METHODS FOR CHANGE POINT DETECTION IN SEQUENTIAL DATA

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Wenyu Zhang

May 2020

METHODS FOR CHANGE POINT DETECTION IN SEQUENTIAL DATA

Wenyu Zhang, Ph.D.

Cornell University 2020

The analysis of numerical sequential data, such as time series, is a frequent practice in both academic and industrial settings. Offline change detection segments the data retrospectively and is useful for uncovering events and systematic behaviors in data analysis tasks. It is applied in a variety of fields including finance, genomics and energy consumption. Furthermore, in the potential presence of change points, utilizing change detection prior to data modeling can help prevent building inappropriate models under the assumption of data homogeneity, and consequently supports improved prediction and statistical inference. In this thesis, we propose three methods that study the offline change point detection problem from different aspects and application domains. The first method is a nonparametric procedure that can provide computational speedups to simultaneously detect multiple change points. The second method models the relationship between the different channels of multivariate observations to detect change points and anomalies. The third method focuses on the specific biomedical domain of cell culture monitoring to detect the transition from cell growth to confluence. All proposed methods are evaluated through simulations and real-world data applications.

BIOGRAPHICAL SKETCH

Wenyu Zhang received the B.A. degree with double major in Applied Mathematics and Statistics from the University of California, Berkeley, Berkeley, CA, USA, in 2013. She then received the Master of Science degree in Computer Science from Cornell University, Ithaca, NY, USA, in 2017. During her Ph.D. studies in Statistics and Data Science at Cornell University, she worked on anomaly and change detection methods for time series and other sequential data. Her research interests are in developing statistical and machine learning methodology for sequential data.

ACKNOWLEDGEMENTS

I would like to thank my advisor David S. Matteson, and my Ph.D. committee members David Ruppert, Karthik Sridharan and Sumanta Basu for their guidance and feedback throughout my Ph.D. studies. I would also like to thank my project collaborators Nicholas A. James, Daniel E. Gilbert and Maryclare Griffin. I have learned many things about research from them.

I am deeply thankful for the continuous support from my friends and parents, as well as my pet Buddy. Their love and support helped me stay motivated on this journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Change detection involves segmenting sequential data such that observations in the same segment share some desired properties. In this thesis, we propose three methods that study the offline change point detection problem from different aspects and application domains. We describe these methods and evaluations in following chapters.

In Chapter 2, we propose a pruning approach for approximate nonparametric estimation of multiple change points. This general purpose change point detection procedure 'cp3o' applies a pruning routine within a dynamic program to greatly reduce the search space and computational costs. Existing goodness-of-fit change point objectives can immediately be utilized within the framework. We further propose novel change point algorithms by applying cp3o to two popular nonparametric goodness of fit measures: 'e-cp3o' uses E-statistics, and 'ks-cp3o' uses Kolmogorov-Smirnov statistics. Simulation studies highlight the performance of these algorithms in comparison with parametric and other nonparametric change point methods. Finally, we illustrate these approaches with climatological and financial applications.

Multivariate change detection continues to be a challenging problem due to the variety of ways change points can be correlated across channels and the potentially poor signal-to-noise ratio on individual channels. In Chapter 3, we are interested in locating additive outliers (AO) and level shifts (LS) in the unsupervised setting. We propose ABACUS, *Automatic BAyesian Changepoints Under Sparsity*, a Bayesian source separation technique to recover latent signals while also detecting changes in model parameters. Multi-level sparsity achieves both dimension reduc-

tion and modeling of signal changes. The procedure is completely automatic and returns both AO and LS changes separately, enabling users to directly assess each type. We show ABACUS has competitive or superior performance in simulation studies against state-of-the-art change detection methods and established latent variable models. We also illustrate ABACUS on two real application, modeling genomic profiles and analyzing household electricity consumption.

In Chapter 4, we focus on the specific biomedical domain of cell culture monitoring. Measurements of many biological processes are characterized by an initial growth period, followed by a sustained equilibrium period. In practice, scientists may wish to quantify features of the growth period, features of the equilibrium period, and the timing of the change point where the growth period gives way to equilibrium. In this work, we assume that the measurements during the growth period are a smooth function of time and assume that the measurements during the equilibrium period are characterized by a simple fractionally integrated time series model. We propose a likelihood-based method to simultaneously estimate the parameters of the growth and equilibrium processes and locate the change point between the two. We find that this method performs well in simulations. Throughout, we are motivated by the specific problems in the study of electrical cell-substrate impedance sensing (ECIS) data. ECIS is a popular new technology used to study cell behavior, which non-invasively measures cell behavior at a high temporal resolution. Previous studies have found that different cell types can be classified by their behavior during the equilibrium period, called confluence in the ECIS literature. However, it can be challenging to identify when the equilibrium period/confluence has been reached, and to quantify the relevant features during this period. Our method allows us to obtain better estimates of measures of cell behavior during confluence, and accordingly better cell classification. Addition-

ally, our method produces estimates of the change points themselves, which we find offer additional gains in classification performance.

CHAPTER 2

PRUNING AND NONPARAMETRIC MULTIPLE CHANGE POINT DETECTION

Contents in this chapter are published in [66].

2.1 Introduction

The analysis of time ordered data, or time series, has become a frequent practice in both academic and industrial settings. When analysis is performed it is generally assumed that the data adheres to some form of homogeneity. However, it may not be appropriate, or practical, to apply the same analytical procedure to many different types of time series. The resulting statistical bias from such model misspecification is one of the reasons for the current resurgence of change point analysis, which attempts to partition a time series into homogeneous segments.

A popular approach is to fit the observed data to a parametric model. In this setting a change point corresponds to a change in the monitored parameter(s) [43, 14]. Parametric approaches rely heavily upon the assumption that the data behaves according to the predefined distribution model. Otherwise the degree of bias in the obtained results is usually unknown [54]. In practice, it is almost always difficult to test for adherence to these assumptions.

Nonparametric analysis is a natural way to proceed. Since nonparametric approaches make much weaker assumptions than their parametric counterparts, they can be used in a much wider variety of settings; for example, the analysis of internet traffic data, where there is no commonly accepted distributional model.

Another challenge in multiple change point analysis is that it can easily become computationally intractable. There are $\mathcal{O}(T^k)$ possible segmentations in a length T time series containing k change points. Naive approaches to find the best segmentation quickly become impractical. Moreover, the number of true change points is usually not known.

In this work, we address the challenge of designing a customizable procedure that can detect a wide range of changes while appropriately balancing detection accuracy and speed. We introduce a new change point search framework called cp3o (**C**hange **P**oint **P**rocedure via **P**runed **O**bjectives). The cp3o framework is a general purpose search procedure, which means it can be used with a large class of goodness-of-fit metrics to detect change points. For instance, additional knowledge about the data, such as the type of changes which are to be detected, or computational time considerations might direct a user to particular goodness-of-fit metrics. This plug-and-play idea is similar to that in [5], such that the users can specify their own goodness-of-fit metrics, or pick from available options based on performance with training data. The cp3o procedure makes use of dynamic programming with search space pruning. This allows the number of change points to be quickly determined, while simultaneously generating all other optimal segmentations as a byproduct.

We further propose two new change point algorithms, named e-cp3o and ks-cp3o, by incorporating two popular nonparametric goodness-of-fit metrics, namely E-statistics and the Kolmogorov-Smirnov statistic, within the cp3o search procedure. Results from a variety of simulations show that in most cases the proposed cp3o algorithms provide a good balance between speed and accuracy in comparison with parametric and other nonparametric change point methods.

Both e-cp3o and ks-cp3o algorithms are freely available in the `ecp` *R* package on CRAN.

2.2 Related Works

2.2.1 Multiple Change Point Detection Methods

Most existing procedures for performing retrospective multiple change point analysis can be classified as belonging to one of two groups: those that return *approximate* solutions and those that return *exact* solutions.

Approximate search algorithms tend to rely heavily on a subroutine for finding a single change point. Estimates for multiple change point locations are produced by iteratively applying this subroutine. Examples include binary segmentation and its adaptations such as the Circular Binary Segmentation approach of [51] and the E-Divisive approach of [44]. Approximate procedures tend to produce suboptimal segmentations of the given time series, but have much lower computational complexity than exact procedures.

Exact search algorithms return segmentations that are optimal with respect to a pre-specified goodness-of-fit metric. In order to achieve a reasonable computational cost, the utilized goodness-of-fit metrics often satisfy Bellman's Principle of Optimality [8], and can thus be optimized through the use of dynamic programming. Examples of exact algorithms include the Kernel Change Point algorithm, [27] and [2], and the MultiRank algorithm [42].

2.2.2 Pruning Methods

The runtime of traditional dynamic programming change point detection approaches is still at least quadratic in the length of the time series. However, many of the calculations performed during the dynamic programs do not result in the identification of a new change point. These calculations can be viewed as excessive and they quickly compound to slow down analysis. One way to tackle this is by continually pruning the set of potential change point locations. [57] proposes a pruning method that can be used when the goodness-of-fit metric is convex. The PELT method [38] is a parametric method which incorporates a pruning step in its dynamic program, such that the expected running time is linear in the length of the time series under certain conditions. However, these methods restrict the options of goodness-of-fit metrics that can be used due to requirements of convexity and parametric objective formulations.

2.3 Problem Formulation

Let $Z_1, Z_2, \ldots, Z_T \in \mathbb{R}^d$ be a length T sequence of independent d-dimensional time ordered random variables. We denote k as the true number of change points, where the change points are time indices $1 = t_0 < t_1 < \cdots < t_k < t_{k+1} = T + 1$, such that $Z_i \overset{iid}{\sim} F_j$ for $t_j \leqslant i < t_{j+1}$, and $F_j \neq F_{j+1}$, for distributions F_j with $0 \leqslant j \leqslant k$.

Given a series of such observations, the challenge is to select the number of change points and change point locations so that the observations within each segment are identically distributed, and the distributions of observations in adjacent segments are different. We approach this problem through the use of goodness-

of-fit metrics, which are commonly used for exact search procedures. We also incorporate the parameter $w \geqslant 1$, which is a user-defined lower bound for the distance between change points.

We refer to a partition of $Z_1, Z_2, \ldots, Z_T$ with κ segmentation points as a κ-segmentation. With segmentation points $1 = \tau_0 < \tau_1 < \cdots < \tau_\kappa < \tau_{\kappa+1} = T+1$, we quantify the quality of the resulting κ-segmentation with the empirical goodness-of-fit metric:

$$\sum_{j=1}^{\kappa} \widehat{g}_R\left(\tau_{j-1}, \tau_j, \tau_{j+1}\right)$$

where $\widehat{g}_R\left(a, b, c\right) = \widehat{R}\big(Z_a^{b-1}, Z_b^{c-1}\big)$ and $Z_a^b = \{Z_i\}_{i=a}^b$, for $a < b < c$. Here $\widehat{R}(\cdot, \cdot)$ is a sample version of a given population measure $R(\cdot, \cdot)$ of the dissimilarity between the distributions of two random variables.

Empirical goodness-of-fit of the κ-segmentation of a length T sequence with k change points is maximized at

$$\widehat{G}_T(\kappa, w) = \max_{\substack{\tau_1, \tau_2, \ldots, \tau_\kappa \\ \tau_i + w \leqslant \tau_\ell,\ i<\ell \\ \tau_i \in \{1+w, \ldots, T-w+1\}}} \sum_{j=1}^{\kappa} \widehat{g}_R\left(\tau_{j-1}, \tau_j, \tau_{j+1}\right)$$

Calculating $\widehat{G}_T(\kappa, w)$ requires maximization over all κ-tuples containing a strictly increasing sequence of elements from $1+w$ to $T-w+1$ (that are at least w apart), and hence is computationally expensive. We next introduce an approximation procedure that gives significant speed-ups.

2.4 Proposed cp3o Procedure

We adapt the evaluation of $\widehat{G}_T(\kappa, w)$ in two ways to increase computational efficiency:

- Approximation of $\widehat{G}_T(\kappa, w)$ to allow the use of dynamic programming,
- Pruning to reduce the dynamic program search space.

To obtain estimates κ and $\{\tau_i\}_{i=1}^{\kappa}$ for the number of change points k and the change point locations $\{t_i\}_{i=1}^{k}$, we can calculate $\widehat{G}_T(\kappa, w)$ for a range of values $1 \leqslant \kappa \leqslant K$ where $K \geqslant k$ is a user-defined upper bound for k, then select κ based on a chosen rule which we propose in Section 2.4.3.

2.4.1 Dynamic Programming

Since there are $\mathcal{O}(T^{\kappa})$ possible κ-segmentations, a direct computation of $\widehat{G}_T(\kappa, w)$ requires $\mathcal{O}(T^{\kappa})$ evaluations of the goodness-of-fit metric. Instead, we employ dynamic programming in the following fashion. Define

$$H_t(\kappa, w, \tau) = \widetilde{G}_{\tau-1}(\kappa - 1, w) + \widehat{g}_R\left(A_{\tau-1}(\kappa - 1), \tau, t\right).$$

Then, in the κ^{th} iteration, for each subsequence $\{Z_i\}_{i=1}^{t}$ where $1 \leqslant t \leqslant T$, we

define

$$A_t(\kappa, w) = \underset{\tau \in \{1+\kappa^* w, \ldots, t-w+1\}}{\operatorname{argmax}} H_t(\kappa, w, \tau),$$
$$\widetilde{G}_t(\kappa, w) = \max_{\tau \in \{1+\kappa^* w, \ldots, t-w+1\}} H_t(\kappa, w, \tau),$$

where $\widetilde{G}_t(\kappa, w)$ denotes the approximation of the optimal goodness-of-fit for the length t subsequence with κ segmentation points, and $A_t(\kappa, w)$ denotes the location of the κ^{th} segmentation point in this approximation.

$\widetilde{G}_t(\kappa, w)$ is obtained by optimizing over all possible candidates for the κ^{th} segmentation point, and approximating the previous $\kappa - 1$ change points through A. For example, if τ is the κ^{th} segmentation point, then we take $A_{\tau-1}(\kappa - 1)$ as the $(\kappa - 1)^{th}$ segmentation point

Each computation of $\widetilde{G}_t(\kappa, w)$ needs at most t evaluations of the goodness-of-fit metric. Hence there are $\mathcal{O}(T^2)$ evaluations of the goodness-of-fit metric in the κ's iteration to obtain $\widetilde{G}_T(\kappa, w)$. This is significantly lower than the $\mathcal{O}(T^\kappa)$ evaluations prior to approximation.

2.4.2 Pruning

In the κ^{th} iteration of dynamic programming, $A_t(\kappa, w)$ and $\widetilde{G}_t(\kappa, w)$ require searching for the optimal κ^{th} segmentation point of $\{Z_i\}_{i=1}^t$ from candidates $\{1 + \kappa^* w, \ldots, t - w + 1\}$. To cut computations further, we reduce this search space by only searching in $S_t(\kappa, w)$, defined below.

For the first iteration, we initialize $S_t(1, w) = \{1 + w, \ldots, t - w + 1\}$ which

is the largest possible search space. For the $(\kappa+1)^{th}$ iteration, the search space $S_t(\kappa+1, w)$ is the result of pruning the search space $S_t(\kappa, w)$ from the previous iteration, as we want to iteratively pinpoint the most optimal change point before t. The pruning rule is:

$$S_t(\kappa+1, w) = \{\tau \in S_t(\kappa, w) : \\ H_t(\kappa+1, w, \tau) \geqslant H_t(\kappa+1, w, t-w+1)\}.$$

In the above expression, the inequality compares the goodness-of-fit of two valid $(\kappa+1)$-segmentations for the length t subsequence, one with the last change point at τ and the other at $t-w+1$. If the former is less than the latter, then τ is a less optimal segmentation point than $t-w+1$, hence τ can be pruned away from the set of candidate change points.

We choose to benchmark the goodness-of-fit induced by τ against that induced by $t-w+1$ because $t-w+1$ is the last possible change point location before t. Keeping $t-w+1$ in the search space would maximize the total number of change points possible for the sequence.

2.4.3 Algorithm

We outline the complete cp3o procedure in Algorithm 1. Aside from the notation described in Sections 2.4.1 and 2.4.2, we use $cps_t(\kappa)$ to denote the set of κ change points estimated for the subsequence $\{Z_i\}_{i=1}^{t}$.

The cp3o algorithm iterates through κ from 1 to a user-defined upper bound

Algorithm 1 cp3o

Input : Data sequence $z_1, z_2, \ldots, z_T \in \mathbb{R}^d$
Upper bound on number of changes K
Minimum distance between changes w

Initialize: Search space $S_t(1) = \{1 + w, \ldots, t - w + 1\}$
Set of change points $cps_t(0) = \varnothing$
Previous change point $A_t(0) = 1$ before t

for κ *from 1 to* K **do**
  for t *from* 2^*w *to* T **do**
    $\tau^* = \underset{\tau \in S_t(\kappa)}{\operatorname{argmax}}\, H_t(\kappa, w, \tau)$
    $\widetilde{G}_t(\kappa, w) = H_t(\kappa, w, \tau^*)$
    Update $cps_t(\kappa) = cps_{\tau^*}(\kappa - 1) \cup \{\tau^*\}$
    Update $A_t(\kappa) = \tau^*$
    Update $S_t(\kappa + 1) = \{\tau \in S_t(\kappa) : H_t(\kappa + 1, w, \tau) \geqslant H_t(\kappa + 1, w, t - w + 1)$
  end
end
Pick optimal number of change points κ^*

Output : $cps_T(\kappa^*)$

K. In each iteration κ, for each subsequence $\{Z_i\}_{i=1}^t$, cp3o finds τ^* as the κ^{th} change point in the subsequence. The other $\kappa - 1$ change points are as found in previous iterations. The goodness-of-fit $\widetilde{G}_t(\kappa, w)$ of the length t subsequence is calculated with these κ change points, and $cps_t(\kappa)$ is updated to save these change points. In preparation for future iterations, the search set $S_t(\kappa + 1)$ is obtained by discarding candidate points which produce a worse goodness-of-fit than segmenting at $t - w + 1$, which is the last possible segmentation point before t. Note that in each iteration κ, when $t = T$ is reached, we obtain estimates for κ change points for the entire data sequence.

We now describe the criteria for picking the optimal number of change points κ^*. Empirically, $\widetilde{G}_t(\kappa, w)$ usually increases with κ, and tends to be kinked at k. This is expected since partitioning beyond the optimal number of partitions should not increase the goodness-of-fit at the same rate as before. We fit a piecewise linear function with two pieces on the empirical $\widetilde{G}_t(\kappa, w)$ values, and estimate the number of change points to be the κ at which the function transitions from one piece to

the other. This is similar to techniques used to determine cutoff values from scree plots.

2.5 Divergence Metrics

We now offer some guidelines for selecting the divergence metric R and its sample counterpart $\widehat{R}$, then propose two metrics, namely the energy statistic and $\mathcal{A}$-distance, which satisfy these guidelines. The energy statistic and Kolmogorov-Smirnov statistic, a special case of the $\mathcal{A}$-distance, are incorporated into the cp3o procedure. We refer to the two resulting algorithms as e-cp3o and ks-cp3o.

2.5.1 Selection Guidelines

We use the notation $X \overset{d}{=} Y$ to mean X and Y are identically distributed.

Property 1. *Convergence of empirical divergence to true divergence. Let* $\boldsymbol{X}_n = \{X_i\}_{i=1}^n$ *and* $\boldsymbol{Y}_m = \{Y_j\}_{j=1}^m$ *be two sets of independent random variables;* $X_i \overset{d}{=} X$ *for all* i*, and* $Y_j \overset{d}{=} Y$ *for all* j*.* $\widehat{R}(\boldsymbol{X}_n, \boldsymbol{Y}_m) \xrightarrow{a.s.} R(X,Y)$ *as the sample size* $\min(n,m) \to \infty$*, where* $R(X,Y) \geqslant 0$*, and equality holds iff* $X \overset{d}{=} Y$*.*

Property 2. *Single change point detection. For* $0 < \gamma < 1$*, suppose* $Z_1, \ldots, Z_{\lfloor \gamma T \rfloor} \overset{d}{=} X$ *and* $Z_{\lfloor \gamma T \rfloor + 1}, \ldots, Z_T \overset{d}{=} Y$ *for a sequence of any length* T*. For* $0 < \eta < 1$*, let* $A(\eta) = \{Z_i\}_{i=1}^{\lfloor \eta T \rfloor}$ *and* $B(\eta) = \{Z_j\}_{j=\lfloor \eta T \rfloor + 1}^{T}$*.* $\widehat{R}(A(\eta), B(\eta)) \xrightarrow{a.s.} \Theta_0^1(\eta|\gamma) R(X,Y)$ *as* $T \to \infty$*, where* $\Theta_0^1(\eta|\gamma)$ *maps from the interval* $(0,1)$ *to* $\mathbb{R}$*, and has a unique maximizer at* $\eta = \gamma$*.*

Property 1 concerns the convergence of the empirical divergence to the true

divergence metric. It is reasonable to enforce that the non-negative divergence be 0 when applied on two identically distributed random variables. Property 2 implies that for a large enough sample size with one change point, the empirical divergence metric will attain its maximum value when the estimated change point location η and true change point location γ coincide.

2.5.2 Energy Statistics

The E-statistics introduces by [60] are indexed by $\alpha \in (0, 2)$ and allows for the detection of *any* type of distributional change[1]. For a given α, the only distributional assumption made is that the observations have finite absolute α^{th} moments.

Suppose $\boldsymbol{X}_n = \{X_i\}_{i=1}^n$ and $\boldsymbol{Y}_m = \{Y_j\}_{j=1}^m$ are iid samples from distributions with probability measures F_X and F_Y, respectively. Then the population distance is

$$\mathcal{E}(X, Y|\alpha) = 2E|X - Y|^\alpha - E|X - X'|^\alpha - E|Y - Y'|^\alpha.$$

This is equivalent to

$$\mathcal{D}(X, Y|\alpha) = \int_{\mathbb{R}^d} |\phi_X(t) - \phi_Y(t)|^2 \omega(t|\alpha)\, dt$$

with an appropriately chosen positive weight function ω, where ϕ_X and ϕ_Y are the characteristic functions associated with distributions F_X and F_Y, respectively.

[1] If the detection of only mean changes is desired $\alpha = 2$ is used.

The empirical counterpart to $\mathcal{E}(X, Y|\alpha)$ is

$$\widehat{\mathcal{E}}(\boldsymbol{X}_n, \boldsymbol{Y}_m|\alpha) = \frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m}|x_i - y_j|^\alpha - \binom{n}{2}^{-1}\sum_{1\leqslant i<j\leqslant n}|x_i - x_j|^\alpha - \binom{m}{2}^{-1}\sum_{1\leqslant i<j\leqslant m}|y_i - y_j|^\alpha$$

Let γ denote the proportion of observations from F_X in the limit as $\min(n, m) \to \infty$. Then we define our divergence metrics as

$$R(X, Y|\alpha) = \gamma(1-\gamma)\mathcal{E}(X, Y|\alpha),$$
$$\widehat{R}(\boldsymbol{X}_n, \boldsymbol{Y}_m|\alpha) = \frac{mn}{(m+n)^2}\widehat{\mathcal{E}}(\boldsymbol{X}_n, \boldsymbol{Y}_m|\alpha).$$

Theorem 1. *Properties 1 and 2 are satisfied by the divergence metric based on the E-statistic.*

Proof. Using the result of [44, Theorem 1] we have that

$$\widehat{R}(A(\widehat{\gamma}), B(\widehat{\gamma})|\alpha) \xrightarrow{a.s.} \widehat{\gamma}(1-\widehat{\gamma})h(\widehat{\gamma};\gamma)\mathcal{E}(X, Y|\alpha)$$

where $h(\widehat{\gamma};\gamma) = \left(\frac{\gamma}{\widehat{\gamma}}\mathbb{1}_{\widehat{\gamma}\geqslant\gamma} + \frac{1-\gamma}{1-\widehat{\gamma}}\mathbb{1}_{\widehat{\gamma}<\gamma}\right)^2$. Therefore, $R(X, Y|\alpha) = \gamma(1-\gamma)\mathcal{E}(X, Y)|\alpha)$ and $\Theta_0^1(\widehat{\gamma}|\gamma) = \frac{\widehat{\gamma}(1-\widehat{\gamma})}{\gamma(1-\gamma)}h(\widehat{\gamma};\gamma)$, which can be shown to have a unique maximizer at $\widehat{\gamma} = \gamma$.

By definition, $\mathcal{D}(X, Y|\alpha) \geqslant 0$, with equality if and only if $F_X = F_Y$ by the uniqueness of characteristic functions. The rest of the proof follows from the equality of $\mathcal{D}(X, Y|\alpha)$ and $\mathcal{E}(X, Y|\alpha)$. □

Empirically, we use an incomplete U-statistic version of $\widehat{R}$ to reduce the number of samples needed the compute the pairwise distances. We define a window size δ, within which all pairwise distances are included, and outside which only adjacent points have their pairwise distances included. That is, suppose $\boldsymbol{X}_n = \{Z_a, Z_{a+1}, \ldots, Z_{a+n-1}\}$ and $\boldsymbol{Y}_m = \{Z_{a+n}, Z_{a+n+1}, \ldots, Z_{a+n+m-1}\}$, and define the following sets:

$$
\begin{aligned}
W_X^\delta &= \{(i,j) : a+n-\delta \leqslant i < j < a+n\} \cup \\
&\qquad \bigcup_{i=0}^{n-\delta-1} \{(a+i, a+i+1)\} \\
W_Y^\delta &= \{(i,j) : a+n \leqslant i < j < a+n+\delta\} \cup \\
&\qquad \bigcup_{i=\delta-1}^{m-2} \{(a+n+i, a+n+i+1)\} \\
B^\delta &= (\{a+n-1, \ldots, a+n-\delta\} \times \\
&\qquad \{a+n, \ldots, a+n+\delta-1\}) \cup \\
&\qquad \left(\bigcup_{i=\delta+1}^{m \wedge n} \{(a+n-i, a+n+i-1)\} \right)
\end{aligned}
$$

The incomplete U-statistic $\widetilde{\mathcal{E}}$ is then

$$
\begin{aligned}
\widetilde{\mathcal{E}}(\boldsymbol{X}_n, \boldsymbol{Y}_m | \alpha, \delta) &= \frac{2}{\#B^\delta} \sum_{(i,j) \in B^\delta} |X_i - Y_j|^\alpha - \\
&\frac{1}{\#W_X^\delta} \sum_{(i,j) \in W_X^\delta} |X_i - X_j|^\alpha - \frac{1}{\#W_Y^\delta} \sum_{(i,j) \in W_Y^\delta} |Y_i - Y_j|^\alpha
\end{aligned}
$$

This reduces computation of $\widehat{R}(\boldsymbol{X}_n, \boldsymbol{Y}_m | \alpha)$ from $\mathcal{O}\left(n^2 \bigvee m^2\right)$ to $\mathcal{O}\left(\delta^2 + n \bigvee m\right)$. Letting $\delta \leqslant C\lfloor\sqrt{T}\rfloor$ for some constant C results in a computational complexity of

$\mathcal{O}(T)$. Note that $\delta < w$, so we set $\delta = w - 1$. It is shown [49] that a strong law of large numbers result holds for incomplete U-Statistics, thus the incomplete U-statistic version of $\widehat{R}$ shares the same almost sure limit as $\widehat{R}$.

