expect the delay between the two vehicles to increase or decrease as a function of time. This may explain the results we obtained in Section 4.6.

In addition, in the future, we would like to extend this methodology to allow for the prediction of different types of changes such as variance and trend. It would also be interesting to extend the model to allow for more than one explanatory time series. Another avenue of future research could be to amend the algorithm for an on-line setting.

4.A Appendix

Proof of Proposition 4.2.3

The expectation of x_t is given by: $\mathbb{E}[x_t] = \mu_1 \mathbf{I}_{1\leq t\leq \tau_X} + \mu_2 \mathbf{I}_{\tau_X+1\leq t\leq n}$. Then, the expectation of y_t is given by

$$\mathbb{E}[y_t] = \sum_{i=0}^{D} \delta_i \mathbb{E}[x_{t-i}] = \sum_{j=1}^{|S|} \delta_{S_j} \mathbb{E}[x_{t-S_j}] = \sum_{j=1}^{|S|} \left[\mu_1 \mathbf{I}_{1+S_j\leq t\leq \tau_X+S_j} + \mu_2 \mathbf{I}_{\tau_X+S_j+1\leq t\leq n+S_j}\right].$$

Separating these out gives the following expression:

$$\mathbb{E}[y_t] = \sum_{j=1}^{|S|} \delta_{S_j}\mu_1 \mathbf{I}_{1\leq t\leq \tau_X+S_1} + \left(\sum_{j=1}^{1} \delta_{S_j}\mu_2 + \sum_{j=1}^{|S|} \delta_{S_j}\mu_1\right)\mathbf{I}_{\tau_X+S_1+1\leq t\leq \tau_X+S_2} + \dots$$
$$+ \left(\sum_{j=1}^{|S|-1} \delta_{S_j}\mu_2 + \sum_{j=1}^{1} \delta_{S_j}\mu_1\right)\mathbf{I}_{\tau_X+S_{|S|-1}+1\leq t\leq \tau_X+S_{|S|}} + \sum_{j=1}^{|S|} \delta_{S_j}\mu_2 \mathbf{I}_{\tau_X+S_{|S|}+1\leq t\leq n}.$$

Therefore the changes in mean in y_t are given by $\tau_X + S_j$, $j = 1, \dots, |S|$.

Proof of Proposition 4.2.5

By imposing that the minimum segment length be greater than D, where $D := \max_j S_j$, the multiple changepoint case follows immediately from the single changepoint scenario.

Proof of Proposition 4.3.1

Let $\tau^X_{m_X}$ be the location of the most recent changepoint location in x_t. Then, from Proposition 4.2.5 the $|S|$ most recent changes in y_t are given by $\tau^Y_{m_Y+i} = \tau^X_{m_X} + S_i$. The result follows by considering the location of each of these changes in relation to the end of in-sample period, n, and the minimum segment length, g.

Chapter 5

Wavelets

In the previous chapters of this thesis we have seen, that more often than not, many of the data sets we encounter are non-stationary in nature. We have also seen that in many important application areas, e.g. time series forecasting in Chapter 4, it is important to capture the (temporal) dependence structure between observations adequately, otherwise future predictions may be unreliable. In this chapter, we turn our attention to a non-parametric framework in which we model such non-stationary time series. Specifically, we introduce wavelets (Section 5.1) and review the literature surrounding their application within locally stationary time series modelling (Section 5.2). Finally, in Section 5.3 we review the literature surrounding detecting change-points using the model described in Section 5.2. These ideas will be used in Chapter 6 for proposing a new method for detecting changes in variance, and in Chapter 7 we extend this into detecting changes in autocovariance.

5.1 Wavelets

Wavelets, as the name suggests, can be described as "little waves". This is because they are compactly supported functions. In contrast, the basis functions in a Fourier transform have global support, i.e. the sines and cosines constitute "big waves". It is the compact support of wavelets that naturally lend them to modelling time series whose properties vary over time. When we use the term term wavelet, we are typically referring to the mother wavelet, $\psi(x)$. Following Meyer and Salinger (1992), we introduce wavelets in the following.

Definition 5.1.1. *Let $m \in \mathbb{N}$ and $x \in \mathbb{R}$. Then a function $\psi(x)$ is called a mother wavelet of order m if the following properties hold:*

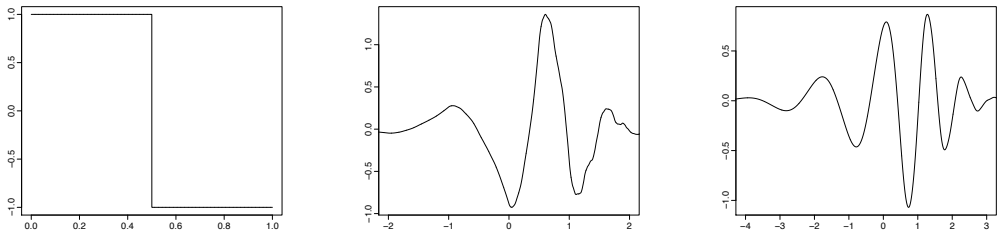
1. *If $m = 0$, $\psi(x) \in L^\infty(\mathbb{R})$. If $m \geq 1$, then $\psi(x)$ and all its derivatives up to order m belong to $L^\infty(\mathbb{R})$.*

2. *$\psi(x)$ and all its derivatives up to order m decrease rapidly as $x \to \pm\infty$.*

3. *For each $k \in \{0, \ldots, m\}$,*
$$\int_{-\infty}^{\infty} x^k \psi(x) \mathrm{d}x = 0.$$

4. *The collection $\{\psi_{j,k}\}_{j,k\in\mathbb{Z}}$ forms an orthonormal basis of $L^2(\mathbb{R})$, the $\psi_{j,k}$ being constructed from the mother wavelet using the identity*
$$\psi_{j,k}(x) = 2^{-j/2}\psi(2^{-j}x - k).$$

In Definition 5.1.1, condition 1 expresses the smoothness of the wavelet. Figure 5.1 shows three examples of mother wavelets of increasing order. Conditions 2 and 3 address the localisation and oscillation of ψ. Condition 3 is often referred to as the vanishing moments property. Finally, the parameters j and k, in condition 4,



(a) (b) (c)

Figure 5.1: Examples of Daubechies extremal phase mother wavelets with (a) one, (b) four and (c) nine vanishing moments.

correspond to the dilation (scale) and translation (location), respectively.

The *Haar wavelet* is a popular example of a wavelet. Haar wavelets are generated from the following mother wavelet, of order zero

$$\psi(x) = \begin{cases} 1 & \text{if } 0 \leq x < 1/2; \\ -1 & \text{if } 1/2 \leq x < 1; \\ 0 & \text{otherwise .} \end{cases}$$

The Haar mother wavelet is shown in Figure 5.1a.

In order to perform a wavelet transform, we rely on a multi-resolution analysis (MRA). This provides a framework for examining functions at different scales. It enables us to understand wavelet bases and construct new examples. Following Mallat (1989) we define a multi-resolution analysis as follows.

Definition 5.1.2. *A multiresolution analysis (MRA) is a nested sequence of closed subspaces,* $V_{j\in\mathbb{Z}} \subset \mathbb{L}_2(\mathbb{R})$*,*

$$\ldots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \ldots$$

such that

1. *the spaces have an intersection which is trivial:*

$$\cap_{j\in\mathbb{Z}} V_j = \{0\};$$

2. *the spaces have a union which is dense in* $\mathbb{L}_2(\mathbb{R})$*:*

$$\cup_{j\in\mathbb{Z}} V_j = \mathbb{L}_2(\mathbb{R});$$

3. *the spaces are constructed such that the following self similar relations exist:*

   (a) $f(x) \in V_j \iff f(2x) \in V_{j+1} \qquad \forall j \in \mathbb{Z}$;

   (b) $f(x) \in V_0 \iff f(x-k) \in V_0 \qquad \forall k \in \mathbb{Z}$;

4. *there exists a unique scaling function,* $\phi(x) \in V_0$*, whose integer translations span the space* V_0*, and for which* $\{2^{-j/2}\phi(2^{-j}x-k)|k\in\mathbb{Z}\}$ *is an orthonormal basis of* V_j*.*

Since $V_0 \subset V_1$, we can express the function $\phi(x) \in V_0$ as a linear combination of functions from V_1. Consequently, due to the conditions in Definition 5.1.2, we can express

$$\phi_j(x) = \sum_{k\in\mathbb{Z}} h_k 2^{-j/2}\phi(2^{-j}x-k) = \sum_{k\in\mathbb{Z}} h_k \phi_{1,k}(x), \tag{5.1}$$

for some coefficients h_k. Equation (5.1) is called the *scaling equation* and an individual element from the basis, for any location k, is denoted by

$$\phi_{j,k}(x) = 2^{-j/2}\phi(2^{-j}x-k). \tag{5.2}$$

The coefficients $\mathbf{h} = \{h_k\}_{k\in\mathbb{Z}}$ are often referred to as wavelet filters. When associated with an orthogonal MRA, they have two important properties, *normalisation* and

orthogonality:

$$\sum_{k\in\mathbb{Z}} h_k = \sqrt{2} \qquad \text{and} \qquad \sum_{k\in\mathbb{Z}} h_k h_{k-2l} = \delta_l. \tag{5.3}$$

We refer the reader to Vidakovic (2009) for proofs of (5.1).

The scaling equation (5.1) allows us to obtain an approximation of $f(x)$ at a particular scale, or resolution, j. Since $\{\phi_{j,k}\}$ is a basis for V_j, we can write an approximation to $f(x)$ at scale j as

$$f_j(x) = \sum_{k\in\mathbb{Z}} c_{j,k}\phi_{j,x}(x) = \mathcal{P}_j f, \tag{5.4}$$

for some coefficients $\{c_{j,k}\}_{k\in\mathbb{Z}}$, where $\mathcal{P}_j$ is the projection operator introduced by Daubechies (1988).

As the $\{\phi_{j,k}\}_{k\in\mathbb{Z}}$ are orthonormal, the coefficients $\{c_{j,k}\}$ are given by the inner product of (a) the function $f(x)$ which we are approximating, and (b) the basis elements $\phi_{j,k}$,

$$c_{j,k} =< f, \phi_{j,k} >= \int_{\mathbb{R}} f(x)\phi_{j,k}(x)\mathrm{d}x. \tag{5.5}$$

Figure 5.2 shows approximations obtained when applying the Haar MRA to a piecewise polynomial used by Nason and Silverman (1995) and available in the `wavethresh` R package (Nason, 2012). We can see that as j increases, the approximation becomes increasingly coarser. This is because the projection in equation (5.1) is constructed using the basis function (5.1). As j increases, the approximation uses less information because of the $2^{-j}x$ term in equation (5.1).

5.1.1 Fourier Properties of the Scaling Function

It can often be useful to consider the Fourier properties of the scaling function $\phi(x)$. Following Vidakovic (2009), the Fourier transform of the scaling equation (5.1) is

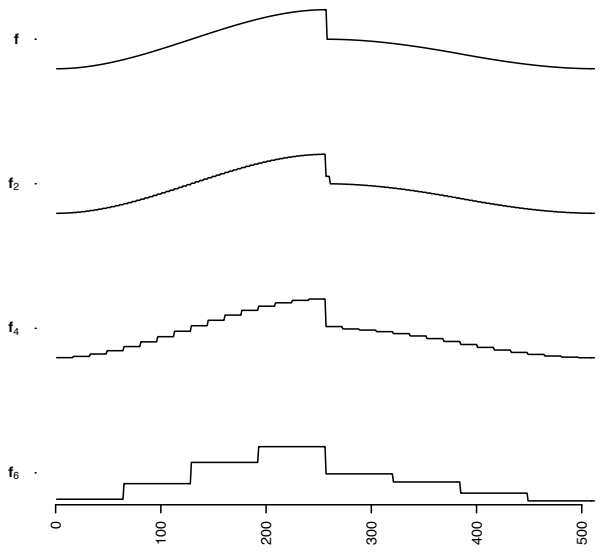


Figure 5.2: A piecewise polynomial function f with successive approximations, f_2, f_4, f_6 using the Haar MRA, obtained using the `wavelets` R package (Aldrich, 2013). The coarsest approximation is f_6 and the finest scale approximation is f_2.

given by

$$\Phi(\omega) = m_0\left(\frac{\omega}{2}\right)\Phi\left(\frac{\omega}{2}\right),$$

where $\Phi(\cdot)$ should not be confused with the CDF for the Normal distribution used in Chapter 4. Here the function, m_0, describes the behaviour of the filter, h_k, in the frequency domain, given by

$$m_0(\omega) = \frac{1}{\sqrt{2}}\sum_{k\in\mathbb{Z}} h_k e^{-ik\omega}.$$

Daubechies (1988) shows that the orthonormal properties of the scaling function leads to the following condition

$$|m_0(\omega)|^2 + |m_0(\omega+\pi)|^2 = 1, \tag{5.6}$$

and we necessarily have $|m_0(0)| = 1$ and $|m_0(\pi)| = 0$. Equation (5.1.1) is the Fourier domain equivalent of Definition 5.1.2 part 4. I.e. that the translates of the scaling

function, $\{\phi(x-k)|k \in \mathbb{Z}\}$, form an orthonormal basis for V_0. In the following we describe how to derive a wavelet from the scaling function.

5.1.2 Deriving a Wavelet Function from a MRA

Multi-resolution analysis is key to deriving a wavelet function. Specifically, a mother wavelet is derived from a scaling function. This is achieved by considering the information which is lost when we move from one resolution space V_{j+1}, down to a coarser space V_j. Definition 5.1.3 formalizes this idea.

Definition 5.1.3. *The detail space W_j is the orthogonal complement of V_j in V_{j+1}*

$$V_{j+1} = V_j \oplus W_j. \tag{5.7}$$

This expresses the subspace V_{j+1} as a composition of detailed information, W_j, and coarse information, V_j. The detail space W_j captures information we would otherwise lose if we considered a coarser scale.

Now, recall that the integer translations of the scaling function $\phi_j(x)$ (5.1) form an orthonormal basis of V_j. Analogously, a function $\psi_j(x)$ can be found such that its integer translates form an orthonormal basis of W_j. This *wavelet function* $\psi_{j,k}(x)$ is defined as

$$\psi_{j,k}(x) = 2^{-j/2}\psi(2^{-j}x - k), \quad j,k \in \mathbb{Z}. \tag{5.8}$$

The set $\{\psi_{j,k}\}_{j,k\in\mathbb{Z}}$ forms an orthonormal basis of $\mathbb{L}_2(\mathbb{R})$.

In order to derive a wavelet function $\psi(x)$ from the scaling function $\phi(x)$, we use the fact that $W_j \subset V_{j+1}$ and hence $\psi_{j,k}(x) \in V_{j+1}$. Therefore, we can represent the wavelet

function at scale j as a linear combination of the basis functions for the subspace V_{j+1}:

$$\psi_j(x) = \sum_{k\in\mathbb{Z}} g_k 2^{-(j+1)/2}\phi(2^{-(j+1)}x - k). \tag{5.9}$$

The Fourier representation of the wavelet function (5.1.2) is given by

$$\Psi(\omega) = m_1\left(\frac{\omega}{2}\right)\Phi\left(\frac{\omega}{2}\right),$$

where the function m_1 describes the filter g_k in the frequency domain, i.e. $m_1(\omega) = 2^{-1/2}\sum_{k\in\mathbb{Z}} g_k e^{-ik\omega}$, and must satisfy the following orthogonality conditions in relation to the function m_0

$$|m_0(\omega)|^2 + |m_1(\omega)|^2 = 1,$$

and

$$m_0(\omega)\overline{m_1(\omega)} + m1(\omega+\pi)\overline{m_0(\omega+\pi)} = 0.$$

We refer the reader to Vidakovic (2009) for a proof.

It follows, that the filters $\mathbf{h} = \{h_k\}_{k\in\mathbb{Z}}$ and $\mathbf{g} = \{g_k\}_{k\in\mathbb{Z}}$, associated with the scaling relations (5.1) and (5.1.2) respectively, are related to one another by the so called *quadrature mirror filter relation*: $g_k = (-1)^k h_{1-k}$, in which the coefficients h_k, $k \in \mathbb{Z}$ combine to form a low-pass (averaging) filter and the g_k coefficients form a high pass filter. This relation allows us to take any scaling function, $\phi_{j,k}(x)$, which satisfies the MRA properties and use it to derive a wavelet function using equation (5.1.2).

Now that we have established representations for both the detailed information, W_j, and the coarse information, V_j, it is possible to extend the approximate representation of a function $f(x)$ in equation (5.1) into an exact wavelet representation.

We can extend the representation in (5.1), using the decomposition in equation (5.1.3),

to also include detailed information

$$f_{j+1}(x) = f_j(x) + \sum_{k\in\mathbb{Z}} d_{j,k}\psi_{j,k}(x) = \sum_{k\in F} c_{j,k}\phi_{j,k}(x) + \sum_{k\in\mathbb{Z}} d_{j,k}\psi_{j,k}(x), \tag{5.10}$$

where $d_{j,k} = \int_{\mathcal{R}} f(x)\phi_{j,k}(x)$.

For some coarser level scale $j_0 < j$, we can repeat this decomposition

$$f_{j+1}(x) = \sum_{k\in\mathbb{Z}} c_{j_0,k}\phi_{j_0,k}(x) + \sum_{l=j_0}^{j}\sum_{k\in\mathbb{Z}} d_{l,k}\psi_{l,k}(x),$$

illustrating that a function can be approximated at scale $j+1$ by the approximation at the coarser scale j_0 plus the detailed information in between.

The above decomposition can be iterated for an increasing number of scales and, as $j \to \infty$, more and more detail is included in the approximation yielding an exact wavelet representation of the function

$$f(x) = \sum_{k\in\mathbb{Z}} c_{j_0,k}\phi_{j_0,k}(x) + \sum_{l=j_0}^{\infty}\sum_{k\in\mathbb{Z}} d_{j,k}\psi_{j,k}(x).$$

It is often the case that we do not want to decompose a continuous function $f(x)$, but instead we wish to analyse a set of discrete observations $(x_1, x_2, \ldots)$. In the following section we outline the procedure for performing a wavelet transform on discrete data.

5.1.3 The Discrete Wavelet Transform

It is often the case that the data we are analysing is of a discrete nature. In such a case the continuous representation in equation (5.1) is inappropriate. Instead we turn to the the discrete wavelet transform (DWT), proposed by Mallat (1989). The DWT filters a sequence of data $\{x_1, \ldots, x_N\}$ of dyadic length $N = 2^J$ into a wavelet de-

composition sequence $\{c_{J,0}, d_{J,0}, d_{J-1,0}, d_{J-1,1}, \ldots, d_{1,0}, \ldots, d_{1,\frac{N}{2}-1}\}$, in which the $\{c_{j,k}\}$ and $\{d_{j,k}\}$ are defined as in equation (5.1) and (5.1.2) respectively. The coefficients $\{c_{j,k}\}$ and $\{d_{j,k}\}$ are commonly known as the smooth and detail coefficients of the transformation.

If the father wavelet $\phi(x)$ satisfies the properties of a MRA, then Mallat (1989)'s pyramidal algorithm can be used to efficiently calculate the smooth and detail coefficients at scale $j+1$, from the smooth coefficients at scale j. We describe this algorithm below, following the notation of Nason (2010).

Recall that $c_{j+1,k} = \int_{\mathcal{R}} f(x)\phi_{j+1,k}(x)dx$, since the $\{\phi_{j,k}\}_{k\in\mathbb{Z}}$ is an orthonormal basis for V_{j+1}. Then using the scaling equation (5.1) and the relationship in equation (5.1), we can re-write $c_{j+1,k}$, in terms of $c_{j,k}$, as

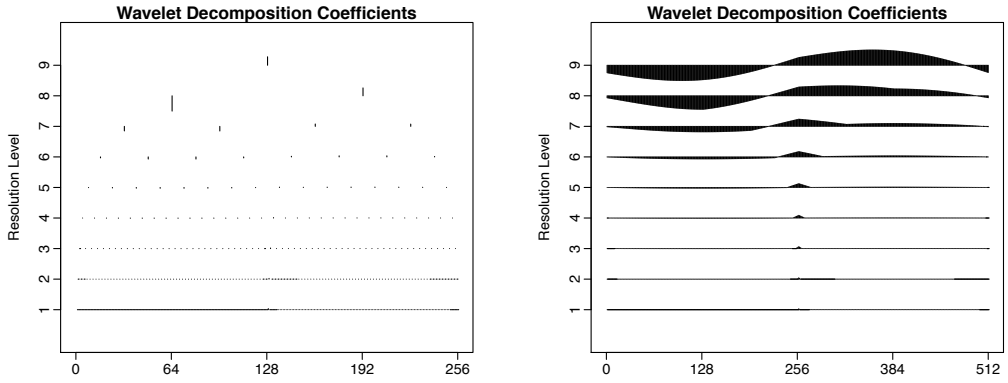
$$c_{j+1,k} = \sum_{n\in\mathbb{Z}} h_n c_{j,n+2k} = \sum_{n\in\mathbb{Z}} h_{n-2k} c_{j,n}. \tag{5.11}$$

In a similar way, we can obtain the detail coefficients $d_{j+1,k}$ at scale $j+1$, from the smooth coefficients at scale j. In this case, instead of using the scaling equation (5.1), we can use the wavelet function (5.1.2) to obtain

$$d_{j+1,k} = \sum_{n\in\mathbb{Z}} g_{n-2k} c_{j,n}. \tag{5.12}$$

The relations in (5.1.3) and (5.1.3) can be applied recursively for $j = 1, \ldots, J$ to obtain the DWT coefficients $\{c_{J,0}, d_{J,0}, d_{J-1,0}, d_{J-1,1}, \ldots, d_{1,0}, \ldots, d_{1,\frac{N}{2}-1}\}$. Figure 5.3a shows the detail coefficients of the DWT for the function f, considered in Figure (5.2), for $J = 8$. We can see that as the scale increases from j to $j+1$, the number of coefficients halves.

