UNIVERSITY OF CALIFORNIA,
IRVINE

Statistical Inference of Change Points and Its Applications in Neuroscience Research

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Statistics

by

Tong Shen

Dissertation Committee:
Professor Zhaoxia Yu, Chair
Professor Michele Guindani
Professor Babak Shahbaba

2021

DEDICATION

To my parents.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

First I would like to deeply thank my advisor, Professor Zhaoxia Yu, for her support and excellent guidance during my research in the past few years. Without her, It will be impossible to finish this dissertation. I would like to thank my committee members, Professor Michele Guindani, Professor Babak Shahbaba for their help and kindness. I am also grateful to Professor Hernando Ombao for his collaboration. I would also like to thank Professor Daniel Gillen and other faculties in our department. My thank also goes to Dustin Pluta and Xu gao, who collaborated and helped a lot in my research.

I am grateful to all of my cohorts not listed below at UCI Statistics department for the accompany. I want to thank Yannan Tang, Wei Hu and Yadong Lu for all the chats about life and help; Di Zhang, Fan Yin, Mingwei Tang, Chengcen Sha and Lechuan Hu for all the gatherings and sports times; Tianyu Pan, Tong Zou, Yue Wang and Zhuoran Zhang for the happy hours in my last year at UCI. My special thank goes to Yiwen Dai, Kathy Qin and Siliangyu Cheng for the emotion support and great game times together.

Moreover, I want to thank all of my old friends from my hometown and Zhejiang University. The friendship and good memories always give me energy when I'm lost. Last but not least, I dedicate this dissertation to my family, who have been supporting me for 28 years and always there for me.

CURRICULUM VITAE

Tong Shen

EDUCATION

Doctor of Philosophy in Statistics **2017–2021**
University of California, Irvine *Irvine, California, USA*

Master of Science in Statistics **2015–2017**
University of California, Irvine *Irvine, California, USA*

Bachelor of Science in Statistics **2011–2015**
Zhejiang University *Hangzhou, Zhejiang, China*

RESEARCH EXPERIENCE

Graduate Student Researcher **2017–2021**
University of California, Irvine *Irvine, California, USA*

TEACHING EXPERIENCE

Teaching Assistant **2016–2020**
University of California, Irvine *Irvine, California, USA*

PUBLICATIONS

Tong Shen, Kevin Johnston, Gyorgy Lur, Michele Guindani, Hernando Ombao, Xiangmin Xu, Zhaoxia Yu. Time-varying ℓ_0 optimization for Spike Inference from Multi-Trial Calcium Recordings. *Submitted*, 2021.

Tong Shen, Gyorgy Lur, Xiangmin Xu, Zhaoxia Yu. To deconvolve, or not to deconvolve: inferences of neural activities using calcium imaging data. *Submitted*, 2021.

Jiao, Shuhao, **Tong Shen**, Zhaoxia Yu, and Hernando Ombao. Change-point detection using spectral PCA for multivariate time series. *Unpublished manuscript*, 2021.

Kevin G. Johnston, Steven F. Grieco, Zhaoxia Yu, Suoqin Jin, **Tong Shen**, Rachel Crary, John Guzowski, Todd Holmes, Qing C. Nie, and Xiangmin Xu. Robust Population Single Neuronal Calcium Signal Extraction Using SCOUT Allows for Longitudinal Analysis of Behavior-associated Neural Ensemble Dynamics. *Submitted*, 2020.

Dustin Pluta, **Tong Shen**, Gui Xue, Chuansheng Chen, Hernando Ombao, Zhaoxia Yu. Ridge-Penalized Adaptive Mantel Test and Its Application in Imaging Genetics. *Under*

revision, 2020.

Dustin Pluta, Zhaoxia Yu, **Tong Shen**, Chuansheng Chen, Gui Xue, and Hernando Ombao. Statistical methods and challenges in connectome genetics. Statistics & Probability Letters 136 (2018): 83-86.

ABSTRACT OF THE DISSERTATION

Statistical Inference of Change Points and Its Applications in Neuroscience Research

By

Tong Shen

Doctor of Philosophy in Statistics

University of California, Irvine, 2021

Professor Zhaoxia Yu, Chair

Change point detection is a critical analysis in various scientific fields such as finance, medicine, and climatology. Despite the recent developments in methods and algorithms, it remains challenging in many problems. In this dissertation, we address and apply the detection of change points in two research problems. The first problem was motivated by identifying the epileptic seizure onset time using multi-channel EEG data and detecting abrupt changes in stocks that might characterize major events in the financial market. We propose a change point method using spectral principal component analysis on multivariate time series. By combining multiple time series and allowing for lead-lag relationships, our method achieves not only improved detectability but also more precise estimate of the locations of change points. In the second problem, the goal was to detect the exact time points at which a neuron fires using observed noisy calcium fluorescence recordings. We solve this problem by developing a time-varying ℓ_0 penalized approach to jointly detect spikes using a dynamic change point detection algorithm and estimate firing rates using a Gaussian-boxcar smoother. Our simulated and real studies demonstrate that improved accuracy can be achieved by robustly integrating the evolving neural dynamics within and across recording sessions in a longitudinal setting.

Chapter 1

Introduction

"*The only constant in life is change.*" – Heraclitus.

"*Change alone is eternal, perpetual, immortal.*" – Arthur Schopenhauer

Changes happen everywhere in our daily life. Being able to detect abrupt and significant changes is an important first step toward understanding the mechanisms of various scientific processes. The earliest work, which introduced a cumulative sum method, in change point detection was proposed in the 1950's (Page, 1954). In the following decades, methods of change point detection have been adopted and further developed in many scientific disciplines. Here we present three motivating examples in which the detection of change points is of high relevance in their specific scientific settings.