2.5.3 $\mathcal{A}$-distance

The $\mathcal{A}$-distance is introduced in [37]. It is a generalization of the Kolmogorov-Smirnov statistic, which is often used to quantify the distance between two empirical distribution functions.

We use the same notations as in Section 2.5.2. Let $\mathcal{A}$ be a collection of measurable sets from their domain. Then the $\mathcal{A}$-distance is defined as

$$d_{\mathcal{A}}(F_X, F_Y) = 2 \sup_{A \in \mathcal{A}} |F_X(A) - F_Y(A)|$$

The empirical $\mathcal{A}$-distance is

$$\widehat{d}(\boldsymbol{X}_n, \boldsymbol{Y}_m | \mathcal{A}) = 2 \sup_{A \in \mathcal{A}} \left| \frac{|\boldsymbol{x}_n \cap A|}{n} - \frac{|\boldsymbol{y}_m \cap A|}{m} \right|$$

Let γ denote the proportion of observations from F_X in the limit as $\min(n, m) \to \infty$. Then we define our divergence metrics as

$$\begin{aligned} R(X, Y | \mathcal{A}) &= \gamma(1 - \gamma) d_{\mathcal{A}}(F_X, F_Y), \\ \widehat{R}(\boldsymbol{X}_n, \boldsymbol{Y}_m | \mathcal{A}) &= \frac{mn}{(m+n)^2} \widehat{d}(\boldsymbol{X}_n, \boldsymbol{Y}_m | \mathcal{A}). \end{aligned}$$

In particular, for $\mathcal{A} = \{(-\infty, r) | r \in \mathbb{R}\}$, $\widehat{d}(\boldsymbol{X}_n, \boldsymbol{Y}_m | \mathcal{A})$ is the Kolmogorov-Smirnov

statistic.

Theorem 2. *Property 1 is satisfied by the divergence metric based on the $\mathcal{A}$-distance.*

Proof. We note that $\widehat{d}(\boldsymbol{X}_n, \boldsymbol{Y}_m|\mathcal{A}) \xrightarrow{a.s.} d_{\mathcal{A}}(F_X, F_Y)$ if $\mathcal{A}$ has a finite VC-dimension. From [37], for $M = \min(n, m)$,

$$P[|d_A(F_X, F_Y) - \widehat{d}(\boldsymbol{X}_n, \boldsymbol{Y}_m|\mathcal{A})| \geqslant \epsilon] < \pi_{\mathcal{A}}(2M)4e^{-M\epsilon^2/4},$$

where for domain D, $\pi_{\mathcal{A}}(n) = \max\{\ |\{A \cap B : A \in \mathcal{A}\}| : B \subseteq D \text{ and } |B| = n\}$. For finite VC-dimension c, $\pi_{\mathcal{A}}(n) < n^c$ by Sauer's Lemma. In particular, $\mathcal{A} = \{(-\infty, r)|r \in \mathbb{R}\}$ has $c = 2$. Hence,

$$P[|d_A(F_X, F_Y) - \widehat{d}(\boldsymbol{X}_n, \boldsymbol{Y}_m|\mathcal{A})| \geqslant \upsilon] < (2M)^c 4e^{-M\upsilon^2/4}.$$

We then note that for any $\upsilon > 0$,

$$\sum_{M=1}^{\infty} (2M)^c 4e^{-M\upsilon^2/4} = 4(2^c)Li_{-c}\left(e^{-\upsilon^2/4}\right) < \infty$$

where $Li_{-c}(x)$ is the polylogarithm function. Hence, $\widehat{d}(\boldsymbol{X}_n, \boldsymbol{Y}_m|\mathcal{A}) \xrightarrow{a.s.} d_{\mathcal{A}}(F_X, F_Y) \geqslant 0$, and if $F_X = F_Y$, then $\widehat{d}(\boldsymbol{X}_n, \boldsymbol{Y}_m|\mathcal{A}) \xrightarrow{a.s.} 0$.

The proof concludes with noticing $\frac{n}{m+n} \to \gamma$. □

Theorem 3. *Property 2 is satisfied by the divergence metric based on the $\mathcal{A}$-distance.*

Proof. As $\min(n, m) \to \infty$,

$$\widehat{R}(A(\widehat{\gamma}), B(\widehat{\gamma})|\mathcal{A}) \xrightarrow{a.s.} \widehat{\gamma}\,(1-\widehat{\gamma})\,g\,(\widehat{\gamma};\gamma)\,d_{\mathcal{A}}(F_X, F_Y)$$

where $g\,(\widehat{\gamma};\gamma) = \left(\frac{\gamma}{\widehat{\gamma}}\mathbb{1}_{\widehat{\gamma}\geqslant\gamma} + \frac{1-\gamma}{1-\widehat{\gamma}}\mathbb{1}_{\widehat{\gamma}<\gamma}\right)$. Therefore $\Theta_0^1(\widehat{\gamma}|\gamma) = \frac{\widehat{\gamma}(1-\widehat{\gamma})}{\gamma(1-\gamma)}g\,(\widehat{\gamma};\gamma) = \left(\frac{1-\widehat{\gamma}}{1-\gamma}\mathbb{1}_{\widehat{\gamma}\geqslant\gamma} + \frac{\widehat{\gamma}}{\gamma}\mathbb{1}_{\widehat{\gamma}<\gamma}\right)$, and it is maximized at $\widehat{\gamma} = \gamma$. □

2.6 Simulation Study

To assess the performance of the segmentations, we use Fowlkes and Mallows' adjusted Rand index [21]. This value is calculated by comparing an estimated segmentation to the true segmentation. The index takes into account both the number of change points as well as their locations, and lies in the interval $[0, 1]$, where it is equal to 1 if and only if the two segmentations are identical.

We also include two measures of discrepancy between the true and estimated change point locations as an assessment of estimation accuracy. T2E is the average shortest distance from a true change point to the estimated change points, and E2T is the average shortest distance from an estimated change point to the true change points. A low T2E shows that all the true change points are well-estimated, and a low E2T shows that all the estimated change points are close to true change points.

For each simulation scenario, we apply various methods to 100 randomly generated time series, each with three evenly-spaced change points. We compare our methods with E-divisive [35] and NPCP-F [31] (nonparametric, approximate/bisection search), and PELT [38] (parametric, exact/dynamic programming search).

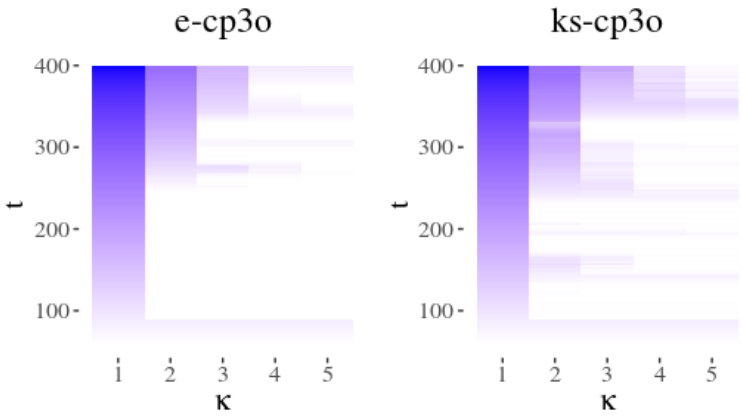
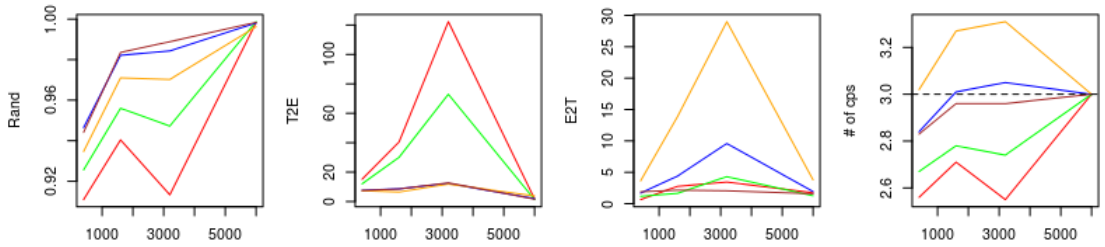


Figure 2.1: Color represent number of candidate change points in search space $S_t(\kappa)$ at each time index t and iteration κ. The darker the color, the higher the number. $S_t(\kappa)$ is at it's maximum at $\kappa = 1$, and is rapidly pruned in subsequent iterations.

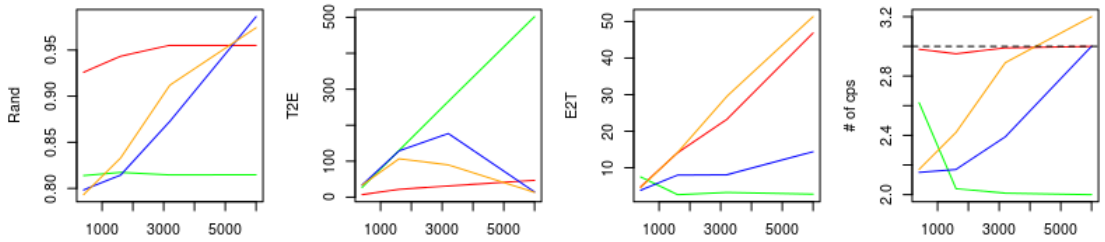
All methods were run with their default parameter values unless otherwise specified. For E-Divisive and e-cp3o this corresponds to $\alpha = 1$. For PELT, e-cp3o and ks-cp3o, the upper bound of number of changes K was set to 5. For all methods, the minimum segment size was set to approximately $1.5\sqrt{T}$ observations, that is, $w = 30, 60, 90, 120$ for time series of length $T = 400, 1600, 3200, 6000$, respectively. All experiments were run on a standard desktop computer.

2.6.1 Effects of Pruning

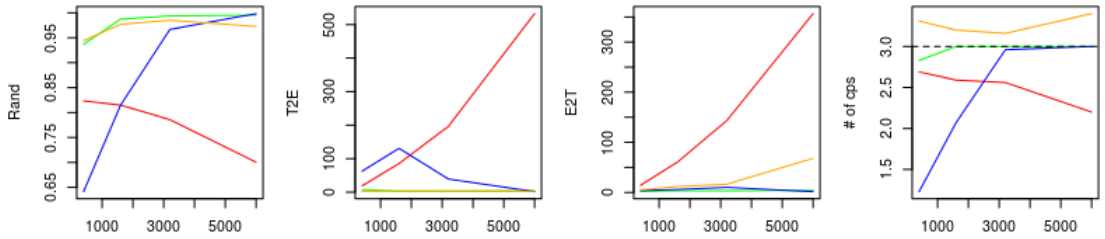
We demonstrate the effects of the pruning step within the dynamic program on the search space $S_t(\kappa)$ in Figure 2.1. The darker the color, the bigger the search space. The search space is pruned significantly within a few iterations.



(a) Simulation 1 with mean and variance changes in Gaussian distribution.



(b) Simulation 2 with distribution, mean and tail changes.



(c) Simulation 3 with changes in t and Cauchy distribution.

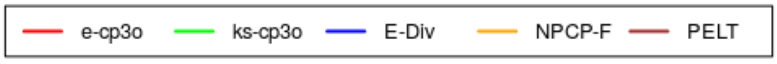


Figure 2.2: Average Rand index, discrepancy values and number of change points detected against length of time series. True number of change points denoted by black dotted line. Good performance is reflected by Rand close to 1, small T2E and E2T, and estimated change point number close to 3.

T	e-cp3o	ks-cp3o	E-Div	NPCP-F	PELT
400	0.119	0.918	5.285	5.475	0.002
1600	1.811	77.869	118.672	85.288	0.020
3200	7.977	683.003	756.503	346.632	0.087
6000	27.855	4951.490	1841.570	1357.562	0.342

Table 2.1: Average runtimes (s) of the first univariate simulation from Section 2.6.2 with mean and variance changes in Gaussian distributions.

2.6.2 Simulation 1

This set of simulations consist of independent Gaussian observations which undergo changes in their mean and variance. The distribution parameters were chosen so that $\mu_j \overset{iid}{\sim} Unif(-10, 10)$ and $\sigma_j^2 \overset{iid}{\sim} Unif(0, 5)$.

As can be seen from Table 2.1 and Figure 2.2a, PELT was fast and suffered little loss in accuracy in identifying change points in longer time series, as observed from the Rand and discrepancy (T2E and E2T) values. At $T = 400, 1600$ and 3200, from the lower estimated number of change points and the higher values of T2E, we notice that e-cp3o and ks-cp3o did not always detect all the changes. But from the lower values of E2T, we see that the points which e-cp3o and ks-cp3o did identify as changes are amongst the closest to the true changes. At $T = 6000$, e-cp3o and ks-cp3o performed comparably with the competing methods in terms of segmentation quality.

The computation time of the ks-cp3o procedure did not scale well with time series length because of the sorting step required to calculate the statistic. In the `ecp` *R* package, we also provide a faster version of ks-cp3o which only computes the Kolmogorov-Smirnov statistic using points within a window of size δ around

each candidate segmentation point. Its average runtime at $T = 6000$ is $14.483s$, but it detected a slightly lower average number of change points (2.620), and hence is excluded from the reporting.

2.6.3 Simulation 2

Time series in this simulation study contain a change in distribution, mean, and tail index. The data transitions from a exponential distribution $Exp\left(\frac{1}{3}\right)$ to a normal distribution $N(3, 1)$ to a standard normal distribution $N(0, 1)$. The tail index change is caused by a final transition to a t-distribution with 2.01 degrees of freedom.

We do not include PELT in the following experiments since it is a parametric method that detects only mean and variance changes.

We expect that all methods included will be able to easily detect the mean change and will have more difficulty detecting the change in tail index. Results for this set of simulations can be found in Figure 2.2b. Runtimes are similar to those in Table 2.1 with PELT excluded.

At $T = 400, 1600$ and 3200, e-cp3o was not only significantly faster than all other procedures, but also managed to generate the best segmentations on average. While most procedures tended to miss the tail index change, e-cp3o detected the most number of change points with averages within 0.05 of the true number 3. e-cp3o had higher E2T since it picked out the third change point more often than the other methods, but the accuracy of detecting the third change was not as high as those for the first two changes. At $T = 6000$, E-Divisive overtook in terms of segmentation quality, but e-cp3o was much faster and hence provided a better

balance between speed and accuracy.

2.6.4 Simulation 3

The data transitions from a t-distribution $t_{0.1}$ to $t_{1.9}$ to a Cauchy distribution $Cauchy(-2, 1)$ to $Cauchy(0, 1)$. We use $\alpha = 0.09$ instead since we need $\alpha < 0.1$ for the moment assumptions of E-statistics to hold. Complete results are shown in Figure 2.2c. Runtimes are similar to those in Table 2.1 with PELT excluded.

In the short time series setting ($T = 400$), NPCP-F and ks-cp3o performed comparatively. In the long time series setting ($T = 6000$), E-Divisive and ks-cp3o performed comparatively. In general, ks-cp3o had the most consistent performance by almost always achieving the highest Rand and lowest discrepancy values. In fact, ks-cp3o picked out the correct number of change points in every sample series from $T = 1600$ onwards. It demonstrated great potential in change point detection in general datasets where commonly desired distributional properties cannot be assumed.

Due to the small value of α which makes the E-statistics smaller in magnitude and therefore more difficult to distinguish, e-cp3o does not perform as well as the other methods. Moreover, it is not straightforward to determine the best α to use in practice, especially when extreme observations are present. Hence, it is important to select a goodness-of-fit that is appropriate for the data.

2.7 Applications to Real Data

2.7.1 Temperature Anomalies

We examine the HadCRUT4 dataset of [48]. This dataset consists of monthly global temperature anomalies from 1850 to 2014. From looking at the plot of the tropical land air anomaly time series it is suspected that there is some dependence between observations. This assumption is quickly confirmed by looking at the auto-correlation plot. As a result, we apply the change point procedure to the differenced data which visually appears to be piecewise stationary. The auto-correlation plot for the differenced data shows that much of the linear dependence has been removed, however, the same plot for the differences squared still indicates some dependence. We believe that this indicated dependence can be attributed to changes in distribution.

The e-cp3o and ks-cp3o procedures were applied with a minimum segment length of 5 years, corresponding to $w = 60$; a maximum of $K = 5$ change points were fit. We chose default values $\alpha = 1$ for e-cp3o. e-cp3o identified change points at April 1917 and April 1969, and ks-cp3o identified changes at February 1864, May 1878 and September 1898. These are shown in Figure 2.3.

The May 1878 change point may be a result of a large climate disruption in 1877-1878, which may be caused by a major El Niño episode. The April 1969 change point occurs around the United Nations Conference on the Human Environment. This conference, which was held in June 1972, focused on human interactions with the environment.

e-cp3o used $3.659s$ and ks-cp3o used $376.138s$. With the same parameters,

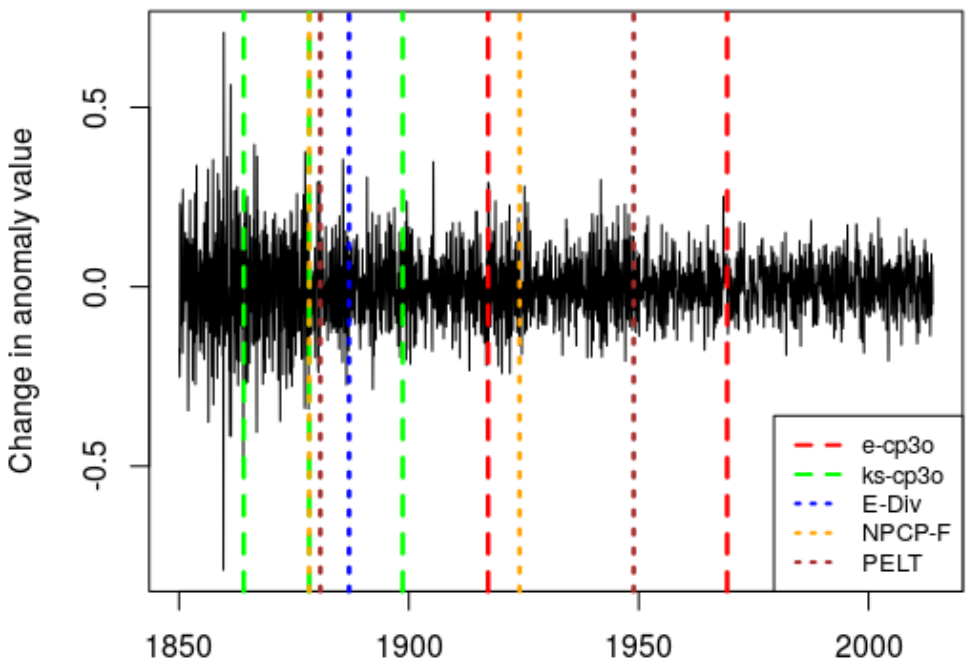


Figure 2.3: Change in land air temperature anomalies for the Tropical climate zone from February 1850 to December 2013. Estimated change point locations indicated by dashed vertical lines.

competing methods E-divisive, NPCP-F and PELT used 135.795s, 111.377s and 0.078s respectively. Segmentation results vary and the true change points are unknown, which make it difficult to compare methods. However, we note that even though ks-cp3o took the longest time, it is the only method that identified the 1864 change point, which visually does look like a true change.

2.7.2 Exchange Rates

We apply e-cp3o to a set of spot foreign exchange (FX) rates obtained through the `R` package `Quandl` [47], and compare results with multivariate methods E-divisive and NPCP-F. We consider the 3-dimensional time series consisting of monthly FX rates for Brazil (BRL), Russia (RUB), and Switzerland (CHF) against the United

States (USD). The time horizon spanned is September 30, 1996 to February 28, 2014, which results in a total of 210 observations. We look at the change in the log rates, such that marginal processes appear to be piecewise stationary.

The e-cp3o procedure is applied with a minimum segment length of 12 observations (a year), which corresponds to a value of $w = 12$. Furthermore, we have chosen to fit at most $K = 5$ change points, and default values of $\alpha = 1$ is used. This specific choice of values resulted in change points being identified at December 1998, August 2002 and April 2008. These results are depicted in Figure 2.4.

Changes in Russia's economic standing leading up to the 1998 ruble crisis may be the cause of the December 1998 change point. The August 2002 and April 2008 change points may be the results of the 2002 South American economic crisis and 2008 financial crisis respectively. The change point identified at August 2002 also coincides with an economic shift in Brazil. In January 1999 the Brazilian Central Bank announced that they would be changing to a free float exchange regime, thus their currency was no longer pegged to the USD. This change devalued the currency and helped to slow the ongoing economic downturn.

e-cp3o used $0.026s$, while E-Divisive and NPCP-F used $1.59s$ and $1.213s$ respectively using the same parameters. e-cp3o and E-Divisive found similar change points, but e-cp3o is much faster. NPCP-F seemed to have underestimated the number of change points and only identified one whereas the other methods identified three or more.

2.8 Conclusion

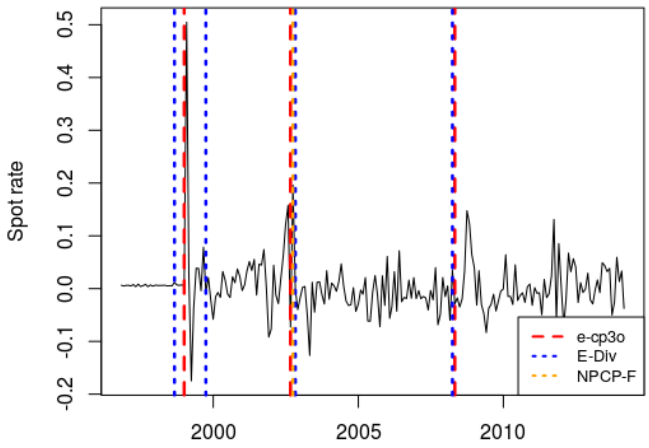
We have presented an approximate search procedure that incorporates pruning in order to reduce the amount of unnecessary calculations and to dramatically reduce computational costs. This search method can be used with almost any goodness-of-fit metric in order to identify change points in univariate and multivariate time series. In addition, this is accomplished without the user having to specify any sort of penalty parameter or function.

By combining the cp3o search algorithm with E-statistics and Kolmogorov-Smirnov statistics, we proposed e-cp3o and ks-cp3o algorithms, respectively. These methods can perform nonparametric multiple change point analysis that can detect *any* type of distributional change. The e-cp3o algorithm makes mild distribution (moment) assumptions, and its runtime scales well with the length of time series. The ks-cp3o algorithm makes no assumption about the data distribution. Although it is less sensitive in detecting changes at the tails of the distribution and its runtime does not scale as well with the series length, ks-cp3o is still valuable in detecting changes in scenarios where we have no knowledge about the data distribution and the runtime is reasonable on shorter time series.

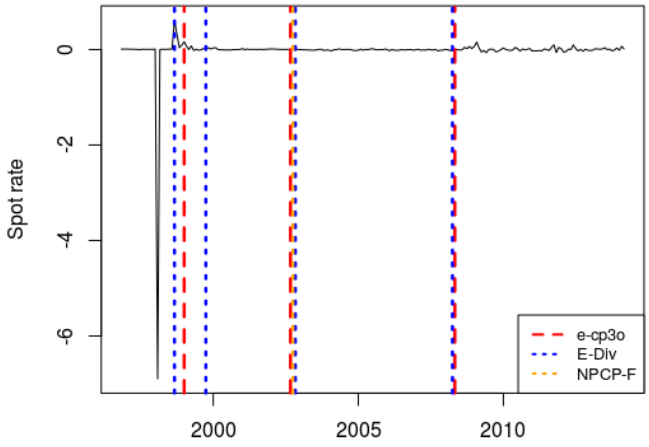
As the simulation studies demonstrate, the cp3o procedures do not uniformly record the best running time, average Rand values or average discrepancy values. However, when accuracy and computation time are viewed together across different data scenarios, the cp3o procedures are either better or comparable to almost all other competitors. Moreover, greater care in choosing a goodness-of-fit metric that is suitable to the data application is likely to improve performance further in terms of accuracy and/or speed. Hence, we would advocate the cp3o procedure as a general purpose change point procedure.

Acknowledgement

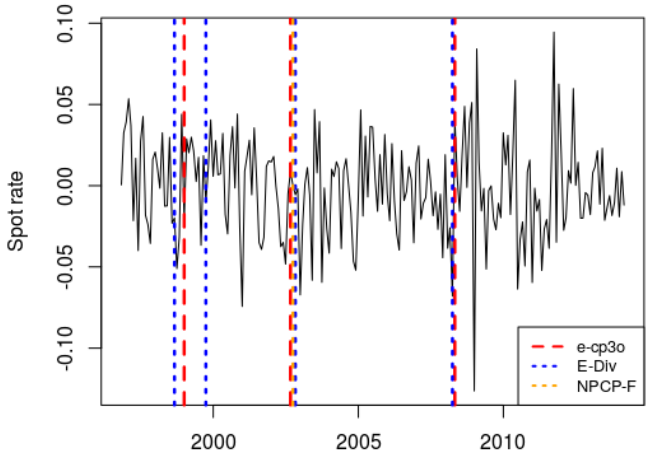
Matteson was supported by an NSF CAREER award (DMS-1455172), a Xerox PARC Faculty Research Award, and the Cornell University Atkinson Center for a Sustainable Future (AVF-2017).



(a) Brazil



(b) Russia



(c) Switzerland

Figure 2.4: Time series for FX spot rates for each of the three countries' currencies versus the USD. Estimated change point locations indicated by dashed vertical lines.

CHAPTER 3

UNSUPERVISED MULTIVARIATE CHANGE DETECTION VIA BAYESIAN SOURCE SEPARATION

Contents in this chapter are published in [65].

3.1 Introduction

Change detection segments sequential data such that observations in each segment share the same characteristics. We can view it as a specific form of clustering where sequential data points tend to cluster together. Two common sequential orderings are time and physical location. Change detection methods have been developed for many types of data including video [63], social network [46] and numerical time series [28, 44, 62]. Online change detection processes streaming data from a system and raises an alert as soon as it estimates a state change. This is useful for monitoring purposes such as fall detection [26], health monitoring and network and machine monitoring [63, 67].

Offline change detection segments the data retrospectively and is useful for uncovering events and systematic behaviors in data analysis tasks. It is applied in a variety of fields including energy consumption [28], genomics [51] and finance [20]. Furthermore, in the potential presence of change points, utilizing change detection prior to data modeling can help prevent building inappropriate models under the assumption of data homogeneity, and consequently supports improved prediction and statistical inference.