It is often notationally convenient to express the operations described by equations



(a) (b)

Figure 5.3: The (a) DWT and (b) NDWT using the Haar wavelet for the function f considered in Figure 5.2.

(5.1.3) and (5.1.3) using operators. Following Nason and Silverman (1995), let $\mathcal{H}$ and $\mathcal{G}$ represent the *convolutions* associated with the filters $\mathbf{h}$ and $\mathbf{g}$ respectively, then for some sequence $\{x_i\}$,

$$(\mathcal{H}x)_k = \sum_{n\in\mathbb{Z}} h_{n-k}x_n,$$
$$\text{and } (\mathcal{G}x)_k = \sum_{n\in\mathbb{Z}} g_{n-k}x_n.$$

Then, in order to obtain the operations described by (5.1.3) and (5.1.3), we could first apply the convolution operators $\mathcal{H}$ and $\mathcal{G}$ to the coefficients $c_{j,n}$ and then choose even elements of the new sequence. This second operation is known as dyadic decimation and this can also be expressed as an operator. Following Nason and Silverman (1995), define the (even) dyadic decimation operator $\mathcal{D}_0$ by $(\mathcal{D}_0x)_k = x_{2k}$. Then, using the convolution operators $\mathcal{H}$ and $\mathcal{G}$, we can re-express equations (5.1.3) and (5.1.3) as

$$c_{j+1} = \mathcal{D}_0\mathcal{H}c_j \text{ and } d_{j+1} = \mathcal{D}_0\mathcal{G}c_j, \tag{5.13}$$

in which it is clear that the length of the sequence at scale $j+1$, is half that at scale

j.

The DWT is an orthogonal transformation, however it is not *translation equivariant*. To see this, note that in equation (5.1.3), we chose to perform an even dyadic decimation step. Instead, we could have chosen to select every odd element of the sequence, however this simple shift leads to a non-trivial change in the wavelet transform. A wavelet transform which is translation equivariant is the non-decimated wavelet transform (NDWT), which we describe in the following section.

5.1.4 Non-decimated Wavelet Transform

In Section 5.1.3 we described the decimated discrete wavelet transform (DWT), in which we highlighted that the transformation, despite being orthogonal, is not translation equivariant. At each scale, we choose to perform an even or an odd decimation step, but how about if we chose to perform both? If we do this, we are able to obtain and make use of extra information from the transform. This is the key idea of the non-decimated wavelet transform (NDWT). Following Nason and Silverman (1995), we can implement the NDWT in the following way.

Let $\mathcal{Z}$ denote the operator which pads out a sequence with zeros as follows:

$$(\mathcal{Z}x)_{2j} = x_j \text{ and } (\mathcal{Z}x)_{2j+1} = 0.$$

Then the NDWT uses filters which are defined recursively as follows:

$$\mathcal{H}^{[0]} = \{h_k\}_{k\in\mathbb{Z}}, \qquad \mathcal{G}^{[0]} = \{g_k\}_{k\in\mathbb{Z}},$$
$$\mathcal{H}^{[r]} = \mathcal{Z}\mathcal{H}^{[r-1]}, \qquad \mathcal{G}^{[r]} = \mathcal{Z}\mathcal{G}^{[r-1]}.$$

Suppose c_j and d_j are the coarse and detail coefficients at scale j respectively. Then

the coarse and detail coefficients at each scale are defined recursively as

$$c_{j+1} = \mathcal{H}^{[j]} c_j \text{ and } d_{j+1} = \mathcal{G}^{[j]} c_j.$$

The decomposition of c_J is then given by the sequence $\{c_J, d_J, d_{J-1}, \ldots, d_1\}$ of length $2^J(J+1)$.

The NDWT retains an equal number of wavelet coefficients at each scale, Figure 5.3b shows the NDWT for the test function in Figure 5.2. This additional information means that at medium and low resolution levels, it is more informative than the DWT. However, this extra information comes at a cost, because the transformation is a redundant, non-orthogonal representation of the original data.

In the following section we turn our attention to the use of wavelets in statistics and in particular, how we can use these non-decimated wavelet transforms to model locally stationary time series.

5.2 Locally Stationary Time Series

The remainder of this thesis applies wavelets in a time series context. This section introduces a wavelet based approach to modelling locally stationary time series introduced by Nason et al. (2000). Section 5.2.2 introduces the locally stationary time series framework we assume in this context. We first provide an introduction to classical (stationary) time series as a basis for the idea of locally stationarity.

5.2.1 Stationary Time Series

Suppose we have an observed (discrete) time series of length n which we denote by $x_1, \ldots, x_n$, and let X_t be the corresponding random variable. Various assumptions

are made about the properties of a time series, to enable the modelling of the autocovariance structure. For example, a *strictly stationary* time series is one for which the probabilistic behaviour of $\{X_{t_1}, \ldots, X_{t_n}\}$ is identical to $\{X_{t_1+h}, \ldots, X_{t_n+h}\}$, for all t_i, n and h. This precise form of stationarity is too strong, and so it is often sufficient to impose an assumption of *weak stationarity* on a time series.

A time series X_t is said to be *second-order (weakly) stationary* if it has constant expectation, $\mathbb{E}(X_t) = \mu$, and its autocovariance is not dependent upon time, i.e. $\gamma(h) = \mathrm{Cov}\,(X_t, X_t + h) = \mathbb{E}[(X_{t+h} - \mu)(X_t - \mu)]$. In practice, we can estimate the auto covariance function using the *sample autocovariance*, defined as

$$\hat{\gamma}(h) = \frac{1}{n}\sum_{t=1}^{n-h}(x_{t+h} - \bar{x})(x_t - \bar{x}), \quad \text{for } h = 0, 1, \ldots, n-1.$$

If a time series is a *stationary stochastic process* then it has the following Fourier representation (Priestley, 1988)

$$X_t = \int_{-\pi}^{\pi} A(\omega)\exp(i\omega t)\mathrm{d}\xi(\omega), \tag{5.14}$$

where $A(\omega)$ is the amplitude of the process and $\mathrm{d}\xi(\omega)$ is an orthonormal increments process. In equation (5.2.1), the amplitude $A(\omega)$ does not depend on time. Realistically, for many applications, assuming that a time series is stationary over time is a misconception. It may be reasonable however to assume that in some window the series is stationary, however globally this property may not hold. This means, for many applications, model (5.2.1) is not appropriate and time dependence needs to be introduced.

There are many ways to represent local stationarity stemming from Dahlhaus et al. (1997). Nason et al. (2000) enable time dependence by replacing the set of Fourier functions $\{\exp(i\omega t)\}$, $\omega \in (-\pi, \pi)$, by a set of discrete non-decimated wavelets. We

describe the model of Nason et al. (2000) in the following section.

5.2.2 Locally Stationary Wavelet (LSW) Processes

In order to capture the time dependence within a locally stationary setting, Nason et al. (2000) introduce the compactly supported discrete wavelets.

Definition 5.2.1. *Let* **h** *and* **g** *be the low and high pass quadrature mirror filters used in the construction of wavelet functions as described in Section 5.1.2. Nason et al. (2000) construct the compactly supported* discrete *wavelets* $\psi_j = (\psi_{j,0}, \ldots, \psi_{j,(N_j-1)})$ *of length* N_j *for scale* $j < 0$ *using the following:*

$$\psi_{1,n} = \sum_k g_{n-2k}\delta_{0,k} = g_n, \quad \textit{for } n = 0, \ldots, N_1 - 1,$$

$$\psi_{j,n} = \sum_k h_{n-2k}\psi_{j,k} = g_n, \quad \textit{for } n = 0, \ldots, N_j - 1,$$

$$N_j = (2^j - 1)(N_h - 1) + 1.$$

Here $\delta_{0,k}$ *is the Kronecker delta, and* N_h *is the number of non-zero elements of* h_k*. We define the quantity* $\psi_{j,k}(\tau)$ *to be* $\psi_{j,k-\tau}$*, the* $(k-\tau)^{\text{th}}$ *element of the vector* ψ_j*.*

As we describe below, Nason et al. (2000) use discrete *non-decimated* wavelets to construct locally stationary stochastic processes. These permit a wavelet to appear at each time point at each scale, so that $\psi_{j,k}(\tau) = \psi_{j,k-\tau}$.

Following Nason et al. (2000), a LSW process $\{X_{t,T}\}_{t=0,\ldots,T-1}$, for a dyadic length of time $T = 2^J \geq 1$, is a doubly indexed stochastic process having the following representation in the mean-square sense:

$$X_{t,T} = \sum_{j=1}^{J} \sum_k \omega_{j,k;T}\psi_{j,k}(t)\xi_{j,k}, \tag{5.15}$$

where $\{\psi_{j,k}(t) = \psi_{j,k-t}\}_{j,k}$ is a set of discrete non-decimated wavelets and the parameter j represents the scale of the corresponding wavelet. The $\{\omega_{j,k;T}\}$ are a set of amplitudes or weights, which may be interpreted as a transfer function. Finally, $\{\xi_{j,k}\}$ is a random orthonormal increment sequence.

The use of the notation $X_{t,T}$ rather than the traditional X_t is to emphasize the triangular stochastic array across different T, although in practice dependence on T is often suppressed within notation.

Nason et al. (2000) specify three sets of assumptions required of model (5.2.2). The first and second assumptions concern the orthonormal increment process $\xi_{j,k}$, and the third is placed upon the amplitudes $\{\omega_{j,k}\}$:

1. $\mathbb{E}[\xi_t] = 0$;
2. $\text{cov}(\xi_{j,k}, \xi_{l,m}) = \delta_{j,l}\delta_{k,m}$;
3. For each scale j there exists a Lipschitz function $W_j(z) : [0, 1) \to \mathcal{R}$ such that:

   - $\sum_{j=1}^{\infty} |W_j(z)|^2 < \infty$ where $z \in (0, 1)$ is rescaled time $z = \frac{k}{T}$;
   - $\sum_{j=1}^{\infty} 2^j L_j < \infty$;
   - $\exists$ constants C_j satisfying $\sum_{j=1}^{\infty} C_j < \infty$ such that, for each T,

$$\sup_{k=0,\ldots,T-1} |\omega_{j,k;T} - W_j(k/T)| \leq C_j/T.$$

The first assumption ensures that $\{X_{t,T}\}_{t=0,\ldots,T-1}$ is a zero mean process. The second assumption means that the orthonormal increment sequence $\xi_{j,k}$ is uncorrelated. This results in a complete description of the dependence structure in $\omega_{j,k;T}$. The final set of assumptions control the evolution of the weights $\omega_{j,k}$, ensuring that they can change over time but this must happen slowly.

The non-decimated wavelet system is over-complete and hence the coefficients $\omega_{j,k}$ cannot be uniquely determined. However, the assumptions that Nason et al. (2000)

place upon the LSW model, described above, allow for the *asymptotic evolutionary wavelet spectrum* (EWS) to be determined uniquely. The EWS measures the local power in an LSW process. It is given by

$$S_j(z) := |W_j(z)|^2 = \lim_{T\to\infty} |\omega_j(z)|^2,$$

on the rescaled time interval $z = k/T \in (0, 1)$.

The wavelet spectrum $S_j(k/T) = |W_j(k/T)|^2$ is estimated from the raw wavelet periodogram, given by the squares of the detail coefficients of the non-decimated wavelet transform:

$$I_{j,k} := |d_{j,k}|^2 = \left| \sum_t X_{t,T} \psi_{j,k}(t) \right|^2 .$$

The vector of periodograms is hence given by $\mathbf{I}_k := \{I_{j,k}\}_{j=1,\ldots,J}$.

Due to the redundancy of the non-decimated wavelet transform, the wavelet spectrum is biased. In fact, as (Nason et al., 2000, Proposition 4) establish, the expectation of $I_{j,k}$ is given by

$$\mathbb{E}(I_{j,k}) = \sum_l A_{j,l} S_l(z) + \mathcal{O}(T^{-1}) \qquad \forall z \in (0, 1,).$$

Here the operator $A_{j,l}$ is the *inner product of the autocorrelation wavelets*: $A_{j,l} := \langle \Psi_j, \Psi_l \rangle = \sum_\tau \Psi_j(\tau)\Psi_l(\tau)$ where the autocorrelation wavelets are defined by

$$\Psi_j(\tau) := \sum_k \psi_{j,k}(0)\psi_{j,k}(\tau). \tag{5.16}$$

Hence, in order to obtain an unbiased estimate of the wavelet periodogram, we need to correct the periodogram by multiplying by the inverse of the inner product matrix

of autocorrelation wavelets.

$$\mathbf{L}_k := \{L_{j,k}\}_{j=1,\ldots,J} = A_J^{-1}\mathbf{I}_k.$$

In addition to bias, the raw wavelet periodogram is also an inconsistent estimator. This is due to the periodogram's asymptotically non-vanishing variance. Nason et al. (2000) show that the variance of the wavelet periodogram is given by

$$\text{var } I_{j,k} = 2\left\{\sum_l A_{jl}\right\}.$$

Therefore in order to obtain consistency, the wavelet periodogram needs to be smoothed. In practice, Nason et al. (2000) recommend that we smooth and then correct as this is theoretically easier to analyse. Nason et al. (2000) also recommend using wavelet shrinkage to smooth the wavelet periodogram, whilst more recent work by Fryzlewicz and Nason (2006) suggests using wavelet-Fisz transforms. Please see (Nason, 2010, Chapter 6) for further details.

As we consider detecting changes in the local autocovariance estimates in Chapter 7, we introduce here the time varying measure of the autocovariance of a time series as described in Nason et al. (2000).

For a locally stationary wavelet process with evolutionary wavelet spectrum $\{S_j(z)\}$, the local autocovariance function is given by

$$c(z,\tau) = \sum_{j=1}^{\infty} S_j(z)\Psi_j(\tau).$$

Here the $\Psi_j(\tau)$ are the autocorrelation wavelets at lag τ (5.2.2). Nason et al. (2000) show that the autocovariance of $X_{t,T}$, defined by,

$$c_T(z,\tau) = \text{cov}(X_{\lfloor zT\rfloor,T}, X_{\lfloor zT\rfloor+\tau,T}) \tag{5.17}$$

converges to $c(z, \tau)$ as $T \to \infty$.

Having focussed on non-stationary time series we now turn to consider the closely related challenge of changepoint estimation for piecewise stationary time series.

5.3 Changepoint Detection using Locally Stationary Wavelet Models

In this section we summarise the literature surrounding changepoint detection using the LSW model. We focus specifically on detecting changes in second order structure using the LSW framework. This is one, of several potential changepoint approaches using wavelet methods. However, for the purposes of this thesis we will focus on LSW-based approaches due to its explicit modelling of non-stationary time series structure. Those interested in learning about the other wavelet-based methods are referred to: Whitcher et al. (2000), who detect changes in variance in long memory processes using binary segmentation and the non-decimated wavelet transform to estimate the location of changes; Gabbanini et al. (2004) use the Ljung-Box test for autocorrelation on packets from the Discrete Wavelet Packet Transform, see (Nason, 2010, Section 2.11); Fernandez (2004) uses the cumulative sum of the wavelet variance to detect changes in variance.

We begin our review by describing a particular extension of the LSW model of Nason et al. (2000). This allows us to model processes whose second-order structure evolves over time in a discontinuous fashion. Specifically, Fryzlewicz and Nason (2006) extend the LSW model of Nason et al. (2000) to include time series with a piecewise second order structure. Following Fryzlewicz and Nason (2006)'s modification, a triangular stochastic array $\{X_{t,T}\}_{t=0,\ldots,T-1}$, for a dyadic length of time $T = 2^J \geq 1$, is a LSW

process if there exists a mean-square representation

$$X_{t,T} = \sum_{j=1}^{J} \sum_{k} W_{j,k;T} \psi_{j,k}(t) \xi_{j,k}, \tag{5.18}$$

where $\{\psi_{j,k}(t) = \psi_{j,k-t}\}_{j,k}$ is a set of discrete non-decimated wavelets and the parameter j represents the scale of the corresponding wavelet. The $\{W_{j,k;T}\}$ are a set of time varying amplitudes or weights, each of which is a real-valued piecewise constant function with a finite number of jumps which is unknown a priori. The $\xi_{j,k}$ in (5.3) are zero-mean, orthonormal, identically distributed random variables ensuring $\{X_{t,T}\}$ is a zero mean process. Further, Fryzlewicz and Nason (2006) denote the total magnitude of jumps in $\{W^2_{j,k;T}\}$ by $\mathcal{P}_j$. Consequently, condition 3 in Definition 5.2.2 is replaced with the following assumptions to control the variability of $\{W_{j,k;T}\}$.

1. $\sum_{j=1}^{\infty} W^2_{j,k;T} < \infty$ uniformly in $z = k/T$;
2. $\sum_{j=1}^{J} 2^j \mathcal{P}_j = \mathcal{O}(\log T)$ where $J = \log_2 T$.

The above representation lends itself to modelling time series with discontinuous second-order structure. This is the framework adopted by Cho and Fryzlewicz (2012) to detect changes in second order structure. The methodology relies on the premise that if a time series has piecewise second order structure, then its evolutionary wavelet spectrum will be piecewise constant. Hence, a change in the autocovariance of a time series, will result in a changepoint in at least one of the wavelet periodogram scales.

Cho and Fryzlewicz (2012) apply a BS algorithm to the wavelet periodograms separately at each scale. Once they have done this, to attain consistency in the changepoint locations, they perform a within-scale and across-scale post-processing procedure. Most binary segmentation routines for multiplicative models do not allow for correlated data, see for example (Inclan and Tiao, 1994) and (Chen and Gupta, 1997). However, the wavelet periodogram has a scaled χ^2 distribution, and so Cho and Fry-

zlewicz (2012)'s implementation of BS necessarily allows for correlated data.

Killick et al. (2013) extend the work of Cho and Fryzlewicz (2012) to a parametric likelihood setting. This is motivated by a number of reasons. Firstly, Cho and Fryzlewicz (2012)'s method relies heavily on a variance stabilisation step. As such, if the stabilisation step does not make the variance constant across time, the method begins to break down because the assumptions placed upon the test statistic are violated. Additionally, the method requires the specification of a range of parameters, each of which influence the final result. Finally, it is often the case that a parametric test statistic will outperform a non-parametric equivalent when the modelling assumptions are reasonable.

Killick et al. (2013)'s likelihood approach involves detecting a single change in second order structure using a likelihood ratio test. The log-likelihood is expressed in terms of the wavelet spectrum using the definition of the covariance of an LSW process (5.2.2). Killick et al. (2013) extend this into the multiple changepoint case using BS.

Other advances in the LSW literature include the work of Cho and Fryzlewicz (2015), in which the authors introduce a multivariate extension to the LSW framework in order to detect changes in autocovariance in high dimensional data. In parallel they also develop a new modification to binary segmentation, termed "Sparsified Binary Segmentation" (SBS). The sparsification step in the algorithm consists of only using some of the information regarding changepoints from each of the time series sequences. Before each of the CUSUM statistics for the time series are aggregated, they apply a threshold to each of them, such that any sequences that do not meet the threshold, and so do not contain changepoints, are excluded from the aggregation. They point out that this characteristic is particularly useful in a high dimensional setting. They also improve on Cho and Fryzlewicz (2012) by achieving better rates of convergence for the location estimators of changepoints.

More recently, Korkas and Fryzlewicz (2017) use another modification to BS, namely Wild Binary Segmentation (WBS) developed by Fryzlewicz et al. (2014), in the same setting as in Cho and Fryzlewicz (2012). The motivation for using this modified algorithm is that it outperforms standard BS in cases where there are many changepoints present in the model.

In order to apply the WBS algorithm to the wavelet periodogram, Korkas and Fryzlewicz (2017) have to adapt the procedure for a multiplicative model setting, where the observations are scaled χ^2 random variables. Similarly to Cho and Fryzlewicz (2012) they include an across-scale post-processing step, essentially used for aggregation of the scales. They suggest two methods for this, the first is similar to that used in Cho and Fryzlewicz (2012) and the other is motivated by that used in Cho and Fryzlewicz (2015). The two methods of aggregation can also be used when the standard BS algorithm is implemented.

The work presented in the remainder of this thesis extends the work of Cho and Fryzlewicz (2012) and Killick et al. (2013) into the time domain. Instead of detecting changes in the wavelet periodogram, we consider the local autocovariance function. One elegance of this, is the ability to consider the cases of changes in variance and autocovariance separately. Consequently, Chapter 6 introduces our method for detecting changes in variance and in Chapter 7 we extend this to the case of the autocovariance.

Chapter 6

A Nonparametric Approach to Detecting Changes in Variance in Locally Stationary Time Series

6.1 Introduction

In this chapter we introduce a non-parametric method for detecting changes in variance in the presence of outliers and heavy tails. Data sequences are often prone to outliers and/or heavy tail structures which the majority of approaches are intolerant to. Typically some pre-processing of the data is often performed in an attempt to mitigate these effects (Candemir and Oğuz, 2017). In some cases this is a straightforward adaptation, however given the unprecedented volume of data now being generated, pre-processing is becoming increasingly impractical and often subjective (Taleb et al., 2015). This motivates the need for new methods that are inherently resilient to such features.

The structure of this chapter is as follows. In Section 6.2 we introduce our non-

parametric approach to change in variance detection. This approach is based upon the Locally Stationary Wavelet (LSW) model of Nason et al. (2000), described previously in Section 5.2. The LSW framework is used to provide a local estimate of the variance of a time series. The method we present is then assessed under various simulation scenarios in Section 6.3. Lastly, in Section 6.4 we apply our method to wind speed data collected from a site in the UK.