In the first example, electroencephalogram (EEG) data from a subject with epileptic seizure were recorded for 500 seconds. The signal was acquired through electrodes on the patient's scalp at 100 Hz. Presented in Figure 1.1 is the recording of an example channel among other 30 recorded channels. Visual inspection shows increased fluctuations of EEG values between

300-400s. Accurate detection of seizure-associated changes in brain activity helps us better understand the neurological mechanism and provide early warnings for both patients and caregivers.

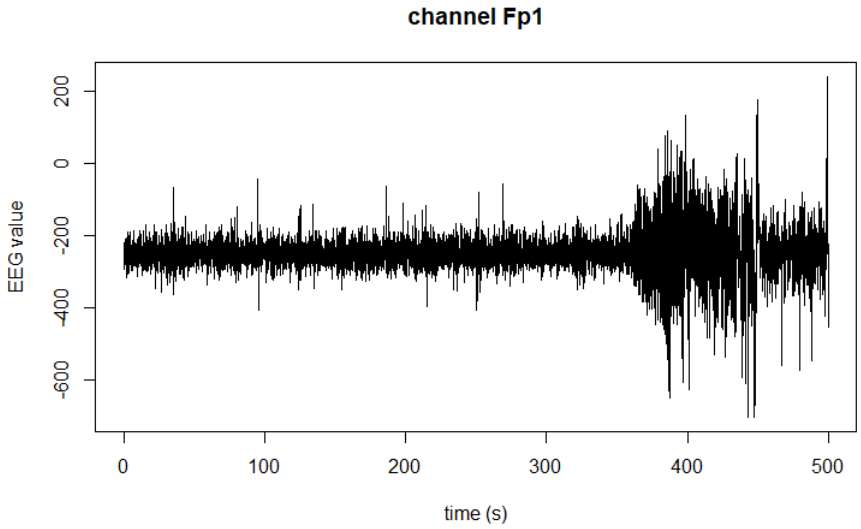

Figure 1.1: EEG recording of a single channel (Fp1, located above the left eye) before and during a spontaneous epileptic seizure of a subject monitored at the epilepsy center of University of Michigan. There are 31 channels in total. The data was sampled at 100 Hz and lasted for 500 seconds.

One difficulty in this problem is how to analyze all the 31 channels jointly to increase statistical efficiency in detecting the onset of epileptic seizure. The problem is further complicated by potential lead-lag relationships between channels. A similar setting is the stock data in the financial market. The stocks in $S\&P$ 100 are the most established companies in the $S\&P$ 500 index, which measures the stock performance of 500 large companies listed on stock exchanges in USA. This index has often been used to forecast the direction of the economy. Figure 1.2 shows the log returns of Bank of America in the past two decades. It is clear that the time series is not stationary. For example, the time series showed abnormal fluctuations at several periods such as the financial crisis around 2008. A highly relevant question is to use multiple stocks to identify these changes.

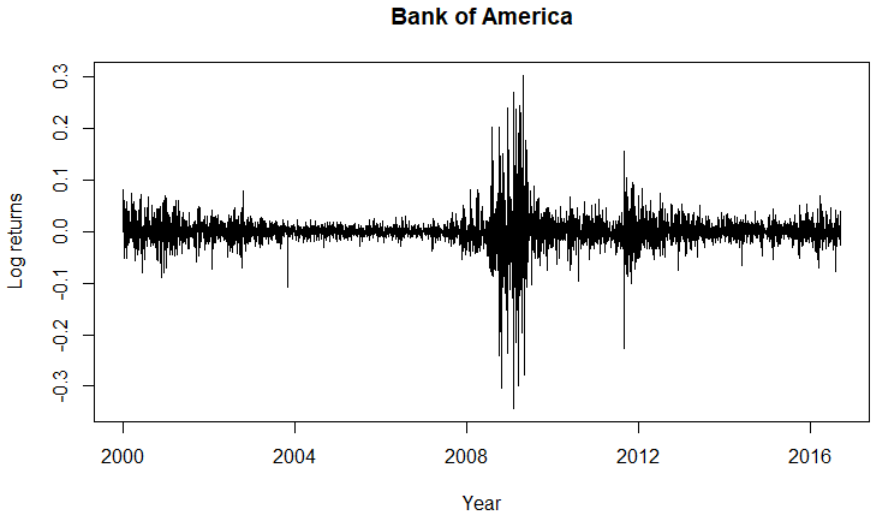

Figure 1.2: Log return of the daily closing values of Bank of America between January 2000 and August 2016.

In imaging analysis, deconvolution methods are often applied to extract useful information from the noisy observed data. The deconvolution problem in estimating the times of action potentials from calcium fluorescence traces can be treated as a special change point problem due to the almost instantaneous rise and slow decay of a calcium transient in the presence of a neural spike. Figure 1.3 shows 21 observed calcium fluorescence traces, recorded at 15 Hz for 2 minutes, of an example neuron from a mouse during a fear conditioning task designed to study the neuron ensembles associated with contextual discrimination. In this figure, 11 traces were obtained during the learning stage (red) and 10 were recorded during the relearning stage (blue). Identifying the neural spike times is a critical but still challenging problem and existing spike detection methods analyze one trace at a time, which ignores the information shared between a trace and those from other sessions.