In this work, we study offline multiple change detection in multivariate data, specifically where the data exhibit mean changes that can occur simultaneously in

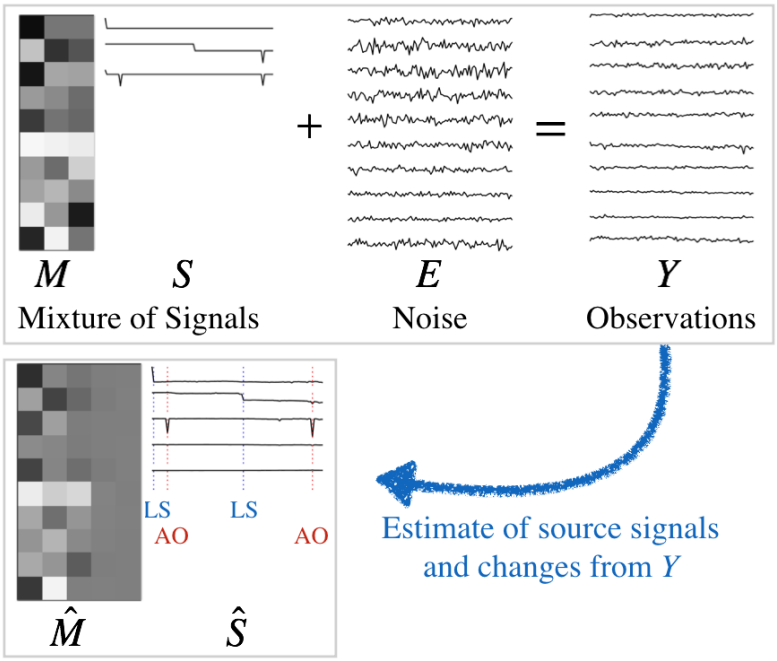


Figure 3.1: Given observations generated by the linear mixing of signals contaminated by noise, ABACUS estimates the source signals and detect additive outliers (AO, red) and level shifts (LS, blue). In M, darker and lighter cells represent negative and positive values respectively, and medium gray cells represent zero.

several channels. The direction and magnitude of change can be different across channels. Here, we refer to mean changes lasting a single time unit with an immediate return as additive outliers (AO), and mean changes with duration two or greater as level shifts (LS). We assume that the multivariate data are generated by low-dimensional latent source signals through linear mixing according to the model $Y = MS + E$, shown in Figure 3.1, similar to the general linear setting used in the blind source separation literature [33, 45]. Notation-wise, M is the mixing matrix, and Y, S and E are the observations, source signals and noise, respectively. Observed mean changes manifest from the latent space, and we detect changes by estimating these latent source signals, which possess 'semantic' meaning of the underlying states and are free of noise.

Multivariate data are readily observed in many applications in today's world,

and mean changes are of particular interest since the mean is often a salient aspect of the system state. Multivariate data can be observations from multiple channels monitoring a single system, or a collection of univariate data streams from multiple related systems. Examples of the first scenario include household power consumption measured with sub-meters [28], and wine quality based on physicochemical test variables [1]. Examples of the second scenario include array comparative genomic hybridization measurements from several patients with the same medical condition [44]. In these and other examples, change points in multivariate data sometimes occur simultaneously in multiple channels because the signals may be driven by the same underlying processes. It is of interest to identify these shared change points to further analyze the relationship between channels. Running univariate change detection on each channel does not encourage identification of such shared changes.

Finding changes in multidimensional data is known to be a difficult problem. If the magnitude of change as measured by symmetric Kullback-Leibler divergence is kept constant, detectability of the change worsens when the data dimension P increases. This can hinder detection even at dimensions as low as $P = 10$ [1]. Another issue arises when the data dimensions P exceeds the sample size N. If one wishes to use hypothesis testing to test for homogeneity, naive calculations of familiar test statistics such as the Hotelling's t-squared statistic are prohibitive. Several approaches tackle multivariate data by incorporating a dimensionality reduction step [63, 62], but these either project the data onto a single dimension or require the user to select the reduced dimensionality.

Our main contribution is to successfully integrate sparse Bayesian blind source separation with a change detection framework. No previous work on latent variable

modeling explicitly considered source signals with unconstrained mean changes. Bayesian variations of principle component analysis (PCA) are capable of automatic dimensionality selection [9, 68], and shrinkage priors also achieve desirable properties in trend filtering [39]. In our Bayesian latent model, we use horseshoe priors to recover the lower-dimensional source signals and to simultaneously model the change points. The two tasks complement each other since the source signals exhibit changes. We propose ABACUS, *Automatic BAyesian Changepoint Under Sparsity*, an automatic procedure that simultaneously detects additive outliers and level shifts via estimating components from the source separation problem. Figure 3.1 gives an example where ABACUS recovers the true latent change space of size three by estimating values in the appropriate dimensions of M and S to zero, and ABACUS also locates relevant change points. We show through simulations and real data applications that ABACUS achieves better performance in both change detection and source recovery.

3.2 Related Works

3.2.1 Multivariate Change Detection

Change detection methodologies usually consider a data collection style where either P or N may be varied depending on time and resource constraints. The first scenario is common when the data consists a set of univariate data streams of a fixed length, and more data streams can be collected if desired. The second scenario happens when the number of observation channels is fixed, but the data length can be varied by adjusting the sampling rate or by adjusting the scale of

time or space for which the system is observed.

Authors of [11] formulated multivariate change detection as a group fused Lasso, and showed empirically that detection probability approaches one with increasing P when noise is small. Variants of binary segmentation produce approximately optimal segmentations by iteratively detecting single change points [51, 44]. Dynamic programming with a suitable multivariate goodness-of-fit metric can recursively segment the data [66]. The above methods directly segment the observations and some assume independence across channels [62, 22]. We recover the latent change space with prior belief that only the latent signals are independent given model parameters.

Some works use a two-step procedure with data compression onto a low dimension $K \ll P$ followed by change detection. Projection onto a single dimension enables univariate change detection [22]. For $K > 1$, [55] applies univariate change detection on each latent signal after Independent Component Analysis (ICA). Random projection where the projection is either fixed or varied across time has been paired with hypothesis testing [63]. Using compressive measurements, where the projection matrix is a random projection or drawn from a Gaussian ensemble, [3] derives the number of observations required for a target detection delay. For the above methods, the user needs to specify the compression ratio through K. Our proposed method ABACUS is more robust to the specification of K due to automatic dimensionality selection by our sparsity assumptions. In contrast to the latent variable model that we employ, these methods also ignore estimating the mixing matrix.

Bayesian approaches in change detection typically rely on using indicator variables to denote the presence of change points. The BCP method [20, 6] assumes

that observations in each segment are independent and identically distributed as Gaussian, and updates posterior segment means conditional on the segmentation at each iteration of an MCMC scheme. A uniform prior $\mathrm{U}(0, q)$ is put on the change point probabilities, and the user tunes the chances of discovering shorter or longer segments through q. In [28], given the segmentation informed by the indicator variables, a Wilcoxon rank sum test is performed at each index of the data and the resulting p-values are modeled as a Beta-Uniform mixture. The data likelihood is written as a composite marginal likelihood of the p-values. The formulation makes no assumption on the distributional form of the data.

ABACUS similarly utilizes the sparsity of changes by applying horseshoe priors, modeling the presence and absence of changes, but also the change directions and magnitudes. We utilize the horseshoe prior as it is known for robustness and superior shrinkage properties [12, 25]. Empirically, differences in neighboring non-change location means are effectively shrunk to zero.

3.2.2 Latent Variable Model

We consider the following setting: $Y = MS + E$, and we would like to find the decomposition of Y into the $P \times r$ mixing matrix M, the $r \times N$ source signals matrix S, and the $P \times N$ noise matrix E. In general, the problem has no unique solution. For any orthogonal matrix Q and solution pair M and S, MQ and $Q^T S$ is an equally viable solution. The true scaling of the source signals cannot be recovered as well. There are a variety of relevant methods in the literature on topics such as blind source separation and factor analysis, and their distinction is mainly in the assumptions made on the decomposed components. No previous work dealt specifically with our setup where S is piecewise-constant.

Independent Component Analysis (ICA) is a popular approach in blind source separation which assumes mutually independent non-Gaussian latent variables. ICA recovers the latent signals by maximizing the non-Gaussianity of each signal or minimizing the dependence between the signals. In contrast, Factor Analysis (FA) assumes spherical Gaussian latent variables. This assumption is not valid in our case where the source signals are assumed piecewise-constant. FA estimates M using the covariance matrix of the observed variables, and cannot be used directly when the covariance matrix is singular. Both ICA and FA require a user-specified K as an estimate for r, and do not have any sparsity assumption on M. FA does not work with large K since the covariance matrix does not have sufficient information to estimate M [33].

PCA does not specifically model data variability from the noise component, and is often used for the purpose of dimensionality reduction. Probabilistic PCA (PPCA) [19], on the other hand, includes this component and can additionally be used to recover latent signals. Both the latent variables and the noise are given a spherical Gaussian distribution. PPCA estimates the parameters by maximum likelihood estimation. Bayesian PCA (BPCA) [9] is a Bayesian treatment on the basis of PPCA and further places a spherical Gaussian prior on each column of M, which automatically shrinks entire columns of M to recover the effective dimensionality of the latent space. Further works include shrinking column i of M and row i of S simultaneously by assigning them mean-zero spherical Gaussian priors with the same covariance [68], and capturing sparsity only in S through a three parameter Beta prior acting on the local, factor-specific and global level [23]. In our case, the source signals are piecewise-constant, and hence not sparse, Gaussian or independently and identically distributed. We apply the sparsity assumption on the change magnitudes instead, and use multiple levels of sparsity on both the

change magnitudes and M.

3.3 Problem Formulation

We observe $Y \in \mathbb{R}^{P \times N}$, a P-dimensional data stream of length N. Each column take the form $Y_{\cdot n} = MS_{\cdot n} + E_{\cdot n}$, where $M \in \mathbb{R}^{P \times r}$ is the mixing matrix, $S_{\cdot n}$ is the r-dimensional source signal, and $E_{\cdot n}$ is the P-dimensional noise vector, at index n. This is the general formulation of the cocktail party problem with P microphones and r conversations observed for N time points. Here, Y is not necessarily a time series, but data which are indexed sequentially. S is assumed to have full row rank.

We assume that the source signals are piecewise-constant. Each segment can be of any length, and adjacent segments have different means. Latent variables are driven by the same underlying system state, and hence may share change locations, but change directions and magnitudes are not necessarily the same. We assume that the linearly-mixed signals are corrupted by independent Gaussian noise, but noise variances are not necessarily the same across channels. In the cocktail party analogy, this means that each microphone is subject to a different amount of noise due to the environment and microphone quality. The Gaussian assumption is standard in parametric change detection models [63, 51, 6].

We aim to decompose Y into its components without further information. Although the decomposition solution is not unique, [23] reports that sparsity formulations in their Bayesian latent variable model helped to stabilize fitting. We similarly apply multiple levels of sparsity in our model, as described in the next section.

3.4 Proposed Method: ABACUS

We introduce our Bayesian data model and estimation method, as well as our change detection approach which makes use of MCMC posterior samples.

3.4.1 A Bayesian Latent Variable Model

We decompose source signals further into components consisting of either additive outliers (AO) or level shifts (LS). Additive outliers are abrupt mean changes lasting for only one index, while level shifts persist for two or more indices. This decomposition allows us to naturally distinguish between the two types of changes, such that they can be studied separately, e.g., a user may remove additive outliers and retain level shifts for analysis. Let K be a user-specified upper bound for $\text{rank}(S) = r$ such that $r \leqslant K < P$. Then our modified formulation is

$$
\begin{aligned}
Y_{\cdot n} &= MS_{\cdot n} + E_{\cdot n} \\
S_{\cdot n} &= S^{(0)}_{\cdot n} + S^{(1)}_{\cdot n} \\
S^{(0)}_{\cdot n} &= V^{(0)}_{\cdot n} \text{ and } \triangle S^{(1)}_{\cdot n} = V^{(1)}_{\cdot n}
\end{aligned}
$$

where M is the $P \times K$ mixing matrix, S is the $K \times N$ source signal matrix, E is the $P \times N$ error matrix, $S^{(0)}$ and $S^{(1)}$ are the $K \times N$ component matrices of S, $V^{(0)}$ and $V^{(1)}$ are $K \times N$ 'sparse' matrices, and $\triangle$ is the differencing operator. The diagonal covariance matrix of $E_{\cdot n}$ is denoted by $\Psi = \text{diag}\,(\psi)$, so $E_{\cdot n} \sim \text{N}\,(0, \Psi)$.

We place sparse group priors on the columns of M and rows of $V^{(0)}$ and $V^{(1)}$ for dimensionality reduction of the latent space. Furthermore, we place sparse

group priors on the columns of $V^{(0)}$ and $V^{(1)}$ to select a subset of indices as change locations. We also use elementwise sparsity on $V^{(0)}$ and $V^{(1)}$ to allow sparse changes for each latent variable.

We choose to use horseshoe priors because the horseshoe-shaped shrinkage profile discovers null values without diminishing strong signals. [12]. We extend the global-local shrinkage hierarchy to impose sparsity in the model at the element and group level.

For $1 \leqslant i \leqslant P$ and $1 \leqslant h \leqslant K$ and $1 \leqslant n \leqslant N$ and $d \in \{0, 1\}$, we set priors as

$$
\begin{aligned}
M_{\cdot h} | \lambda_h^{(0)}, \lambda_h^{(1)},\ \tau^{(0)}, \tau^{(1)}, \Psi &\sim \mathrm{N}\left(0, \lambda_h^{(0)} \lambda_h^{(1)} \tau^{(0)} \tau^{(1)} \Psi\right) \\
V_{hn}^{(d)} | \phi_n^{(d)}, \lambda_h^{(d)}, \gamma_{hn}^{(d)},\ \tau^{(d)} &\sim \mathrm{N}\left(0, \phi_n^{(d)} \lambda_h^{(d)} \gamma_{hn}^{(d)} \tau^{(d)}\right) \\
\psi_i &\sim \Gamma^{-1}\left(1,\ 1\right) \\
\tau^{(d)} | \xi^{(d)} &\sim \Gamma^{-1}\left(\frac{1}{2},\ \frac{1}{\xi^{(d)}}\right) \\
\lambda_h^{(d)} | \eta_h^{(d)} &\sim \Gamma^{-1}\left(\frac{1}{2},\ \frac{1}{\eta_h^{(d)}}\right) \\
\phi_n^{(d)} | \omega_n^{(d)} &\sim \Gamma^{-1}\left(\frac{1}{2},\ \frac{1}{\omega_t^{(d)}}\right) \\
\gamma_{hn}^{(d)} | \zeta_n^{(d)} &\sim \Gamma^{-1}\left(\frac{1}{2},\ \frac{1}{\zeta_{hn}^{(d)}}\right) \\
\xi^{(d)}, \eta_h^{(d)}, \omega_n^{(d)}, \zeta_{hn}^{(d)} &\sim \Gamma^{-1}\left(\frac{1}{2},\ 1\right)
\end{aligned}
$$

where $N()$ denotes the Gaussian distribution and $\Gamma^{-1}()$ denotes the Inverse Gamma distribution. Marginally, the shrinkage parameters $\tau^{(d)}, \lambda_h^{(d)}, \phi_n^{(d)}$ and $\gamma_{hn}^{(d)}$ are half-Cauchy, as in the horseshoe setup. Given the shrinkage parameters, we impose the prior belief that the source signals are independent, but the posterior is not necessarily so.

Let $D^{(1)}$ be the matrix representation of $\triangle$ such that $S^{(1)}\left[D^{(1)}\right]^T = V^{(1)}$, and let $D^{(0)} = I$ such that $S^{(0)} = V^{(0)}$. Now, we define the expression $F = SS^T + \text{diag}\left(\tau^{(0)}\tau^{(1)}\lambda^{(0)}\lambda^{(1)}\right)^{-1}$, which appears below.

For $1 \leqslant i \leqslant P$, $1 \leqslant n \leqslant N$, and $d \in \{0, 1\}$, we derive the full conditionals for the posterior distribution of the main model components below. Distributions of all additional parameters are provided in the Supplementary Materials. First,

$$\begin{aligned} M_{i\cdot}|\cdot &\sim \text{N}\left(F^{-1}SY_{i\cdot}, \psi_i F^{-1}\right) \\ \psi_i|\cdot &\sim \Gamma^{-1}\left(1 + \frac{N}{2},\ 1 + \frac{1}{2}(Y_{i\cdot} - M_{i\cdot}S)^T(Y_{i\cdot} - M_{i\cdot}S)\right) \end{aligned}$$

and for $V_{\cdot n}^{(d)}$, the full conditional distribution is

$$\text{N}\left(\left[B^{(n)}\right]^{-1} M^T\Psi^{-1}C^{(n)}\left[D^{(d)}\right]_{\cdot n}^{-1}, \left[B^{(n)}\right]^{-1}\right)$$

where

$$\begin{aligned} B^{(n)} &= M^T\Psi^{-1}M\left(\left[D^{(d)}\right]_{n\cdot}^{-T}\left[D^{(d)}\right]_{\cdot n}^{-1}\right) + \\ &\qquad \text{diag}\left(\phi_n^{(d)}\lambda^{(d)}\gamma_{\cdot n}^{(d)}\tau^{(d)}\right)^{-1} \\ C^{(n)} &= Y - MS + MV_{\cdot n}^{(d)}\left[D^{(d)}\right]_{n\cdot}^{-T}. \end{aligned}$$

We use Gibbs sampling to approximate the posterior. The procedure is easily parallelized. Furthermore, the number of model components and parameters depend on K and correctly setting a small K can significantly reduce computational time.

In our modified $Y = MS + E$ model, multiple levels of sparsity regulate the transformations each solution pair M and S can take to reach a different solution pair, but we cannot identify the sign and scaling of M and S. To recover the components and parameters empirically, we use the median of the posterior samples to provide robustness against possible movements of the sampling path between different solutions.

3.4.2 Change Detection

In our data model, $V^{(0)}$ and $V^{(1)}$ contain the changes for each latent variable at each index. The matrices are sparse since only entries which correspond to changes are nonzero. Let $f_n^{(d)}$ be the element with the largest magnitude in $V_{\cdot n}^{(d)}$. At any index n, $f_n^{(d)}$ is nonzero if and only if there is a change of type d in at least one latent variable. Finding all such indices is equivalent to finding the change locations. We use the median defined

$$\widehat{g}_n^{(d)} = \text{median}\left(\widehat{f}_n^{(d)}\right)$$

for robustness with empirical samples.

Since we impose horseshoe priors on $V^{(d)}$, the entries are shrunk to approximately zero but not exactly zero. To identify the approximately zero values in the estimated $\widehat{g}^{(d)}$, we apply kernel density estimation on $|\widehat{g}^{(d)}|$ with a rectangular kernel and set the cutoff to be at the first minimum in the density function such that the minimum value is below threshold δ. The threshold ensures that the approximately zero and non-zero values are sufficiently different. We set $\delta = 10^{-10}$ for all our experiments.

3.5 Implementation

We fit the full Bayesian latent variable model in Section 3.4.1 by first fitting a partial model. The partial model differs only in that it does not include $S^{(0)}$ or $V^{(0)}$ and their associated parameters, and hence we drop the superscripts when referring to its components and parameters. Changepoints *cpt* detected by the partial model are a mix of additive outliers (AO) and level shifts (LS), with the former being detected as two consecutive mean changes of opposite signs in $\widehat{g}$. We distinguish between the two types of changes according to this observation with Algorithm 1, and produce additive outliers *cpt*0 and level shifts *cpt*1. We decompose the estimated components and parameters from the partial model according to *cpt*0 and *cpt*1, and pass them to the full model as initialization. For example, $V^{(0)}$ is initialized with values from $\widehat{V}$ at *cpt*0, and $V^{(1)}$ is initialized with values from $\widehat{V}$ at *cpt*1.

Algorithm 1: Separating AO and LS changes

Data: Estimated $\widehat{g}$, ordered change points *cpt*
Result: Additive outliers *cpt*0, level shifts *cpt*1

```
cpt0 = cpt1 = {};
i = 1;
while not at end of cpt do
    condition 1: cpt[i + 1] − cpt[i] = 1;
    condition 2: ĝ corresponding to cpt[i] and cpt[i + 1] are of opposite
     signs;
    if condition 1 and 2 are True then
        add cpt[i] to cpt0;
        i = i + 2;
    else
        add cpt[i] to cpt1;
        i = i + 1;
    end
end
```

The partial model is smaller and hence can quickly estimate components and parameters for initialization. This step stabilizes fitting the full model and helps to

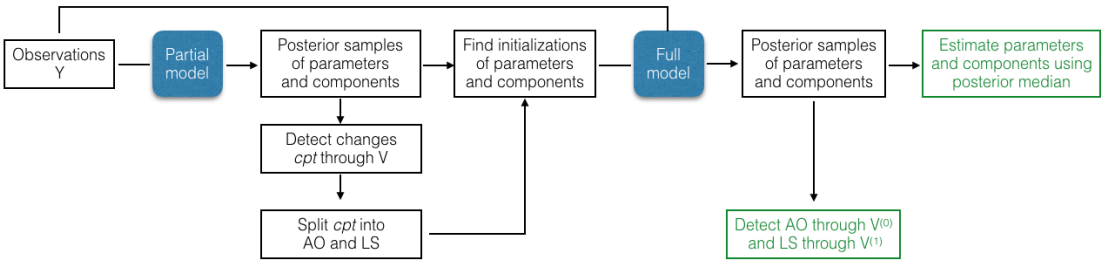


Figure 3.2: Implementation procedure. From observations Y, a partial model is first fit and its estimations initialize the full Bayesian model. Final estimates of source signals and change points are obtained from the median of MCMC samples.

achieve better distinction between the two types of changes. The entire procedure is shown in Figure 3.2. The final two boxes in green indicate the final outputs for change detection and source recovery.

3.6 Simulation Study

We conduct several experiments according to the model $Y = MS + E$ described in Section 3.3. We fix the latent space dimensionality $r = 3$, and vary N and P. Some methods require a user-specified K as an estimate for r, and we test their robustness to the selection of K. Although r cannot be controlled in real data applications, we also include an experiment where r is varied to simulate different data generating processes. Entries of M are drawn independently from $\text{Unif}(-1, 1)$, and each noise variance as $\psi_i \sim \text{Unif}(0.1, 5)$. Given the number of additive outliers and level shifts, change locations are sampled uniformly at random from $\{2, 4, 6, \ldots, N-1\}$. This ensures that level shifts are at least of length two and that we do not unintentionally construct level shifts through consecutive additive outliers. To construct sparse changes, at each change location, the number of latent signals experiencing change is selected uniformly at random. Change magnitudes are drawn from $\text{Unif}(1, 5)$ with the sign being equally likely to be positive or

negative.

We compare ABACUS against state-of-the-art change detection techniques and popular latent variable models which are marked by $\times$ and $\circ$, respectively, in plots in this section. We use default parameters in software packages unless otherwise specified. To find additive outliers, we set the minimum segment length parameter to one where possible in competing change detection implementations. The detected changes are categorized into additive outliers and level shifts using Algorithm 1 without Condition 2, except for TSO mentioned below which automatically outputs different types of changes. For all MCMC procedures, number of iterations is 3000 and burn-in is 500. Each simulation is run 100 times, and we report the average performance according to the evaluation metrics in Section 3.6.1.

Amongst competing multivariate change detection methods, GFLseg [11] finds candidate mean changes by group fused Lasso followed by selection via dynamic programming. E-divisive [44] uses binary segmentation to iteratively locate each single change point through measuring between-segment distance by the energy statistic. We specify its moment index parameter $\alpha = 2$ to find level shifts, and $min.size = 2$ the smallest segment length allowed, which implies E-divisive is unable to find additive outliers. BCP [20] is a Bayesian method which models the presence of mean change at each location through an indicator variable and uses MCMC sampling to infer the posterior probability of change. BCP outputs a set of change points corresponding to each posterior sample, hence for evaluation we compute the average metric across all these sets. We also combine BPCA [9] and BCP to obtain a two-step Bayesian approach to first compress and then detect.

Inspect [62] transforms observations into a univariate series through cumulative sum transformation before applying wild binary segmentation. We also test three

univariate methods by first applying PCA to the observations. PELT [38] is a popular parametric approach that uses dynamic programming to efficiently find the segmentation that minimizes the negative log-likelihood plus a penalty. We refer to the non-parametric version as np-PELT, which uses the empirical distribution instead [30]. A third method, TSO, jointly estimates ARIMA model parameters and change effects due to additive outliers and level shifts [15].

To fit the latent variable model, we tested against well-established methods including Independent Component Analysis (ICA), Factor Analysis (FA) and Bayesian Principal Component Analysis (BPCA). Note that ICA and FA do not impose sparsity assumptions, whereas BPCA imposes sparsity on the columns of M. For ICA, we use the FastICA implementation which measures non-Gaussianity using negentropy [34]. For FA, we use the factanal function in R [56] which automatically checks for identifiability given K and does not fit a model if K is too large to fit a unique model.

3.6.1 Evaluation Criteria

We evaluate the detection of additive outliers and level shifts separately since some competing methods [44] detect one but not the other. We report precision and recall, and treat an estimate as accurate if it is within w of a true change location. We set $w = 1$ for the small sample experiment in Section 3.6.2, and $w = 3$ for the larger sample experiments in Section 3.6.3, 3.6.4 and 3.6.5.

We evaluate the quality of model recovery through components M and S, and noise variance parameter ψ. Given true mixing matrix M and estimate $\widehat{M}$, we center and scale each row of the matrices and measure their dissimilarity using the

squared trace metric in [23],

$$\epsilon_M = \frac{1}{P^2} Tr\left(MM^T - \widehat{M}\widehat{M}^T\right).$$

The metric ϵ_M is invariant to orthogonal rotation and allows cases where either MM^T or $\widehat{M}\widehat{M}^T$ is singular. Next, given true source signals S and estimate $\widehat{S}$, we measure their dissimilarity using a variation of averaged squared Euclidean distance

$$\epsilon_S = \frac{1}{r}\sum_{i=1}^{r}(1 - |\rho_i|)$$

where ρ_i is the Pearson correlation coefficient between $S_{i\cdot}$ and some $\widehat{S}_{j\cdot}$, and each pair is found greedily by descending magnitude of correlation. This measure is invariant to sign and label switching. Finally, given true noise variance ψ and estimate $\widehat{\psi}$, the difference is measured by their scaled squared norm

$$\epsilon_E = \frac{1}{P}\|\psi - \widehat{\psi}\|_2^2.$$

3.6.2 Simulation 1: Variations in P

We test the case of small sample size $N = 100$ and varying $P \in \{10, 30, 60, 90, 110\}$. Each sample has two additive outliers and two level shifts, and K is set to 5.