6.2 A Nonparametric Approach to Detecting Changes in Variance

In this section we describe our non-parametric method for detecting changes in variance. Our approach is based on the key insight that detecting a change in variance in the time domain can be transformed into detecting a change in mean in a transformed domain, given a suitable transformation. We are by no means the first to consider this, see for example, Darkhovski (1994); Inclan and Tiao (1994). In contrast to this earlier work we adopt a wavelet based approach.

6.2.1 Locally Stationary Wavelet Framework

Our method for detecting changes in variance relies upon capturing the local behaviour of a time series' variance. This could be achieved using a rolling window estimate of the variance, but would require choice of a window size. Instead we choose to adopt the locally stationary wavelet (LSW) framework which is built upon non-decimated wavelets.

The advantage of the LSW framework is that it encompasses many common time series processes, such as moving average and autoregressive processes. Of particular

interest for this work, we can use the LSW framework to attain a local time-varying measure of the variance.

Suppose we have a LSW process, X_t, which has a representation as in equation (5.3). Then, the variance of X_t is given by

$$\text{var}(X_t) = \sum_{j,k} W_j^2(k/N)\psi_{j,k-t}^2. \tag{6.1}$$

The dependence on time in equation (6.2.1) is introduced indirectly via the compact support of the wavelet. Using the wavelet spectrum, Nason et al. (2000) introduce the **localized variance function** for a LSW process of length $N = 2^J$. This is defined to be

$$\sigma^2(z) = \sum_{j=1}^{J} S_j(z), \tag{6.2}$$

where $z = k/N \in (0, 1)$ is rescaled time. If a time series has a constant variance, then the dependence on z in equation (6.2.1) is lost and the localised measure becomes a global one.

The time-varying estimate of the variance (6.2.1) can be interpreted as a windowed rolling estimate of the variance of the time series. However, unlike a usual rolling estimate, no consideration of the window length is required. The benefit of a wavelet approach is that a variety of window sizes are used in the wavelet transform. Through the compact support of the wavelets the representation in equation (6.2.1) is unique given the wavelet (Nason et al., 2000).

Figure 6.1 shows an example of a process with (a) constant variance and (b) piecewise variance and their associated smoothed and unsmoothed local variance functions in (c), (d) and (e), (f) respectively. Figure 6.1(c) demonstrates that smoothing the spectral estimate masks the abrupt change that is clearly visible in (b) and (f). For this reason, the following section presents a method based on the unsmoothed localised

variance.

6.2.2 The NPLE Method

As previously described, if a time series is second order stationary then its evolutionary wavelet spectrum will be constant across each scale. Similarly, if a time series is piecewise second order stationary, then the spectrum will be piecewise constant (Fryzlewicz and Nason, 2006). Consequently, as the localised variance function in equation (6.2.1) is the sum of the spectrum over scales, this means that the localised variance function will also be piecewise constant. In order to exploit this property for changepoint detection, we need to translate it into a practical setting. Thus our estimate of the un-smoothed local variance function is defined as

$$\hat{\sigma}^2(z) = \sum_{j=1}^{J} \sum_{l=1}^{J} A_{j,l}^{-1} d_{l,z}^2. \tag{6.3}$$

Recall, that the $A_{j,l}$ are the inner products of the autocorrelation wavelets (5.2.2), and the $d_{l,z} = \sum_{t=1}^{N} X_t \psi_{l,z-t}$, are the empirical wavelet coefficients of an LSW process $X_{t,N}$.

Due to the compact support of the wavelets it is clear that, for a signal with piecewise constant variance, this estimate is also piecewise constant. The following section outlines the method for detecting these changes in the localised variance.

The Nonparametric Model

The localised variance function (6.2.2) is a sum of correlated χ^2 random variables. In practice it is difficult to obtain the distribution for this (Gordon and Ramig, 1983). Therefore, we choose to adopt a non parametric approach to this changepoint detection problem.

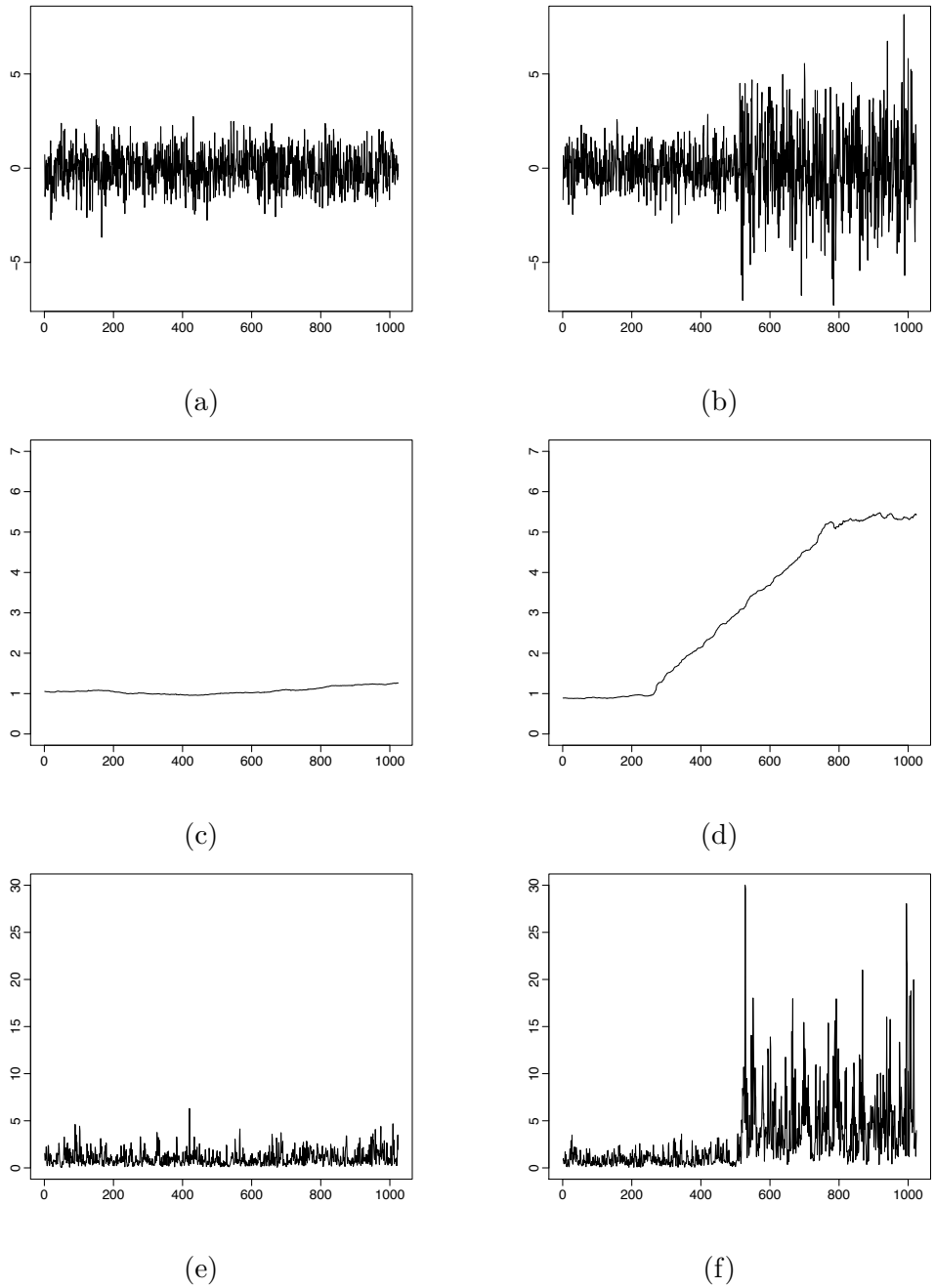


Figure 6.1: A time series with (a) constant variance (b) piecewise variance with their associated smoothed local variance function in (c) and (d) respectively, and their unsmoothed local variance function in (e) and (f) respectively.

We consider the localized variance, σ^2, and model its cumulative distribution function, $G(u) = \mathbb{P}(\sigma^2 \leq u)$, for quantile u using the empirical CDF

$$\hat{G}(u) = \frac{1}{n}\left(\sum_{t=1}^{n} \mathbf{I}_{\{\hat{\sigma}_t^2 < u\}} + \frac{1}{2}\mathbf{I}_{\{\hat{\sigma}_t^2 = u\}}\right),$$

where the $\hat{\sigma}_t^2$ are assumed to be independent and $\mathbf{I}(\cdot)$ is the indicator function.

Then, for n i.i.d data points with CDF $G(u)$, for a fixed value of u, the empirical CDF satisfies $n\hat{G}(u) \sim \text{Binomial}(n, G(u))$. Hence, following Zou et al. (2014), the maximum log likelihood of $G(u)$ is given by

$$n\{\hat{G}(u)\log\hat{G}(u) + (1-\hat{G}(u))\log(1-\hat{G}(u))\}.$$

In order to identify changepoints, we can take a penalised cost function approach (Section 2.4) and minimise the following

$$\sum_{i=1}^{m+1}\left[\mathcal{C}(\hat{\sigma}^2_{\{\tau_{i-1}+1\}:\tau_i})\right] + \beta f(m),$$

where the cost function for segment i is given by the negative of the empirical log likelihood of the CDF of the localised variance estimate:

$$-\mathcal{L}(\hat{\sigma}^2_{\{\tau_{i-1}+1\}:\tau_i}; u) = (\tau_i - \tau_{i-1}) \times \left[\hat{G}_i(u)\log\hat{G}_i(u) + (1-\hat{G}_i(u))\log\left(1-\hat{G}_i(u)\right)\right].$$

The above cost function only uses information about the CDF evaluated for a single value of u. This choice of u can result in differing segmentations. To overcome this, Zou et al. (2014) recommend an integrated form of the cost function:

$$\int_{-\infty}^{\infty} -\mathcal{L}(\hat{\sigma}^2_{\{\tau_{i-1}+1\}:\tau_i}; u)\; dw(u), \tag{6.4}$$

where $w(\cdot)$ is a weight function, dependent upon the CDF of the data set, such that the integral is finite. The consistency of this approach is detailed in Zou et al. (2014). This allows information across all time points to be incorporated into the cost function.

The computational cost of the cost function suggested by Zou et al. (2014) is of order $\mathcal{O}(Mn^2+n^3)$, where M is a specified maximum number of changepoints (Haynes et al., 2017b). Zou et al. (2014) suggest a screening step to help reduce this computational time; however this jeopardizes the accuracy of the locations of the changepoints.

Haynes et al. (2017b) suggest an improved segment cost that involves approximating the integral in (6.2.2) by a sum with some fixed number of terms K. This improves the computational time taken to calculate the cost for a given segment to $\mathcal{O}(\log n)$. The suggested approximation is as follows.

Following Haynes et al. (2017b), we fix a K and define $\gamma = \frac{-log(2n-1)}{K}$. Time is then rescaled according to quantiles dependent upon the choice of K. Let $\{t_k\}_{k=1,\ldots,K.}$ be equal to the $(1 + (2n-1)\exp\{\gamma(2k-1)\})^{-1}$ empirical quantile of the data. The approximation to the integral in equation (6.2.2) is then given by:

$$\mathcal{C}_K(\hat{\sigma}^2_{\{\tau_{i-1}+1\}:\tau_i}) = \frac{2\log(2n-1)}{K}\sum_{k=1}^{K}\mathcal{L}_{np}(\hat{\sigma}^2_{\{\tau_{i-1}+1\}:\tau_i}; t_k). \tag{6.5}$$

We could use any search function in order to identify the changepoints using equation (6.2.2). However, Haynes et al. (2017b) show that this cost function is compatible with PELT (Killick et al., 2012), a computationally efficient search for changepoints. We therefore use this search method in our simulation study in Section 6.3.

Based upon the above description, we choose to call the method outlined here Non-Parametric change in variance detection using Localised Estimates, abbreviated to NPLE.

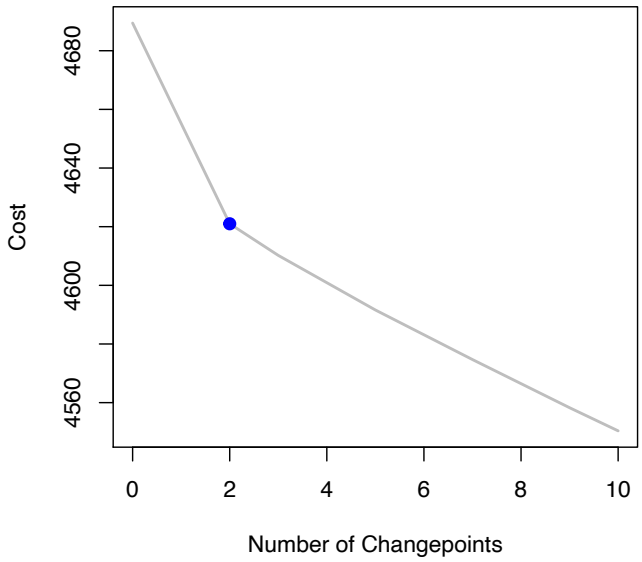


Figure 6.2: Example plot of the number of changepoints against the cost function for a model with two changes in variance. From the plot we can correctly identify the true number of changes to be two.

6.2.3 Penalty Choice

Penalty choice is a practical challenge in many changepoint settings. We choose to take an adaptive approach to penalty selection following that of Lavielle (2005). Intuitively, this approach involves selecting the segmentation which causes the most significant decrease in the cost function. This can be presented graphically in an analogous way to a scree plot used in Principal Components analysis (Jolliffe, 2002). Figure 6.2 shows an example plot of a cost function against the number of changepoints identified for a model with two true changepoints. It is visible that the true segmentation occurs at the point of maximum curvature, or 'elbow', of the plot; where the largest relative decrease in the cost function occurs. The procedure of identifying this 'elbow' can be formalized, and automatized, as follows.

In line with Lavielle (2005), let m_{MAX} be an upper bound on the number of change-

points in the model. The PELT search algorithm results in a single optimal segmentation for a given penalty value. In order to obtain segmentations for a range of penalty values efficiently we utilize the CROPS method (Haynes et al., 2017a). From this range of segmentations we then wish to determine $\hat{m}$; the estimated number of changepoints in the model. Following Lavielle (2005), we obtain $\hat{m}$ using the following procedure:

1. For $0 \leq m \leq m_{\text{MAX}}$ let

$$\tilde{\mathcal{J}}_m = \frac{\mathcal{J}_{m_{\text{MAX}}} - \mathcal{J}_m}{\mathcal{J}_{m_{\text{MAX}}} - \mathcal{J}_0} m_{\text{MAX}} + 1,$$

   where $\mathcal{J}_m$ is the cost for the segmentation corresponding to m changepoints at locations $\tau_{1:m}$. The associated costs have now been normalised between 1 and $m_{\text{MAX}} + 1$.

2. Then, for $1 \leq m \leq m_{\text{MAX}} - 1$, let

$$D_m = \tilde{\mathcal{J}}_{m-1} - 2\tilde{\mathcal{J}}_m + \tilde{\mathcal{J}}_{m+1},$$

   and $D_1 = \infty$.

3. The estimate for the true location of the changepoint is given by the largest value of m such that the second derivative of $\mathcal{J}_m$, D_m, is greater than some threshold S,

$$\hat{m} = \max \{0 \leq m \leq m_{\text{MAX}} - 1 | D_m > S\}.$$

The above procedure has also been implemented for penalty choice in a wavelet context by Killick et al. (2013). The intuition behind this approach is that true changes will be added to the segmentation first as they will result in the largest improvement to the cost function. Following this we will start to add spurious changes to the data,

which are just due to noise, and so the improvement in fit will be small. The aim of the choice of S is to put a threshold on the rate of change in the scaled test statistic as the number of changes increases. For an individual dataset we would do this using the changepoint equivalent of a scree plot.

6.3 Simulation Study

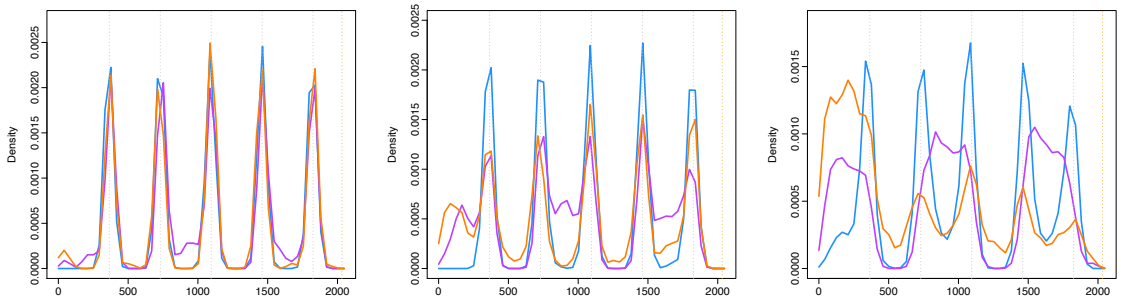
In the following simulation study we test the robustness of NPLE against the log likelihood of a Normal distribution with changing variance (MLvar) (Chen and Gupta, 2013) and the non parametric Cumulative Sums of Squares (CSS) (Inclan and Tiao, 1994). This allows for a comparison between both a parametric and non-parametric method. Each of these are implemented using the `changepoint` package (Killick et al., 2015; Killick and Eckley, 2014) in R (R Core Team, 2018). For the calculation of the localised variance estimate we utilize the `wavethresh` package (Nason, 2012). The study also considers departures from the idealised Normal distribution change in variance setting. Specifically, the simulations study provides a practical assessment of the resilience to departures from Normality, including outliers and heavy tailed dependence structure.

6.3.1 Random Outliers

In this first study we seek to test how each of the methods performs with varying degrees of outliers. To this end, we simulate time series with different proportions of outliers. Specifically, we simulate epidemic changes in variance, $\sigma = (1, 3, 1, 3, 1, 3)$ from a Normal distribution of length 2048 with changes at $365i$ for $i = 1, \ldots, 5$. The timing of the outliers are simulated from a Unif$(1, 2048)$ distribution. To create outliers at these time points, we add a fixed constant, 15, to the existing observations.

	P	0	1	2	3	4	5	6	7	≥ 8
NPLE	0.00%	0.00	0.00	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00
	0.01%	0.00	0.00	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00
	1.00%	0.00	0.00	0.00	0.00	0.02	**0.98**	0.00	0.00	0.00
	5.00%	0.01	0.00	0.01	0.08	0.07	**0.63**	0.08	0.12	0.00
MLvar	0.00%	0.00	0.00	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00
	0.01%	0.00	0.00	0.00	0.01	0.01	**0.74**	0.04	0.20	0.00
	1.00%	0.07	0.03	0.06	0.14	0.13	**0.32**	0.13	0.12	0.00
	5.00%	0.28	0.00	0.21	0.06	0.17	**0.06**	0.11	0.11	0.00
CSS	0.00%	0.00	0.00	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00
	0.01%	0.00	0.00	0.01	0.28	0.00	**0.66**	0.00	0.01	0.04
	1.00%	0.00	0.00	0.12	0.20	0.02	**0.28**	0.05	0.10	0.23
	5.00%	0.00	0.01	0.14	0.11	0.04	**0.10**	0.00	0.07	0.53

Table 6.1: Proportion of changepoints detected for different percentages of outliers.



(a) $P = 0.01\%$ (b) $P = 1\%$ (c) $P = 5\%$

Figure 6.3: Density of detected changepoint locations using (blue) NPLE, (purple) MLvar and (orange) CSS, when the percentage of outliers is equal to (a) 0.01% (b) 1% and (c) 5%.

We repeat this for $P = 0.01\%, 1\%$ and 5% density of outlying observations within each data set as well as the no outlier case for comparison. The choice to use additive outliers instead of multiplicative outliers means that the size of the outliers will vary less across segments with differing variances.

Table 6.1 shows the number of changepoints detected by each of the methods for the four values of P over 500 repetitions. As expected, the performance of each method degrades as the percentage of outlying values increases. However, this degradation is not uniform across the methods. NPLE detects the correct number of changepoints 63% of the time when 5% of observations are outliers, in comparison, CSS achieves a similar rate when only 0.01% of observations are outliers.

Figure 6.3 shows the density of detected changepoint locations for each of the methods for P equal to 0.01%, 1% and 5%. NPLE maintains accurate changepoint locations as P increases, whereas the other methods are drawn to the outliers.

The results of these simulations demonstrate that NPLE is less sensitive to outliers than the other methods. When using MLvar, and similarly CSS, the outliers contribute to both the likelihood and the sum of squares directly and distort the estimates.

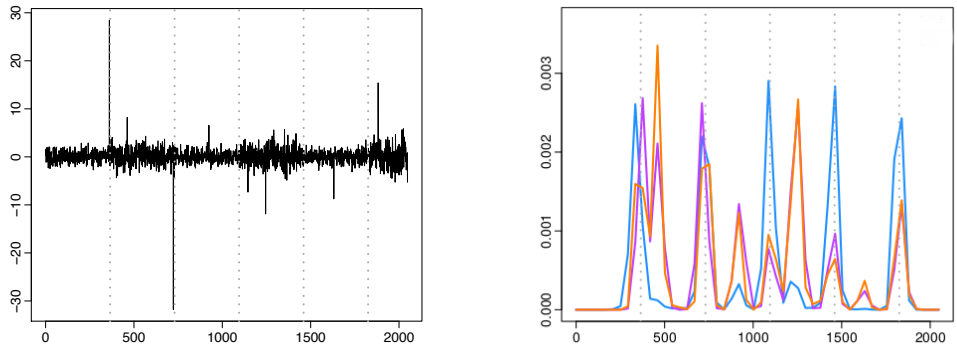
In the next simulation study, we consider another model with outliers, however they are located at fixed points in time.