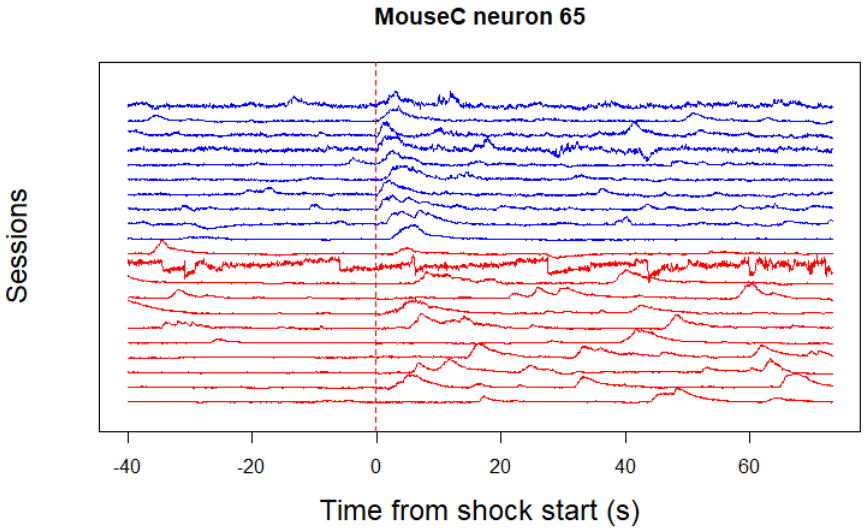

Figure 1.3: Calcium traces of an example neuron in mouse foot shock data. Recorded at 15 Hz for 2 minutes. The red and blue traces were obtained during the learning (first 11 sessions) and relearning stages (last 10 sessions), respectively. The long dashed vertical line denotes the time when a foot shock was applied.

Motivated by the difficulties in change point detection problems and the limitation in current solutions, we develop novel statistical methods to address the above mentioned change point detection problems. The remaining of this dissertation is organized as follows. Chapter 2 introduces the problem of change point detection and provides a review of general solutions. Chapter 3 presents a new spectral principal component method for detecting change points using multivariate time series. Chapter 4 proposes a multi-trial time-varying ℓ_0 penalized algorithm to iteratively detect spikes using a dynamic change point detection algorithm and estimate firing rates using a Gaussian-boxcar kernel. Chapter 5 is a follow up of Chapter 4. This chapter assesses, in problems where calcium traces are often analyzed directly (such as clustering, PCA, and population decoding), the necessity of estimating the underlying spike trains.

Chapter 2

Background on Change Point Detection

2.1 Change Point Problem

The change point problem is often defined as follows. Suppose we have a data series $x(1), x(2), \ldots, x(T)$ with distributions $F_1, F_2, \ldots, F_T$. Then the change point problem can be viewed as testing the null hypothesis $H_0 : F_1 = F_2 = \cdots = F_T$ against the alternative. The simplest case is the single change point problem, with the corresponding alternative hypothesis as $H_1 : F_1 = \cdots = F_{\tau_1} \neq F_{\tau_1+1} = \cdots = F_T$ where τ_1 is the change point. Another useful definition of change points is called epidemic change which can be expressed using the following alternative hypothesis

$$H_1 : F_1 = \cdots = F_{\tau_1} = \alpha \neq F_{\tau_1+1} = \cdots = F_{\tau_2} = \beta \neq F_{\tau_2+1} = \cdots = F_T = \alpha$$

where τ_1, τ_2 are the two change points. In this situation, the distribution of the sequence deviates from baseline and goes back to the initial state. It has been applied in qualify control

and medical applications (Chen and Gupta, 2011; Kirch et al., 2015; Levin and Kline, 1985; Yao, 1993). The more general case is the multiple change point problem which can be defined as:

$$H_1 : F_1 = \cdots = F_{\tau_1} \neq F_{\tau_1+1} = \cdots = F_{\tau_2} \neq F_{\tau_2+1}$$
$$= \cdots F_{\tau_m} \neq F_{\tau_m+1} \cdots = F_T$$

where $\tau_1, \tau_2, \ldots, \tau_m$ are unknown change points to be estimated. In the remaining of this section, we will discuss the application of change point detection, main categories of the approaches and change point detection in multivariate time series in this chapter.

2.1.1 Online and Offline Approaches in Change Point Detection

There are two general categories of change point detection: online change point detection and offline change point detection. In online detection, the approach is sequential as new data points are becoming available. The goal is to identify change point right after it occurs. Literature reviews, including methodological developments and applications in different fields such as biomedicine, economics, and human activity, can be found in Basseville et al. (1993), Chakraborti et al. (2001), Lai (2001). As a comparison, in an office analysis, the entire data is available and the goal is to identifies change points that have already occurred.

In this dissertation we focus on offline change point detection. Next, we review commonly used methods.

2.2 Detection of A Single Change Point

In change point detection, the most basic case is detecting a single change point. Two common types of methods are cumulative sum (CUSUM) and likelihood-ratio based methods.

2.2.1 CUSUM

Cumulative sum (CUSUM) test statistic was first introduced to detect a change in mean in a sequential setting (Page, 1954). The data point where the CUSUM statistic is maximized and exceeds a certain threshold is considered as a candidate change point. The CUSUM statistic for change in mean of time series data, $x(1), \ldots, x(T)$, is defined as

$$\mathcal{C}(x) = \left(\sqrt{\frac{T-b}{Tb}} \sum_{t=1}^{b} x(t) - \sqrt{\frac{b}{T(T-b)}} \sum_{t=b+1}^{T} x(t) \right)$$

where $b \in \{1, \ldots, T\}$ are the candidate change points.