As seen from Figure 3.3, competing methods have high precision but low recall on additive outliers. As P increases, ABACUS can locate most of the additive outliers, and is one of the best-performing methods for level shifts. Both precision and recall on level shifts decrease as P increases for BCP, possibly because parameters such as the prior on change probabilities need to be adjusted. BPCA + BCP

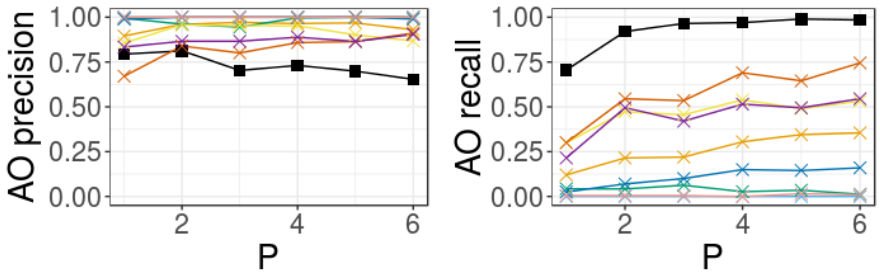
has more consistent performance, indicating the advantage of detecting changes on latent signals. In terms of model recovery, our method also gives the lowest errors for M, S and ψ, see Figure 3.4.

3.6.3 Simulation 2: Variations in N

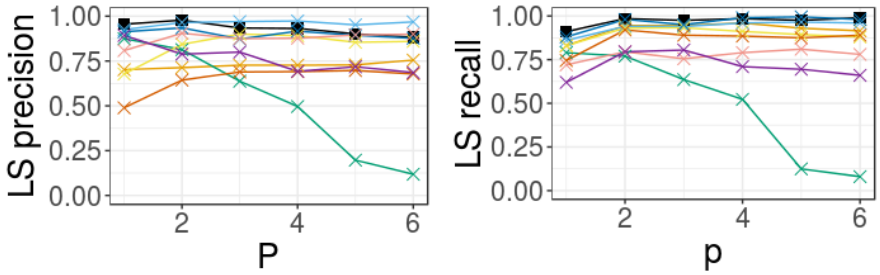
We fix $P = 10$ and vary $N \in \{600, 800, 1000, 1200, 1400, 1600\}$. Each sample has $\frac{N}{100}$ additive outliers and $\frac{N}{100}$ level shifts, and K is set to 5. Performance of all methods is consistent across N, as shown in Figures 3.5 and 3.6. BCP shows deteriorating performance in detecting level shifts just as it did in Section 3.6.2, again possibly because model parameters need to be adjusted according to the sample size. Overall, ABACUS offers the best balance of precision and recall on additive outliers while all other competing change detection methods tend to miss them. ABACUS has the highest recall for level shifts, and almost always has the lowest errors for model recovery.

3.6.4 Simulation 3: Variations in K

We fix $P = 10$ and $N = 1000$. Each sample has ten additive outliers and ten level shifts. We vary the user-specified estimate of the latent space dimensionality K between 2 and 9. The true dimensionality r is 3. The horizontal lines in Figures 3.7 and 3.8 correspond to results of methods which do not have the parameter K. According to Figure 3.7, the change detection results of ABACUS are consistent across K. From Figure 3.8, ABACUS has much more consistent error ϵ_S in S compared to competing latent variable models, whose ϵ_S increases sharply at $K \geqslant r$.

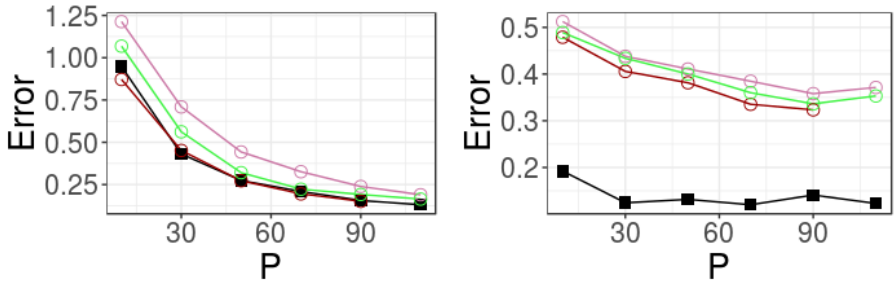


(a) Additive outliers



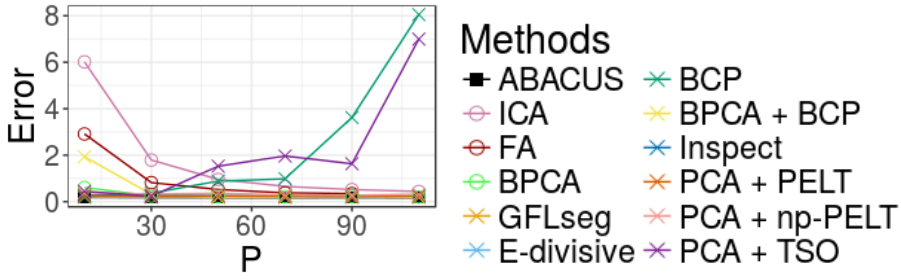
(b) Level shifts

Figure 3.3: Average errors in change detection as data dimensionality P is varied; $N = 100$ and $K = 5$ are fixed.



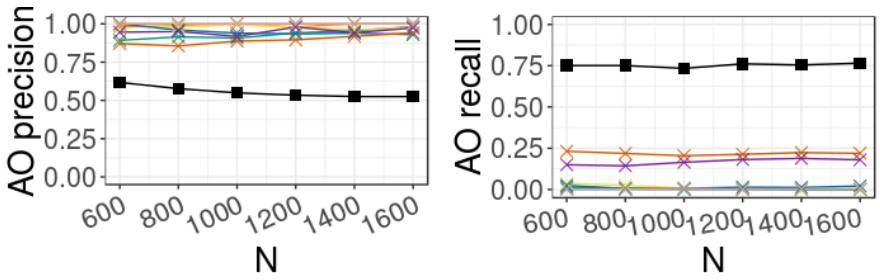
(a) Error ϵ_M for M

(b) Error ϵ_S for S

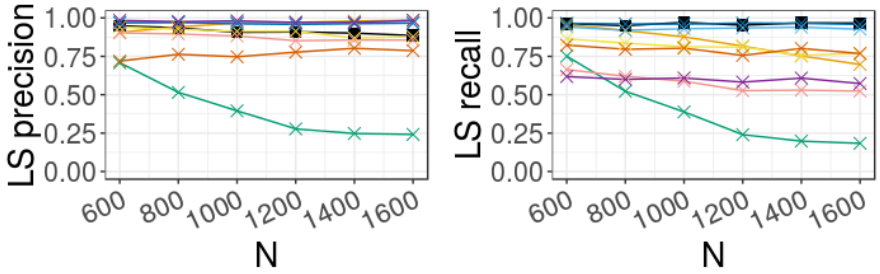


(c) Error ϵ_E for ψ

Figure 3.4: Average errors in model recovery as data dimensionality P is varied; $N = 100$ and $K = 5$ are fixed. FA does not support computations for $P = 110$ due to non-identifiability.

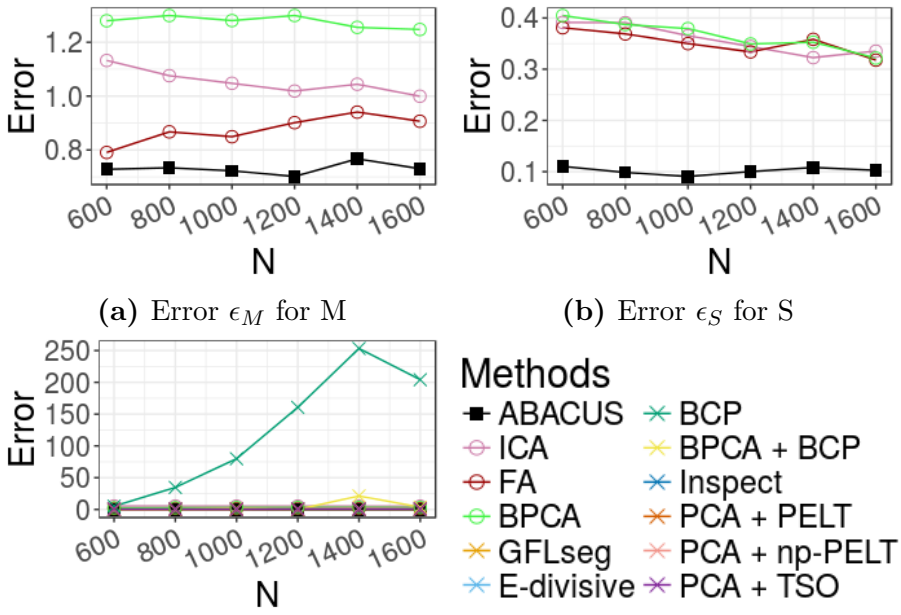


(a) Additive outliers



(b) Level shifts

Figure 3.5: Average errors in change detection as sample size N is varied; $P = 10$ and $K = 5$ are fixed.

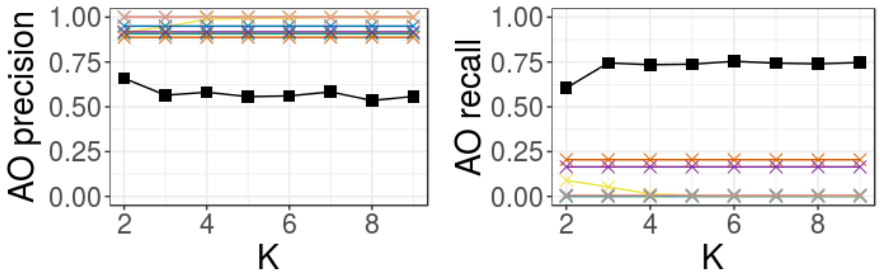


(a) Error ϵ_M for M

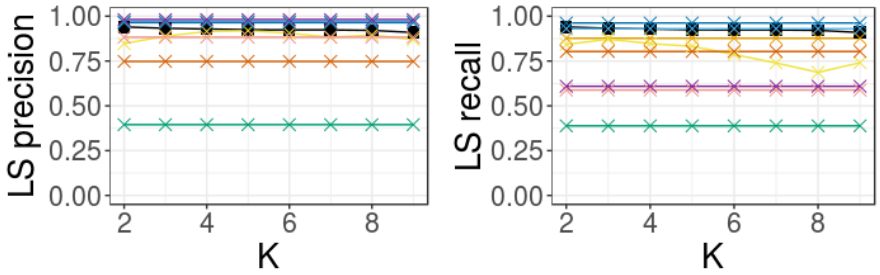
(b) Error ϵ_S for S

(c) Error ϵ_E for ψ

Figure 3.6: Average errors in model recovery as sample size N is varied; $P = 10$ and $K = 5$ are fixed.

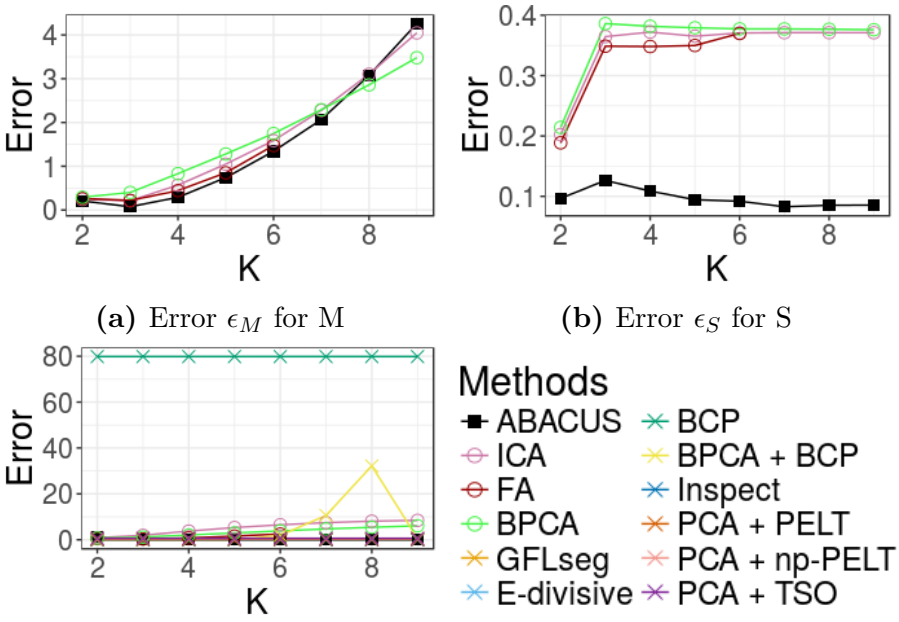


(a) Additive outliers



(b) Level shifts

Figure 3.7: Average errors in change detection as estimated latent space dimensionality K is varied; fixed $N = 1000$ and $P = 10$.

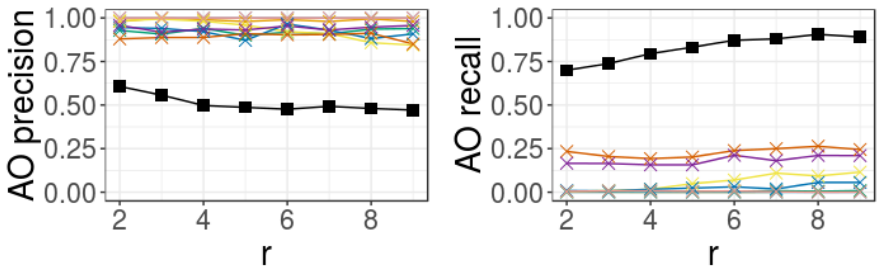


(a) Error ϵ_M for M

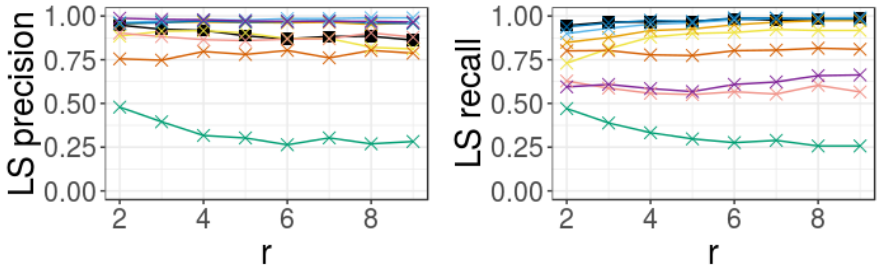
(b) Error ϵ_S for S

(c) Error ϵ_E for ψ

Figure 3.8: Average errors in model recovery as latent space dimensionality parameter K is varied; $N = 1000$ and $P = 10$ are fixed. FA does not support computations for $K \geqslant 7$ due to non-identifiability.

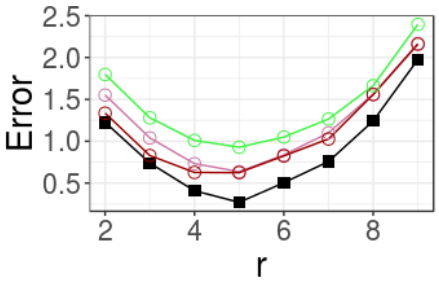


(a) Additive outliers

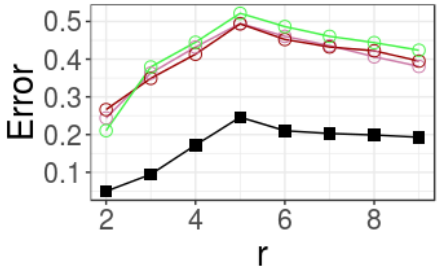


(b) Level shifts

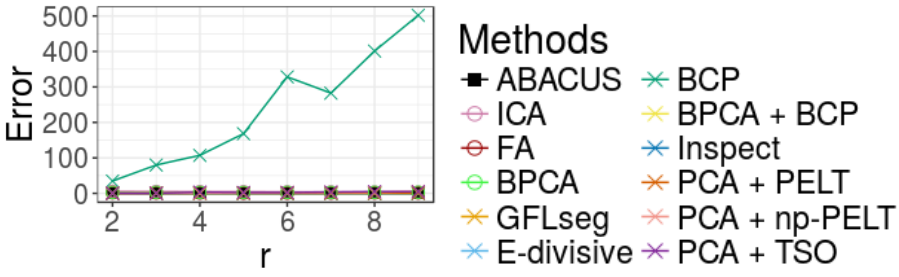
Figure 3.9: Average errors in change detection as latent dimensionality r is varied; $N = 100$, $P = 10$ and $K = 5$ are fixed.



(a) Error ϵ_M for M



(b) Error ϵ_S for S



(c) Error ϵ_E for ψ

Figure 3.10: Average errors in model recovery as latent space dimensionality r is varied; $N = 100$, $P = 10$ and $K = 5$ are fixed.

3.6.5 Simulation 4: Variations in r

We fix $P = 10$, $N = 1000$ and $K = 5$. Each sample has ten additive outliers and ten level shifts. We vary the latent space dimensionality r between 2 and 9. According to Figure 3.9, the change detection results of all methods are quite consistent across r. In Figure 3.10, ABACUS has the lowest errors for model recovery. For all methods, the error ϵ_M is lowest at $r = K$, since otherwise the dimensionality of the estimate $\widehat{M}$ is either overspecified or underspecified. As r increases from 2 and 5, the error ϵ_S increases and peaks at $r = K$ since a higher number of latent signal needs to be recovered. At $r = K$, the estimated signals $\widehat{S}$ needs to match the true signals $\widehat{S}$ in terms of correlation for each signal. As $r > K$, the average taken to calculate ϵ_S is only over the top K matching signals, hence ϵ_S decreases slightly as there are more options in the true signals for matching.

3.7 Application to Real Data

In both data applications below we set $K = 5$ and also study the robustness of ABACUS to different K values.

3.7.1 aCGH Data

Array-based comparative genomic hybridization (aCGH) is a technique for studying copy number alterations in event of diseases. We obtain the dataset from the R package ecp [35], which has already removed sequences with more than 7% missing values, and leaves 43 samples of different individuals with bladder tumor.

Each sample has 2215 probes measuring the log2 ratio between the number of transcribed DNA copies from tumorous cells and from a healthy reference [28]. A negative ratio indicates deletion, a positive ratio indicates amplification, and zero indicates an unaltered segment. We expect shared change locations for individuals with the same medical condition.

To reduce computations and ease visualization, we thin the samples by taking every 20^{th} value. We arrive at a dataset with $P = 43$ and $N = 111$. ABACUS takes approximately one minute to run on a standard desktop computer, and finds three additive outliers and seven level shifts. An additive outlier here indicates a shorter segment of genetic aberration compared to a level shift. A plot of all 43 samples with the estimated change points overlaid is in the Supplementary Materials.

At least 99% of the variance of our estimated latent signals can be explained by four principal components, while those found by ICA and FA require all five. As observed in Figure 3.11, the third latent source signal recovered exhibits no evident changes. We map the four other signals to unique sets of genetic aberrations in different stages of bladder tumor in Table 3.1. For instance, patients with genetic aberrations on chromosome arms 2q, 3q and 20p/q simultaneously tend to be in tumor stage pT_1, hence the changes detected can be indicative of diseases for new patients. The mapping is established based on a bladder tumor research article [10] which lists the frequent genomic alterations by chromosome arm in tumor stages pT_a, pT_1 and pT_{2-4}. Each stage is determined pathologically depending on the size and location of the tumor.

ABACUS performs consistently across different K. Figure 3.12 shows that for $K \in \{10, 15, 20, 25, 30\}$, the change points and latent source signals recovered are very similar to those found with $K = 5$.

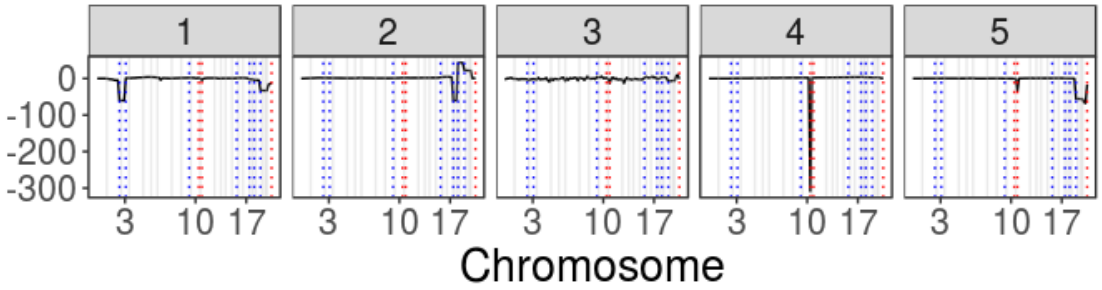
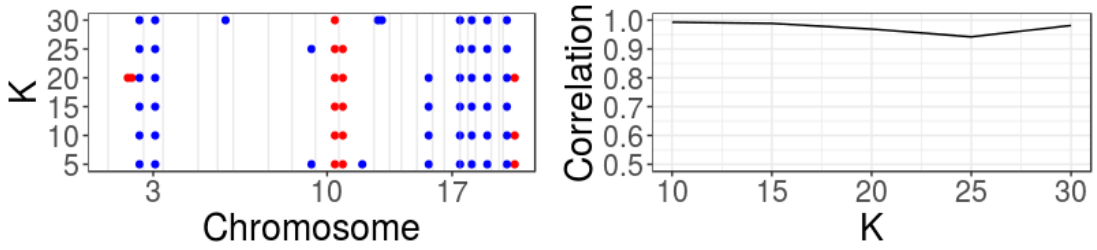


Figure 3.11: aCGH: Latent source signals (1-5) recovered (black), and additive outliers (red) and level shifts (blue) detected. Gray lines indicate the boundaries between chromosome pairs.

S	Chromosome arm with changes	Tumor stage
1	2q, 3q, 20p/q	pT_1
2	17p/q, 18p/q, 19p/q, 20p/q	pT_1
4	10q	pT_a, pT_1, pT_{2-4}
5	11p, 20p/q	pT_{2-4}

Table 3.1: aCGH: Genetic aberrations corresponding to changes detected on latent source signals. To read the table, 20p is the short arm of chromosome 20, and 20q is the long arm. Tumor stages range from a, 1 to 4 in order of severity.



(a) Additive outliers (red) and level shifts (blue)

(b) Average correlation to latent signals at $K = 5$

Figure 3.12: aCGH: Changes and latent source signals recovered by ABACUS are similar regardless of the specification of K.

3.7.2 Electric Power Consumption Data

This dataset contains per-minute measurements of electric power consumption in one household and is available on the UCI Machine Learning Repository [16]. The data has seven dimensions including global active power (GAP), global reactive power (GRP), voltage (V), global intensity (GI), and three sub-meterings corresponding to the kitchen (S1), laundry room (S2) and heating system (S3). We expect shared change points since the seven dimensions are related arithmetically, and some electrical appliances tend to be used simultaneously. For instance, $\frac{1000}{60}\text{GAP}-\text{S1}-\text{S2}-\text{S3}$ is the power consumed by appliances outside of the kitchen, laundry room and heating system. We analyze a full day's worth of data, that is, the observation matrix has $P = 7$ and $N = 1440$. ABACUS takes approximately fifteen minutes to run on a standard desktop computer.

The Supplementary Materials contain a plot of the standardized data with estimated changes overlaid. Although the data does not follow our model assumptions exactly since the amount of fluctuations or noise is more significant in the first half of the day, and there are minor trend changes in the second half of the day, ABACUS is robust and with post-processing it finds one additive outlier and sixteen level shifts. We post-process by dynamic programming to prune the initially estimated level shifts. This is similar to GFLseg [11], except that we apply the procedure on the latent source signals which are less contaminated by noise.

The change points are indicative of the household's pattern of electricity usage, which concentrates in the first half of the day as illustrated in Figure 3.13. The fourth latent signal reflects the usage fluctuations and trends which differ across the two halves of the day as measured by GAP and GI. ABACUS performs consistently across different specifications of K. Figure 3.14 shows that for $K \in \{2, 3, 4, 6, 7\}$,

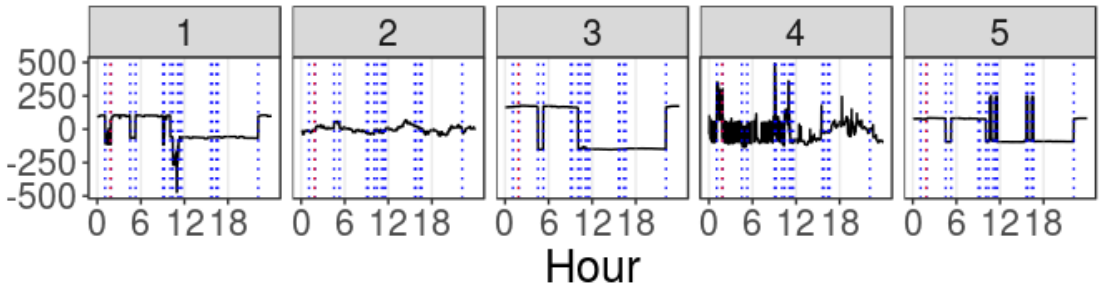
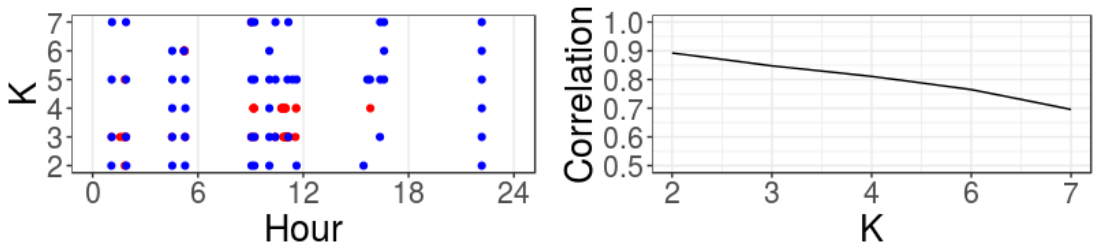


Figure 3.13: Power: Latent source signals (1-5) recovered (black), and additive outliers (red) and level shifts (blue) detected.



(a) Additive outliers (red) and level shifts (blue)

(b) Average correlation to latent signals at $K = 5$

Figure 3.14: Power: Changes and latent source signals recovered by ABACUS are similar regardless of the specification of K.

the estimated change points and latent source signals recovered are similar to those found at $K = 5$.