6.3.2 Fixed Outliers

In this section we test the robustness of the model for increasingly sized changes in variance, using variance changes that are more difficult to detect than those in Section 6.3.1. We simulated 500 repetitions of a Normal distribution of length 2048 with changepoints at $365i$ for $i = 1, \ldots, 5$ and $\sigma = (1, 1.6, 1, 1.8, 1, 2)$. We also consider the effect of proximity of outliers to changepoint locations. Hence, we introduce multiplicative outliers located at times (361, 462, 723, 924, 1244, 1630, 1881) with inflation factors (20, -20, 16, 18, 20, 10, 7).

Figure 6.4a shows a realisation of this model where it is important to note the location of the outlier in relation to the location of the changepoint. The first and third outliers occur close to changepoint locations; whereas the remaining outliers are firmly within segments. Despite the locations of the outliers being fixed, in comparison to the uncertain locations in Section 6.3.1, the size of the outliers are more variable as a consequence of their multiplicative nature.

Figure 6.4b shows the density of detected changepoints and Table 6.2 gives the cor-



(a) Realisation (b) Changepoint Locations

Figure 6.4: For the outliers model, (a) a realisation of the data and (b) density plots for detected changes in variance using (blue) NPLE (purple) MLvar and (orange) CSS for the outliers model.

responding numbers of changepoints detected. NPLE detects the true number of changes 87% of the time, whereas MLvar and CSS achieve only 13% and 14% respectively.

Turning consideration to the locations of the changes. For the first change, for MLvar and CSS, the presence of the outliers near the changepoint means that there are two distinct peaks corresponding the location of the change. This is not the case for NPLE, but the outlier appears to result in the change being detected slightly early. At the third change, MLvar and CSS often detect a change either side of the true changepoint location.

All three methods perform similarly when detecting the second change.

Despite being the largest changes, the last three are detected correctly the least by MLvar and CSS, this is probably a consequence of the methods detecting a larger number of changes elsewhere, induced by the outliers. The large outliers at 462, 924, and 1244 have clearly resulted in spurious changes for both MLvar and CSS.

Our final simulation study considered data which instead of having outliers, exhibits heavy tail behaviour.

	0	1	2	3	4	5	6	7	≥ 8
NPLE	0.00	0.00	0.00	0.02	0.00	**0.82**	0.01	0.15	0.00
MLvar	0.02	0.01	0.19	0.04	0.30	**0.13**	0.15	0.15	0.00
CSS	0.09	0.00	0.22	0.02	0.21	**0.14**	0.11	0.21	0.00

Table 6.2: Proportion of changepoints detected for the outliers model.

	ξ	0	1	2	3	4	5	6	7	≥ 8
NPLE	0.00	0.00	0.00	0.00	0.01	0.01	0.11	0.02	**0.85**	0.00
	0.25	0.00	0.00	0.00	0.03	0.04	0.19	0.08	**0.57**	0.08
	0.45	0.00	0.02	0.00	0.06	0.08	0.16	0.15	**0.31**	0.22
MLvar	0.00	0.00	0.01	0.00	0.02	0.01	0.13	0.02	**0.80**	0.01
	0.25	0.00	0.03	0.02	0.11	0.04	0.20	0.10	**0.30**	0.20
	0.45	0.00	0.06	0.08	0.12	0.08	0.20	0.11	**0.15**	0.2
CSS	0.00	0.00	0.00	0.00	0.01	0.00	0.06	0.00	**0.87**	0.05
	0.25	0.02	0.08	0.01	0.12	0.02	0.17	0.02	**0.33**	0.22
	0.45	0.06	0.07	0.09	0.15	0.05	0.16	0.07	**0.15**	0.19

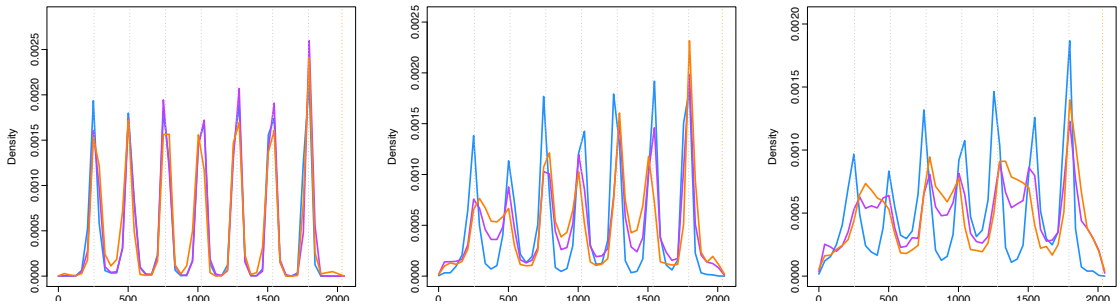
Table 6.3: Proportion of changepoints detected for the simulated Generalised Extreme Value data.

6.3.3 Heavy Tail Structure

In this section we consider data which is generated from a Generalised Extreme Value (GEV) distribution, with zero mean ($\mathbb{E}(X_t) = 0$), that exhibits varying changes in variance. The changes are located at times $256i$, $i = 1 \dots 7$ and the sequence of standard deviations is given by $\sigma = (1, 1.6, 1, 1.8, 1, 2, 1, 2.5)$. We consider three values of the shape parameter for the GEV distribution: $0, 0.25$ and 0.45. Note that as σ is a function of the shape and scale parameters, we keep the shape constant and only modify the scale across the segments to obtain the required σ. The tails become heavier as the shape parameter, ξ, increases across simulations.

Table 6.3 shows the number of changepoints detected and Figure 6.5 show the corresponding densities for the locations. As expected, as the tails become heavier the detection rate decreases for all methods. Whilst they all perform similarly for $\xi = 0$ as the shape parameter increases, NPLE is most resilient to the heavy tails providing around double the detection rate as $\xi = 0.25$ and 0.45.

This illustrates NPLE's reduced sensitivity to heavy tailed distributions. For MLvar



(a) $\xi = 0$ (b) $\xi = 0.25$ (c) $\xi = 0.45$

Figure 6.5: Density plots for detected changes in variance using (blue) NPLE (purple) MLvar and (orange) CSS for 500 realisations of simulated Generalised Extreme Value data.

we are using a Gaussian assumption and so we expect this method to perform poorly but CSS does not have any tail assumptions.

6.4 Application to Wind Speed Characterization

We now turn to consider the detection of variance changepoints within a time series of wind speeds. The data we analyse were obtained at a UK wind farm location during November 2005. Each measurement represents the average wind speed obtained from an anemometer at the farm. The series contains 4261 observations, as depicted in Figure 6.6a.

Changepoint methods have been used extensively to derive insight for a number of important environmental and ecological applications. See, for example, the important work of Andersen et al. (2009); Evans et al. (2016); Hilborn et al. (2017); Richardson et al. (2018). Here, we consider the problem of detecting changes in variance within wind speed data related to challenges arising within the renewable energy sector. Specifically, in recent years there has been an increasing focus on detecting damage in wind turbine blades. As Chou et al. (2013) report, damage to these blades can cause up to 19.4% of wind turbine damage. Such damage can be caused by various factors

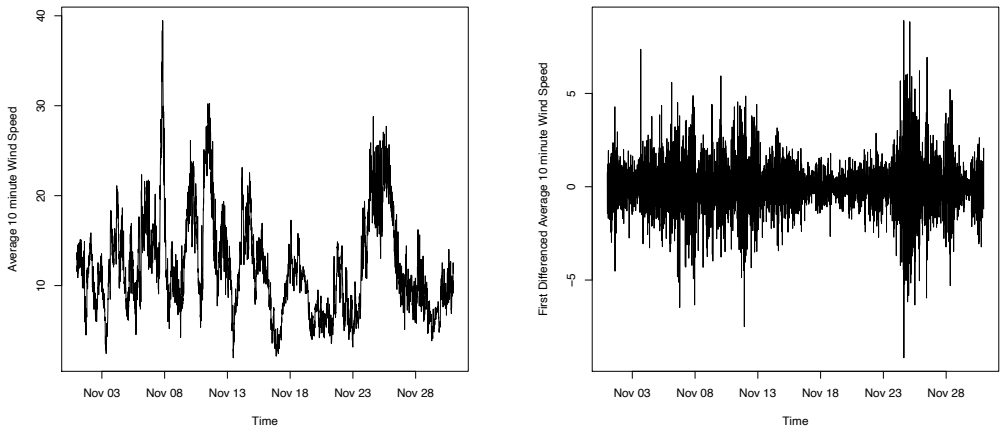
including severe environmental conditions such as gusty winds, lightening strikes and storms (Herr and Heidenreich, 2015; Hoell and Omenzetter, 2015). Amongst a variety of different analyses one might undertake, it may be of interest to segment the wind speed observed at a given location into regions of differing variability to allow better understanding of the wind gusts experienced by the turbine. Data of this form may be heavy tailed, and subject to outliers.

To explore whether any changes in variance exist within this wind speed data, we begin by taking first differences to remove the mean behaviour. The resulting series has a very clear, non-constant variance structure (Figure 6.6b). There also appear to be some anomalous observations that could potentially affect changepoint estimation. Next, we apply both the NPLE and MLvar methods to the differenced wind speeds. To provide a fair comparison between the methods we use the Lavielle (2005) method for penalty choice for both methods. The diagnostic plots are give in Figure 6.7 where it is clear that the elbow in the curve for NPLE is at 9 changes and for MLvar is at 8 changes.

The resulting changepoint plots for NPLE and MLvar are given in Figure 6.8. Note, in particular, how MLvar appears to be inflating the variance estimate for the first segment of data in response to the anomalous points. This results in a later changepoint than the NPLE method which chooses to use two changepoints to capture the period of smaller variability. For operational decisions the segmentation provided by NPLE is preferred.

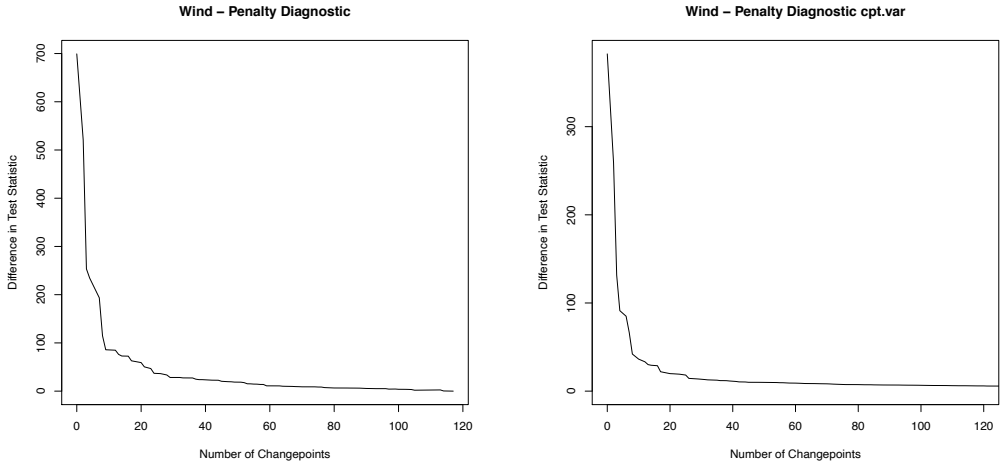
6.5 Conclusion

In this Chapter we have introduced a novel changepoint detection procedure to detect changes in variance (NPLE). The key benefits of our nonparametric approach are its



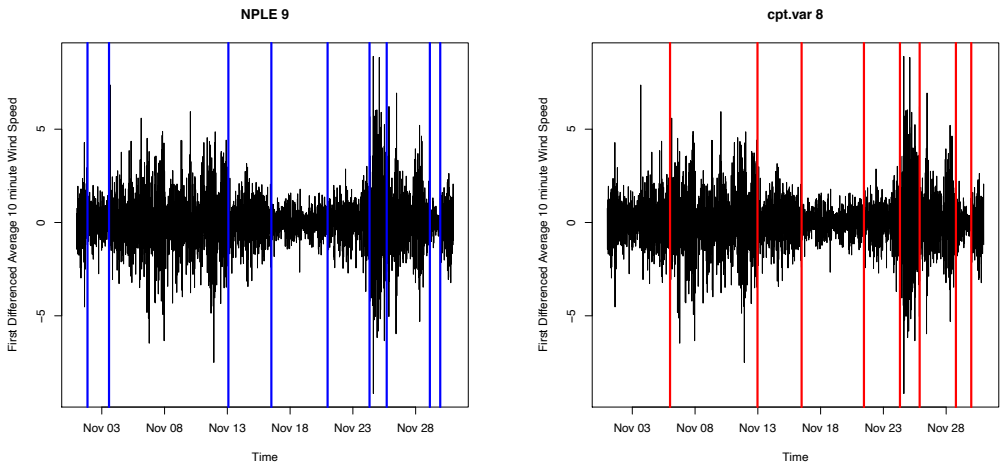
(a) (b)

Figure 6.6: (a) Original Wind Speed data, (b) Difference of the data from (a).



(a) (b)

Figure 6.7: Diagnostic plots for (a) NPLE and (b) MLvar.



(a) (b)

Figure 6.8: Changepoint plots for (a) NPLE with 9 changes and (b) MLvar with 8 changes following the method in Lavielle (2005).

capacity to provide changepoint estimates that are resilient to outliers and departures from normality.

This method is shown to perform well against an established nonparametric method (CSS) and penalised likelihood approaches (MLvar) in all simulated settings. We also considered the utility of NPLE on data obtained from a UK wind farm. In Chapter 7 we extend our approach to detect changes in the local autocovariance.

Chapter 7

A Nonparametric Approach to Detecting Changes in Autocovariance in Locally Stationary Time Series

7.1 Introduction

In Chapter 6 we considered the problem of detecting changes in the variance of a locally stationary time series. In this chapter we extend the work presented in Chapter 6 to the case of changes in autocovariance.

The problem of detecting changes in autocovariance structure has been studied relatively little in the literature. Notable contributions include the likelihood approach taken by Davis et al. (2006) and Gombay (2008). These two approaches are specifically manufactured for detecting changes in autoregressive (AR) models. Davis et al. (2006) introduce an Auto-PARM algorithm in which the test statistic is a penalised

likelihood ratio and the penalty is the Minimum Description Length (MDL). The solution space is searched using a genetic algorithm.

In contrast, previous non-parametric approaches to detecting changes in autocovariance include Ombao et al. (2001) and Ahamada et al. (2004), the latter of which is considered in a Fourier setting. More recently Cho and Fryzlewicz (2012) and Killick et al. (2013) propose methods based in the locally stationary wavelet setting, the former is non-parametric and the latter is parametric, each of these adopt a binary segmentation (BS) (Scott and Knott, 1974) approach to changepoint detection. Most recently, Korkas and Fryzlewicz (2017) propose an improvement to the approach of Cho and Fryzlewicz (2012) by using an adaptation of the binary segmentation algorithm which they call wild binary segmentation (WBS).

In this chapter we extend the work of Chapter 6 by developing a non-parametric method of detecting changes in autocovariance in the presence of outliers. Our method relies on modelling the data as a locally stationary wavelet (LSW) process; the framework of which is built upon non-decimated wavelets. Our approach differs to that of Killick et al. (2013) and Korkas and Fryzlewicz (2017) in that we use the LSW framework to obtain a local measure of the autocovariance function over time. Hence, within this setting, we conduct the changepoint analysis in the time domain and not in the frequency domain.

The chapter is structured as follows, in Section 7.2 we describe our method for detecting changes in autocovariance. This is based upon the Locally Stationary Wavelet model of Nason et al. (2000), previously described in Chapter 5. The method is then tested with various simulation studies in Section 7.3. Lastly, Section 7.4 applies our method to Telematics data collected from a car journey.

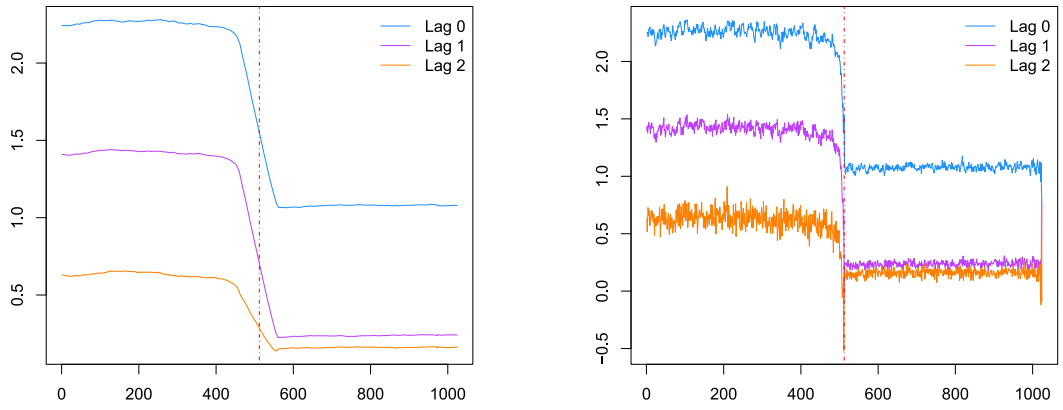


Figure 7.1: The (a) smoothed and (b) unsmoothed local autocovariance function for a piecewise MA(2) model.

7.2 A Nonparametric Approach to Detecting Changes in Autocovariance

The Locally Stationary Wavelet (LSW) process has the ability to capture many dependence structures. In particular, piecewise second-order stationarity can be captured from piecewise constant sequences in the local wavelet periodogram. This feature has already been used by Cho and Fryzlewicz (2012), Killick et al. (2013) and Fryzlewicz et al. (2014) to detect changes in the second-order structure of a time series. In addition to this, piecewise second-order stationarity can also be captured from a piecewise constant local autocovariance function. Here we outline methodology for detecting changes in second-order structure using the local autocovariance function.

Figure 7.1 shows the local autocovariance function for a time series with piecewise constant autocovariance. Note that the local autocovariance estimates in Figure 7.1a are not strictly piecewise due to the slope induced around the change in autocovariance at time 512. This is a consequence of smoothing the local autocovariance estimates to obtain consistency. Figure 7.1b shows the unsmoothed estimates which has no such slope. Consequently, for detecting changes in autocovariance, we will use an estimate

of the unsmoothed local autocovariance function.

For a LSW time series, $\{X_t\}_{t=0,\ldots,N-1}$, of length $N = 2^J$, the estimated unsmoothed **local autocovariance function** at lag ν is defined as:

$$\hat{c}_z(\nu) = \sum_{j=1}^{J} \sum_{l=1}^{J} A_{j,l}^{-1} d_{l,z}^2 \Psi_j(\nu). \tag{7.1}$$

Here $A_{j,l}$ is the inner product of the autocorrelation wavelets: $\Psi_j(\nu) := \sum_k \psi_{j,k}(0)\psi_{j,k}(\nu)$. The $d_{l,z}$ are the detail coefficients of the discrete non-decimated wavelet transform: $d_{l,z} := \sum_t X_{t,N}\psi_{l,z}(t)$. Finally, $z = k/N \in (0,1)$, is rescaled time.

For a time series with piecewise second order structure, the estimate in equation (7.2) will be piecewise constant for at least one $\nu \in \{0,\ldots\}$. It is likely that the change in the local autocovariance function will occur for a multitude of lags simultaneously. For example, the autocovariance of an autoregressive model decays to zero as the lag increases. The rate of this decay is dependent upon the coefficients of the model. Hence, any changes in these coefficients which cause changes in the autocovariance at multiple lags. Therefore, when we perform change detection, we will detect changes simultaneously in the local autocovariance estimates for a range of lags $\nu \in \{0,\ldots,\nu_{\text{MAX}}\}$. The following section outlines the method for detecting these changes in the local autocovariance function.

7.2.1 The Non-parametric Model

The local autocovariance function is a weighted sum of correlated chi-squared random variables. As such, modelling the distribution of this weighted sum of this correlated sequence of random variables is notoriously complex and often this sum is approximated (Chuang and Shih, 2012). Alternatively, Nason (2013) shows that the smoothed local autocovariance estimates are approximately normal due to the asymp-

totic Gaussianity of the running mean smoother used to obtain consistency. However, because we are choosing not to smooth the estimates of the local autocovariance, the results from Nason (2013) do not hold. Therefore, we choose to use a non-parametric log-likelihood to model the local autocovariance function and extend the approach outlined in Chapter 6. We summarise this in the following.

In Chapter 6 we detected changes in the local variance function of a LSW time series using the ED-PELT algorithm of Haynes et al. (2017b) (Section 6.2). This methodology is a special case of detecting changes in the local autocovariance. This is because the local variance is the lag zero case of the local autocovariance.

With the above in mind, instead of only detecting changes in the local variance estimates, we extend the methodology outlined in Section 6.2 and detect changes (simultaneously) in all lags of the local autocovariance function. That is, we aim to minimise the following:

$$\sum_{i=1}^{m+1} \sum_{\nu=0}^{\nu_{\max}} \left[\mathcal{C}(\hat{c}_{\{\tau_{i-1}+1\}:\tau_i}(\nu))\right] + \beta f(m). \tag{7.2}$$

Where the cost function for segment i at lag ν is given by the negative of the log-likelihood of the empirical CDF of the localised autocovariance estimate (6.2.2). We minimise equation (7.2.1) using a multivariate implementation of the ED-PELT algorithm.

We call this extension to the method presented in Chapter 6, Non-Parametric change detection using Localised AutoCovariance Estimations, abbreviated to NP-LACE.