CUSUM has also been extended to different cases. Inclan and Tiao (1994) discussed the change point detection problem in the variance. CUSUM test in auto-regressive model has also been developed: Bai (1994) discussed the change in ARMA model, Kokoszka and Leipus (1999) develops change point detection in the parameters of an autoregressive conditional heteroscedasticity (ARCH) process. We introduce the frequency-specific CUSUM statistic for spectrum in Chapter 3.

2.2.2 Likelihood-Ratio Based Detection

The likelihood-ratio test for change point models was first proposed in Hinkley (1970). In single change point detection, we can define a hypothesis test problem: the null hypothesis of no change point vs the alternative of a single change point. The log likelihood ratio can be

calculated as follows:

$$\begin{aligned}\Lambda &= -2\log\frac{L_{H_0}(\hat{\boldsymbol{\theta}})}{\max_{\tau} L_{H_1}\left(\hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2\right)} \\ &= -2\log\frac{L\left(y_{1:n} \mid \hat{\boldsymbol{\theta}}\right)}{\max_{\tau} L\left(y_{1:\tau} \mid \hat{\boldsymbol{\theta}}_1\right) L\left(y_{\tau+1:n} \mid \hat{\boldsymbol{\theta}}_2\right)}\end{aligned}$$

where $\hat{\boldsymbol{\theta}}$ is the maximum likelihood estimator for the parameters under the null hypothesis, $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$ are for the data before and after the change point τ. The null hypothesis is rejected if $\Lambda > c$ for a threshold value c. Under some regularity conditions, Λ follows asymptotic chi-squared distribution when the null hypothesis is true (Wilks, 1938). However in this setting the change point τ is discrete, which violates the assumptions in Wilk's theorem. Alternatively, Yao and Davis (1986) proposed a more accurate approximation procedure to the null distribution.

The likelihood ratio test has been applied to detect change points in data under different situations. For example, Hinkley (1970) applied this method to normally distributed data. Haccou et al. (1987) applied the test to exponentially distributed data to detect the change in the rate parameter. Kim and Siegmund (1989) considered the test in linear regression and detect changes in the intercept and slope. The likelihood ratio test has also been used in other conditions as binomial and exponential processes (Worsley, 1983, 1986), autoregressive moving-average (ARMA) models (Robbins et al., 2016).

2.3 Multiple Change Point Detection

In this section we extend discussion from single change point detection to multiple change points detection. Ideally we have to consider all combinations of the number and locations of change points. However this problem is very computationally challenging. Computation-

ally efficient approaches have been developed, which include two main algorithms: binary segmentation and dynamic programming. Next, we provide the two algorithms in details.

2.3.1 Binary Segmentation

The Binary Segmentation (BS) algorithm was first introduced in Scott and Knott (1974) and Vostrikova (1981). It has been used in many works to detect multiple change points. (Cho and Fryzlewicz, 2012; Fryzlewicz and Rao, 2014; Venkatraman, 1993). The general binary segmentation algorithm is initiated by setting the interval boundaries to $s = 1$ and $e = T$ with a threshold w_0 and a test statistic $W_{s,e}^{\tau}$ based on the time series $x(s), ...x(e)$ where τ is change point candidate. The optimal threshold depends on the specific change point test statistic and the length of the time series. We will defer the discussion on how to choose it to Chapter 3. The change point test is applied to the subseries before and after the change point. This process is repeated until no more change points are detected. The computation complexity of the algorithm is $\mathcal{O}(T \log T)$. The general algorithm is described below:

Algorithm 1 Binary segmentation

```
function BINARY SEGMENTATION(s,e,w_0)
    if e − s ≥ 2 then
        τ* = arg max_τ W_{s,e}^τ
        if W_{s,e}^{τ*} > w_0 then
            Add τ* to the set of estimated change points.
            Binary segmentation(s,τ*,w_0).
            Binary segmentation(τ* + 1,e,w_0).
        end if
    end if
end function
```

BS conducts change point detection sequentially, which significantly reduces computational complexity. However, because it tests for a change point at a time, small changes in large segments are often undetectable. In the following, we introduce two extensions, namely the circular Binary Segmentation (CBS) and the wild Binary Segmentation (WBS).

The Circular Binary Segmentation was proposed by Olshen et al. (2004). CBS considers connecting the segments at the two ends to form a circle and searches for two change points at a time in an epidemic change. The computational cost of the method is $\mathcal{O}(T^2)$. Venkatraman and Olshen (2007) proposed a faster CBS algorithm with a hybrid approach which calculates the p-values of the test statistic in linear time using a tail probability approximation.

The wild Binary Segmentation (WBS) was proposed by Fryzlewicz et al. (2014). WBS runs their test on multiple subsamples as $x(s), \ldots, x(e)$ where $1 \leq s < e \leq T$, and identify a change point on each subsample. Then the change point with the largest test statistic is chosen as the change point candidate and tested against a threshold value. The process is repeated recursively to the left and right side of the first change point. There is a trade-off between the detecting accuracy and computational efficiency which is also discussed in Fryzlewicz et al. (2014) on how to choose the number of subsamples.

2.3.2 Dynamic Programming

Dynamic programming is another commonly used approach in multiple change point detection. The idea is to solve an optimization problem for segments based on a chosen cost function to each segment and minimizing the total cost. Common choices of cost functions include negative log likelihood, cumulative sums (Page, 1954), residual sum of squares (Inclan and Tiao, 1994; Rigaill, 2015) and minimum description length (Davis et al., 2006; Rissanen, 1989).