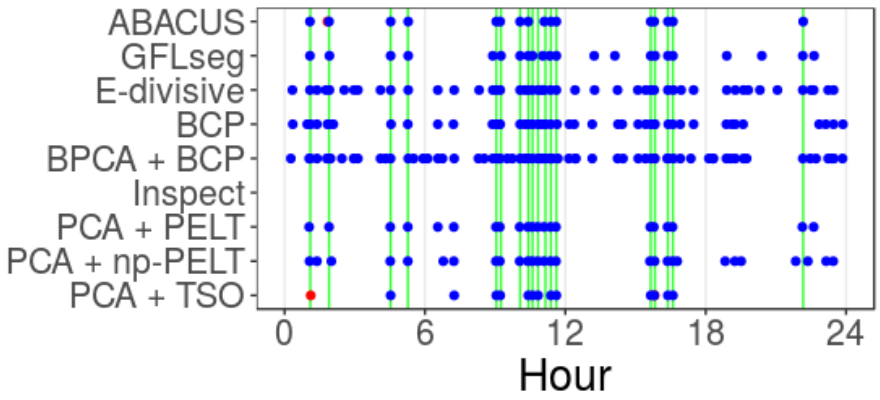
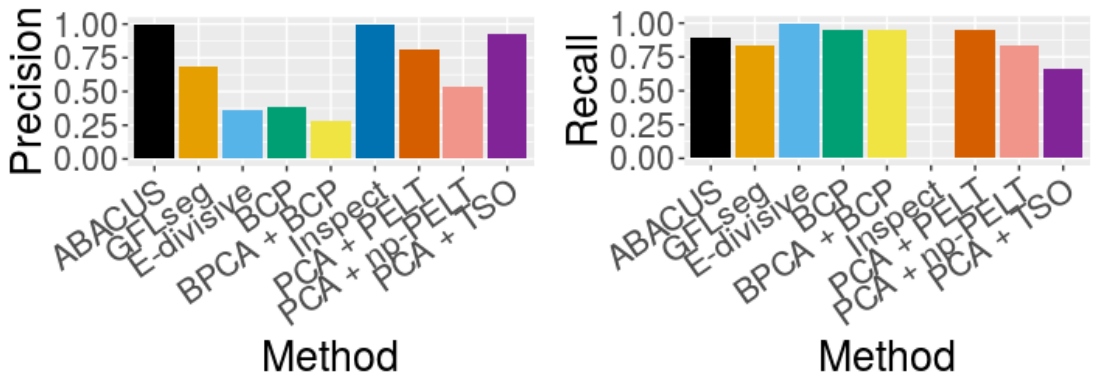


Figure 3.15: Power: Additive outliers (red) and level shifts (blue) estimated vs ground truth level shifts (green).

Since the sub-meterings S1, S2 and S3 demonstrate distinct level shifts when the respective appliances are utilized, we extract ground truths for level shifts by



(a) Precision **(b)** Recall

Figure 3.16: Power: Performance in estimating level shifts.

finding positions where these signals deviate from their base levels. Compared to other change detection methods in Figure 3.15 and 3.16, ABACUS has the best overall performance with precision = 1 and recall = 0.889.

3.8 Conclusion

In this work, we propose ABACUS, an automatic change detection procedure which makes use of Bayesian latent variable modeling. Due to the separation of additive outlier and level shift effects in the model, ABACUS naturally identifies these two types of changes separately, unlike many competing approaches.

In simulations, ABACUS shows competitive or superior performance in both change detection and model recovery. In two real data applications, ABACUS found relevant change points and source signals. It is robust to over-specification of K, an important property since the true value is rarely known to the user in practice.

Acknowledgement

NSF grant DMS-1455172, a Xerox PARC Faculty Research Award, and Cornell University Atkinson's Center for a Sustainable Future AVF-2017 is gratefully acknowledged.

3.9 Supplementary Materials

3.9.1 Posterior Distributions

Let $D^{(1)}$ be the matrix representation of $\triangle$ such that $S^{(1)}\left[D^{(1)}\right]^T = V^{(1)}$. Also $D^{(0)} = I$ such that $S^{(0)} = V^{(0)}$. We define the following expressions for the full conditionals of the posterior distributions:

$$F = SS^T + \text{diag}\left(\tau^{(0)}\tau^{(1)}\lambda^{(0)}\lambda^{(1)}\right)^{-1}$$

$$G^{(0)} = \sum_{i=1}^{p}\sum_{h=1}^{K}\frac{M_{ih}^2}{2\lambda_h^{(0)}\lambda_h^{(1)}\tau^{(1)}\psi_i} + \sum_{n=1}^{N}\sum_{h=1}^{K}\frac{\left[V_{hn}^{(0)}\right]^2}{2\phi_n^{(0)}\lambda_h^{(0)}\gamma_{hn}^{(0)}}$$

$$H_h^{(0)} = \sum_{i=1}^{p}\frac{M_{ih}^2}{2\tau^{(0)}\tau^{(1)}\lambda_h^{(1)}\psi_i} + \sum_{n=1}^{N}\frac{\left[V_{hn}^{(0)}\right]^2}{2\phi_n^{(0)}\gamma_{hn}^{(0)}\tau^{(0)}}$$

For $1 \leqslant i \leqslant P$ and $1 \leqslant h \leqslant K$ and $1 \leqslant n \leqslant N$ and $d \in \{0, 1\}$, we derive the full conditionals for the posterior distributions below. We leave out $\tau^{(1)}$ and $\lambda_h^{(1)}$ since their full conditional distributions are similar in form to those of $\tau^{(0)}$ and $\lambda_h^{(0)}$ respectively.

$$
\begin{aligned}
M_{i\cdot}|\cdot &\sim \mathrm{N}\left(F^{-1}SY_{i\cdot}, \psi_i F^{-1}\right) \\
\psi_i|\cdot &\sim \Gamma^{-1}\left(1+\frac{N}{2},\ 1+\frac{1}{2}(Y_{i\cdot}-M_{i\cdot}S)^T(Y_{i\cdot}-M_{i\cdot}S)\right) \\
\tau^{(0)}|\cdot &\sim \Gamma^{-1}\left(\frac{1+K(p+N)}{2},\ \frac{1}{\xi^{(0)}}+G^{(0)}\right) \\
\xi^{(d)}|\cdot &\sim \Gamma^{-1}\left(1, 1+\frac{1}{\tau^{(d)}}\right) \\
\lambda_h^{(0)}|\cdot &\sim \Gamma^{-1}\left(\frac{1+p+N}{2},\ \frac{1}{\eta_h^{(0)}}+H_h^{(0)}\right) \\
\eta_h^{(d)}|\cdot &\sim \Gamma^{-1}\left(1, 1+\frac{1}{\lambda_h^{(d)}}\right) \\
\phi_n^{(d)}|\cdot &\sim \Gamma^{-1}\left(\frac{1+K}{2},\ \frac{1}{\omega_n^{(d)}}+\sum_{h=1}^{K}\frac{\left[V_{hn}^{(d)}\right]^2}{2\lambda_h^{(d)}\gamma_{hn}^{(d)}\tau^{(d)}}\right) \\
\omega_n^{(d)}|\cdot &\sim \Gamma^{-1}\left(1, 1+\frac{1}{\phi_n^{(d)}}\right) \\
\gamma_{hn}^{(d)}|\cdot &\sim \Gamma^{-1}\left(1,\ \frac{1}{\zeta_{hn}^{(d)}}+\frac{\left[V_{hn}^{(d)}\right]^2}{2\lambda_h^{(d)}\phi_n^{(d)}\tau^{(d)}}\right) \\
\zeta_{hn}^{(d)}|\cdot &\sim \Gamma^{-1}\left(1, 1+\frac{1}{\gamma_{hn}^{(d)}}\right)
\end{aligned}
$$

For $V_{\cdot n}^{(d)}$, the full conditional distribution is

$$
\mathrm{N}\left(\left[B^{(n)}\right]^{-1} M^T\Psi^{-1}C^{(n)}\left[D^{(d)}\right]_{\cdot n}^{-1}, \left[B^{(n)}\right]^{-1}\right)
$$

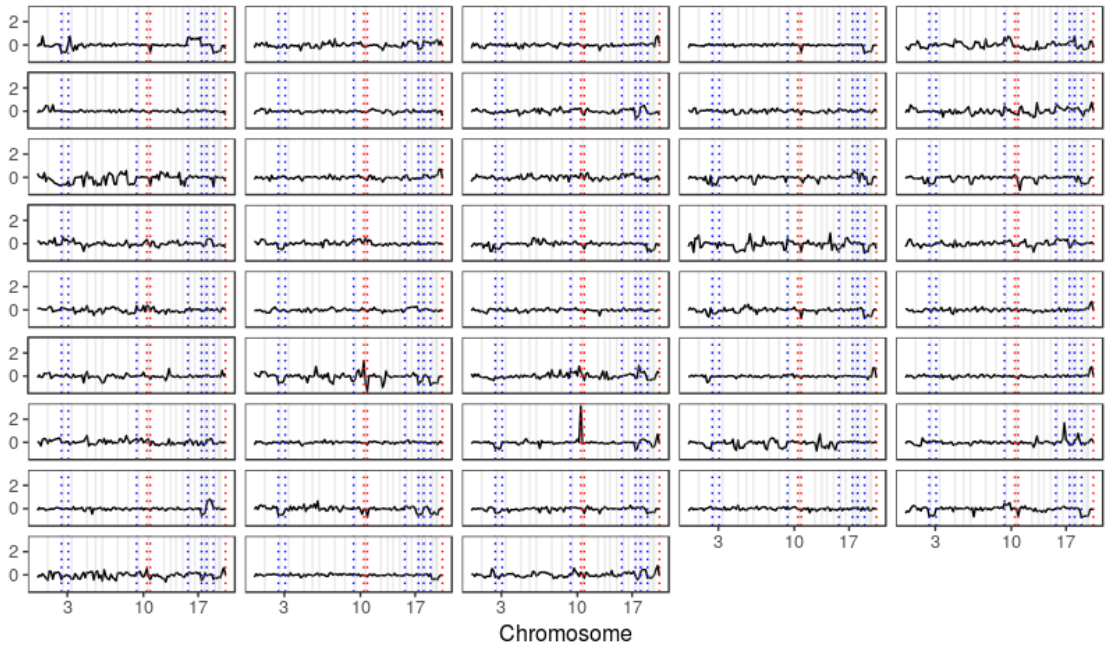


Figure 3.17: aCGH: Additive outliers (red) and level shifts (blue) detected by ABACUS. Gray lines indicate the boundaries between chromosome pairs. Additive outliers correspond to shorter segments of genetic aberrations and level shifts correspond to longer segments.

where

$$B^{(n)} = M^T \Psi^{-1} M \left(\left[D^{(d)} \right]_{n\cdot}^{-T} \left[D^{(d)} \right]_{\cdot n}^{-1} \right) + \operatorname{diag}\left(\phi_n^{(d)} \lambda^{(d)} \gamma_{\cdot n}^{(d)} \tau^{(d)} \right)^{-1}$$

$$C^{(n)} = Y - MS + MV_{\cdot n}^{(d)} \left[D^{(d)} \right]_{n\cdot}^{-T}$$

3.9.2 Additional Plots for aCGH Data

Figure 3.17 plots all 43 samples with the estimated change points overlaid.

For comparison, we include again the recovered latent source signals with the estimated change points overlaid in Figure 3.18.

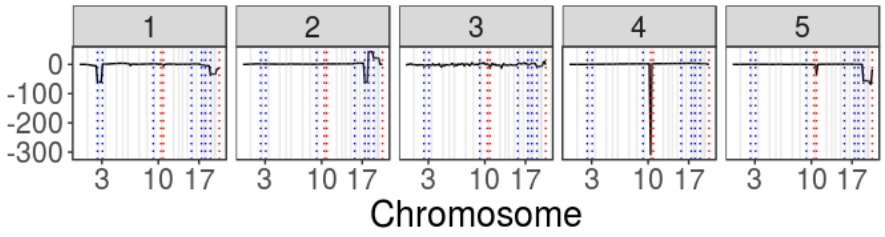


Figure 3.18: aCGH: Latent source signals (1-5) recovered (black), and additive outliers (red) and level shifts (blue) detected. Gray lines indicate the boundaries between chromosome pairs.

3.9.3 Additional Plots for Electric Power Consumption Data

Figure 3.19 plots the standardized data from the electric power consumption dataset with estimated changes overlaid. ABACUS is run on the standardized dataset.

For comparison, we include again the recovered latent source signals with the estimated change points overlaid in Figure 3.20.

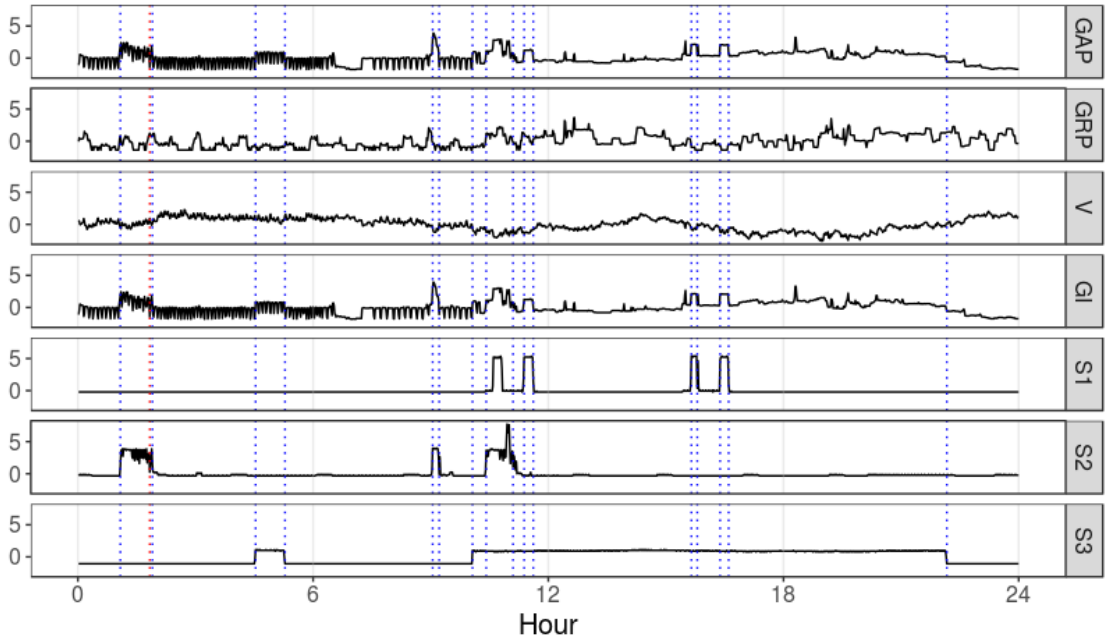


Figure 3.19: Power: Additive outliers (red) and level shifts (blue) detected by ABACUS. The level shifts detected correspond well with appliance usages in sub-meterings S1, S2 and S3.

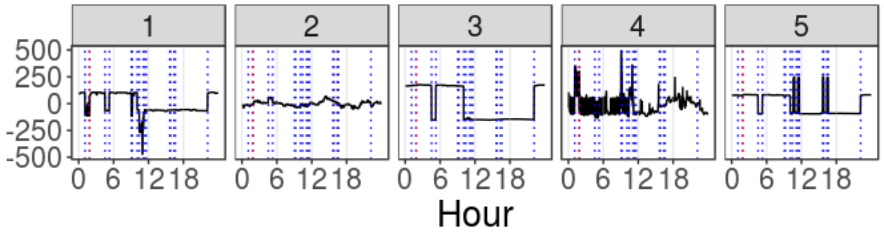


Figure 3.20: Power: Latent source signals (1-5) recovered (black), and additive outliers (red) and level shifts (blue) detected.

CHAPTER 4

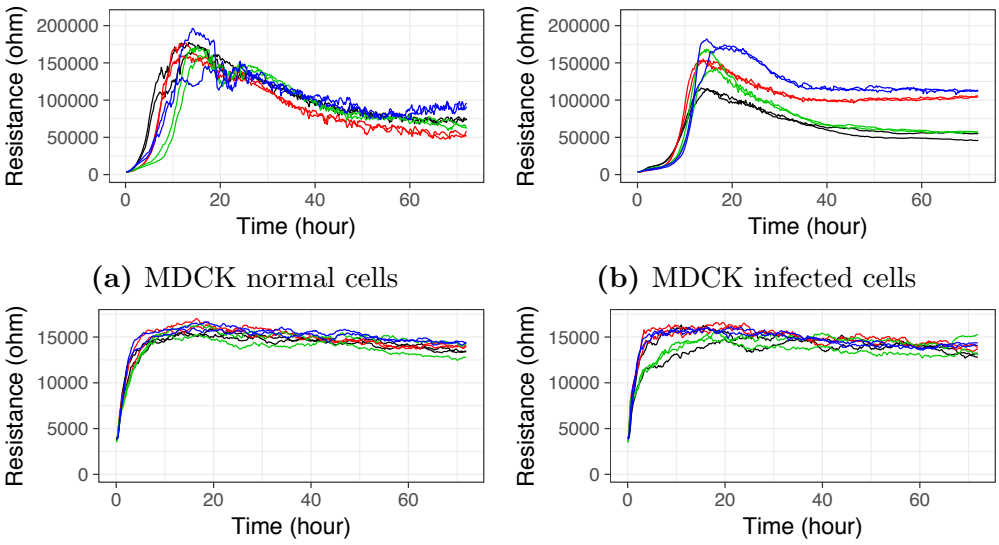
CHANGE DETECTION OF CELL CONFLUENCE WITH LONG-MEMORY DEPENDENCE IN ECIS DATA

4.1 Introduction

We propose a model for time-series data that is characterized by two consecutive regimes, which correspond to a highly nonstationary and nonlinear growth period and a stable, equilibrium period. Often, researchers are interested in estimating the features of each regime, as well as the timing of the transition or change point between the two.

We are motivated by the analysis of data measured using electric cell-substrate impedance sensing (ECIS). ECIS is a relatively new non-invasive method used to study cell attachment, growth, morphology, function and motility [36]. Cells are grown in medium within culture dish wells on top of small gold-film electrodes. Alternating current is applied between the electrodes at different points in time, and electrical impedance is measured. The electrical impedance measurements can be interpreted as an indirect measure of cell growth; increasing impedance indicates the presence of more cells covering the electrode. The ECIS technique has been applied in numerous cell biology studies, such as cancer biology [32] and cytotoxicity [52].

The ECIS data that we analyze in this paper consists of an initial growth regime, followed by an equilibrium regime. During the growth regime, impedance measurements increase as cells multiply and fill the well. Eventually, the cells fill the well completely and confluence occurs. The equilibrium regime is characterized



(a) MDCK normal cells

(b) MDCK infected cells

(c) BSC normal cells

(d) BSC infected cells

Figure 4.1: Example of resistance measurements at 500 hertz for cell samples cultivated in gel from Experiment 1 (black), 2 (red), 3 (green), 4 (blue). Growth patterns differ across experiments, while cell behaviors after confluence are more similar.

by a plateau in the measurements due to physical constraints preventing further growth imposed by the well walls. Fluctuations of measurements after confluence are caused by cell micromotion, and are believed to be less sensitive to initial conditions and other sources of possible batch effects. The equilibrium regime persists until the cells eventually exhaust their resources and begin to die. Figure 4.1 plots example samples from the MDCK and BSC cell lines that we analyze in this work. Resistance measurements for the MDCK cells show distinct peaks while regime transitions for BSC cells are less obvious visually, which also translate to higher difficulty for change detection. The growth patterns differ across batches, but cell behaviors after confluence are more similar.

It has been hypothesized that ECIS data can be used for cell classification. For instance, features constructed from measurements after confluence have been used for classification of cell lines [24], and to differentiate between cancerous and

noncancerous cells [41]. After confluence, ECIS measurements have been found to demonstrate long memory dependence, i.e. correlations that decay very slowly over time [41, 61]. Long-memory dependence, which has also been observed in wind speed and inflation data [29, 17] is often modeled as a Gaussian fractionally integrated (FI) or long-memory process. Importantly, the FI process involves a very small number of parameters - an overall mean μ, variance σ^2, and a scalar long-memory parameter d that governs how quickly temporal correlations decay. Ideally, if these parameters could be estimated well, they might be useful for classifying cells of different types. In our case, we classify infected versus normal cells as plotted in Figure 4.1.

The long-memory parameter d is notoriously difficult to estimate in finite samples. Furthermore, the onset of confluence, which determines the amount of data available to estimate d, is typically not precisely known in practice. Standard practice is use a fixed time point, e.g. 20 hours, as a conservative estimate of the start of the confluence regime [61]. This under-utilizes the data, potentially resulting in poorer estimates of the parameters of interest. Furthermore, such a conservative estimate could incorrectly characterize the growth phase.

In this work, we focus on estimating the change point from growth to confluence, which consequently allows us to extract the growth trend and statistically characterize the confluent state. By more accurately pinpointing the start of the confluence regime, features can be extracted more accurately.

Identifying the change point from growth to confluence requires unsupervised methods, because labeled data indicating the true change point are not available. Among the unsupervised methods, we are not aware of existing work in detecting the transition from a nonstationary model to a long-memory model. [18] detects the

parameter change within a long-memory model. [13] detects outliers and changes in ARMA (autoregressive moving-average) models, which are for short-memory processes. Statistical time series change detection methods typically assume independent and identically distributed observations within a segment [44]. Some methods restrict the distribution of the data [38], and some methods are only formulated to find specific types of change points such as outliers and level shifts [65]. Change detection methods in the biomedical field are often domain-specific. In the paper by Olshen et al., the authors propose circular binary segmentation to detect DNA sequence copy number alterations [51], and in the paper by Nika et al., the proposed method identifies important changes in serial magnetic resonance images and rejects unimportant changes by comparing image representations against a learned dictionary [50]. The data distributions assumed by these methods are not suitable for the highly nonlinear and nonstationary growth processes we observe in the ECIS data.

In Section 4.3, we propose a novel method named Growth-to-Confluence Detector (G2CD) for time series data that exhibit a strong growth pattern that transitions into a confluent state. We consider an exact search method which assumes complete separation between the growth and equilibrium regimes, and an approximate search method which has better computational feasibility. The methods have a likelihood-based formulation and model the long-range dependence structure in the second regime with an FI process. The FI process and its generalizations have been popularly applied in econometrics and climate science [4, 17, 29, 53]. The G2CD method can be naturally extended to multivariate data when replicate samples are available. In Section 4.5, we demonstrate the performance of G2CD in simulations. Finally, in Section 4.6 we demonstrate the performance of G2CD on two real-world ECIS datasets. We further use the estimated change point and

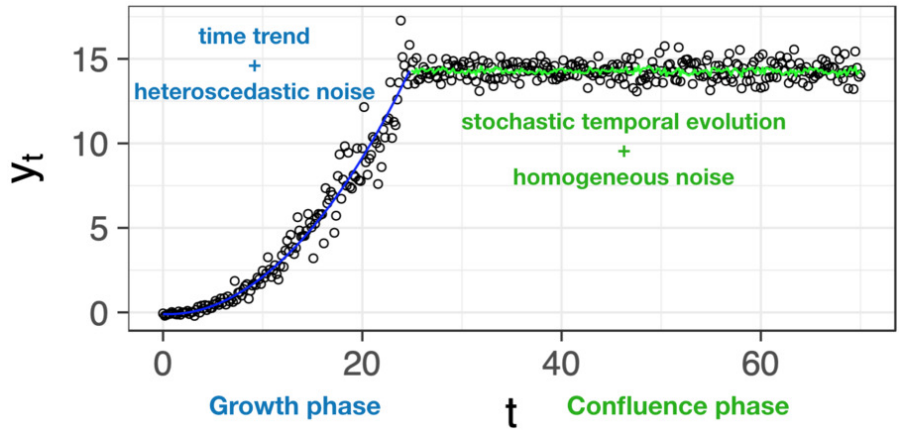


Figure 4.2: Overview of data generating process: growth phase consists of a non-stationary time trend and heteroscedastic noise, and the confluence phase consists of a stochastic temporal evolution and homogeneous noise.

long-memory parameter for a downstream task of classifying infected versus normal cells, and achieve an increase in accuracy of over 18% and 7% for the two datasets, respectively, compared to a feature-engineering method for cell classification [24].

4.2 Problem Formulation

We consider the data generating process motivated by observations from ECIS data for cell growth. Let $y_1, y_2, \ldots, y_T \in \mathbb{R}$ be a sequence of time-ordered observations at $t = 1, 2, \ldots, T$, respectively. We assume that the measurements y_t belong to two successive regimes according to Figure 4.2.

Formally, let τ denote the change point, then:

$$y_t = f(t; \beta) + \eta_t \qquad \text{for } t < \tau \tag{4.1}$$

$$y_t = g(y_\tau, \ldots, y_{t-1}; \mu, d) + \epsilon_t \qquad \text{for } t \geqslant \tau \tag{4.2}$$

where $\eta_t \sim \mathrm{N}(0, \sigma_t^2)$ and $\epsilon_t \sim \mathrm{N}(0, \nu^2)$. The noise terms are independent within

and across the two regimes. They encompass both measurement errors as well as random fluctuations due to continuous cell growth, motility, death and other functions. The log-likelihood of y_t is then:

$$\ell\left(y_t\right) = \ell^{(1)}\left(y_t\right) \mathbb{1}_{t<\tau} + \ell^{(2)}\left(y_t\right) \mathbb{1}_{t\geqslant\tau} \tag{4.3}$$

where $\ell^{(1)}(y_t)$ and $\ell^{(2)}(y_t)$ is the log-likelihood of y_t under the first and second regime, respectively. The dependencies on model parameters and historical values of y are left out of the expression for simplicity.

During the first (growth) regime, the measurement at time t will be centered around a growth curve $f\left(t;\beta\right)$ which is a function of time t and fixed but unknown parameters β. The noise terms $\eta_t \sim \mathrm{N}\left(0,\sigma_t^2\right)$ are possibly heteroscedastic, to reflect different degrees of uncertainty in the measurements when the cell culture undergoes different rates of growth. During the second (equilibrium) regime, the measurement at time t will be centered about a function $g\left(y_\tau,\ldots,y_{t-1};\mu,d\right)$ of previous measurements $y_\tau,\ldots,y_{t-1}$ and fixed but unknown parameters μ, and d. The noise terms $\epsilon_t \sim \mathrm{N}\left(0,\nu^2\right)$ are homoscedastic with fixed but unknown variance ν^2, since the cell culture is in equilibrium and not undergoing drastic changes.

4.3 Model

We describe our modeling choices of the two regimes in Section 4.2.

4.3.1 The first regime: Growth

Resistance measurements in the first regime, or the growth phase, are characterized by a trend of initial steep increase followed by a plateau, as well as heteroscedasticity with higher variance at the stage of rapid cell growth. The growth curve is denoted as $f(t;\beta)$. Depending on the growth trend, any appropriate parametric, semi-parametric or nonparametric model can be used to fit the first regime. The exact formulation of the growth curve can depend on the application domain and the choice of the user. For the ECIS application that we focus on in this work, we assume a smooth growth curve. This is in line with visual inspection of real ECIS data in Figure 4.1, and that cell growth, motility, death and other functions are continuous processes. We utilize penalized splines for their flexibility to capture the ECIS growth trend, since it is highly nonstationary.

We similarly use penalized splines in modeling the noise variance. The heteroscedastic noise term is denoted $\eta_t \sim \mathrm{N}(0, \sigma_t^2)$ and we define the variance σ_t^2 as a function of time and fixed but unknown parameters θ. In particular, $\log(\sigma_t^2) = h(t;\theta)$.