7.3 Simulation Study

In this simulation study we compare the performance of NP-LACE, against the wavelet-based likelihood (WL) method of Killick et al. (2013), Auto-PARM (AP) presented by Davis et al. (2006) and the wild binary segmentation method of Korkas and Fryzlewicz (2017), referred to as KF. We replicate simulations from those presented in Killick et al. (2013), and in addition to these studies, we include the same models but subject the processes to outliers. This will allow us to assess the relative robustness of the models.

In all of the simulations presented, we report results using the Haar wavelet, however similar results were obtained using different wavelets. For each of the simulation scenarios, we choose the maximum lag for NP-LACE, $\nu_{\max}$, to be three. We use an adaptive penalty following that of Lavielle (2005) (described in Section 6.2.3). For Auto-PARM, we use the default values as specified in Davis et al. (2006). For the methods outlined in Korkas and Fryzlewicz (2017) we use the R package `wbs` (Korkas and Fryzlewicz, 2018), in which the number of intervals drawn for WBS is selected to be a linear function of the sample size of the time series. For calculation of the local autocovariance estimate, we use the `wavethresh` package (Nason, 2012) in R (R Core Team, 2018).

In each case we report both the location and the number of changepoints. Tables 7.1, 7.2 and 7.3 report the number of changepoint detected in each scenario and Figure 7.3 displays the density of identified changepoints for each of the models. We detail the scenarios below.

(A) Stationary AR(1) process with no changepoints This scenario is designed to test the methods when there are no changepoints in the process and to evaluate the

Table 7.1: Results for scenario (A). We report the percentage of repetitions that identified that number of changepoint with the true number in bold. Note: NP-LACE has been abbreviated to NP.

Model A								
num	-0.7				-0.1			
cps	NP	WL	AP	KF	NP	WL	AP	KF
0	**97**	**100**	**100**	**94**	**100**	**100**	**100**	**94**
1	2	0	0	1	0	0	0	5
≥2	1	0	0	5	0	0	0	1
no. of	0.4				0.7			
cps	NP	WL	AP	KF	NP	WL	AP	KF
0	**91**	**100**	**100**	**95**	**91**	**91**	**100**	**91**
1	5	0	0	2	6	9	0	9
≥2	4	0	0	3	3	0	0	0

extent to which they identify false positives. We simulate from the following model

$$X_t = aX_{t-1} + \epsilon_t, \tag{7.3}$$

for a range of parameter values a.

Table 7.1 shows the number of changepoint detected for Model A using each of the methods. Auto-PARM performs best. NP-LACE performs better when the coefficient in (7.3) is smallest and is comparable to WL and Auto-PARM. Generally, KF detects the largest number of false positives.

(B) Piecewise stationary AR process with clearly observable changes We simulate from the following model

$$X_t = \begin{cases} 0.9X_{t-1} + \epsilon_t & \text{if } 1 \leq t \leq 512, \\ 1.68X_{t-1} + -0.81X_{t-2} + \epsilon_t & \text{if } 513 \leq t \leq 768, \\ 1.32X_{t-1} - 0.81X_{t-2} + \epsilon_t & \text{if } 769 \leq t \leq 1024. \end{cases}$$

From the results in Table 7.2 we can see that for Model B, NP-LACE and Auto-PARM detect the true number of changes 94% of the time, in the cases that the incorrect number of changes are identified, NP-LACE tends to underestimate the number of

Table 7.2: Results for scenarios (B)-(D). We report the percentage of repetitions that identified that number of changepoint with the true number in bold. Note: NP-LACE has been abbreviated to NP.

num	Model B				Model C				Model D			
cps	NP	WL	AP	KF	NP	WL	AP	KF	NP	WL	AP	KF
0	5	0	0	0	8	0	0	0	12	4	0	14
1	1	0	0	12	3	0	0	1	**87**	**95**	**100**	**60**
2	**94**	**98**	**94**	**53**	**88**	**94**	**100**	**73**	1	1	0	16
3	0	2	6	22	1	6	0	20	0	0	0	10
≥4	0	0	0	13	0	0	0	6	0	0	0	0

changes whereas Auto-PARM overestimates the number of changes. WL detects the true number of changes 98% of the time and KF often overestimates the number of changes and detects the true number around half of the time.

From the density plot in Figure 7.3 (a) it can be seen that all of the methods capture the location of the first change equally well with KF doing poorly on the second change, this can be attributed to KF overestimating the number of changes.

(C) Piecewise stationary AR process with less clearly observable changes We simulate from the following model

$$X_t = \begin{cases} 0.4X_{t-1} + \epsilon_t & \text{if } 1 \leq t \leq 400, \\ -0.6X_{t-1} + \epsilon_t & \text{if } 401 \leq t \leq 612, \\ 0.5X_{t-1} + \epsilon_t & \text{if } 613 \leq t \leq 1024. \end{cases}$$

The changes in this model are less clear and no longer occur at dyadic locations in time. From Figure 7.3 we can see that all of the methods perform similarly in terms of the locations of the changes detected. KF identifies extra changes in the centre of the true changepoints more often than the other methods. Table 7.2 shows that KF is overestimating the number of changes and NP-LACE underestimates the number of changes 11% of the time.

(D) Piecewise stationary AR process with a short segment We simulate from

Table 7.3: Results for scenarios (E)-(G). We report the percentage of repetitions that identified that number of changepoint with the true number in bold. Note: NP-LACE has been abbreviated to NP.

num	Model E				Model F				Model G			
cps	NP	WL	AP	KF	NP	WL	AP	KF	NP	WL	AP	KF
0	29	0	0	2	6	0	0	0	0	0	0	0
1	28	21	9	12	7	20	51	12	**92**	**100**	**99**	**94**
2	**40**	**51**	**33**	**19**	27	22	33	32	3	0	1	6
3	2	24	31	23	**60**	**35**	**16**	**49**	5	0	0	0
≥4	1	4	27	44	0	23	0	7	0	0	0	0

the following model

$$X_t = \begin{cases} 0.75X_{t-1} + \epsilon_t & \text{if } 1 \leq t \leq 50, \\ -0.5X_{t-1} + \epsilon_t & \text{if } 51 \leq t \leq 1024, \end{cases}$$

This model has a short segment at the beginning of the time series. From Figure 7.3 (c) we can see that NP-LACE has the highest density for the changepoint location, however we see that NP-LACE catches the change slightly early. On average, NP-LACE captures the changepoint at $t = 42$. In Table 7.2 we can see that NP-LACE underestimates the number of changes 12% of the time, KF identifies the true number of changes 60% of the time and both underestimates and overestimates the number of changes frequently.

(E) Piecewise stationary AR process with high autocorrelation We simulate from the following model

$$X_t = \begin{cases} 1.399X_{t-1} - 0.4X_{t-2} + \epsilon_t, & \epsilon_t \sim \mathcal{N}(0, 0.8^2) \quad \text{if } 1 \leq t \leq 400, \\ 0.999X_{t-1} + \epsilon_t, & \epsilon_t \sim \mathcal{N}(0, 1.2^2) \quad \text{if } 401 \leq t \leq 750, \\ 0.699X_{t-1} + 0.3X_{t-2} + \epsilon_t, & \epsilon_t \sim \mathcal{N}(0, 1.2^2) \quad \text{if } 751 \leq t \leq 1024, \end{cases}$$

For this model all of the methods struggle to detect the true number of changes, see Table 7.3. Model E is close to being non-stationary within segments. For this model, the density of changepoint locations for NP-LACE, Figure 7.3 (d), is different to WL

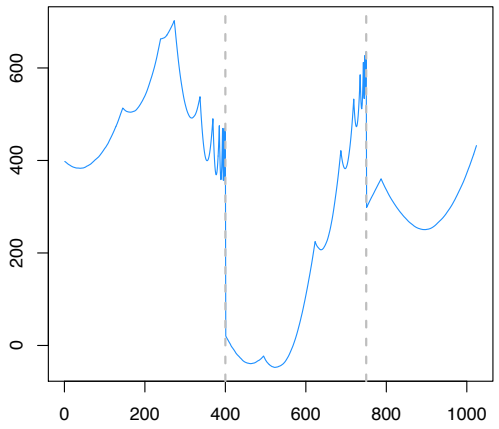


Figure 7.2: The local autocovariance function for Model E.

and KF, with four peaks instead of two. Auto-PARM has a similar peak to NP-LACE at the beginning of the time series and WL seems to also be detecting some changes around $t = 600$. If we inspect the local autocovariance function for Model E in Figure 7.2 we can see that does not have the piecewise constant structure our method relies upon.

(F) Piecewise stationary ARMA(1,1) process We simulate from the following model

$$X_t = \begin{cases} 0.7X_{t-1} + \epsilon_t + 0.6\epsilon_{t-1} & \text{if } 1 \leq t \leq 125, \\ 0.3X_{t-1} + \epsilon_t + 0.3\epsilon_{t-1} & \text{if } 126 \leq t \leq 352, \\ 0.9X_{t-1} + \epsilon_t & \text{if } 353 \leq t \leq 704, \\ 0.1X_{t-1} + \epsilon_t - 0.5\epsilon_{t-1} & \text{if } 705 \leq t \leq 1024. \end{cases}$$

This model is different to the others because it incorporates a moving average term. Figure 7.3 (e) shows the density of changepoint locations for Model F and Table 7.3 displays the number of changepoints detected. All of the methods prefer the last change to the other two - this is because the change in autocovariance is largest- and in general they all struggle to detect the correct number of changes. NP-LACE detects

the true number of changes more often than the other methods, Auto-PARM detects the last change the best, but this is somewhat attributed to it underestimating the number of changes 84% of the time.

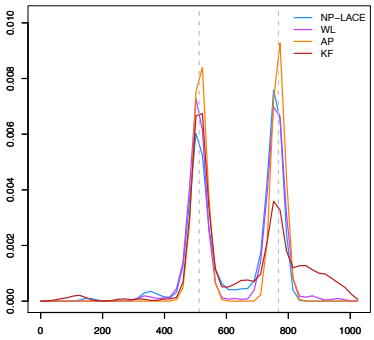
(G) Piecewise stationary MA process

$$X_t = \begin{cases} \epsilon_t + 0.8\epsilon_{t-1} & \text{if } 1 \leq t \leq 128, \\ \epsilon_t + 1.68\epsilon_{t-1} - 0.81\epsilon_{t-2} & \text{if } 129 \leq t \leq 256. \end{cases}$$

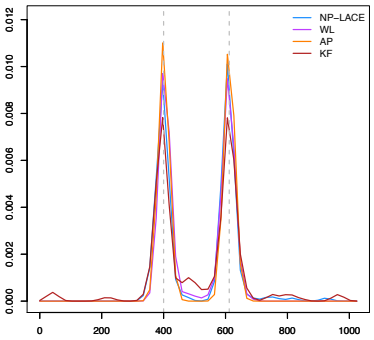
For the moving average process all methods are comparable with AP and NP-LACE most accurately identifying the location of the change, shown in Figure 7.3 (f). In Table 7.3, we can see that NP-LACE identifies the true number of changes the least at 92% of the time. However, this may be a consequence of the choice of maximum lag considered for the study. For consistency we chose ν_{MAX} to be three. In the case of autoregressive processes, this is suitable because long memory is induced, resulting in changes being present in each lag of the local autocovariance function. However, for a moving average model of order q, the estimates of the local autocovariance function at lags greater than q, should be zero. We revisit this in Appendix A.

In summary, the simulation study shows that the fully non-nonparametric NP-LACE method is comparable with the other methods presented. We perform particularly well in Model F, the ARMA model, and in the case of a short segment. However, NP-LACE is inclined to underestimate the number of changepoints. This could be rectified by lowering the adaptive penalty threshold selected in the methodology of Lavielle (2005), however this would increase the number of false positives obtained in the no changepoint scenarios.

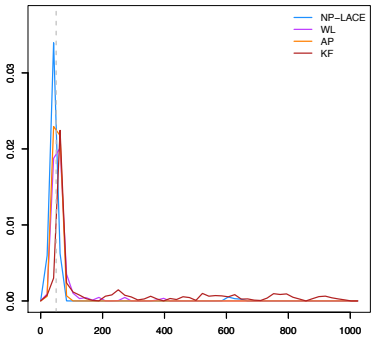
In the following, we turn our attention to including outliers in the models we have tested.



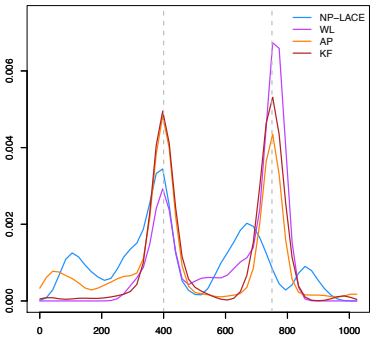
(a) Model B



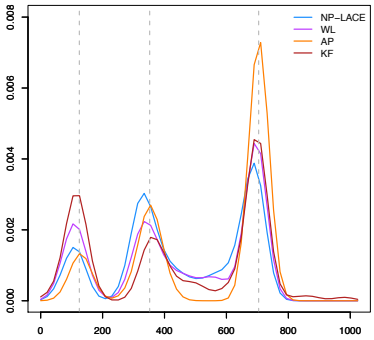
(b) Model C



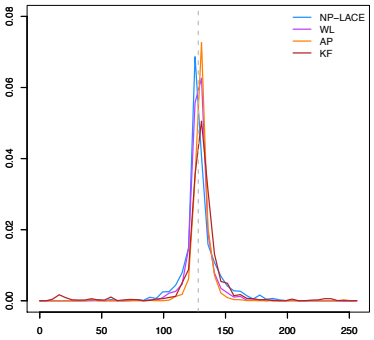
(c) Model D



(d) Model E



(e) Model F



(f) Model G

Figure 7.3: Density of changepoint locations detected for Models (B)-(G) using from our method (NP-LACE), the wavelet-based likelihood method (WL) from Killick et al. (2013), Auto-PARM (AP) and KF from Korkas and Fryzlewicz (2017).

Outliers

In order to test how the methods perform when the data exhibits outliers, for the models (B)-(G) above, we replace 1% of each of the data sets with additive outliers, the locations of which are drawn randomly from a Uniform(a,b) distribution where $a = 1$ and b is equal to the length of the data set.

The density of the changepoint locations in the case of models (B) to (G) are shown in Figure 7.4 and the associated number of changepoints detected are given in Tables 7.4 and 7.5. When the data is exposed to outliers we can see that the accuracy of NP-LACE is largely unaffected. As a consequence of the outliers, the other methods tend to overestimate the number of changepoints.

From Figure 7.4 we can see that WL tends to find spurious changes at either end of the time series. Auto-PARM also has some resilience to outliers, in terms of the location of the changepoints, for Models C, D and E.

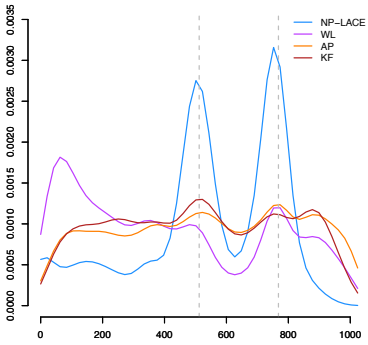
Table 7.4: Results for scenarios (B)-(D) when subjected to 1% outliers. Note: NP-LACE has been abbreviated to NP.

num	Model B				Model C				Model D			
cps	NP	WL	AP	KF	NP	WL	AP	KF	NP	WL	AP	KF
0	5	0	0	0	4	0	0	0	18	0	0	2
1	4	0	0	0	3	2	0	0	**28**	**1**	**0**	**1**
2	**68**	**1**	**0**	**0**	**26**	**0**	**0**	**1**	3	1	0	3
3	2	1	0	0	16	0	0	7	14	1	0	14
≥4	21	98	100	100	51	98	100	92	37	97	100	80

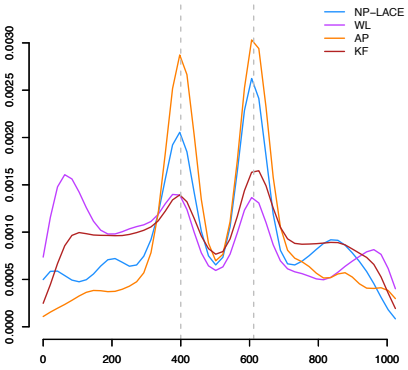
Table 7.5: Results for scenarios (E)-(G) when subjected to 1% outliers. Note: NP-LACE has been abbreviated to NP.

num	Model E				Model F				Model G			
cps	NP	WL	AP	KF	NP	WL	AP	KF	NP	WL	AP	KF
0	11	0	0	0	2	0	0	0	15	0	0	4
1	16	0	0	0	41	0	0	0	**60**	**1**	**0**	**34**
2	**33**	**0**	**0**	**0**	31	1	0	0	12	6	0	39
3	13	0	0	2	**15**	**0**	**0**	**0**	13	9	0	19
≥4	27	100	100	98	11	99	100	100	0	84	100	4

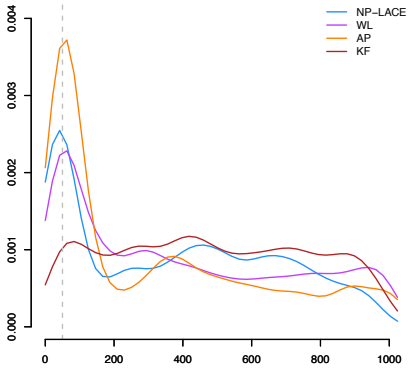
The results of this simulation study show that NP-LACE has more resilience to outliers



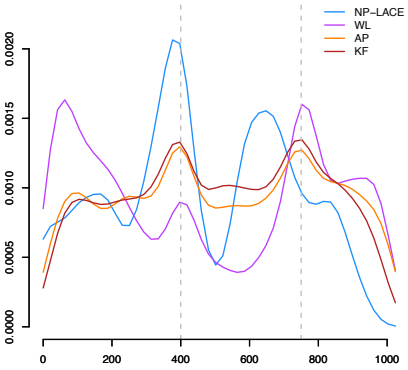
(a) Model B



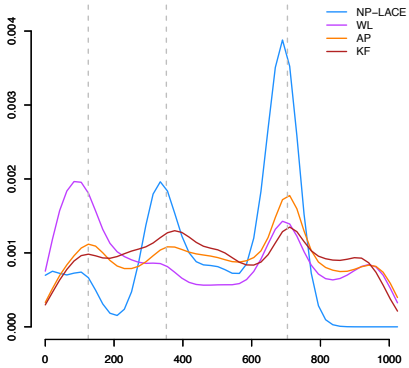
(b) Model C



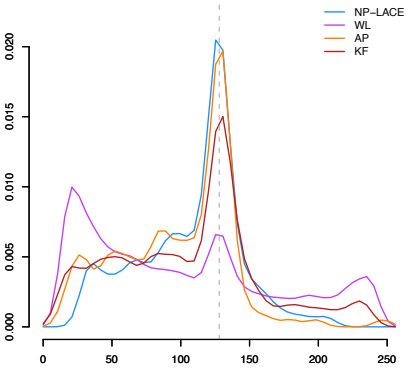
(c) Model D



(d) Model E



(e) Model F



(f) Model G

Figure 7.4: Density of changepoint locations detected for Models (B)-(G), when 1% of the observations are replaced with additive outliers, from our method (NP-LACE), the wavelet-based likelihood method (WL) from Killick et al. (2013), Auto-PARM (AP) and KF from Korkas and Fryzlewicz (2017).

than the other methods considered.

In the following section, we apply each of the method to detecting changes in autocovariance in Telematics data.

7.4 Application to Telematics Data

We now consider detecting changes in the autocovariance of acceleration data from a car journey. Figure 7.5 shows an example of acceleration data for a car journey of approximately 55 miles beginning in Lancaster and ending in the Lake District.

The data was received as longitude and latitude coordinates with associated time stamps. For each pair of coordinates we used the `geosphere` R package (Hijmans et al., 2017) to calculate the distance between them. This transformation highlighted that the data had been transmitted equidistantly. Then, in order to obtain acceleration data, we divided the distances by the squared change in time.

From visually inspecting the data, it appears that there may be outliers. Telematics data is often prone to outliers due to multipath propagation (Mikulski, 2013).

Figure 7.6 shows the route of the car journey and the detected changes in autocovariance for (a) NP-LACE, (b) WL, (c) Auto-PARM, and (d) KF.

From visually inspecting the maps in Figure 7.6, NP-LACE *appears* to pick out the changes in driving behaviour due to road type. In contrast, the segmentations of the other methods do not appear to have any physical meaning. NP-LACE detects no changes within the motorway segment of the journey and the two regions where it detects three changes together are locations of roundabouts. The other methods place changes within the motorway part of the journey due to the presence of outliers in the acceleration data. This robustness of NP-LACE to outliers was demonstrated in

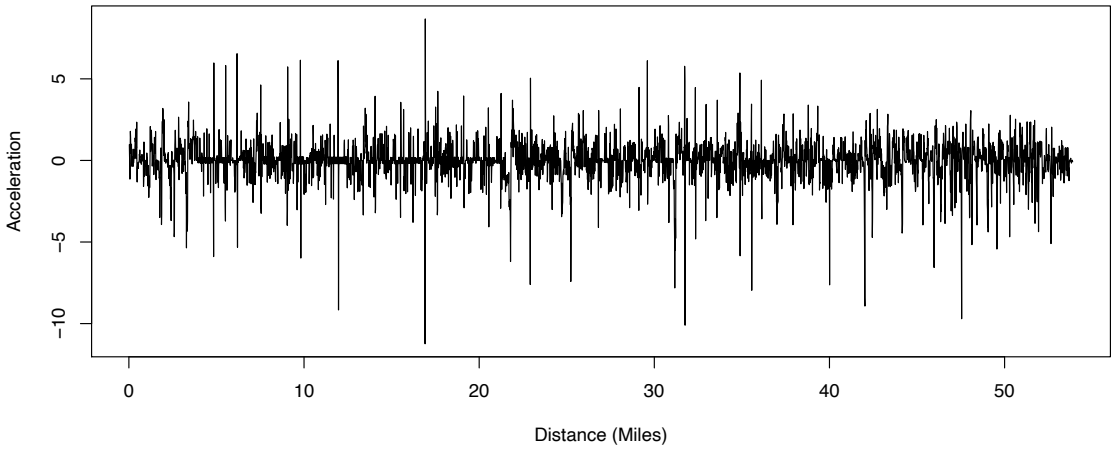


Figure 7.5: Acceleration data for a car journey beginning in Lancaster and ending in the Lake District.