There are two ways to formulate the optimization problem. The first is the constrained minimization problem for a fixed number of change points K:

$$\min_{\tau_1,\ldots,\tau_K} \left[\sum_{j=0}^{K} \mathcal{C}\left(x_{(\tau_j+1):\tau_{j+1}}\right) \right]$$

where $\mathcal{C}(\cdot)$. is the cost function, $\tau_1, ..., \tau_K$ are the K change points to be determined. An alternative is to solve a penalized minimization problem where a penalty term is added to each new segment:

$$\min_{k,\tau_1,...,\tau_k} \left\{ \sum_{j=0}^{k} \left(\mathcal{C}\left(x_{(\tau_j+1):\tau_{j+1}} \right) + \lambda \right) \right\}$$

where λ is the penalty value. Since both minimization problems are computationally expensive, several algorithms have been proposed. Auger and Lawrence (1989) developed the Segment Neighbourhood (SN) search algorithm to solve the constrained minimization problem. The computational cost is $\mathcal{O}(KT^2)$. Jackson et al. (2005) developed the Optimal Partitioning (OP) method to solve the penalized minimization problem in a recursive way. The cost function at each time point t is calculated as:

$$F(t) = \min_{0 \leq \tau < t} \left\{ F(\tau) + \mathcal{C}\left(x_{(\tau+1):t} \right) + \lambda \right\}$$

The details of the derivation is included in Chapter 4. The computational cost of the recursions is $\mathcal{O}\left(T^2\right)$. To further improve the efficiency, Killick et al. (2012) proposed Pruned Exact Linear Time (PELT) which is a pruned version of OP. The computational cost can be near $\mathcal{O}\left(T\right)$ under mild conditions. Rigaill (2015) proposed a pruned algorithm based on SN and has linear cost in the single parameter case.

2.4 Change Point Detection in Multivariate Time Series

In real world, multivariate time series is frequently encountered in change point analysis (Aue et al., 2009a; Chen and Gupta, 2011; Cho and Fryzlewicz, 2015; Horváth and Hušková, 2012; Ombao et al., 2005). Change point detection in multivariate time series is very important as the real data are often correlated. By jointly analyzing them, we are able to make full use of

the information in the time series. On the other hand, the analysis in multivariate time series faces many challenges. For example, the computational cost for multivariate time series is much higher than that for univariate time series. This motivated us to develop computation efficient methods, which is presented in Chapter 3.

Chapter 3

Change-point Detection Using Spectral PCA for Multivariate Time Series

3.1 Introduction

Detecting change point in time series is an important goal in many scientific disciplines such as medicine, financial markets, and climate (Scheffer et al., 2009). For instance, much effort has been dedicated to capture the sudden changes in brain due to epilepsy. The analysis of epilepsy helps people understand disruptions in normal brain functioning and develop more precise diagnosis, improve therapy, and develop effective early-warning systems for onset of seizure activity (Schröder and Ombao, 2019). Epileptic seizure prediction has been a notably challenging problem (Mormann et al., 2006). Recent studies showed that prediction might be feasible by analyzing ectroencephalography (EEG) time series (Chisci et al., 2010). These automatic algorithms for seizure detection aim to provide real-time detection and onset

warnings, which may greatly improve clinical and monitoring diagnosis.

In the financial field, Aue et al. (2009a,b) developed rigorous change-point detection methods. They introduced an asymptotic test procedure to assess the stability of volatilities and cross-volatilites. It is of prime importance to detect the potential change in stock market and during financial crisis, which enables people to better understand the relationship between financial events and stocks. (Choudhry, 1996) studied the 1987 financial crisis. Generalized autoregressive conditional heteroskedasticity-in-mean (GARCH-M) model was used to investigate the volatility and changes in volatility for emerging markets before and after the crisis. In Dooley and Hutchison (2009), the subprime crisis circa 2008 was analyzed. A long decline was observed in U.S. equities at the start of subprime crisis in mid-2007 through September 2008. There was also a dramatic change in U.S. stock market in September 2008 with increased volatility after the crisis.

In the literature, both parametric and nonparametric methods have been proposed to detect change points using either univariate or multivariate time series (Chen and Gupta, 2011; Chen et al., 2010; Davis et al., 2006). Parametric models are efficient if the model assumptions are correct. Kirch et al. (2015) developed an approach where a score type of test statistic based on vector autoregressive (VAR) models was used for identifying change points. Nonparametric change points are also used for detecting changes in the spectral characteristics of the time series such as those based on cumulative sum-type tests (Adak, 1998; Ombao et al., 2005; Terrien et al., 2013). For example, Ombao et al. (2005) developed a procedure for segmentation of a non-stationary time series via model selection using the SLEX library which is a collection of orthonormal time-localized Fourier waveforms. One limitation of these methods is they lack the interpretability for frequency as we expect to detect change points for EEG seizure on frequency bands as well. Frequency-specific change point detection can help people gain more insights in brain activity at different frequencies (Alarcon et al., 1995; Blondin and Greer, 2011; Schmitt et al., 2012). Moreover, when we extend methods from univariate to

multivariate time series case, existing methods can be computationally inefficient.

In practice, multivariate time series are often correlated. Analyzing or modeling them jointly, rather than individually or pairwisely, allows us to take the full multivariate information into consideration (Ombao et al., 2005); as a result, multivariate approaches are often more efficient. Besides, in many time series data, spectral analysis is often able to extract critical spectral characteristics of the data by examining different frequency bands (Kaplan et al., 2001). Motivated by these previous work, we develop a two-stage detection method for multivariate time series which we call the Spectral PCA change point (Spec PC-CP) method. The main advantage of our method is that it reduces computational complexity and is more likely to give lower reconstruction error (Wang et al., 2019) especially when there is lead-lag relationship between components of the time series.