A spline of degree D with Q distinct interior knots $\{u_1, \ldots, u_Q\}$ is a function formed by connecting polynomial segments of degree D with the constraints [58] that:

- the function is continuous,
- the function has $D-1$ continuous derivatives over the entire range of the data,
- the D^{th} derivative is constant between knots.

We construct the spline from B-splines $B_{i,k}(u)$ which can be defined recursively by the Cox-de-Boor formula:

$$B_{i,0} = \begin{cases} 1 & \text{if } u_i \leqslant u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$B_{i,k} = \frac{u - u_i}{u_{i+k} - u_i} B_{i,k-1}(u) + \frac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} B_{i+1,k-1}(u)$$

Each $B_{i,D}(u)$ is non-zero on $[u_i, u_{i+D+1})$, and there are a total of $Q + D + 1$ basis functions.

We denote the matrix of B-spline basis functions $B_{i,D}(u)$ for the growth regime as $\mathbf{X} \in \mathbb{R}^{(\tau-1)\times(Q+D+1)}$. Then the spline is $s(t) = \mathbf{x}_t'\beta$ for coefficients $\beta \in \mathbb{R}^{Q+D+1}$, where $\mathbf{x}_t$ is row t of $\mathbf{X}$. We impose the smoothness penalty $\lambda \int \hat{s}''(u)^2 du$ on the spline estimate. This is equivalent to assuming a mean-zero prior distribution for the spline coefficients with covariance matrix $\lambda^{-1}\mathbf{M}^{-1}$, where $\mathbf{M}_{ij} = \int B''_{i,D}(u) B''_{j,D}(u) du$ and $\lambda > 0$ is a scalar that corresponds to how aggressively smoothness is encouraged.

Let $\mathbf{X}$ denote the $(\tau - 1) \times (Q_f + D_f + 1)$ matrix of basis functions for the trend, and let $\mathbf{V}$ be the $(\tau - 1) \times (Q_h + D_h + 1)$ matrix of basis functions for the log variances of the noise terms, and let $\mathbf{x}_t$ and $\mathbf{v}_t$ refer to row t of the corresponding matrix $\mathbf{X}$ or $\mathbf{V}$. The model for the first regime takes the form

$$y_t = \mathbf{x}_t'\beta + \eta_t$$
$$\log\left(\sigma_t^2\right) = \mathbf{v}_t'\theta$$

where $\eta_t \sim N(0, \sigma_t^2)$, and priors $\beta \sim N(\mathbf{0}, \lambda_f^{-1}\mathbf{M}_f^{-1})$, and $\theta \sim N(\mathbf{0}, \lambda_h^{-1}\mathbf{M}_h^{-1})$ under

the equivalent Bayesian formulation.

4.3.2 The second regime: Equilibrium

The second regime, or the confluence phase, is fit with a fractionally integrated process to capture long-memory dependence of the observations at confluence. We assume that the noise variance ν^2 and mean μ of the FI process are constant.

We assume that

$$g\left(y_\tau, \ldots, y_{t-1}; \mu, d\right) = \mu - \sum_{i=1}^{\infty} \binom{d}{i} (-1)^i \left(y_{t-i} - \mu\right) \mathbb{1}_{t-i \geqslant \tau}, \tag{4.4}$$

where the parameter d plays the role of the long memory parameter in a Gaussian fractionally integrated (FI) model [59], which is used for time series data that display long memory, i.e. slowly decaying correlations over time.

The FI process is stationary and invertible if $|d| < 0.5$. The process is nonstationary for $d \geqslant 0.5$. When $d \in (0, 0.5)$, the process is said to have long memory, and the autocovariance function exhibits hyperbolic decay: $\gamma_k \sim ck^{2d-1}$ as $k \to \infty$ where c is a finite nonzero constant. When $d \in (-0.5, 0)$, the process is said to have anti-persistence, long-range negative dependence or to be 'overdifferenced', and the inverse autocorrelations decay hyperbolically.

When $d \in (0.5, 1.5)$, we follow the standard treatment of taking a first difference of the series such that $z_t = y_t - y_{t-1}$ is a zero-mean stationary FI($\tilde{d}$) process where $\tilde{d} \in (-0.5, 0.5)$ [59].

4.3.3 Multivariate scenario

A general extension to the multivariate case assumes different parameters for each dimension. We impose the additional condition that a subset of the parameter set for the second regime is the same across all dimensions. Let $\boldsymbol{Y} \in \mathbb{R}^{T \times p}$. We assume that

$$y_{t,j} = f\left(t; \beta_j\right) + \eta_{t,j} \qquad \text{for } t < \tau_j \tag{4.5}$$

$$y_{t,j} = g\left(y_{\tau,j}, \ldots, y_{t-1,j}; \mu_j, d\right) + \epsilon_{t,j} \qquad \text{for } t \geqslant \tau_j \tag{4.6}$$

where $\eta_{t,j} \sim \mathrm{N}\left(0, \sigma_{t,j}^2\right)$ with $\log\left(\sigma_{t,j}^2\right) = h\left(t; \theta_j\right)$, $\epsilon_{t,j} \sim \mathrm{N}\left(0, \nu_j^2\right)$, and priors $\beta_j \sim N(\mathbf{0}, \lambda_{fj}^{-1}\mathbf{M}_f^{-1})$, and $\theta_j \sim N(\mathbf{0}, \lambda_{hj}^{-1}\mathbf{M}_h^{-1})$. This corresponds to the univariate model for each time series on column j of $\mathbf{Y}$, but with a shared long memory parameter d.

In the ECIS application, each dimension is a replicate sample, and the shared parameter reflects the shared behavior when the cells reach the confluence regime. We do not enforce shared parameters in the first regime β_j and θ_j or shared change points τ_j because varying initial conditions, such as the number of cells deposited, can be unintentionally introduced and can affect the growth rates and time of transition between the two regimes.

4.4 Estimation

We describe parameter estimation for the model outlined in Section 4.3. We describe the methods in terms of the univariate scenario before moving on to the

multivariate scenario in Section 4.4.3.

4.4.1 G2CD-exact: exact search

G2CD-exact is a version of the proposed G2CD method which performs an exact search for the transition point between the two regimes that maximizes the log-likelihood of the data.

In the univariate case, the log-likelihood of y_t under the first regime with the change point candidate c has the form:

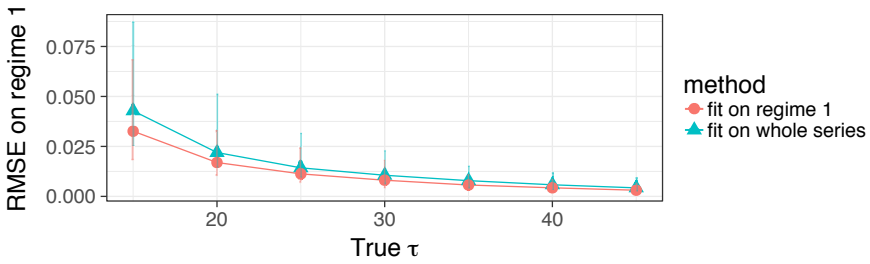
$$-\frac{1}{2}\left(\log(2\pi) + h(t;\theta) + \frac{\left(y_t - f\left(t;\beta\right)\right)\right)^2}{\exp\left(h(t;\theta)\right)}\right) \tag{4.7}$$

The conditional log-likelihood for y_t under the second regime has the form:

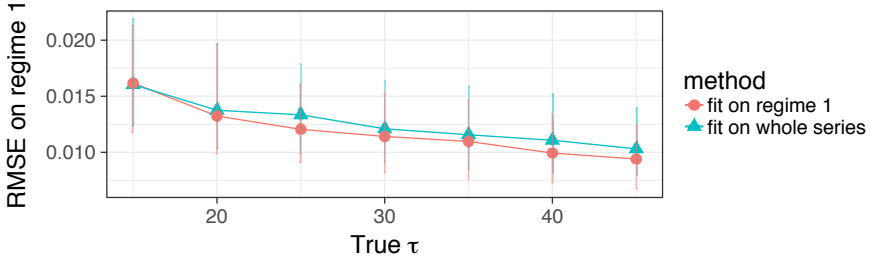
$$-\frac{1}{2}\left(\log(2\pi) + \log(\nu^2) + \frac{\left(y_t - g\left(y_c, \ldots, y_{t-1};\mu,d\right)\right)^2}{\nu^2}\right) \tag{4.8}$$

All parameters are functions of c and are estimated for each candidate change point.

Fitting the first regime by likelihood maximization will result in overfitting. Hence we enforce smoothness of the spline fits by regularization on the β and θ parameters as described in Section 4.3.1, and select the regularization hyperparameters λ_f and λ_h by leave-one-out cross-validation. We use an iterative Feasible Generalized Least Squares (FGLS) procedure [40] to estimate the spline coefficients β. A more detailed explanation of the FGLS procedure is provided in the Supplementary Materials.



(a) The first regime generated via degree-5 polynomials.



(b) The first regime generated via Gaussian process with squared exponential kernel.

Figure 4.3: RMSE on trend estimation by fitting the first regime with the entire series versus the true first regime across 2000 simulated series (100 simulated series for each of 20 d's) per τ. Medians are plotted and error bars indicate the upper and lower quartiles.

In G2CD-exact, we estimate β and θ for each candidate change point c. We find that estimating the spline coefficients on the entire time series and fixing them across c does not significantly degrade model fitting performance, and hence can be considered as a faster alternative. The spline bases are flexible to fit local trends and we show through simulations that fitting on the entire series is almost as good as fitting on the true first regime in Figure 4.3. The simulation set-up is detailed in Sections 4.5.1 and 4.5.2.

Given the candidate change location, we fit the second regime parameters d, μ, and ν^2 by maximum likelihood. Note that a cross-validation procedures is not suitable for the second regime because the FI process is dependent on past observations.

The estimate change point $\hat{\tau}$ is selected as the time index that maximizes the

sum of the log-likelihood for the two regimes. In practice, it may be reasonable to restrict the candidate change locations to $c \in [\tau_a, \tau_b]$, where the lower and upper bounds on the change point τ_a and τ_b are specified by the user.

4.4.2 G2CD-fast: approximate search

G2CD-fast models the step transition function between the two regimes with a smooth transition function, and offers computational speed-ups since it does not need to evaluate model fit at each candidate change point. We define $w(t;\alpha)$ to be a continuous sigmoid or logistic function

$$w(t;\alpha) = \text{sigmoid}(\alpha_0 + \alpha_1 t), \tag{4.9}$$

where $\text{sigmoid}(x) = \frac{1}{1+e^{-x}} \in (0,1)$. The function is parameterized by a pair of continuous, real-valued parameters $\alpha = \{\alpha_0, \alpha_1\}$ such that the function monotonously increases from 0 to 1. We can think of G2CD-fast as doing an approximate search for the change point. Because the log-likelihood of the data can be differentiated with respect to α, GCD-fast has the potential to be a computationally simpler alternative to G2CD-exact.

Incorporating the transition function, then the objective function at y_t becomes:

$$\ell(y_t) = \ell^{(1)}(y_t)(1 - w(t;\alpha)) + \ell^{(2)}(y_t)\, w(t;\alpha) \tag{4.10}$$

where $\ell^{(1)}(y_t)$ and $\ell^{(2)}(y_t)$ is the log-likelihood of y_t under the first and second

regime, respectively, as in Equation 4.3. The expression for the FI process becomes:

$$g\left(y_1, \ldots, y_{t-1}; \mu, d\right) = \mu - \sum_{i=1}^{\infty} \binom{d}{i} (-1)^i \left(y_{t-i} - \mu\right) w\left(t - i; \alpha\right), \tag{4.11}$$

First regime parameters are estimated in the same way as in G2CD-exact. The remaining parameters are estimated by maximizing the log-likelihood jointly with respect to α, d, μ, and ν^2. The constraint that the gradual transition between the two regimes lies between the user-specified lower- and upper-bounds τ_a and τ_b can be enforced by adding a penalty $C(w(\tau_b; \alpha) - w(\tau_a; \alpha))$ with fixed penalty parameter $C > 0$ to the objective function. Larger values of C more strongly encourage the transition to occur within the user-specified range $[\tau_a, \tau_b]$ by encouraging $w(\tau_a; \alpha) = 0$ and $w(\tau_b; \alpha) = 1$. For a given application, C can be set to be on the order of the log-likelihood component using an initial estimate of the transition function, such as $\tilde{\alpha}_0 = -\frac{\tau_a + \tau_b}{2}$ and $\tilde{\alpha}_1 = 1$ so that $w\left(\frac{\tau_a + \tau_b}{2}; \tilde{\alpha}\right) = \frac{1}{2}$.

4.4.3 Multivariate Scenario

When there are p replicates of the sequences, the log-likelihood objective function is a sum of the log-likelihoods of the individual sequences. The only constraint is that the long-memory parameter d is shared across dimensions as described in Section 4.3.3.

When change locations are allowed to differ across replicates, the number of possible combinations for change locations is m^p, where m is the number of time indices in $[\tau_a, \tau_b]$. An exhaustive search for the best combination is often computationally prohibitive. For this reason, we use the following two-step procedure.

First, we run G2CD-exact or G2CD-fast on each univariate sequence to obtain estimates of β_j, θ_j, τ_j, and an initial time series specific value of d_j, μ_j, ν_j^2. Fixing the estimates for β_j, θ_j, and τ_j at these values, we then optimize over d having initialized at the mean univariate estimate across all of the time series $p^{-1}\sum_{j=1}^{p}\hat{d}_j$. The remaining parameters μ_j's and ν_j^2's are re-estimated simultaneously with d.

4.5 Simulation Study

We test for the performance of the proposed G2CD methods in estimating τ and d under difference scenarios, comprising both univariate and multivariate sequences. We set up the simulations to be similar to the ECIS data described in Section 4.1.

For the univariate experiments, each simulated series has $T = 400$ observations, with time indices scaled to 70 hours to match the ECIS applications. The value for τ ranges from 15-hour to 45-hour at intervals of 5, the exact value differs slightly due to spacing of the time indices. The trend in the first regime is generated using a degree-5 polynomial or Gaussian process. Heteroscedastic noise is added to the first regime, where the standard deviation is between 0.1 and 2, with larger standard deviation at higher value of the trend. The linear relationship is $\sigma_i = \frac{2-0.1}{\max\{y_j\}_{t_j=1}^{\tau}-\min\{y_j\}_{j=1}^{\tau}}\left[y_i - \min\{y_j\}_{j=1}^{\tau}\right] + 0.1$. In the second regime, the value for d ranges from -0.45 to 1.45 at intervals of 0.2. The standard deviation of noise is constant at $\nu = 0.5$. For each combination of τ and d, 100 simulations are conducted. The candidate range of τ is set to $[10, 50]$. We use spline basis of degree 3 with knots at every integer value of t when fitting β, and knots at every integer multiple of 5 when fitting θ. The regularization hyperparameters λ_f and λ_h are found by leave-one-out cross-validation. We use the `smooth.spline` function in `R`

[56] to fit the splines. The function optimizes for the regularization hyperparameter through a Golden-section search routine. When implementing G2CD-fast, we fix $C = 1000$.

For the multivariate experiment, the dimensionality is set to $p = 3$. The series on each dimension is generated following the same procedure as in the univariate experiments described above with the first regime generated using a Gaussian process. The three series have different change locations, at 15, 25 and 45, respectively. For each d, 100 simulations are conducted.

Since we are not aware of current methods tackling the change point problem involving the transition from a non-stationary model to a long-memory process, we compare the performance of G2CD with a 2-step procedure of change detection and estimation of d, as well as methods which estimates d with a given change location. Change detection is carried out using the popular E-Divisive algorithm [44], a nonparametric procedure which uses the energy statistics as a distance metric for binary segmentation. We use ECP and ECP.fdiff to denote the procedure where E-Divisive is applied on the original and differenced sequence, respectively. We specify ECP to find a maximum of 3 change points to have a higher chance of finding a change point in the τ candidate range. The estimated τ is set to be the first, which is also the most significant change point, found within the τ candidate range. After τ is estimated, d is estimated with the MLE of FI(d). For comparison, we also include FixedTau where the estimate of τ is fixed at the end of candidate range $\tau_b = 50$, and TrueTau where the true change location is given and used. TrueTau gives the best d estimate that can be attained through maximum likelihood estimation. FixedTau sets the bar for estimating d with conservative data usage.

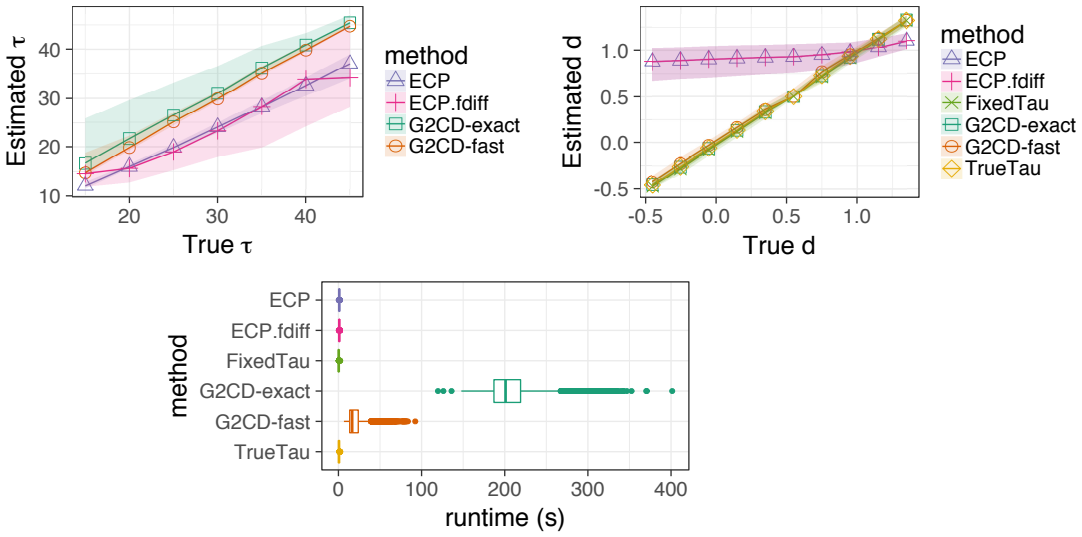
We present the simulation results collectively in Figure 4.4 and provide details for each simulation experiment in the respective sections below. We discuss further the performance of G2CD in Section 4.5.5.

4.5.1 The first regime generated via polynomial

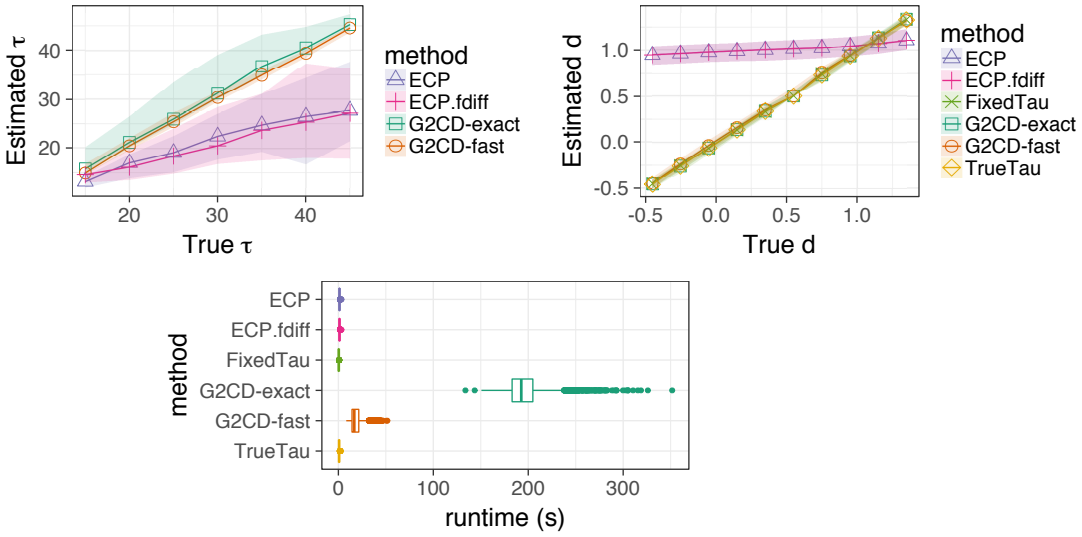
The series component in the first regime is generated from degree-5 polynomials. The coefficients are produced by drawing randomly from a normal distribution $N(0, 0.1^2)$ and further shrinking the output for coefficients corresponding to high degrees. For degree j where $j \geqslant 3$, the output is shrunk by a factor of 0.1^j. This ensures an expressive series that is not dominated by higher-order terms, since the higher-order terms can result in the first regime being scaled disproportionately to the second regime.

From Figure 4.4a, we see that the proposed G2CD methods recovers τ and d better than the ECP methods. ECP assumes that observations in a segment are independently and identically distributed, which does not hold for data in our use-case, and tends to segment too early. This results in poor estimates for τ and subsequently d. The G2CD methods obtain as good estimates of d as TrueTau which is given the true value of τ used. This demonstrates that G2CD is capable of additionally estimating τ without compromising the estimation for d. We notice that G2CD-exact tends to overestimate τ, which is not surprising since splines are capable of flexible fits. We discuss this phenomenon further in Section 4.5.5.

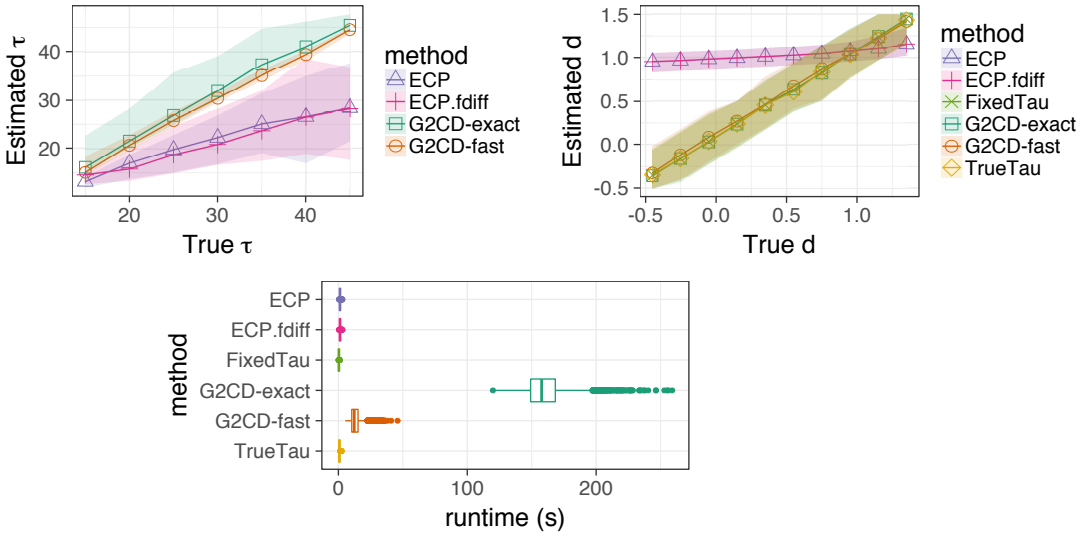
In terms of run-time, G2CD-fast roughly 10 times faster than G2CD-exact on average, which makes it more comparable with ECP in term of computation time while giving much better parameter estimates.



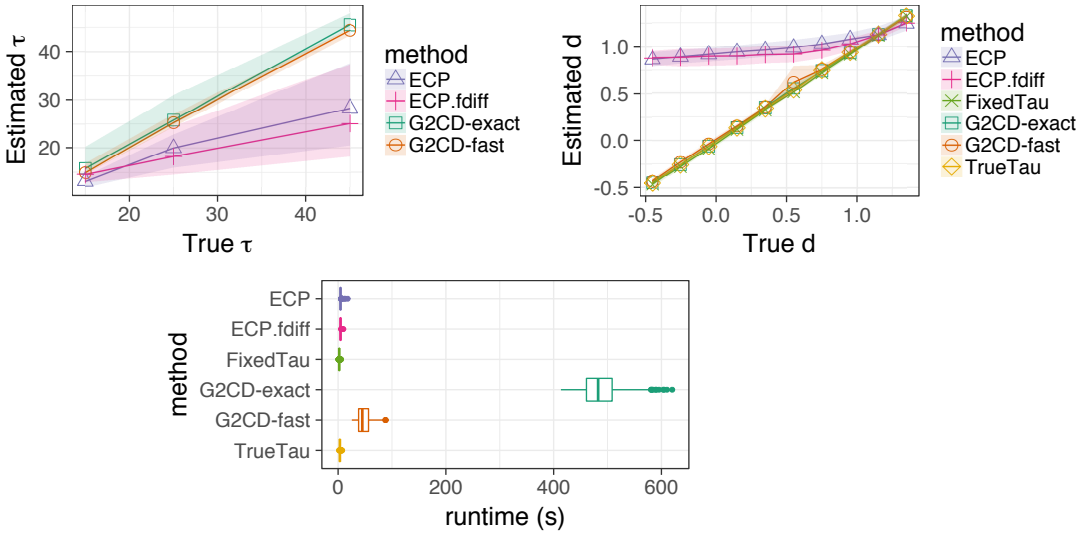
(a) The first regime generated via degree-5 polynomials; the second regime generated via FI(d).



(b) The first regime generated via Gaussian process with squared exponential kernel; the second regime generated via FI(d).



(c) The first regime generated via Gaussian process with squared exponential kernel; the second regime generated via ARFIMA(1,d,1).



(d) Multivariate simulations with dimensionality 3, with change points at 15, 25, and 45. For each series, the first regime is generated via Gaussian process with squared exponential kernel, and the second regime is generated via FI(d).

Figure 4.4: Estimations for change location τ and long-memory parameter d, and computation time across 100 simulated series per each combination of τ and d for each simulation configuration. In plots for τ and d, medians are plotted and error bars indicate the upper and lower quartiles.

4.5.2 The first regime generated via Gaussian process

The series component in the first regime is generated from a Gaussian process with the squared exponential kernel $K_{SE}(s,t) = 10\exp(-0.5(s-t)^2)$. The sequences generated from Gaussian process is more varied as compared to polynomials in terms of trend. Performances as shown in Figure 4.4b are similar to those in Section 4.5.1.

4.5.3 The first regime generated via Gaussian process, the second regime generated via ARFIMA(1,d,1)

This set of experiments aims to test G2CD's robustness to model misspecification in the second regime. The series component in the first regime is generated from a Gaussian process as described in Section 4.5.2. The series component in the second regime is generated from ARFIMA(1,d,1) where the autoregressive and moving-average coefficients are drawn from a uniform $Unif(0,1)$ distribution.

From Figure 4.4c, as compared to the results in Section 4.5.2 with no model misspecification, we see here that the G2CD methods obtained similar performance for estimating τ but show higher inaccuracy for the estimation of d. However, the level of accuracy is still on par with TrueTau.