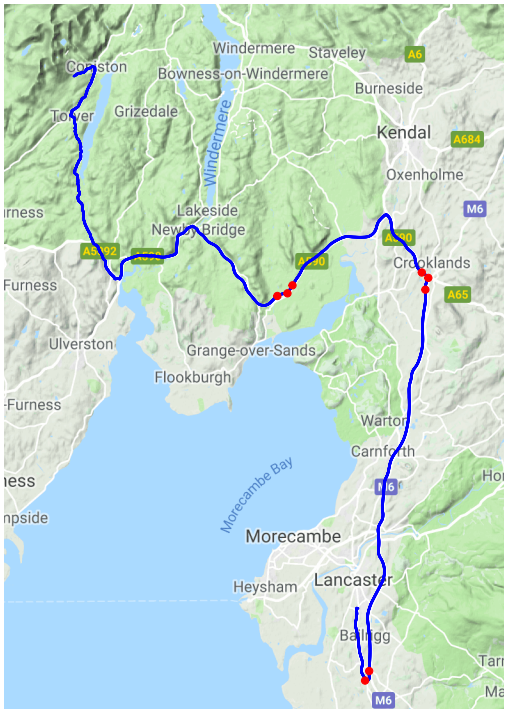
the simulation study in Section 7.3.

The results suggest that detecting changes in the second-order structure of acceleration data could represent the transitions between types of road. However, more research would be required, and spatial variations and dynamics would need to be carefully considered, in order to validate such an approach.

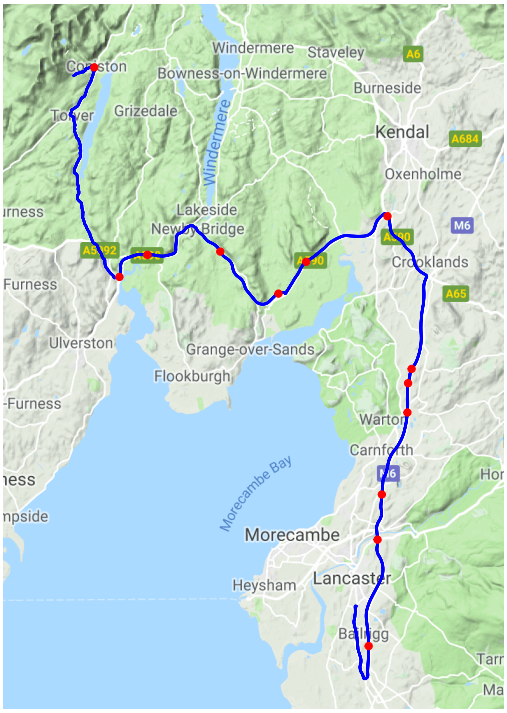
7.5 Conclusion

In this chapter we have extended the methodology from Chapter 6 and developed a method (NP-LACE) for segmenting a time series that does not assume independence of the observations. It is non-parametric and, further, does not require us to impose a time series model form.

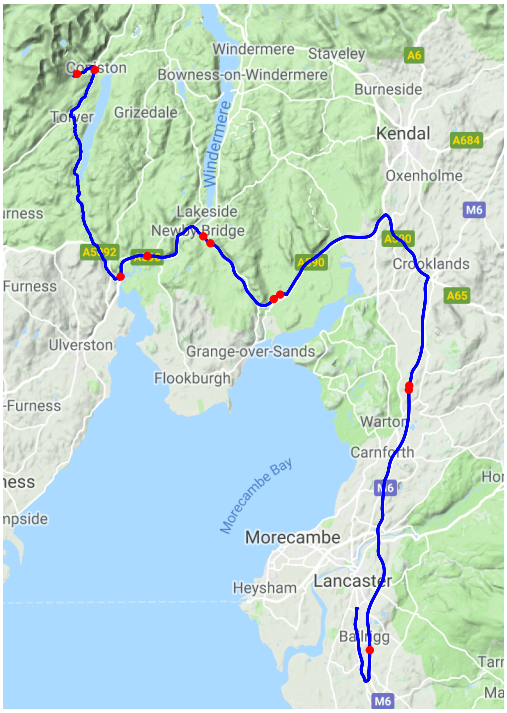
In a practical setting, where the data at hand are prone to outliers, NP-LACE maintains performance whilst the performance of the other methods is shown to degrade. By exploiting this feature, we were able to segment a journey in an interesting way. It



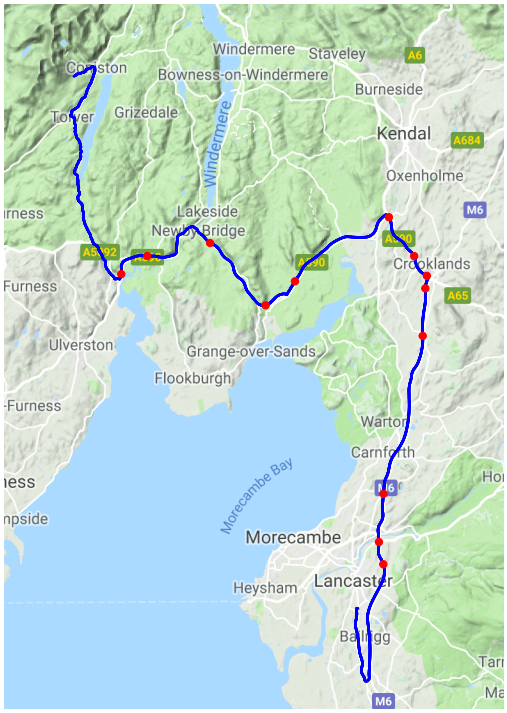
(a) NP-LACE



(b) WL



(c) Auto-PARM



(d) KF

Figure 7.6: A map of the route for the data in Figure 7.5. The solid blue line indicates the route taken by the car and the red dots indicate detected changes in second-order structure using (a) NP-LACE (b) WL (c) Auto-PARM and (d) KF. Maps produced using the `ggmap` R package (Kahle and Wickham, 2013).

is possible that this approach could segment a journey into road types, which would be of great interest to, for example, insurance and haulage companies. With many of these companies now opting to use Telematics devices in their vehicles, the ability to analyse such data automatically is becoming increasingly important. However, more work would be needed to make such an analysis feasible. This is left as an intriguing avenue for future research.

One interesting feature of the proposed method is that NP-LACE has the benefit that we could consider the autocovariance at specified lags and hence specifically identify at which lags the changes in autocovariance occur. In this case, we could specify whether the second-order change occurred in the variance or the autocovariance, and if so at which lag it occurred. It may be the case, that changes in certain lags represent different driving behaviours and these features could be used to better classify a person's driving behaviour. This is an avenue for further work, for which an initial investigation is presented in Appendix A.

Chapter 8

Conclusion

This thesis has focussed on the development of off-line changepoint algorithms for times series. Specifically, the detection of changes in second order structure. One of the many important aspects of time series is forecasting. As such, we have considered the use of our changepoint methodology to improve forecasts and have developed methodology to forecast changepoints themselves.

Chapter 3 integrated the cost function for an ARMA model into the PELT framework (Killick et al., 2012). This provided an approach to using changepoints to improve forecasts and the resulting methodology was used to forecast GDP data, with good results.

Chapter 4 was also concerned with forecasting, however in contrast to Chapter 3, the goal was to forecast changepoint times themselves. For the case of changes in mean, a transfer function model was built between an impulse and response variable. Changepoints in the response variable could then be forecast using the changepoints in the impulse variable. One key element of this procedure was the pre-whitening of the impulse variable. Drawing upon methodology from Chapter 3, a modified approach to pre-whitening was introduced, which accounted for changes in second order structure

of the explanatory time series. The changepoint prediction methodology was used to predict changes in mean speed of a HGV.

Chapter 6 introduced a non-parametric approach to detecting changes in the variance of a time series and Chapter 7 extended this work to changes autocovariance. Each of these methods were robust to the presence of outliers and consequently, informative segmentations for wind speed data and vehicle acceleration data were obtained.

This thesis is concluded by considering various avenues of future research, many of which have already been discussed within individual chapters. For example, Chapter 4 predicted future changes in mean. It would be practically useful to extend this work to allow for any type of change to occur. In doing so we could consolidate the Telematics applications in Chapters 4 and 7 to allow for both the detection and prediction of changepoints. It would also be useful to associate confidence intervals with the predicted changepoint locations.

In Chapter 7 we used the local autocovariance function to provide interesting segmentations for car journeys. Further work could include the classification of these segmentations into road types. This could be useful in other application areas such as freight transport. For example, Bonham et al. (2018) consider Automatic Identification System (AIS) data from freight ships. They use this data to segment ship journeys.

The local autocovariance measure used in Chapter 7 has been investigated further in Appendix A. It was found that if the time series can not be represented as a locally stationary wavelet process, then the estimates at individual lags can be misleading. If the methodology in Chapter 7 were to be extended to consider changes in individual lags of the autocovariance, then some correction of the estimates would be required. In Appendix A, we outline an initial approach to correcting the local autocovariance at individual lags.

A measure of the local cross-covariance (Park et al., 2014) could be used in Chapter 4 in order to detect changes in the delay between two time series. This would allow for a dynamic changepoint prediction model. However, if the measure of the local cross-covariance is susceptible to the same misrepresentation issues as those considered in Appendix A, then these would first need to be addressed.

Finally, it would be useful to incorporate the changepoint prediction model into a wider system in the Haulage industry. For example, the ability to consider a fleet of vehicles would require extensions to multiple explanatory or response time series. Additionally, in order to pro-actively control the vehicles, the algorithm would need to operate in an online setting.

Appendix A

Local Autocovariance Estimation

Chapter 7 exploited the properties of the local autocovariance of Locally Stationary Wavelet (LSW) processes proposed by Nason et al. (2000) to detect changes in piecewise second order stationary time series. The simulation study in Chapter 7.3 highlighted that in the case of a moving average model, it is important to choose the maximum lag to be the order of the moving average model. If we consider lags larger than this in the changepoint detection routine, changes in autocovariance are harder to detect because the local autocovariance estimates at those lags should be zero and constant. We discovered that in practice some structure is estimated at these lags.

In the following we provide initial research that investigates the local autocovariance of Locally Stationary Wavelet (LSW) processes proposed by Nason et al. (2000). We show that under certain conditions the estimates of the local autocovariance are not representative of the true autocovariance.

The structure of this Appendix is as follows. In Section A.1 we introduce the *curtailed local autocovariance function* (CLACV) and present a definition of finite sample LSW representability. In particular, we illustrate how the CLACV, based upon the Haar wavelet, can be expressed in terms of stationary autocovariances. In Section A.2 we

generalise this for all wavelet families. Then in Section A.3 we describe an initial approach to rectifying the representability problems discovered in Sections A.1 and A.2. These results constitute work in progress, in Section A.4 we conclude by highlighting potential avenues for further research.

A.1 The Curtailed Local Autocovariance Function

Recall that for a locally stationary wavelet process $X_{t,T}$ with wavelet spectrum $S_j(z)$, the local autocovariance function is given by

$$c(z, \nu) = \sum_{j=1}^{\infty} S_j(z)\Psi_j(\nu).$$

Here $\Psi_j(\nu) := \sum_k \psi_{j,k}\psi_{j,k}(\nu)$ are the autocorrelation wavelets at lag ν and $z = k/T$ is rescaled time.

In reality, we do not observe a time series of infinite length and for this reason Eckley (2001) introduces the *curtailed local autocovariance* (CLACV) measure of LSW processes, defined by

$$c_J(z, \nu) = \sum_{j=1}^{J} S_j(z)\Psi_j(\nu), \tag{A.1}$$

where $J = \log_2(T) < \infty$. The curtailed local autocovariance (A.1) is estimated analogously to the local autocovariance.

Using the curtailed local autocovariance function in (A.1), we can introduce the notion of finite sample LSW representability.

Definition A.1.1. *A LSW process $\{X_{t,T}\}_{t=0,\ldots,T-1}$, for a dyadic length of time $T = 2^J \geq 1$, has an autocovariance which is finite sample (sparsely) LSW representable if*

the following expectation holds

$$\mathbb{E}[c_\infty(z,\nu)] = \sum_{j=1}^{\infty} S_j(z)\Psi_j(\nu) = \sum_{j=1}^{J} S_j(z)\Psi_j(\nu) = \mathbb{E}[c_J(z,\nu)].$$

In which case the curtailed estimator is equivalent to the standard estimator and the finite sample estimator is unbiased. As a consequence, the summation over scales larger than J will be equal to zero

$$\mathbb{E}\left[\sum_{j=J+1}^{\infty} S_j(z)\Psi_j(\nu)\right] = 0.$$

It is possible to generate processes which are finite LSW representable by constructing moving average processes from wavelet filter coefficients. In the following, we define a *Haar moving average process*. This is generated from the filter coefficients of the discrete Haar wavelet.

Definition A.1.2. *A Haar moving average process of order q, HaarMA(q), is a moving average process of order $2^q - 1$, with coefficients given by the filter coefficients of the discrete Haar wavelet at scale q*

$$X_{t,T} = \sum_{i=0}^{2^{q-1}-1} 2^{-q/2}\epsilon_{t-i} - \sum_{i=2^{q-1}}^{2^q-1} 2^{-q/2}\epsilon_{t-i}$$

where $\epsilon_t \sim \mathcal{N}(0,\sigma^2)$.

The autocovariance of a Haar moving average process of order q is given by the autocovariance wavelets, $\Psi_j(\nu) := \sum_k \psi_{j,k}(0)\psi_{j,k}(\nu)$, at scale q. As a consequence, the autocovariance of a Haar moving average process of order q is always finite sample LSW representable as long as we observe a minimum of $J = q$ scales.

For the case of a second order stationary process, Eckley (2001) expresses the expectation of the CLACV as a weighted sum of stationary autocovariances. Let $R(\kappa)$ be the

autcovariance function, at lag κ, of the underlying stationary process. Eckley (2001) shows that the expected value of the CLACV estimator, based on Haar wavelets, is given by

$$\begin{aligned}\mathbb{E}(\hat{c}_J(z,\nu)) = R(0)\left\{\sum_{j=1}^{J}\sum_{l=2}^{J}\Psi_j(\nu)A_{j,l}^{-1} + \sum_{j=1}^{J}\Psi_j(\nu)A_{j,1}^{-1}\right\} - R(1)\sum_{j=1}^{J}\Psi_j(\nu)A_{j,1}^{-1} \\ + \sum_{j=1}^{J}\sum_{l=2}^{J}\Psi_j(\nu)2^{-l}A_{j,l}^{-1}\left\{2\sum_{\kappa=1}^{2^{l-1}-1}(2^l-3\kappa)R(\kappa)\right. \\ \left. -2\sum_{u=\kappa^{l-1}}^{2^l-1}(2^l-\kappa)R(\kappa)\right\}. \quad \text{(A.2)}\end{aligned}$$

Eckley (2001) highlights that this expression for the expectation of the CLACV clearly demonstrates that the estimator is biased by contributions from lags other than ν. For example, consider a MA(3) process of length $T = 128 = 2^7$ whose true autocovariance structure is given by $R(\nu) = \gamma_0\delta_{0,\nu} + \gamma_1\delta_{1,\nu} + \gamma_2\delta_{2,\nu} + \gamma_3\delta_{3,\nu}$. Here, $\delta_{n,\nu}$ is the Kronecker delta. Then equation (A.1) corresponds to (to two decimal places)

$$\mathbb{E}[\hat{c}_7(z,\nu)] = \begin{cases} 1.00\gamma_0 - 0.01\gamma_1 - 0.01\gamma_2 - 0.01\gamma_3 & \text{for } \nu = 0; \\ 0.00\gamma_0 + 0.99\gamma_1 - 0.01\gamma_2 - 0.01\gamma_3 & \text{for } \nu = 1; \\ 0.00\gamma_0 - 0.01\gamma_1 + 0.81\gamma_2 + 0.30\gamma_3 & \text{for } \nu = 2; \\ 0.00\gamma_0 - 0.01\gamma_1 + 0.35\gamma_2 + 0.27\gamma_3 & \text{for } \nu = 3. \end{cases} \quad \text{(A.3)}$$

The estimates at lags zero and one appear to be reliable however at lags two and three there is contamination from other lags. In the following, we explore this bias generally for other wavelet families.

A.2 Extension to Locally Stationary Processes and Other Wavelet Families

In Section A.1 we presented an expression for bias present in the local autocovariance function for the Haar wavelet in a second order stationary setting. Here we wish to explore the bias for general wavelet families. Recall the definition of the local autocovariance function along with its inverse equation

$$c(z,\nu) = \sum_{j=1}^{\infty} S_j(z)\Psi_j(\nu) \qquad S_j(z) = \sum_{l} A_{j,l}^{-1} \sum_{\nu} c(z,\nu)\Psi_l(\nu). \tag{A.4}$$

In order to investigate the local autocovariance function further, let us replace the local autocovariance function in the inverse equation, with the true classical autocovariance function. Then, combining the two expresses in equations (A.2), we can write

$$c(z,\nu) = \sum_{j=1}^{\infty}\sum_{l=1}^{\infty} A_{j,l}^{-1} \sum_{\kappa} c_T(z,\kappa)\Psi_l(\kappa)\Psi_j(\nu).$$

Using this expression, for each lag ν, the local autocovariance can be interpreted as a linear combination of the classical autocovariance function at lags zero to $\kappa_{\max}$,

$$\begin{aligned} c(z,\nu) &= \sum_{j}^{\infty}\sum_{l}^{\infty} A_{j,l}^{-1} \sum_{\kappa} c_T(z,\kappa)\Psi_l(\kappa)\Psi_j(\nu) \\ &= \quad c_T(z,0)\sum_{j}^{\infty}\sum_{l}^{\infty} A_{j,l}^{-1}\Psi_l(0)\Psi_j(\nu) \\ &\quad + 2c_T(z,1)\sum_{j}^{\infty}\sum_{l}^{\infty} A_{n,l}^{-1}\Psi_l(1)\Psi_n(\nu) \\ &\quad + 2c_T(z,2)\sum_{j}^{\infty}\sum_{l}^{\infty} A_{n,l}^{-1}\Psi_l(2)\Psi_n(\nu) \\ &\quad + \dots \end{aligned} \tag{A.5}$$

$$+ 2c_T(z, \kappa_{\max}) \sum_{j}^{\infty} \sum_{l}^{\infty} A_{j,l}^{-1} \Psi_l(\kappa_{\max}) \Psi_n(\nu).$$

Here $\kappa_{\max}$ is the maximum lag for which the process has non-zero autocovariance. This illustrates that at each lag ν, there are contributions from lags other than ν. We formalise these contributions in the following.

Proposition A.2.1. *The contribution to the local autocovariance function at lag ν, $c(z, \nu)$, from the autocovariance at lag κ, $c_T(z, \kappa)$, is given by*

$$\sum_{n=1}^{\infty} \sum_{l=1}^{\infty} A_{n,l}^{-1} \Psi_l(\kappa) \Psi_n(\nu),$$

where $A_{n,l} := \langle \Psi_n, \Psi_l \rangle = \sum_\nu \Psi_n(\nu) \Psi_l(\nu)$ is the the inner product of the autocorrelation wavelets at lag ν.

Proof. This follows directly from the representation in equation (A.2). □

Definition A.2.2. *The local autocovariance function is related to the classical autocovariance function by*

$$\mathbf{c}(z) = \mathrm{L}\ \mathbf{c_T}(z), \tag{A.6}$$

where $\mathbf{L}$ *is the* misrepresentation matrix. *The $(i, j)^{\text{th}}$ element of the misrepresentation matrix is given by*

$$L_{i,j} = \sum_{n=1}^{\infty} \sum_{l=1}^{\infty} A_{n,l}^{-1} \Psi_l(j-1) \Psi_n(i-1), \tag{A.7}$$

for $i, j = 1, \ldots, \kappa_{\max} + 1$.

In Section A.1 we introduced the curtailed local autocovariance function. Analogously, we can define a curtailed equivalent of the misrepresentation matrix in Definition A.2.2.