The rest of the chapter is organized as follows. In section 3.2 we propose a two-stage Spec PC-CP method and describe three competing methods: Contemporaneous Mixture Method mentioned in Wang et al. (2019), Structural Break Detection (Safikhani and Shojaie, 2017) and Sparsified Binary Segmentation (Cho and Fryzlewicz, 2015). In section 3.3 we conduct simulation studies to evaluate the performance of Spec PC-CP under several simulation settings. In section 3.4, we analyze epileptic seizure EEG data and stock data and show that our method can efficiently detect change points corresponding to the onset of seizure in EEG data and financial issues in stock data. In section 3.5 we conclude this chapter to discuss the potential future directions.

3.2 The Spec PC-CP method

In multivariate analysis, PCA is one of the mostly used techniques to extract linear combinations that provide low dimensional summaries of multivariate data. The first principal

component is given by the linear combination with the largest variance among all possible linear combinations. Each following component has the largest variance under the constraint that it is orthogonal to the preceding components. By extracting a few leading components that account for a desired proportion of the total variance, PCA is often used for visualization and dimension reduction.

Contemporaneous Mixture Method. Consider a zero-mean p-dimensional process $X(t)$. In classical PCA, the PCs as mentioned in Wang et al. (2019) can be represented as an instantaneous linear mixture of the original p-dimensional time series $X(t)$. The calculating procedure is as follows:

- For $X(t)$, compute the eigenvalues-eigenvector pairs of Σ^x as $\{(\lambda_\ell, \mathbf{e}_\ell)\}_{\ell=1}^q$, where $\lambda_1 > \lambda_2 >, ..., > \lambda_q$ and $\|\mathbf{e}_\ell\| = 1$. When Σ^x is not known, we estimate Σ^x by $\hat{\Sigma}^x = \frac{1}{T}\sum_{t=1}^T X(t)X(t)^T$, where T is the length of the series
- The ℓ-th component can be calculated by

$$u_\ell(t) = e_\ell^T X(t)$$

Although widely used in dimension reduction, classical PCA may not be optimal when there is lead-lag relationships between multiple time series. Spectral PCA, which was originally introduced in Brillinger (1964), aims to find components that explain the variance in the spectral density matrix at all frequencies rather than the contemporaneous covariance matrix. Because of the 1-1 relationship between the sequence of auto-covariance matrix at all lags and the spectral density matrix at all frequencies, Spectral PCA is able to capture lead-lag dependence at various lags between the different time series. We will show an example in the simulation section where Spectral PCA gives a better summary of the latent source (Figure 3.3). An additional advantage of the Spectral PCA approach is that it allows us to examine the roles of different frequency bands. Because Spectral PCA has the advantages of both

spectral analysis and PCA, it has been used to analyze multivariate time series data (Ombao and Ho, 2006; Ombao et al., 2005; Stoffer et al., 2002; Wang et al., 2019). In spectral PCA, we summarize a p-dimensional time series $X(t) = (X_1(t), \ldots, X_p(t))'$, by $q(\leq p)$ univariate time series, each of which is a linear combination of all past, present and future observed time series, i.e, a linear filtered version of $X(t)$:

$$u(t) = \sum_{h=-\infty}^{\infty} A(h)X(t-h)$$

Based on $u(t)$, one can construct $X(t)$ by $X(t) = \sum_{h=-\infty}^{\infty} B(h)u(t-h)$. Here $A(h)$ and $B(h)$ are absolutely filters with dimensions $q \times p$ and $p \times q$ respectively. The filters are chosen to minimize the eigenvalues of the approximation error matrix $E[(X(t) - \hat{X}(t))^*(X(t) - \hat{X}(t))]$. Similar to classical PCA, there are up to p components. We use $u_\ell(t)$ to denote the ℓ-th spectral PC. We apply the following algorithm to extract the first q spectral PCs.

Algorithm 2 Spectral PCA Method for Extracting Summaries

function SPECTRAL PCA FOR DATA MATRIX $X(t)$
  for $j = 0, 1, ..., T-1$ **do**
    $d(\omega_j) = T^{-1/2} \sum_{t=1}^{T} X(t) \exp(-2\pi i \omega_j t)$ (Compute Fourier coefficients(FFT)), where $\omega_j = j/T$
    Compute periodogram matrix $I(\omega_j) = d(\omega_j) \cdot d^*(\omega_j)$
    Estimate the spectral density matrix $\widehat{f}(\omega_j) = smooth_r I(\omega_{j+r})$ (use Daniel kernel with window size of length 5)
    Compute the q largest eigenvalues of $\widehat{f}(\omega_j)$ denoted

$$0 < \lambda_q(\omega_j) < \ldots, \lambda_1(\omega_j)$$

    Extract the eigenvectors $V_\ell(\omega_j)$ that correspond to the ℓ-th largest eigenvalues, $\ell = 1, \ldots, q$
  end for
  Compute the filter $b_\ell(h) = \sum_j V_\ell^*(\omega_j) * \exp(2\pi i \omega_j h)$ (compute the filter that correspond to th ℓ-th largest eigenvalue) for $\ell = 1, ..., q$
  Obtain the first q spectral PCs $u_\ell(t) = \sum_{h=-R}^{R} b_\ell(h) * X(t-h)$ for $\ell = 1, ..., q$
return $u(t) = u_1(t), ..., u_q(t)$
end function

In the algorithm of Spectral PCA, we first compute the eigenvectors of the smoothed periodogram (an estimation of spectrum) for each frequency. After that we calculate time-varying filter based on the eigenvectors. Finally by applying the filter to the original data, we are able to get the first q Spectral PCs, which are low dimensional summaries. The low dimensional summaries $u_\ell(t)$ can be thought as the frequency domain analog of the ℓ-th principal component in classical PCA. In this article, we calculate the first three spectral PCs ($\ell = 1, 2, 3$) in the algorithm to conduct change point analysis.