4.5.4 Multivariate scenario

In this experiment, each multivariate sample has dimensionality 3. Each of the 3 sequences are generated from a Gaussian process as in Section 4.5.2 for the first

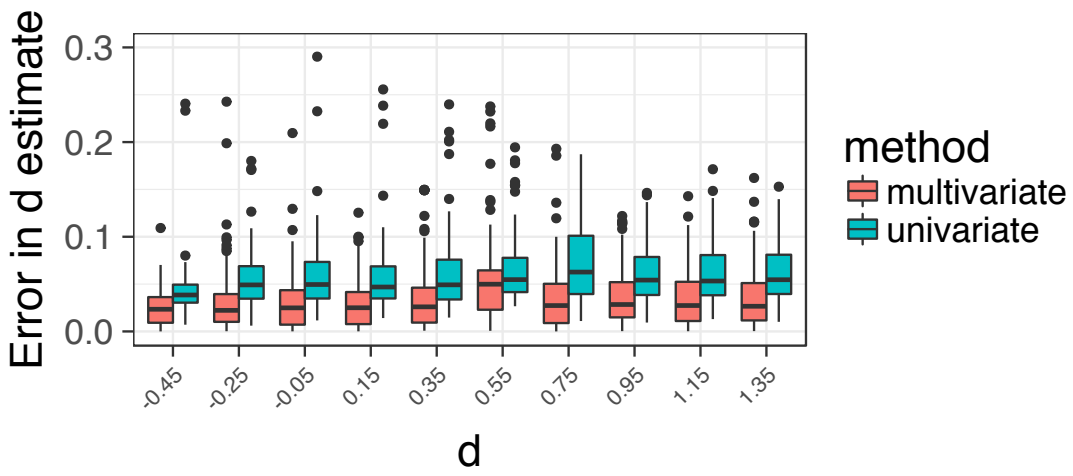


Figure 4.5: Absolute errors in estimates of d by univariate and multivariate implementations of G2CD-exact. The pooled estimates reduced errors.

regime, and a FI process for the second regime, and has a change point at 15, 25, or 45. The long-memory parameter d is shared across the 3 sequences.

For both ECP and G2CD, change points are detected independently for each series. Figure 4.4d reflects that G2CD estimates τ more accurately than the ECP methods, as in the univariate experiments. G2CD-exact's estimation of d is comparable to TrueTau's. G2CD-fast overestimates d slightly when d is just above 0.5, but has computational time roughly 10 times as fast as G2CD-exact. Hence G2CD-fast is still a viable option in providing balance between the quality of estimation and computational speed.

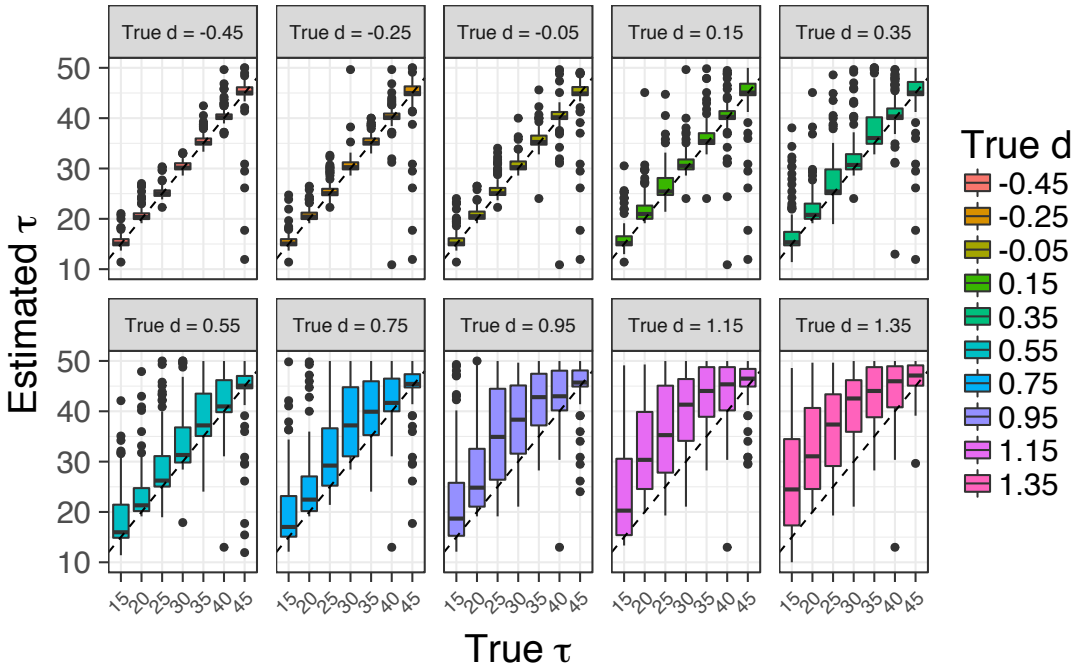
We compare the absolute errors in the estimates of d by the univariate and multivariate implementations of G2CD-exact in Figure 4.5, which demonstrates the benefit of the pooled estimates in reducing errors across all values of d tested.

4.5.5 Discussion

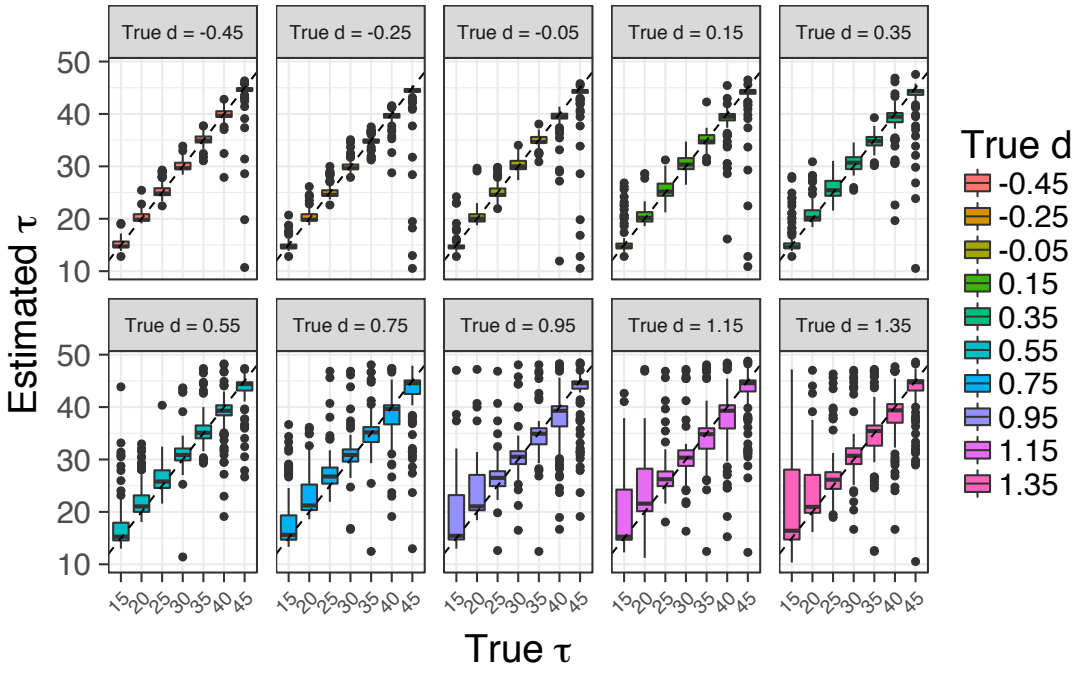
From the simulation experiments, we notice that G2CD-exact tends to overestimate τ. We plot the τ estimated by both G2CD methods for the simulations in Section 4.5.2. From Figure 4.6, we observe that the issue is more prominent in G2CD-exact for larger values of d, as the second regime begins to show more variability which may be picked up by the splines as part of the growth curve. On the other hand, G2CD-fast is affected to a smaller extend. The plot shows the time index when the regime transition function is estimated to cross 0.5. Since G2CD-fast imposes a smooth transition function, it captures uncertainty in segmenting the sequence by estimating a less steep transition function. Figure 4.7 shows the difference between the estimated transition functions for simulated sequences with a small and a large d.

Repeating the analysis on the estimates of d, we see from Figure 4.8 that as the change location τ increases, the estimates of d are slightly more noisy for both G2CD-exact and G2CD-fast. This is expected since as τ increases, less data is available for the estimation of parameters in the second regime.

We further compare the estimates of d by G2CD and the FixedTau method in this same set of simulations in Section 4.5.2 to explore benefits of segmenting the sequence on long-memory parameter estimation. FixedTau segments all sequences at 50. Figure 4.9 plots the absolute errors in the d estimates. Compared to FixedTau, the G2CD methods have lower errors when the ground truth τ is small. In particular, the upper quartile of G2CD-exact error becomes higher than that of FixedTau error only at $\tau = 45$, which is near the upper end of the candidate change point range τ_b. At high values of τ, the G2CD methods may start to segment earlier than the ground truth, causing the observations used for long-

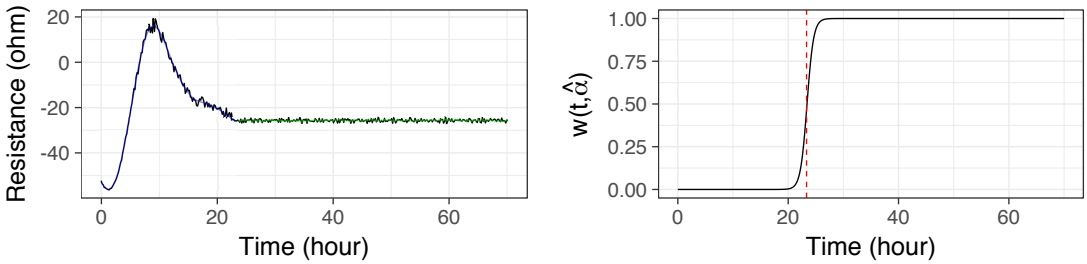


(a) Estimates of τ by G2CD-exact.

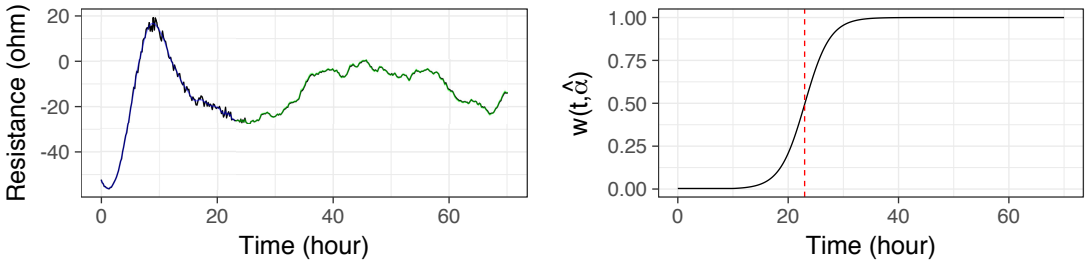


(b) Estimates of τ by G2CD-fast.

Figure 4.6: G2CD estimates of τ for simulation setup where the first regime is generated via Gaussian process with squared exponential kernel and the second regime generated via FI(d). G2CD-exact tends to overestimate τ for larger values of $d > 0.5$ due to non-stationarity of the second regime.

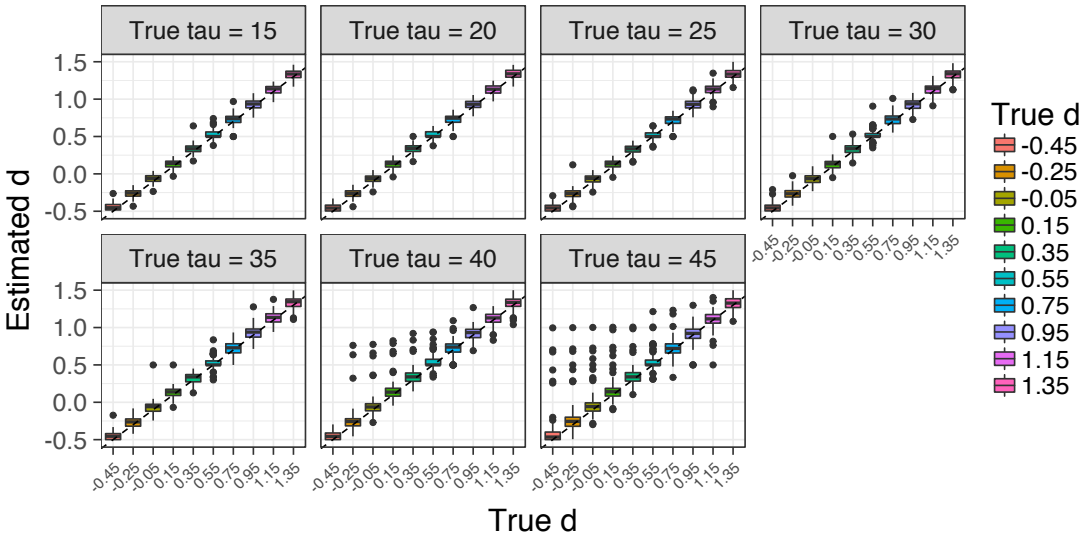


(a) Estimates by G2CD-fast for simulation with $d = -0.45$.

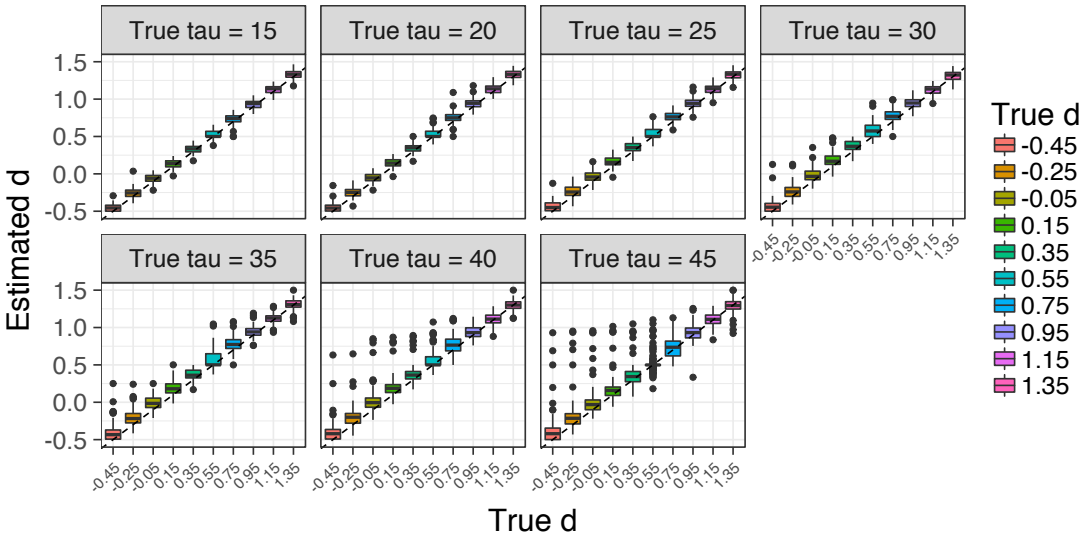


(b) Estimates by G2CD-fast for simulation with $d = 1.35$.

Figure 4.7: G2CD-fast estimates for simulation setup where the first regime is generated via Gaussian process with squared exponential kernel and the second regime generated via FI(d). The blue and green overlaid lines are the fit by G2CD-fast for the growth and confluence phase, respectively. The vertical red dashed line marks the time index when the regime transition function is estimated to cross 0.5. The estimated transition is less abrupt for large d.



(a) Estimates of d by G2CD-exact.



(b) Estimates of d by G2CD-fast.

Figure 4.8: G2CD estimates of d for simulation setup where the first regime is generated via Gaussian process with squared exponential kernel and the second regime generated via FI(d). Estimates of d are slightly noisier as τ increases and less data is available for parameter estimation in the second regime.

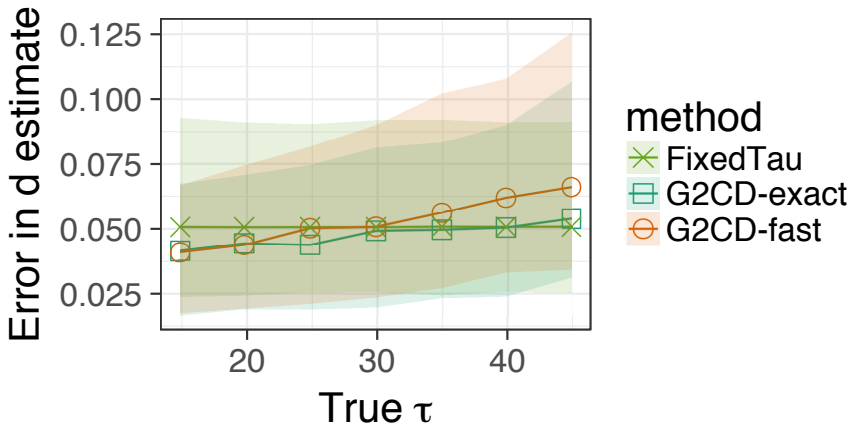


Figure 4.9: Absolute errors in estimates of d for simulation setup where the first regime is generated via Gaussian process with squared exponential kernel and the second regime generated via FI(d). Medians are plotted and error bars indicate the upper and lower quartiles. G2CD reduced errors when the ground truth τ is small. For larger τ, G2CD may underestimate τ, which increases errors in estimating d.

memory parameter estimation to be contaminated with first regime observations. Between the two G2CD methods, G2CD-fast show this effect earlier since phase transition is modeled with a smooth curve.

4.6 Application to ECIS Data

Two sets of data are used, the first is from Madin-Darby Canine Kidney (MDCK) cells, and the second is from BSC cells of African green monkey kidney origin. Both are model mammalian cell line used in biomedical research. Each dataset consists 4 experiments. Each experiment uses a tray with 96 wells, of which some are left empty to act as buffers between differently-treated cells. The wells differ by the serum type (bovine serum albumin BSA vs gel) and the presence of infection by mycoplasma. The effective configuration is 8 wells for BSA and uninfected, 8 wells for gel and uninfected, 12 wells for BSA and infected, and 12 wells for gel and infected. The exception is that Experiment 2 for MDCK cell line has one less

well for BSA and infected.

We study the measurements of resistance, a component of impedance, provided at the frequency of 500 hertz. All wells are observed for at least 72 hours. Figure 4.10 and 4.11 plots the time series and spectrogram for a MDCK sample and a BSC sample, as well as standardized residuals from model fitting with G2CD-exact. The residuals from the first regime are scaled by σ_t estimated. The residual plots show that the normal assumption for the noise terms is met.

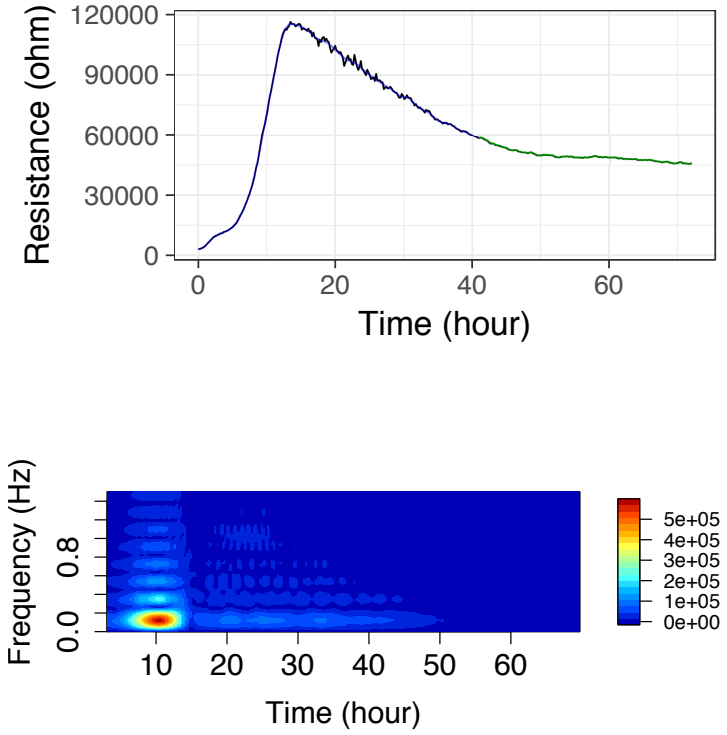
For model fitting, the hyperparameter setups are the same as in Section 4.5 since the simulations are generated to assimilate the ECIS data, except that the candidate change point range for the BSC cell line is set to an earlier period of $[5, 45]$ to capture the earlier peak observed in the dataset.

4.6.1 MDCK cell line

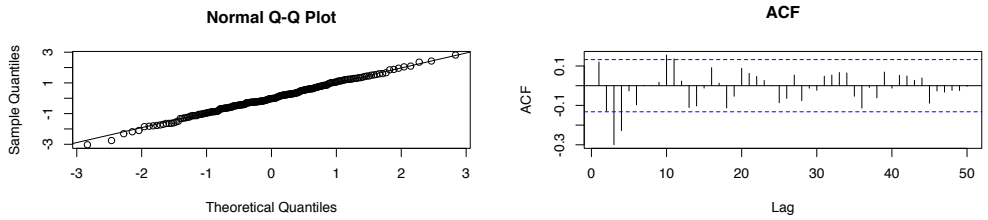
From Figure 4.10, we see that the resistance measurements for the MDCK cell line tend to peak before slightly decreasing to reach a constant trend. The start of the confluence phase is hypothesized to be at or slightly after the peak.

Table 4.1 summarizes the average d estimated by G2CD-exact and G2CD-fast across experiments, serum types and infection status. Except for Experiment 2, the average d for infected samples is always higher than that of normal samples.

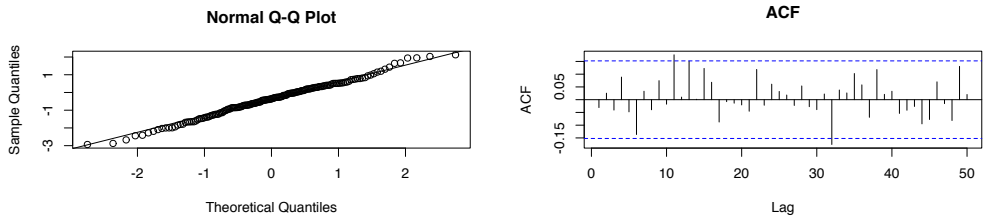
Figure 4.12 plots the τ and d estimated by G2CD-exact for each of the MDCK samples. The estimated change points are scattered within the candidate range of $[10, 50]$, signifying varied initial conditions even in the same batch. All d estimates are above 0.5, which means that the differenced series have intermediate or long-



(a) Time series and spectrogram of resistance measurements recorded at 500 hertz. The blue and green overlaid lines are the fit by G2CD-exact for the growth and confluence phase, respectively.

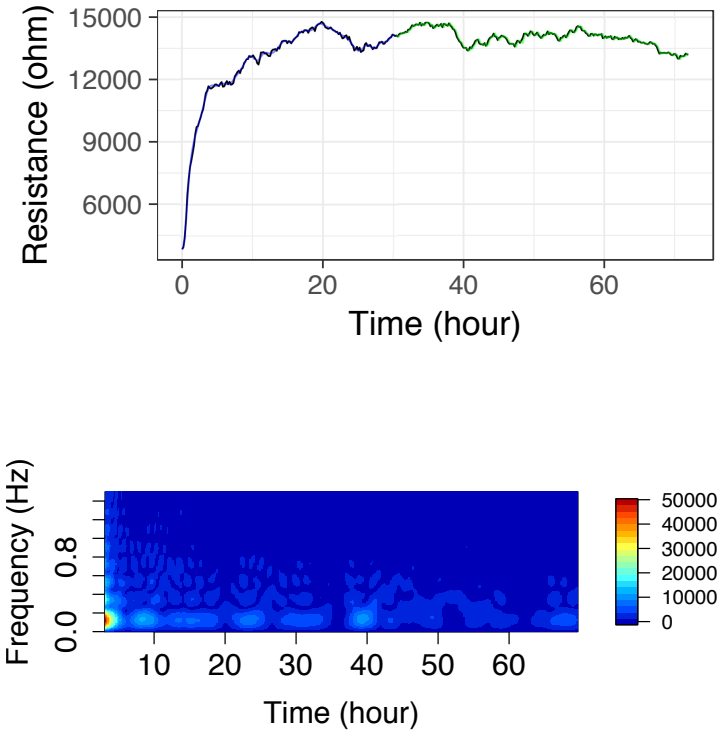


(b) Residual analysis of standardized residuals for the growth phase.

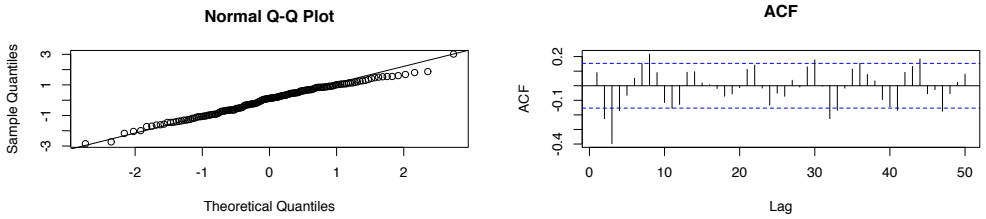


(c) Residual analysis of standardized residuals for the confluence phase.

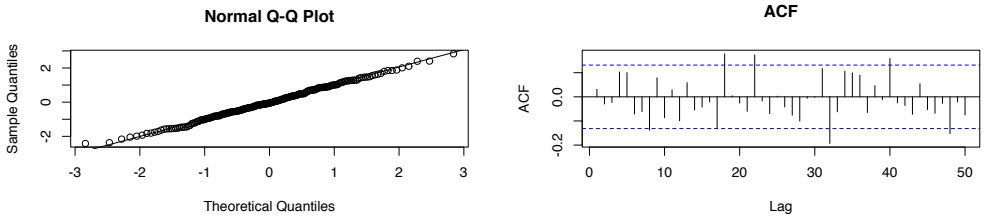
Figure 4.10: MDCK cell; infected and cultivated in gel.



(a) Time series and spectrogram of resistance measurements recorded at 500 hertz. The blue and green overlaid lines are the fit by G2CD-exact for the growth and confluence phase, respectively.



(b) Residual analysis of standardized residuals for the growth phase.



(c) Residual analysis of standardized residuals for the confluence phase.

Figure 4.11: BSC cell; infected and cultivated in gel.

Expt	Serum	Infection	G2CD-exact		G2CD-fast	
			Mean	SD	Mean	SD
1	BSA	No	0.730	0.063	0.728	0.073
		Yes	0.998	0.091	1.019	0.092
	Gel	No	0.781	0.056	0.766	0.070
		Yes	0.873	0.157	0.943	0.101
2	BSA	No	0.686	0.122	0.673	0.114
		Yes	0.656	0.109	0.824	0.099
	Gel	No	0.821	0.051	0.830	0.029
		Yes	0.709	0.127	0.789	0.127
3	BSA	No	0.676	0.125	0.694	0.112
		Yes	0.915	0.097	0.988	0.099
	Gel	No	0.757	0.118	0.782	0.097
		Yes	0.915	0.126	1.040	0.038
4	BSA	No	0.676	0.110	0.702	0.072
		Yes	0.827	0.112	0.886	0.158
	Gel	No	0.756	0.034	0.750	0.049
		Yes	0.805	0.186	0.834	0.160

Table 4.1: MDCK: G2CD estimates of d. Average is taken for samples in the same experiment, serum type and infection status.