Definition A.2.3. *The curtailed local autocovariance function is related to the clas-*

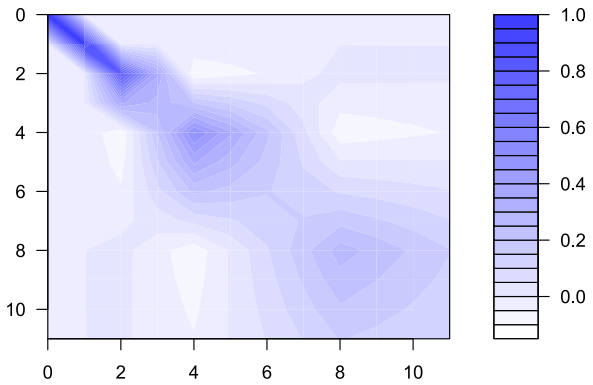


Figure A.1: An illustration of the leakage that occur across lags for the local autocovariance function when using the Haar wavelet.

sical autocovariance function by

$$\mathbf{c}_J(z) = \mathrm{L}^J \ \mathbf{c_T}(z),$$

where $\mathbf{L}^J$ *is the* curtailed misrepresentation matrix. *The* $(i,j)^{\text{th}}$ *element of the* curtailed misrepresentation matrix $\mathbf{L}^J$ *is given by*

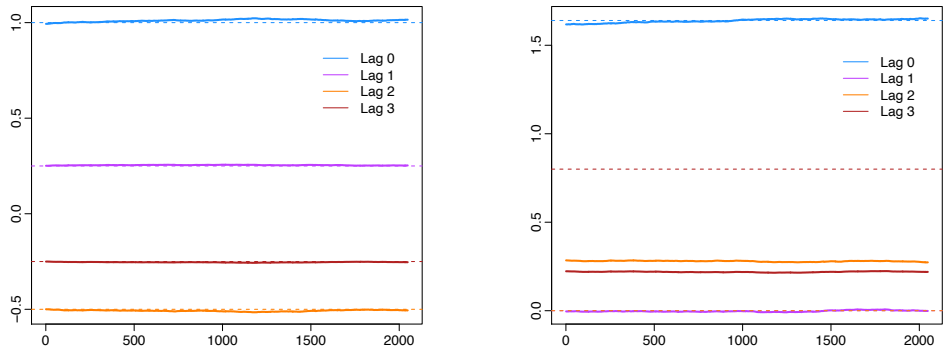
$$L^J_{i,j} = \sum_{n=1}^{J}\sum_{l=1}^{J} A^{-1}_{n,l}\Psi_l(j-1)\Psi_n(i-1), \tag{A.8}$$

for $i,j = 1,\ldots,\kappa_{\max}+1$.

If we evaluate the curtailed misrepresentation matrix (A.2.3) for the Haar Wavelet for $J = 7$ and $\kappa_{\max} = 3$, we recover the system coefficients from (A.1). Figure A.1 shows a heat map of the misrepresentation matrix for $\kappa_{\max} = 10$ for the Haar Wavelet. We can see that as the lag increases, so do the contributions from other lags. In the following example we consider the local autocovariance for a finite sample LSW representable time series.

Example A.2.1

Suppose that the LSW process $X_{t,T}$ is generated from a HaarMA(2) process,



(a) (b)

Figure A.2: The estimated local autocovariance function for (a) a HaarMA(2) process and (b) a MA(3) process.

then we have

$$X_t = \frac{1}{2}\epsilon_t + \frac{1}{2}\epsilon_{t-1} - \frac{1}{2}\epsilon_{t-2} - \frac{1}{2}\epsilon_{t-3}, \tag{A.9}$$

and the autocovariance is given by

$$c_T(\nu) = \begin{cases} 1 & \nu = 0 \\ 0.25 & \nu = 1 \\ -0.5 & \nu = 2 \\ -0.25 & \nu = 3 \end{cases} = \Psi_2(\nu).$$

The stationary local autocovariance estimated using the Haar wavelet, over 100 realisations of the process (A.2) is given by

$$\hat{c}(\nu) = \begin{cases} 1.011 & \nu = 0 \\ 0.254 & \nu = 1 \\ -0.507 & \nu = 2 \\ -0.254 & \nu = 3 \end{cases} \approx \Psi_2(\nu),$$

and Figure A.2a shows the estimated local autocovariance for the LSW process in equation (A.2).

The autocovariance estimates in Example A.2 are unbiased. However, Proposition A.2.1 illustrates that the estimates of the local autocovariance at lags ν, are contaminated by contributions from lags other than ν. This implies that the misrepresentation matrix is a left identity to the autocorrelation wavelet matrix. That is, for a HaarMA(r) process, we have

$$c(\nu) = \mathrm{L}_{1:2^r}\Psi^r = \Psi^r = c_T(\nu).$$

Here $\Psi^r := [\Psi_1(\nu), \Psi_2(\nu), \ldots, \Psi_r(\nu)]$ and is of dimension $2^r \times r$. We generalise this to other wavelet families in the following proposition.

Proposition A.2.4. *For a wavelet moving average process of order r, the misrepresentation matrix is a left identity to the autocorrelation wavelet matrix*

$$\mathrm{L}_{1:s,1:s}\Psi^r = \Psi^r.$$

Here s is the length of the support of the autocorrelation wavelet $\Psi_r(\nu)$:

$$s := \mathbf{card}\left(\{\nu \in \mathbb{R}_{\geq 0} | \Psi_r(\nu) \neq 0\}\right).$$

Whilst $\Psi^r := [\Psi_1(\nu), \Psi_2(\nu), \ldots, \Psi_r(\nu)]$ has dimension $s \times r$.

Proof. For a wavelet moving average process of order r, the dimension of $\mathrm{L}_{1:s,1:s}\Psi^r$ is

$s \times r$ and each element of this product is given by

$$\begin{aligned}
\{\mathrm{L}_{1:s,1:s}\Psi^r\}_{i,j} &= \sum_{\kappa=0}^{s-1} \Psi_j(\kappa) \sum_{n=1}^{\infty}\sum_{l=1}^{\infty} A^{-1}_{n,l}\Psi_l(\kappa)\Psi_n(i-1), \\
&= \sum_{\kappa} \Psi_j(\kappa) \sum_{n=1}^{\infty}\sum_{l=1}^{\infty} A^{-1}_{n,l}\Psi_l(\kappa)\Psi_n(i-1), \\
&= \sum_{n=1}^{\infty}\sum_{l=1}^{\infty}\sum_{\kappa} \Psi_j(\kappa) A^{-1}_{n,l}\Psi_l(\kappa)\Psi_n(i-1), \\
&= \sum_{n=1}^{\infty}\sum_{l=1}^{\infty} A_{j,l}A^{-1}_{n,l}\Psi_n(i-1), \\
&= \sum_{n=1}^{\infty} \Psi_n(i-1)\delta_{n,j}, \\
&= \Psi_j(i-1), \\
&= \{\Psi^r\}_{i,j}.
\end{aligned}$$

□

In the following example we simulate a moving average process which is not constructed using wavelet filter coefficients and estimate its local autocovariance function.

Example A.2.2

Consider the following MA(3) process and its associated autocovariance function:

$$X_t = \epsilon_t + 0.8\epsilon_{t-3}, \qquad c_T(\nu) = \begin{cases} 1.64 & \nu = 0; \\ 0 & \nu = 1; \\ 0 & \nu = 2; \\ 0.8 & \nu = 3, \end{cases}$$

where $\epsilon \sim \mathcal{N}(0,1)$.

We use equation (A.2.2) to evaluate the local autocovariance and also approxi-

mate it over 100 realisations of length $T = 2048$. These are given, respectively, by

$$c(\nu) = \begin{cases} 1.639 & \nu = 0; \\ -0.001 & \nu = 1; \\ 0.287 & \nu = 2; \\ 0.224 & \nu = 3. \end{cases} \quad \text{and} \quad \hat{c}(\nu) = \begin{cases} 1.639 & \nu = 0; \\ -0.003 & \nu = 1; \\ 0.280 & \nu = 2; \\ 0.219 & \nu = 3. \end{cases}$$

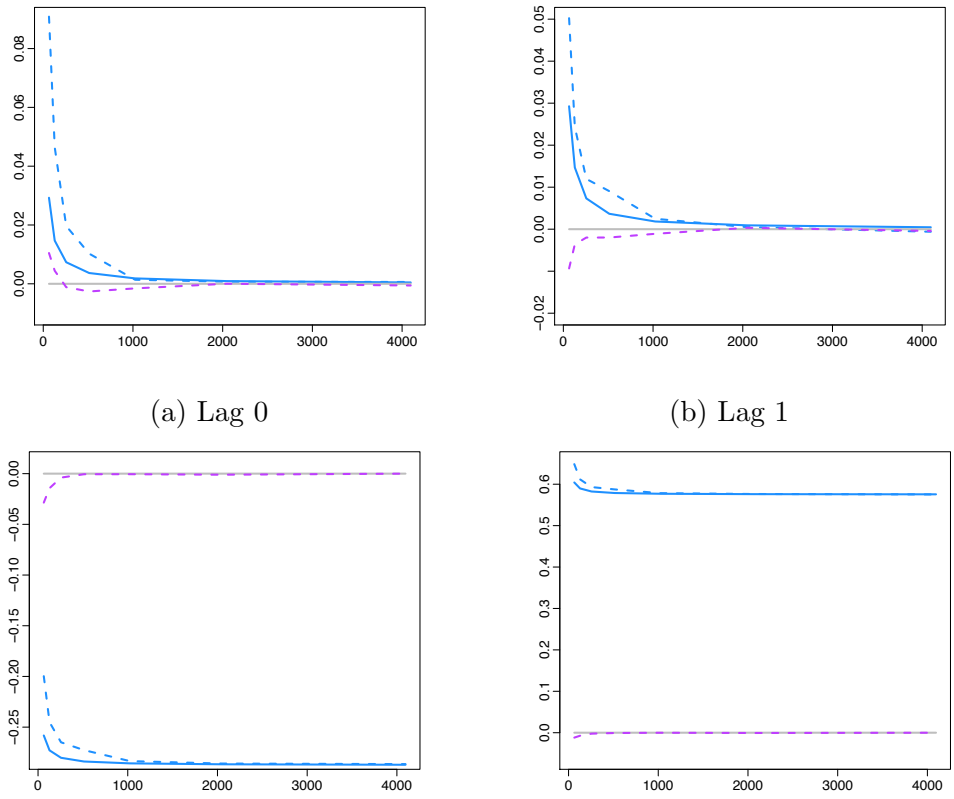
Figure A.2b shows the estimated local autocovariance function. It is clear that $c(\nu)$ and $\hat{c}(\nu)$ do not estimate $c_T(\nu)$ well.

Figure A.3 demonstrates the bias that occurs when estimating the local autocovariance for Example A.2.1 compared to that in Example A.2.2. It can be seen, that for a process which is both generated and analysed by the same wavelet, there is no bias as T increases. However, we encounter some error when we observe too few observations. For a process which is not constructed using wavelet filter coefficients, as T increases, the bias converges to some non-zero value which increases with lag. This exact nature of this bias, as $T \to \infty$, is an avenue for further research.

In the following, we demonstrate an initial approach to correcting for the bias in the local autocovariance estimates.

A.3 Correcting the Local Autocovariance Function

In Section A.2 we established that the local autocovariance and the classical autocovariance are related as: $\mathbf{c}(z) = \mathrm{L}\ \mathbf{c_T}(z)$. Consequently, to correct for the leakage across lags, we can multiply the estimated local autocovariance by the inverse of the



(a) Lag 0

(b) Lag 1

(c) Lag 2

(d) Lag 3

Figure A.3: The bias in the local autocovariance over lags zero to three as the length of the time series increases. The dashed line shows bias using an estimated spectrum for a HaarMA process (pink) and a non LSW process (blue). The solid blue line shows the bias using the theoretical spectrum determined from the theoretical autocovariance for the non LSW process. Finally, the grey line is the bias in the autocovariance when using the theoretical wavelet spectrum for the HaarMA process, there is no bias in this instance.

misrepresentation matrix,

$$\mathbf{c_T}(z) = \mathrm{L}^{-1}\ \mathbf{c}(z).$$

The measure of the stability of the system which relates the local to the classical autocovariance, is given by the condition number of the misrepresentation matrix. This is defined by

$$\mathrm{cond}(\mathrm{L}) = ||\mathrm{L}||\ ||\mathrm{L}^{-1}||.$$

For each wavelet family, this condition number increases as $\kappa_{\max}$ increases. In particular, for the Haar Wavelet, the condition number of the misrepresentation matrix (A.2.2) is unbounded for $\kappa_{\max}$ larger than 3. In such instances, we can take a pseudo inverse of the misrepresentation matrix when performing the correction. If the matrix is indeed invertible, then the pseudo inverse will be the same as the true inverse (Golub and Kahan, 1965).

Define $\ell^{\dagger}_{i,j}$ and and $\ell^{\dagger}_{Ji,j}$ to be the elements of the pseudo inverse of the misrepresentation matrix and the curtailed misrepresentation matrix respectively. Then we can define the misrepresentation corrected LACV and CLACV.

Definition A.3.1. *The misrepresentation corrected local autocovariance function at lag ν is given by*

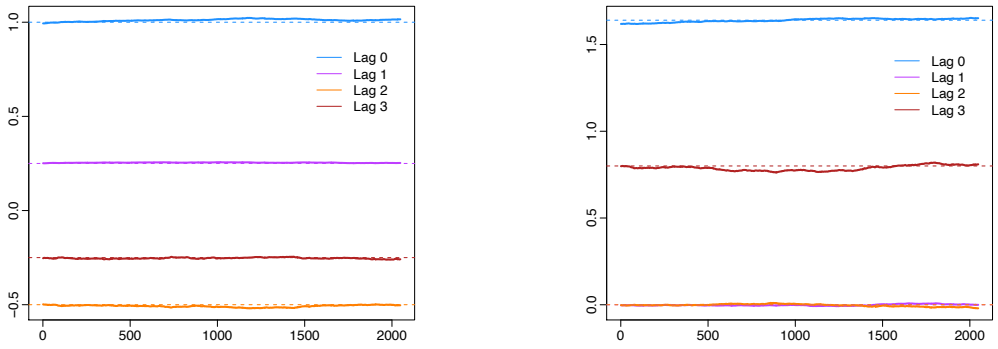
$$c^*(z,\nu) = \sum_{i=0}^{\kappa_{\max}} \ell^{\dagger}_{\nu+1,i+1}\ c(z,i),$$

and the misrepresentation corrected curtailed local autocovariance function at lag ν is given by

$$c^*_J(z,\nu) = \sum_{i=0}^{\kappa_{\max}} \ell^{\dagger}_{J\nu+1,i+1}\ c(z,i),$$

where $\ell^{\dagger}_{i,j}$ and and $\ell^{\dagger}_{Ji,j}$ are the elements of the pseudo inverse of the misrepresentation matrix (A.2.2) *and the curtailed misrepresentation matrix* (A.2.3) *respectively.*

Figure A.4 shows the correction applied to the Haar moving average process A.4a,



(a) (b)

Figure A.4: The corrected estimated local autocovariance function for (a) a HaarMA(2) process and (b) a MA(3) process.

in Example A.2.1 and the MA process A.4b, in Example A.2.2. The first case illustrates that correcting when a correction is in fact unnecessary leaves the estimates unchanged. The latter shows how the correction leads to local autocovariance estimates which are closer to the truth.

A.4 Conclusion

The local autocovariance function is a useful as it allows the autocovariance to be measured over time. It offers a superior solution to a windowed estimate of the classical autocovariance. It is therefore a unfortunate, that in practical circumstances, the LACV can provide misleading estimates.

The bias in the LACV is an interesting avenue for future research. In particular, it impacts the following two topics:

- Forecasting - Fryzlewicz et al. (2003) develop a method for forecasting which uses the local autocovariance function. If the local autocovariance is miss-estimated, then forecasts may not be reliable.
- Changepoint detection - In Chapter 7 we developed a method for detecting

changes in the autocovariance of a time series. If we want to detect changes at particular lags, then we may encounter problems resulting in an increased number of false negatives or positives.

In addition, the LACV correction procedure could also be used as a tool for model selection. If we construct the misrepresentation matrix using the same wavelet family from which the process was generated, then the corrected estimates remain unchanged. This means we could choose the wavelet family which best models our data by:

- Generating a range of misrepresentation matrices from different wavelet families;
- Correcting the estimates of the local autocovariance using each of these matrices and finally,
- Selecting the wavelet family which corrects the estimates least.

This is left as an avenue for further research.

Bibliography

Ahamada, I., Jouini, J., and Boutahar, M. (2004). Detecting multiple breaks in time series covariance structure: a non-parametric approach based on the evolutionary spectral density. *Appl. Econ.*, 36(10):1095–1101.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Trans. Autom. Control*, 19(6):716–723.

Aldrich, E. (2013). *wavelets: A package of functions for computing wavelet filters, wavelet transforms and multiresolution analyses*. R package version 0.3-0.

Alogoskoufis, G. S. and Smith, R. (1991). The phillips curve, the persistence of inflation, and the lucas critique: Evidence from exchange-rate regimes. *The American Economic Review*, pages 1254–1275.

Andersen, T., Carstensen, J., Hernndez-Garca, E., and Duarte, C. M. (2009). Ecological thresholds and regime shifts: approaches to identification. *Trends in Ecology & Evolution*, 24(1):49 – 57.

Andreou, E. and Ghysels, E. (2009). Structural breaks in financial time series. In *Handbook of financial time series*, pages 839–870. Springer.

Aston, J. A., Kirch, C., et al. (2012). Evaluating stationarity via change-point alternatives with applications to fmri data. *Ann. Appl. Stat.*, 6(4):1906–1948.

Auger, I. E. and Lawrence, C. E. (1989). Algorithms for the optimal identification of segment neighborhoods. *Bull. Math. Biol.*, 51(1):39–54.

Bai, J. and Perron, P. (1998). Estimating and testing linear models with multiple structural changes. *Econometrica*, pages 47–78.

Barry, D. and Hartigan, J. A. (1992). Product partition models for change point problems. *Ann. Statist.*, pages 260–279.

Barry, D. and Hartigan, J. A. (1993). A bayesian analysis for change point problems. *J. Am. Stat. Assoc.*, 88(421):309–319.

Baum, L. E. and Eagon, J. A. (1967). An inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology. *Bulletin of the American Mathematical Society*, 73(3):360–363.

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The annals of mathematical statistics*, pages 164–171.

Beaulieu, C., Chen, J., and Sarmiento, J. L. (2012). Change-point analysis as a tool to detect abrupt climate variations. *Phil. Trans. R. Soc. A*, 370(1962):1228–1249.

Beaulieu, C. and Killick, R. (2018). Distinguishing trends and shifts from memory in climate data. *J. Climate*, (2018).

Birgé, L. and Massart, P. (2001). Gaussian model selection. *Journal of the European Mathematical Society*, 3(3):203–268.

Birgé, L. and Massart, P. (2007). Minimal penalties for gaussian model selection. *Probab. Theory Related Fields*, 138(1-2):33–73.

Bodenham, D. A. and Adams, N. M. (2014). Adaptive change detection for relay-like behaviour. In *Intelligence and Security Informatics Conference (JISIC), 2014 IEEE Joint*, pages 252–255. IEEE.

Bonham, C., Noyvirt, A., Tsalamanis, I., and Williams, S. (2018). Analysing port and shipping operations using big data.

Box, G., Jenkins, G., and Reinsel, G. (2013). *Time Series Analysis: Forecasting and Control*. Wiley Series in Probability and Statistics. Wiley.

Brown, R. L., Durbin, J., and Evans, J. M. (1975). Techniques for testing the constancy of regression relationships over time. *J R Stat Soc Series B Stat Methodol*, pages 149–192.

Candemir, C. and Oğuz, K. (2017). A comparative study on parameter selection and outlier removal for change point detection in time series. In *2017 European Conference on Electrical Engineering and Computer Science (EECS)*, pages 218–224. IEEE.

Carlstein, E. G., Müller, H. G., and Siegmund, D. (1994). *Change–point problems.* Change–Point Problems. Institute of Mathematical Statistics.

Chen, J. and Gupta, A. (2013). *Parametric Statistical Change Point Analysis*. Birkhäuser Boston.

Chen, J. and Gupta, A. K. (1997). Testing and locating variance changepoints with application to stock prices. *J. Am. Stat. Assoc.*, 92(438):739–747.

Chernoff, H. and Zacks, S. (1964). Estimating the current mean of a normal distribution which is subjected to changes in time. *Ann. Math. Stat.*, pages 999–1018.

Chib, S. (1996). Calculating posterior distributions and modal estimates in Markov mixture models. *J. Econometrics*, 75(1):79 – 97.

Chib, S. (1998). Estimation and comparison of multiple change-point models. *J. Econometrics*, 86(2):221 – 241.

Cho, H. and Fryzlewicz, P. (2012). Multiscale and multilevel technique for consistent segmentation of nonstationary time series. *Statist. Sinica*, pages 207–229.

Cho, H. and Fryzlewicz, P. (2015). Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *J R Stat Soc Series B*, 77(2):475–507.

Chou, J.-S., Chiu, C.-K., Huang, I.-K., and Chi, K.-N. (2013). Failure analysis of wind turbine blade under critical wind loads. *Eng. Fail. Anal.*, 27:99–118.

Chuang, L.-L. and Shih, Y.-S. (2012). Approximated distributions of the weighted sum of correlated chi-squared random variables. *J. Statist. Plann. Inference*, 142(2):457–472.

Clark, T. E. and McCracken, M. W. (2005). The power of tests of predictive ability in the presence of structural breaks. *J. Econometrics*, 124(1):1–31.

Cleynen, A. and Robin, S. (2016). Comparing change-point location in independent series. *Statistics and Computing*, 26(1-2):263–276.

Dahlhaus, R. et al. (1997). Fitting time series models to nonstationary processes. *The Annals of Statistics*, 25(1):1–37.

Darkhovski, B. S. (1994). Nonparametric methods in change-point problems: A general approach and some concrete algorithms. *Lecture Notes-Monograph Series*, pages 99–107.

Daubechies, I. (1988). Orthonormal bases of compactly supported wavelets. *Commun. Pure Appl. Math.*, 41(7):909–996.

Davis, R. A., Lee, T. C. M., and Rodriguez-Yam, G. A. (2006). Structural break estimation for nonstationary time series models. *J. Am. Stat. Assoc.*, 101(473):223–239.

Eckley, I. (2001). *Wavelet methods for time series and spatial data*. PhD thesis, University of Bristol.