3.2.1 Stage II: Identifying change-points from a Spectral PC using the binary segmentation algorithm

Spectral PCA was proposed to conduct dimension reduction for stationary time series (Brillinger, 1981; Shumway and Stoffer, 2017). Time series with potential change points, such as EEG and stock data collected over a long period time, are naturally non-stationary. Thus, it is necessary to consider segmentation methods when identifying change points. For example, the SLEX library proposed by Ombao et al. (2005) consists of localized waveform where localization is achieved by binary segmentation so that each time block is approximately stationary.

When combined with binary segmentation, the CUSUM statistics can consistently detect multiple change-points in a recursive manner (Cho and Fryzlewicz, 2012). In this chapter, the CUSUM statistic is applied to the spectrum of the time blocks. For each time block, we first compute the smoothed periodogram of the ℓ-th spectral PC $u_\ell(t)$ (See algorithm 2(4,5,6)) Similar to Schröder and Ombao (2019), we let $\hat{f}(\tilde{t}, \omega_j)$ denote the estimate of the spectrum of $u_\ell(t)$ within the $\tilde{t}$-th block at frequency ω_j. Figure 3.1 shows an example of a multivariate time series ($T = 1,000$ and $p = 20$ channels) with two change points at $t = 400$ and 700. The time series is a mixture of AR(2) latent sources with different frequency bands. Between the two change points, the weight of the alpha band (8-12 Hz) is lower while the weights of delta (0-4 Hz) and gamma (30-45 Hz) band remain constant across all time blocks. The estimated spectrum of the first spectral PC is calculated at each time block with length of 100. It is clear that the estimates agree with the latent source. Two change points are detected using the first spectral PC, which agrees with the truth.

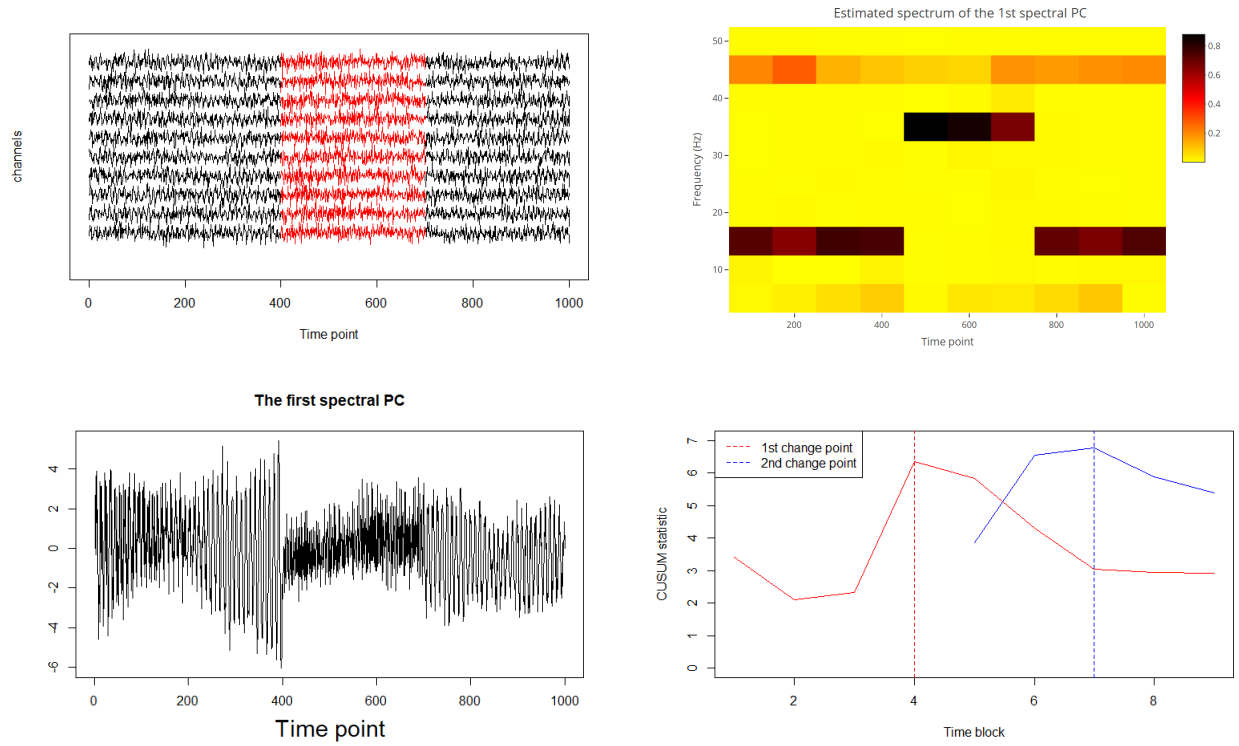

Figure 3.1: Topleft: Time series with $T = 1,000$ and $p = 20$ channels, two change points at $t = 400$ and 700. Topright: The time-frequency plot. Presented are estimated spectrum of the first spectral PC at each time block ($blocklength = 100$). Bottomleft: The first spectral PC and estimated change points. Bottomright: CUSUM statistics using binary segmentation (Red line: first change point. Blue line: second change point.)