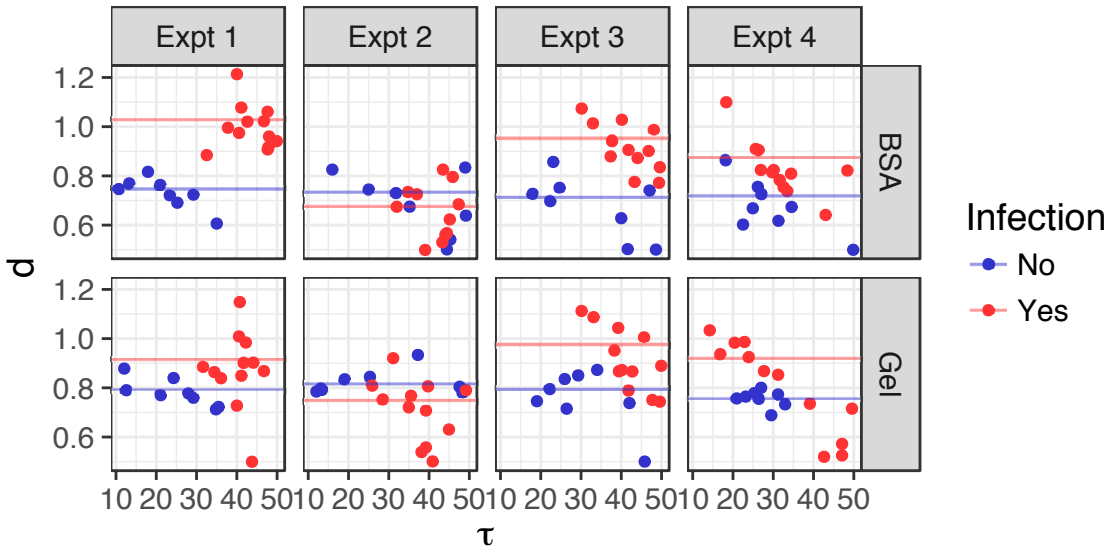


Figure 4.12: MDCK: G2CD-exact estimates for τ and d. Points are estimates from the univariate version of the method, and horizontal lines mark estimates from the multivariate version.

memory at confluence. We notice that Experiments 1, 3 and 4 are the most similar in implying that infected cells tend to have d of larger magnitude. In particular, when $d > 1$, the differenced series demonstrate long-memory properties. The only samples with $d > 1$ are infected samples. When $0.5 < d < 1$, the differenced series demonstrate intermediate-memory. Experiment 2 is the most dissimilar from the other experiments, reflecting the presence of batch effects. The horizontal lines mark the d estimated by the multivariate version of G2CD-exact, which also reflects that the d estimates for infected cells in Experiment 2 are lower than those in the other three experiments.

4.6.2 BSC cell line

From Figure 4.11, we see that the resistance measurements for the BSC cell line tend to increase sharply before plateauing. As compared to the MDCK cell line,

the end of the BSC growth phase is less distinguishable due to fluctuations in the resistance measurement even as it plateaus. This introduces more difficulty in the change point detection problem, as well as the subsequent estimation of the long-memory parameter.

Table 4.2 summarizes the average d estimated by G2CD-exact and G2CD-fast across experiments, serum types and infection status. Contrary to the MDCK cell line, the average d for the infected samples in the BSC cell line is lower than that of normal samples in most cases.

Figure 4.13 plots the τ and d estimated by G2CD-exact for each of the BSC samples. There is less distinct separation between the infected and normal cells. Almost all d estimates are above 1, which means that the differenced series have long-memory at confluence. In contrast to the MDCK cell line, the infected BSC cells tend to have lower d than their uninfected counterparts as observed from Experiments 1 and 2, signifying that the latter has stronger long-memory properties. The horizontal lines mark the d estimated by the multivariate version of G2CD-exact, which also reflects that the long-memory parameters between the infected and normal cells are close in magnitude.

4.6.3 Classification for infection

To demonstrate the quality and usage of the estimated τ and d, we apply the two estimated parameters as features in a downstream task. The task is to classify a sequence by its infection status. We replicate the features and classifiers used in a related work to classify cell lines from ECIS data [24]. The two classifiers tested are Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA).

Expt	Serum	Infection	G2CD-exact		G2CD-fast	
			Mean	SD	Mean	SD
1	BSA	No	1.101	0.077	1.102	0.076
		Yes	1.003	0.056	1.017	0.044
	Gel	No	1.098	0.050	1.106	0.054
		Yes	1.036	0.101	1.015	0.077
2	BSA	No	1.057	0.092	1.057	0.083
		Yes	1.018	0.085	1.041	0.079
	Gel	No	1.089	0.063	1.079	0.066
		Yes	1.021	0.069	1.012	0.047
3	BSA	No	1.056	0.051	1.085	0.080
		Yes	1.079	0.063	1.072	0.053
	Gel	No	1.057	0.092	1.064	0.095
		Yes	1.042	0.050	1.043	0.043
4	BSA	No	1.101	0.051	1.106	0.065
		Yes	1.130	0.063	1.131	0.055
	Gel	No	1.114	0.060	1.119	0.045
		Yes	1.089	0.073	1.086	0.055

Table 4.2: BSC: G2CD estimates of d. Average is taken for samples in the same experiment, serum type and infection status.

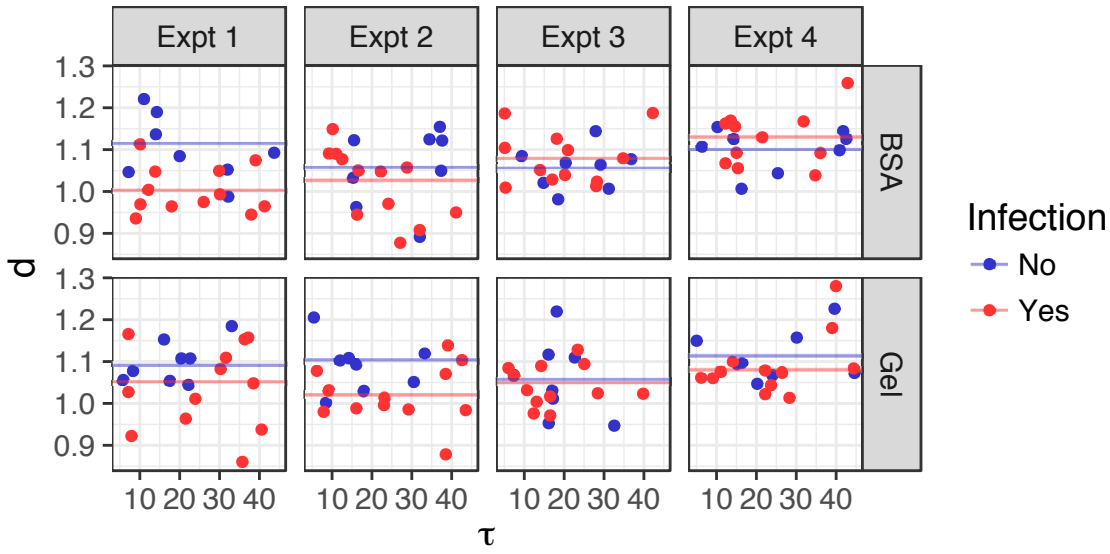


Figure 4.13: BSC: G2CD-exact estimates for τ and d. Points are estimates from the univariate version of the method, and horizontal lines mark estimates from the multivariate version.

The class discriminant scores for LDA ($\rho = 1$) and QDA ($\rho = 0$) in this binary classification task are:

$$\delta_c(X) = \left(X - \bar{X}_c\right)^T \hat{\Sigma}_c^{-1}(\rho) \left(X - \bar{X}_c\right) + \log|\hat{\Sigma}_c(\rho)| \tag{4.12}$$

where

$$\hat{\Sigma}_c(\rho) = (1-\rho)\hat{\Sigma}_c + \rho\hat{\Sigma}$$

for class c. QDA assumes that the features given the class is normally distributed with a class mean and covariance, and LDA makes the further assumption that the class covariances are equal for the two classes.

To account for batch effects between the 4 experiments for each cell line, we test for robustness of the features by using each experiment as the test set in turn,

Cell line	Features	LDA		QDA	
		Mean	SD	Mean	SD
MDCK	Original	0.743	0.272	0.580	0.237
	G2CD-exact	0.880	0.091	0.862	0.136
	G2CD-fast	0.962	0.033	0.975	0.020
BSC	Original	0.563	0.060	0.588	0.072
	G2CD-exact	0.650	0.098	0.630	0.113
	G2CD-fast	0.675	0.108	0.644	0.085

Table 4.3: Classification accuracy for infection status. Average is taken by taking each of the 4 experiments as the test set, and the other 3 as training set. Parameters τ and d estimated by G2CD increased classification accuracy for both MDCK and BSC cell line.

and training on the other 3 experiments. We report the mean test accuracy along with the standard deviation in Table 4.3. The Original features [24] are

- Average resistance at a time fixed time-mark where the measurement tends to peak;
- Maximum average resistance;
- Average resistance at the end of the sequence;

A simple moving average with window length 5 is taken to smoothen the sequence to obtain more stable estimates of the features of interest. The time-mark used for the first feature is 17-hour for MDCK and 2-hour for BSC, selected by visual inspection of the data. The G2CD-exact features are a combination of the τ and d estimated by G2CD-exact and a feature from the Original set that gives the best training accuracy. The concept is the same for the G2CD-fast features. This ensures that the same number of features is input into the classifiers.

From Table 4.3, it is evident that the τ and d estimates from the G2CD methods are useful features that increase classification accuracy for both cell lines over the Original features. For the MDCK cell line, using the LDA classifier, the Original features have a mean classification accuracy of 0.743. Including G2CD-exact and G2CD-fast features improved the mean classification accuracy by 18.4% and 29.4%, respectively. In the BSC cell line, using the QDA classifier, the Original features have a mean classification accuracy of 0.588. Including G2CD-exact and G2CD-fast features improved the mean classification accuracy by 7.1% and 9.5%, respectively. As pointed out in Section 4.6.2, the change points for the BSC samples are less visually obvious, and hence the overall classification accuracy is not as high as that for MDCK samples.

4.7 Conclusion

In this work, we proposed two versions of the method G2CD that automatically segments measurements of cell activity through ECIS data into the growth and confluence phase, as well as quantifies the long-memory behavior in the confluence phase. This reduces the human supervision currently needed to manually segment the data, and ensures that all samples are segmented using the same logic. Practical usage on the MDCK and BSC cell lines shows that G2CD recovers meaningful change points and long-memory parameter in the confluence phase that improve downstream tasks such as classification.

4.8 Supplementary Materials

4.8.1 Feasible Generalized Least Squares

Heteroscedasticity is addressed through Feasible Generalized Least Squares (FGLS). We adopt the following iterative procedure to fit the trend and noise components:

1. Estimate the parameters β in $f(t;\beta)$ assuming homogeneous noise by penalized least squares;
2. Estimate the noise standard deviation $\{\sigma_t\}_{t=1}^{T}$ given $\hat{\beta}_{LS}$ from step 1;
3. Re-estimate β given $\{\hat{\sigma}_t\}_{t=1}^{n}$ from step 2 through FGLS.

Let M denote the penalty on β. Then an application of FGLS [64] to the penalized least squares problem is

$$\hat{\beta}_{FGLS} = \left(X'\widehat{W}^{-1}X + \lambda M\right)^{-1} X'\widehat{W}^{-1}y$$

where $\widehat{W}$ is an estimate of W. For the initial estimate of β, we set $W = I$ under the assumption of homogeneous noise to obtain $\hat{\beta}_{LS}$. We then fit another spline to the log squared residuals $\log\left(y_t - X_{t\cdot}\hat{\beta}_{LS}\right)^2$, and finally taking the exponential of the spline fit to obtain $\hat{\sigma}_t^2$. For the re-estimate of β, we set $\widehat{W}_{t,t} = \hat{\sigma}_t^2$. Step 2 and 3 can be iterated till convergence.

4.8.2 Profile log-likelihood of ARFIMA process

We consider the extension to an AutoRegressive Fractionally Integrated Moving Average (ARFIMA) process. From [17, 7], for process Y_t following an ARFIMA(p,d,q) model,

$$\Phi(B)(1-B)^d(Y_t-\mu_t)=\Theta(B)\epsilon_n$$

where $\Phi(B)=1-\phi_1B-\cdots-\phi_pB^p$ is the p^{th}-order autoregressive polynomial, and $\Theta(B)=1+\theta_1B+\cdots+\theta_qB^q$ is the q^{th}-order moving average polynomial in the lag operator B. The orders $p,q\in\mathbb{Z}_0^+$, and the degree of differencing $d\in\mathbb{R}$. The difference in definition between ARFIMA and ARIMA models is that $d\in\mathbb{Z}_0^+$ in ARIMA models. The fractional difference operator $(1-B)^d$ is the binomial expansion:

$$(1-B)^d=\sum_{i=0}^{\infty}\binom{d}{i}(-B)^i$$

The mean of the process is μ_t and the noise is independently and identically distributed $\epsilon_t\sim\mathrm{N}\left(0,\nu^2\right)$.

Let τ denote the change location. For $d\in(-0.5,0.5)$, the profile log-likelihood of the ARFIMA process in the second regime by expressing the covariance matrix $\Sigma^{(2)}$ in terms of $\Theta^{(2)}=\{\nu,\mu,d\}$ is

$$\begin{aligned}\ell^{(2)}&\left(y_{\tau+1},\ldots,y_T;\Theta^{(2)}\right)\\&=-\frac{n-\tau}{2}\log(2\pi)-\frac{1}{2}\log\left(\left|\Sigma^{(2)}\left(\Theta^{(2)}\right)\right|\right)\\&\quad-\frac{1}{2}\left(\overrightarrow{y}-\mu\right)'\left(\Sigma^{(2)}\left(\Theta^{(2)}\right)\right)^{-1}\left(\overrightarrow{y}-\mu\right)\end{aligned}\tag{4.13}$$

Since the process is stationary, the covariance matrix is Toeplitz. For ARFIMA(p, d, q) where $p = 0$, the s^{th}-autocovariance [59] is

$$\gamma(s) = \nu^2 \sum_{\ell=-q}^{q} \psi(\ell) \frac{\Gamma(1-2d)\Gamma(d+s-\ell)}{\Gamma(d)\Gamma(1-d)\Gamma(1-d-s+l)}$$

where $\Gamma(a)$ is the Gamma function $\Gamma(a) = \int_0^\infty x^{a-1}e^x dx$, and

$$\psi(\ell) = \sum_{i=\max[0,\ell]}^{\min[q,q-\ell]} \theta_i \theta_{i-\ell}$$

When $p \neq 0$, the autocovariance function is a more complex form and we refer readers to Sowell's paper [59] for details.

BIBLIOGRAPHY

[1] Cesare Alippi, Giacomo Boracchi, Diego Carrera, and Manuel Roveri. Change detection in multivariate datastreams: Likelihood and detectability loss. In *IJCAI*, 2016.

[2] Sylvain Arlot, Alain Celisse, and Zaid Harchaoui. Kernel change-point detection. *arXiv*, 2012.

[3] George K. Atia. Change detection with compressive measurements. *Signal Processing Letters*, 22(2):182–186, Feb 2015.

[4] Richard T. Baillie. Long memory processes and fractional integration in econometrics. *Journal of Econometrics*, 73(1):5 – 59, 1996.

[5] Rafal Baranowski, Yining Chen, and Piotr Fryzlewicz. Narrowest-over-threshold detection of multiple change-points and change-point-like feature. *arXiv*, 2016.

[6] Daniel Barry and J. A. Hartigan. A Bayesian analysis for change point problems. *JASA*, 88(421):309–319, 1993.

[7] Christopher Baum. ARFIMA (long memory) models. `http://fmwww.bc.edu/ec-c/s2013/327/EC327.S2013.nn5.slides.pdf`, 2013. Accessed: 2019-06-04.

[8] Richard Bellman. On the theory of dynamic programming. *Proceedings of the National Academy of Sciences of the United States of America*, 38(8):716, 1952.

[9] Christopher M. Bishop. Bayesian PCA. In *NIPS*, pages 382–388, Cambridge, MA, USA, 1999. MIT Press.

[10] Ekaterini Blaveri, Jeremy L. Brewer, Ritu Roydasgupta, Jane Fridlyand, Sandy DeVries, Theresa Koppie, Sunanda Pejavar, Kshama Mehta, Peter Carroll, Jeff P. Simko, and Frederic M. Waldman. Bladder cancer stage and outcome by array-based comparative genomic hybridization. *Clinical Cancer Research*, 11(19):7012–7022, 2005.

[11] Kevin Bleakley and Jean-Philippe Vert. The group fused Lasso for multiple change-point detection. *arXiv*, 06 2011.

[12] Carlos M. Carvalho, Nicholas G. Polson, and James G. Scott. Handling sparsity via the horseshoe. In *AISTATS*, volume 5 of *Proceedings of Machine Learning Research*, pages 73–80. PMLR, 16–18 Apr 2009.

[13] Chung Chen and Lon-Mu Liu. Joint estimation of model parameters and outlier effects in time series. *Journal of the American Statistical Association*, 88(421):284–297, 1993.

[14] Jie Chen and Arjun K Gupta. *Parametric Statistical Change Point Analysis: With Applications to Genetics, Medicine, and Finance*. Springer, 2011.

[15] Javier López de Lacalle. *tsoutliers: Detection of Outliers in Time Series*, 2017. R package version 0.6-6.

[16] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.

[17] Jurgen A. Doornik and Marius Ooms. Inference and forecasting for ARFIMA models with an application to US and UK inflation. *Studies in Nonlinear Dynamics and Econometrics*, 8(2), 2004.

[18] Gilles Dufrenot, Dominique Guegan, and Anne Peguin-Feissolle. Changing-regime volatility : A fractionally integrated SETAR model. *Applied Financial Economics*, 18:519–526, 04 2008.

[19] Tipping Michael E. and Bishop Christopher M. Probabilistic principal component analysis. *JRSS: Series B*, 61(3):611–622, 1999.

[20] Chandra Erdman and John Emerson. bcp: An R package for performing a Bayesian analysis of change point problems. *JSS*, 23(3):1–13, 2007.

[21] Edward B Fowlkes and Colin L Mallows. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78(383):553–569, 1983.

[22] Piotr Fryzlewicz. Wild binary segmentation for multiple change-point detection. *The Annals of Statistics*, 42(6):2243–2281, 2014.

[23] Chuan Gao, Christopher Brown, and Barbara Engelhardt. A latent factor model with a mixture of sparse and dense factors to model gene expression data with confounding effects. *arXiv*, 10 2013.

[24] Megan Gelsinger, Laura Tupper, and David Matteson. Cell line classification using electric cell-substrate impedance sensing (ECIS). *The International Journal of Biostatistics*, 10 2017.

[25] Daniel Gilbert and Martin Wells. Tuning free rank-sparse Bayesian matrix and tensor completion with global-local priors. *arXiv*, 2019.

[26] Abdul Hakim, M. Saiful Huq, Shahnoor Shanta, and B.S.K.K. Ibrahim. Smartphone based data mining for fall detection: Analysis and design. *Procedia Computer Science*, 105:46 – 51, 2017.

[27] Zaid Harchaoui and Oliver Cappe. Retrospective multiple change-point estimation with kernels. In *Statistical Signal Processing, 2007. SSP '07. IEEE/SP 14th Workshop on*, pages 768 –772, 2007.

[28] Flore Harlé, Florent Chatelain, Cédric Gouy-Pailler, and Sophie Achard. Bayesian model for multiple change-points detection in multivariate time series. *IEEE Transactions on Signal Processing*, 64(16):4351–4362, Aug 2016.

[29] John Haslett and Adrian E. Raftery. Space-time modelling with long-memory dependence: Assessing Ireland's wind power resource. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 38(1):1–50, 1989.

[30] Kaylea Haynes, Rebecca Killick, Paul Fearnhead, and Idris Eckley. *changepoint.np: Methods for Nonparametric Changepoint Detection*, 2016. R package version 0.0.2.

[31] Mark Holmes, Ivan Kojadinovic, and Jean-François Quessy. Nonparametric tests for change-point detection à la Gombay and Horváth. *Journal of Multivariate Analysis*, 115:16–32, 2013.

[32] Jongin Hong, Karthikeyan Kandasamy, Mohana Marimuthu, Cheol Soo Choi, and Sanghyo Kim. Electrical cell-substrate impedance sensing as a non-invasive tool for cancer cell study. *Analyst*, 136:237–245, 2011.

[33] Aapo Hyvarinen, Juha Karhunen, and Erkki Oja. Independent component analysis. In *Wiley Interscience*, 2001.

[34] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4):411 – 430, 2000.

[35] Nicholas A. James, Wenyu Zhang, and David S. Matteson. *ecp: An R Package for Nonparametric Multiple Change Point Analysis of Multivariate Data*, 2019.

[36] Charles Keese. ECIS application webinar series. `http://www.biophysics.com/webinar.php`, 2019. Accessed: 2019-04-13.

[37] Daniel Kifer, Shai Ben-David, and Johannes Gehrke. Detecting change in data streams. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30*, VLDB '04, pages 180–191. VLDB Endowment, 2004.

[38] R. Killick, P. Fearnhead, and IA Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

[39] Daniel Kowal, David Matteson, and David Ruppert. Dynamic shrinkage processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 07 2017.

[40] Chung-Ming Kuan. Generalized least squares theory. 2004.

[41] D Lovelady, T Richmond, A Maggi, C-M Lo, and D Rabson. Distinguishing cancerous from noncancerous cells through analysis of electrical noise. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 76:041908, 11 2007.

[42] Alexandre Lung-Yut-Fong, Céline Lévy-Leduc, and Olivier Cappé. Homogeneity and change-point detection tests for multivariate data using rank statistics. *arXiv*, 2011.

[43] Edgard M Maboudou-Tchao and Douglas M Hawkins. Detection of multiple change-points in multivariate data. *Journal of Applied Statistics*, 40(9):1979–1995, 2013.

[44] David S. Matteson and Nicholas A. James. A nonparametric approach for

multiple change point analysis of multivariate data. *Journal of the American Statistical Association*, 109(505):334–345, 2014.

[45] David S. Matteson and Ruey S. Tsay. Independent component analysis via distance covariance. *JASA*, 112(518):623–637, 2017.

[46] Ian McCulloh and Kathleen M. Carley. Social network change detection. 2008.

[47] Raymond McTaggart and Gergely Daroczi. *Quandl: Quandl Data Connection*, 2013. R package version 2.1.2.

[48] Colin P Morice, John J Kennedy, Nick A Rayner, and Phil D Jones. Quantifying uncertainties in global and regional temperature change using an ensemble of observational estimates: The hadcrut4 data set. *Journal of Geophysical Research: Atmospheres (1984–2012)*, 117(D8), 2012.

[49] Masoud M. Nasari. Strong law of large numbers for weighted U-statistics: Application to incomplete U-statistics. *Statistics & Probability Letters*, 82(6):1208 – 1217, 2012.

[50] Varvara Nika, Paul Babyn, and Hongmei Zhu. Change detection of medical images using dictionary learning techniques and PCA. In *Medical Imaging*, 2014.

[51] Adam B. Olshen, E.S. Venkatraman, Robert Lucito, and Michael Wigler. Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics*, 5:557 – 572, 2004.

[52] Daniel Opp, Brian Wafula, Jennifer Lim, Eric Huang, Jun-Chih Lo, and Chun-Min Lo. Use of electric cell–substrate impedance sensing to assess in vitro cytotoxicity. *Biosensors and Bioelectronics*, 24(8):2625 – 2629, 2009.

[53] Jeffrey Pai and Nalini Ravishanker. Maximum likelihood estimation in vector long memory processes via EM algorithm. *Computational Statistics Data Analysis*, 53(12):4133 – 4142, 2009.

[54] Jean-Yves Pitarakis. Least squares estimation and tests of breaks in mean and variance under misspecification. *Econometrics Journal*, 7(1):32–54, 2004.

[55] Theodor D. Popescu. Blind separation of vibration signals and source change detection – application to machine monitoring. *Applied Mathematical Modelling*, 34(11):3408 – 3421, 2010.

[56] R Foundation for Statistical Computing, Vienna, Austria. *R: A Language and Environment for Statistical Computing*, 2018.

[57] Guillem Rigaill. Pruned dynamic programming for optimal multiple change-point detection. *arXiv*, 2010.

[58] Carl James Schwarz. An introduction to splines. `http://people.stat.sfu.ca/~cschwarz/Consulting/Trinity/Phase2/TrinityWorkshop/Workshop-material-Simon/Intro_to_splines/intro_to_splines_notes.pdf`, 2009. Accessed: 2019-08-03.

[59] Fallaw Sowell. Maximum likelihood estimation of stationary univariate fractionally integrated time series models. *Journal of Econometrics*, 53(1-3):165–188, 1992.

[60] Gábor J. Székely and Maria L. Rizzo. Hierarchical clustering via joint between-within distances: Extending Ward's minimum variance method. *Journal of Classification*, 22(2):151 – 183, 2005.

[61] Marco Tarantola, Anna-Kristina Marel, Eva Sunnick, Holger Adam, Joachim Wegener, and Andreas Janshoff. Dynamics of human cancer cell lines mon-

itored by electrical and acoustic fluctuation analysis. *Integrative biology : quantitative biosciences from nano to macro*, 2:139–50, 03 2010.

[62] Wang Tengyao and Samworth Richard J. High dimensional change point estimation via sparse projection. *JRSS: Series B*, 80(1):57–83, 2018.

[63] Yao Xie, Meng Wang, and Andrew Thompson. Sketching for sequential change-point detection. In *GlobalSIP*, pages 78–82, Dec 2015.

[64] Takashi Yamano. Lecture notes on advanced econometrics, 2009.

[65] Wenyu Zhang, Daniel Gilbert, and David S. Matteson. ABACUS: Unsupervised multivariate change detection via Bayesian source separation. *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 603–611, 2019.

[66] Wenyu Zhang, Nicholas A. James, and David S. Matteson. Pruning and nonparametric multiple change point detection. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 288–295, Nov 2017.

[67] Wenyu Zhang, Devesh K. Jha, Emil Laftchiev, and Daniel Nikovski. Multilabel prediction in time series data using deep neural networks. *arXiv*, 2020.

[68] Qian Zhao, Deyu Meng, Zongben Xu, Wangmeng Zuo, and Lei Zhang. Robust principal component analysis with complex noise. In *ICML - Volume 32*, pages II–55–II–63. JMLR.org, 2014.