Eckley, I., Fearnhead, P., and Killick, R. (2011). Analysis of changepoint models. *Bayesian Time Series Models*, pages 205–224.

Elliott, G. (2005). Forecasting when there is a single break. *Manuscript, University of California at San Diego*.

Evans, A. L., Singh, N. J., Friebe, A., Arnemo, J. M., Laske, T. G., Fröbert, O., Swenson, J. E., and Blanc, S. (2016). Drivers of hibernation in the brown bear. *Front. Zool.*, 13(1):7.

Fearnhead, P. (2005). Exact bayesian curve fitting and signal segmentation. *Signal Processing, IEEE Transactions on*, 53(6):2160–2166.

Fearnhead, P. (2006). Exact and efficient bayesian inference for multiple changepoint problems. *Stat. Comput.*, 16(2):203–213.

Fearnhead, P. and Liu, Z. (2011). Efficient bayesian analysis of multiple changepoint models with dependence across segments. *Stat. Comput.*, 21(2):217–229.

Fearnhead, P. and Rigaill, G. (2018). Changepoint detection in the presence of outliers. *J. Am. Stat. Assoc.*, pages 1–15.

Fernandez, V. (2004). Detection of breakpoints in volatility. *Estudios de Administracion*, 11(1).

Fryzlewicz, P. et al. (2014). Wild binary segmentation for multiple change-point detection. *Ann. Statist.*, 42(6):2243–2281.

Fryzlewicz, P. and Nason, G. P. (2006). Haar–fisz estimation of evolutionary wavelet spectra. *J R Stat Soc Series B*, 68(4):611–634.

Fryzlewicz, P. and Subba Rao, S. (2010). Basta: consistent multiscale multiple change-point detection for piecewise-stationary arch processes. *Preprint*.

Fryzlewicz, P. and Subba Rao, S. (2014). Multiple-change-point detection for auto-regressive conditional heteroscedastic processes. *J R Stat Soc Series B*, 76(5):903–924.

Fryzlewicz, P., Van Bellegem, S., and Von Sachs, R. (2003). Forecasting non-stationary time series by wavelet process modelling. *Ann. Inst. Stat. Math.*, 55(4):737–764.

Gabbanini, F., Vannucci, M., Bartoli, G., and Moro, A. (2004). Wavelet packet methods for the analysis of variance of time series with application to crack widths on the brunelleschi dome. *J. Comput. Graph. Statist.*, 13(3):639–658.

Garcia, R., Perron, P., et al. (1991). *An analysis of the real interest rate under regime shifts*. Universite de Montreal, Departement de sciences economiques.

Geweke, J. and Jiang, Y. (2011). Inference and prediction in a multiple-structural-break model. *J. Econometrics*, 163(2):172–185.

Giacomini, R. and Rossi, B. (2009). Detecting and predicting forecast breakdowns. *The Review of Economic Studies*, 76(2):669–705.

Giraitis, L., Leipus, R., and Surgailis, D. (1996). The change-point problem for dependent observations. *J. Statist. Plann. Inference*, 53(3):297–310.

Golub, G. and Kahan, W. (1965). Calculating the singular values and pseudo-inverse of a matrix. *SINUM*, 2(2):205–224.

Gombay, E. (2008). Change detection in autoregressive time series. *J. Multivariate Anal.*, 99(3):451–464.

Gordon, N. and Ramig, P. (1983). Cumulative distribution function of the sum of correlated chi–squared random variables: The sum of correlated chi–squared random variables. *J. Stat. Comput. Simul.*, 17(1):1–9.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711–732.

Haccou, P. and Meelis, E. (1988). Testing for the number of change points in a sequence of exponential random variables. *J. Stat. Comput. Simul.*, 30(4):285–298.

Hannan, E. J. and Quinn, B. G. (1979). The determination of the order of an autoregression. *J R Stat Soc Series B Stat Methodol*, pages 190–195.

Hashem Pesaran, M., Pettenuzzo, D., and Timmermann, A. (2006). Forecasting Time Series Subject to Multiple Structural Breaks. *Rev. Econ. Stud.*, 73(4):1057–1084.

Haynes, K., Eckley, I., and Fearnhead, P. (2017a). Computationally efficient changepoint detection for a range of penalties. *J. Comput. Graph. Statist.*, 26(1):134–143.

Haynes, K., Fearnhead, P., and Eckley, I. (2017b). A computationally efficient nonparametric approach for changepoint detection. *Stat. Comput.*, 27(5):1293–1305.

Hendry, D. F. and Clements, M. P. (2000). Economic forecasting in the face of structural breaks. *Econometric Modelling: Techniques and Applications*, 3:37.

Herr, D. and Heidenreich, D. (2015). How turbulent winds abuse wind turbine drivetrains. *Wind Power Engineering*.

Hijmans, R. J., Williams, E., Vennes, C., and Hijmans, M. R. J. (2017). Package geosphere. *Spherical Trigonometry. v*.

Hilborn, R., Amoroso, R. O., Bogazzi, E., Jensen, O. P., Parma, A. M., Szuwalski, C., and Walters, C. J. (2017). When does fishing forage species affect their predators? *Fish. Res.*, 191:211 – 221.

Hinkley, D. V. (1970). Inference about the change-point in a sequence of random variables. *Biometrika*, 57(1):1–17.

Hinkley, D. V. and Hinkley, E. A. (1970). Inference about the change-point in a sequence of binomial variables. *Biometrika*, 57(3):477–488.

Hinoveanu, L. C., Leisen, F., and Villa, C. (2019). Bayesian loss-based approach to change point analysis. *Computational Statistics & Data Analysis*, 129:61–78.

Hirade, R. and Yoshizumi, T. (2012). Ensemble learning for change-point prediction. In *21st ICPR*, pages 1860–1863. IEEE.

Hoell, S. and Omenzetter, P. (2015). Damage detection in wind turbine blades based on time series correlations. In *7th International Conference on Structural Health Monitoring of Intelligent Infrastructure, SHMII 2015*.

Hosking, J. R. (1981). Fractional differencing. *Biometrika*, 68(1):165–176.

Hsu, C.-C. and Kuan, C.-M. (2001). Distinguishing between trend-break models: method and empirical evidence. *Econom. J.*, 4(2):171–190.

Hsu, D. A. (1979). Detecting shifts of parameter in gamma sequences with applications to stock price and air traffic flow analysis. *J. Am. Stat. Assoc.*, 74(365):31–40.

Huber, P. J. (2011). Robust statistics. In *International Encyclopedia of Statistical Science*, pages 1248–1251. Springer.

Hyndman, R. J., Khandakar, Y., et al. (2007). *Automatic time series for forecasting: the forecast package for R.* Number 6/07. Monash University, Department of Econometrics and Business Statistics.

Hyndman, R. J., Koehler, A. B., Snyder, R. D., and Grose, S. (2002). A state space framework for automatic forecasting using exponential smoothing methods. *Int. J. Forecasting*, 18(3):439–454.

Inclan, C. and Tiao, G. C. (1994). Use of cumulative sums of squares for retrospective detection of changes of variance. *J. Am. Stat. Assoc.*, 89(427):913–923.

Jackson, B., Scargle, J. D., Barnes, D., Arabhi, S., Alt, A., Gioumousis, P., Gwin, E., Sangtrakulcharoen, P., Tan, L., and Tsai, T. T. (2005). An algorithm for optimal partitioning of data on an interval. *IEEE Signal Process. Lett.*, 12(2):105–108.

James, N. A. and Matteson, D. S. (2015). Change points via probabilistically pruned objectives. *arXiv preprint arXiv:1505.04302*.

Jandhyala, V., Fotopoulos, S., MacNeill, I., and Liu, P. (2013). Inference for single and multiple change-points in time series. *J. Time Series Anal.*, 34(4):423–446.

Jiang, Y., Song, Z., and Kusiak, A. (2013). Very short–term wind speed forecasting with Bayesian structural break model. *Renewable Energy*, 50:637–647.

Jochmann, M., Koop, G., and Strachan, R. W. (2010). Bayesian forecasting using stochastic search variable selection in a VAR subject to breaks. *Int. J. Forecasting*, 26(2):326–347.

Jolliffe, I. (2002). *Principal component analysis*. Wiley Online Library.

Jorma, R. (1998). *Stochastic complexity in statistical inquiry*, volume 15. World scientific.

Juang, B. and Rabiner, L. (1991). Hidden markov models for speech recognition. *Technometrics*, 33(3):251–272.

Kahle, D. and Wickham, H. (2013). ggmap: Spatial visualization with ggplot2. *The R Journal*, 5(1):144–161.

Killick, R. and Eckley, I. (2014). changepoint: An r package for changepoint analysis. *J. Stat. Softw*, 58(3):1–19.

Killick, R., Eckley, I., and Jonathan, P. (2013). A wavelet-based approach for detecting changes in second order structure within nonstationary time series. *Electron. J. Stat.*, 7:1167–1183.

Killick, R., Fearnhead, P., and Eckley, I. (2012). Optimal detection of changepoints with a linear computational cost. *J. Am. Stat. Assoc.*, 107(500):1590–1598.

Killick, R., Haynes, K., Eckley, I., Fearnhead, P., and Lee, J. (2015). Package changepoint.

Kim, H.-J., Yu, B., and Feuer, E. J. (2009). Selecting the number of change-points in segmented line regression. *Statist. Sinica*, 19(2):597.

Kirch, C., Muhsal, B., and Ombao, H. (2015). Detection of changes in multivariate time series with application to EEG data. *J. Am. Stat. Assoc.*, 110(511):1197–1216.

Kletting, P. and Glatting, G. (2009). Model selection for time-activity curves: the corrected akaike information criterion and the f-test. *Zeitschrift für medizinische Physik*, 19(3):200–206.

Knab, B., Schliep, A., Steckemetz, B., and Wichern, B. (2003). Model–based clustering with hidden markov models and its application to financial time–series data. In *Between Data Science and Applied Data Analysis*, pages 561–569. Springer.

Ko, S. I., Chong, T. T., Ghosh, P., et al. (2015). Dirichlet process hidden markov multiple change-point model. *Bayesian Analysis*, 10(2):275–296.

Koop, G. and Potter, S. (2007). Estimation and forecasting in models with multiple breaks. *The Review of Economic Studies*, 74(3):763–789.

Korkas, K. and Fryzlewicz, P. (2017). Multiple changepoint detection for non-stationary time series using wild binary segmentation. *Statist. Sinica*, 27(1):287–311.

Korkas, K. and Fryzlewicz, P. (2018). *wbsts: Multiple Change-Point Detection for Nonstationary Time Series*. R package version 2.0.

Kulkarni, V., Al-Rfou, R., Perozzi, B., and Skiena, S. (2015). Statistically significant detection of linguistic change. In *Proceedings of the 24th International Conference on World Wide Web*, pages 625–635. International World Wide Web Conferences Steering Committee.

Lai, T. L. (1995). Sequential changepoint detection in quality control and dynamical systems. *J R Stat Soc Series B Stat Methodol*, 57(4):613–658.

Lai, W. R., Johnson, M. D., Kucherlapati, R., and Park, P. J. (2005). Comparative analysis of algorithms for identifying amplifications and deletions in array CGH data. *Bioinformatics*, 21(19):3763–3770.

Lavielle, M. (2005). Using penalized contrasts for the change-point problem. *Signal Process.*, 85(8):1501–1510.

Lavielle, M. and Lebarbier, E. (2001). An application of MCMC methods for the multiple change–points problem. *Signal Process.*, 81(1):39 – 53. Special section on Markov Chain Monte Carlo (MCMC) Methods for Signal Processing.

Lavielle, M., Ludeña, C., et al. (2000). The multiple change-points problem for the spectral distribution. *Bernoulli*, 6(5):845–869.

Lavielle, M. and Moulines, E. (2000). Least-squares estimation of an unknown number of shifts in a time series. *J. Time Series Anal.*, 21(1):33–59.

Li, S. and Lund, R. (2012). Multiple changepoint detection via genetic algorithms. *J. Climate*, 25(2):674–686.

Liang, F. and Wong, W. H. (2000). Evolutionary monte carlo: Applications to c p model sampling and change point problem. *Statist. Sinica*, pages 317–342.

Liu, J. S. and Lawrence, C. E. (1999). Bayesian inference on biopolymer models. *Bioinformatics (Oxford, England)*, 15(1):38–52.

Lund, R., Wang, X. L., Lu, Q. Q., Reeves, J., Gallagher, C., and Feng, Y. (2007). Changepoint detection in periodic and autocorrelated time series. *Journal of Climate*, 20(20):5178–5190.

Luong, T. M., Perduca, V., and Nuel, G. (2012). Hidden markov model applications in change-point analysis. *arXiv preprint arXiv:1212.1778*.

Lvy-Leduc, C. and Roueff, F. (2009). Detection and localization of change-points in high-dimensional network traffic data. *Ann. Appl. Stat.*, 3(2):637–662.

Maheu, J. M. and Gordon, S. (2008). Learning, forecasting and structural breaks. *J. Appl. Econometrics*, 23(5):553–583.

Maheu, M. and Song, Y. (2014). A new structural break model, with an application to Canadian inflation forecasting. *Int. J. Forecasting*, 30(1):144–160.

Maidstone, R., Hocking, T., Rigaill, G., and Fearnhead, P. (2017). On optimal multiple changepoint algorithms for large data. *Stat. Comput.*, 27(2):519–533.

Mallat, S. G. (1989). Multiresolution approximations and wavelet orthonormal bases of l^2 (r). *Transactions of the AMS*, 315(1):69–87.

Matteson, D. S. and James, N. A. (2014). A nonparametric approach for multiple change point analysis of multivariate data. *J. Am. Stat. Assoc.*, 109(505):334–345.

McCulloch, R. E. and Tsay, R. S. (1993). Bayesian inference and prediction for mean and variance shifts in autoregressive time series. *J. Am. Stat. Assoc.*, 88(423):pp. 968–978.

Meyer, Y. and Salinger, D. (1992). *Wavelets and Operators:*. Cambridge Studies in Advanced Mathematics. Cambridge University Press.

Mikulski, J. (2013). *Activities of Transport Telematics: 13th International Conference on Transport Systems Telematics*. Communications in Computer and Information Science. Springer Berlin Heidelberg.

Nason, G. (2010). *Wavelet Methods in Statistics with R*. Use R! Springer New York.

Nason, G. (2012). wavethresh: Wavelets statistics and transforms. R package v.4.5.

Nason, G. (2013). A test for second-order stationarity and approximate confidence intervals for localized autocovariances for locally stationary time series. *J R Stat Soc Series B Stat Methodol*, 75(5):879–904.

Nason, G. P. and Silverman, B. W. (1995). The stationary wavelet transform and some statistical applications. In *Wavelets and statistics*, pages 281–299. Springer.

Nason, G. P., Von Sachs, R., and Kroisandt, G. (2000). Wavelet processes and adaptive estimation of the evolutionary wavelet spectrum. *J R Stat Soc Series B Stat Methodol*, 62(2):271–292.

Norwood, B. and Killick, R. (2018). Long memory and changepoint models: a spectral classification procedure. *Stat. Comput.*, 28(2):291–302.

Olshen, A., Venkatraman, E. S., Lucito, R., and Wigler, M. (2004). Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics (Oxford, England)*, 5:557–72.

Ombao, H. C., Raz, J. A., von Sachs, R., and Malow, B. A. (2001). Automatic statistical analysis of bivariate nonstationary time series. *J. Am. Stat. Assoc.*, 96(454):543–560.

Page, E. (1954). Continuous inspection schemes. *Biometrika*, pages 100–115.

Park, T., Eckley, I. A., and Ombao, H. C. (2014). Estimating time-evolving partial coherence between signals via multivariate locally stationary wavelet processes. *IEEE Trans Signal Process*, 62(20):5240–5250.

Pesaran, M. H. and Timmermann, A. (2002). Market timing and return prediction under model instability. *J. Empirical Finance*, 9(5):495–510.

Pesaran, M. H. and Timmermann, A. (2004). How costly is it to ignore breaks when forecasting the direction of a time series? *Int. J. Forecasting*, 20(3):411–425.

Pesaran, M. H. and Timmermann, A. (2007). Selection of estimation window in the presence of breaks. *J. Econometrics*, 137(1):134–161.

Picard, F., Robin, S., Lavielle, M., Vaisse, C., and Daudin, J.-J. (2005). A statistical approach for array CGH data analysis. *BMC Bioinf.*, 6(1):27.

Pievatolo, A. and Green, P. J. (1998). Boundary detection through dynamic polygons. *J R Stat Soc Series B*, 60(3):609–626.

Polunchenko, A. S. and Tartakovsky, A. G. (2012). State-of-the-art in sequential change-point detection. *Methodol. Comput. Appl. Probab.*, 14(3):649–684.

Priestley, M. (1988). *Non-linear and non-stationary time series analysis*. Academic Press.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rabiner, L. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proc. IEEE*, 77(2):257–286.

Reeves, J., Chen, J., Wang, X. L., Lund, R., and Lu, Q. (2007). A review and comparison of changepoint detection techniques for climate data. *Journal of Applied Meteorology & Climatology*, 46.

Richardson, A. D., Hufkens, K., Milliman, T., Aubrecht, D. M., Chen, M., Gray, J. M., Johnston, M. R., Keenan, T. F., Klosterman, S. T., Kosmala, M., et al. (2018). Tracking vegetation phenology across diverse north american biomes using phenocam imagery. *Sci. Data*, 5(180028).

Rigaill, G. (2015). A pruned dynamic programming algorithm to recover the best segmentations with 1 to $K_{\max}$ change-points. *Journal de la Société Française de Statistique*, 156(4):180–205.

Rigaill, G., Lebarbier, E., and Robin, S. (2012). Exact posterior distributions and model selection criteria for multiple change-point detection problems. *Statistics and Computing*, 22(4):917–929.

Ruggieri, E., Herbert, T., Lawrence, K. T., and Lawrence, C. E. (2009). Change point method for detecting regime shifts in paleoclimatic time series: Application to δ^{18}O time series of the Plio-Pleistocene. *Paleoceanography and Paleoclimatology*, 24(1).

Schwaller, L. and Robin, S. (2017). Exact bayesian inference for off-line change-point detection in tree-structured graphical models. *Statistics and Computing*, 27(5):1331–1345.

Schwarz, G. et al. (1978). Estimating the dimension of a model. *Ann. Statist.*, 6(2):461–464.

Scott, A. and Knott, M. (1974). A cluster analysis method for grouping means in the analysis of variance. *Biometrics*, 30:507–512.

Shumway, R. H. and Stoffer, D. S. (2000). Time series analysis and its applications. *Studies In Informatics And Control*, 9(4):375–376.

Spokoiny, V. (2009). Multiscale local change point detection with applications to value–at–risk. *Ann. Statist.*, pages 1405–1436.

Stephens, D. (1994). Bayesian retrospective multiple-changepoint identification. *Applied Statistics*, pages 159–178.

Stock, J. H. and Watson, M. W. (1996). Evidence on structural instability in macroeconomic time series relations. *J Bus Econ Stat*, 14(1):11–30.

Sturludottir, E., Gunnlaugsdottir, H., Nielsen, O. K., and Stefansson, G. (2017). Detection of a changepoint, a mean-shift accompanied with a trend change, in short time-series with autocorrelation. *COMMUN STAT-SIMUL C*, 46(7):5808–5818.

Taleb, I., Dssouli, R., and Serhani, M. A. (2015). Big data pre-processing: a quality framework. In *BigData Congress*, pages 191–198. IEEE.

Timmermann, A. (2001). Structural breaks, incomplete information, and stock prices. *Journal of Business & Economic Statistics*, 19(3):299–314.

Venkatraman, E. and Olshen, A. B. (2007). A faster circular binary segmentation algorithm for the analysis of array CGH data. *Bioinformatics*, 23(6):657–663.

Venkatraman, E. S. (1992). *Consistency results in multiple change-point problems*. PhD thesis, Stanford University.

Vidakovic, B. (2009). *Statistical modeling by wavelets*, volume 503. John Wiley & Sons.

Viterbi, A. J. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Inf. Theory*, 13(2):260–269.

Wang, X., Smith, K., and Hyndman, R. (2006). Characteristic-based clustering for time series data. *Data Min. Knowl. Discov.*, 13(3):335–364.

Whitcher, B., Guttorp, P., and Percival, D. B. (2000). Multiscale detection and location of multiple variance changes in the presence of long memory. *J. Stat. Comput. Simul.*, 68(1):65–87.

Whitle, P. (1951). *Hypothesis testing in time series analysis*, volume 4.

Willenbrock, H. and Fridlyand, J. (2005). A comparison study: applying segmentation to array CGH data for downstream analyses. *Bioinformatics*, 21(22):4084–4091.

Yamaguchi, K. (2011). Estimating a change point in the long memory parameter. *J. Time Series Anal.*, 32(3):304–314.

Yao, Y.-C. et al. (1988). Estimating the number of change-points via Schwarz' criterion. *Statistics & Probability Letters*, 6(3):181–189.

Yau, C. Y. and Davis, R. A. (2012). Likelihood inference for discriminating between long-memory and change-point models. *J. Time Series Anal.*, 33(4):649–664.

Zhang, N. R. and Siegmund, D. O. (2007). A modified bayes information criterion with applications to the analysis of comparative genomic hybridization data. *Biometrics*, 63(1):22–32.

Zou, C., Liu, Y., Qin, P., and Wang, Z. (2007). Empirical likelihood ratio test for the change-point problem. *Statistics & probability letters*, 77(4):374–382.

Zou, C., Yin, G., Feng, L., Wang, Z., et al. (2014). Nonparametric maximum likelihood approach to multiple change-point problems. *Ann. Statist.*, 42(3):970–1002.