Suppose that the entire time has been split into $\tilde{T}$ blocks. The CUSUM statistic on the $\tilde{t}$-th block is defined as

$$\mathcal{C}_{\tilde{t}} = \sum_{\omega_j} \mathcal{C}_{\tilde{t}}(\omega_j) * \mathbb{I}(\mathcal{C}_{\tilde{t}}(\omega_j) > \zeta_T) \tag{3.1}$$

where

$$\mathcal{C}_{\tilde{t}}(\omega_j) = |\sqrt{\frac{\tilde{T}-\tilde{t}}{\tilde{T}\tilde{t}}} \sum_{i=1}^{\tilde{t}} \hat{f}(i,\omega_j) - \sqrt{\frac{\tilde{t}}{\tilde{T}(\tilde{T}-\tilde{t})}} \sum_{i=\tilde{t}+1}^{\tilde{T}} \hat{f}(i,\omega_j)|/\hat{\sigma}(\hat{f}(\omega_j)) \tag{3.2}$$

is the CUSUM statistic at frequency ω_j. Here, $\zeta_T = 0.8\log_{1.1}(T)$ is a threshold, whose the theoretical justification has been provided by Schröder and Ombao (2019). $\hat{\sigma}(\omega_j) =$

$\frac{1}{\tilde{T}} \sum_{l=1}^{\tilde{T}} \hat{f}(l, \omega_j)$ is a scaling factor.

In the binary segmentation CUSUM algorithm, for each time block we calculate the periodogram series $\hat{f}(\tilde{t}, \omega_j)$ at all frequency ω_j based on a spectral PC $u_\ell(t)$. Then we identify the first candidate change-point that gives the largest discrepancy of CUSUM statistics. Once a change point is detected at time block t_0 $(1 < t_0 < \tilde{T})$, we repeat our procedure on sub-intervals split by t_0. The following table provides the pseudo code of our algorithm:

Algorithm 3 Binary segmentation CUSUM

Input: Periodogram series $\hat{f}(\omega_j) = (\hat{f}(1, \omega_j), ..., \hat{f}(\tilde{T}, \omega_j))$, start and end point of the series, threshold ζ_T.

function BINARY-CUSUM($\hat{f}(\omega_j)$,$start$,end,ζ_T)
  if $end - start \geq 2$ **then**
    for Block $\tilde{t} = 1, 2, ..., \tilde{T}$ **do**
      Calculate $\mathcal{C}_{\tilde{t}}$ based on $\hat{f}(start, \omega_j), ..., \hat{f}(end, \omega_j)$ using (1), (2)
    end for
    $t_0 = \arg\max_{\tilde{t} \in [start, end]} \mathcal{C}_{\tilde{t}}$
    if $\mathcal{C}_{t_0} > \zeta_T$ **then**
      Add t_0 to the set of estimated change points.
      Binary-CUSUM($\hat{f}(\omega_j)$,$start$,t_0,ζ_T).
      Binary-CUSUM($\hat{f}(\omega_j)$,$t_0 + 1$,end,ζ_T).
    else
      Break.
    end if
  end if
end function

Our two-stage Spec PC-CP method is summarized in Figure 3.1. In the first stage, we partition the whole time series into time blocks of length 100 and compute the first spectral PC at each time interval. In the second stage we compute the periodogram at each time block using the PCs and conduct change point detection using the binary segmentation CUSUM procedure.

3.2.2 Other methods for comparison

Besides Contemporaneous Mixture Method, we also compared our method with two other existing methods, namely Structural Break Detection (Safikhani and Shojaie, 2017) and Sparsified Binary Segmentation (Cho and Fryzlewicz, 2015). Both of the methods aim to detect change points in multivariate time series. The Structural Break Detection method is a parametric method based on a piecewise vector autoregressive model (VAR) with a regularized estimation using the total variation LASSO penalty. The Sparsified Binary Segmentation method is a non-parametric method that aggregates the CUSUM statistics with a sparsifying step that reduces the effect of noisy contributions. To reduce the impact of the channels with no changes, it applies a threshold to the CUSUM statistics for each single channel and aggregates only the CUSUMs that exceeds the threshold.

3.3 Simulation Studies

3.3.1 Generate the latent sources

In this section, we present simulation studies to examine the performance of our proposed Spec PC-CP method and three existing methods. We consider several methods for generating latent source. To obtain realistic signals, we follow the approach in Gao et al. (2016) to generate latent sources using the AR(2) model. As demonstrated in Gao et al. (2016), the AR(2) model is able to produce signals with the desired oscillatory properties. Following the convention for analyzing electroencephalograms (EEGs), we generated sources with localized power in the following frequency bands: delta (0-4) Hz, theta (4-8) Hz, alpha (8-12) Hz, beta (12-30) Hz, gamma (30-45) Hz. The synthetic signals are generated as follows:

- We first generate multivariate AR(2) latent sources using $\mathbf{S}(t) = \Phi_1\mathbf{S}(t-1)+\Phi_2\mathbf{S}(t-2)+$

$\eta(t)$, where $\Phi_1, \Phi_2 \in \mathbb{R}^{k\times k}$ are diagonal matrices and noise $\eta(t)$ is independent Gaussian to guarantee independence of the sources, $\eta(t)$ are Gaussian noise and each column follows $N(\mathbf{0_{k\times 1}}, \mathbf{I}_k)$. Note that we consider five (k = 5) different latent AR(2) processes that represent different EEG oscillations at the five frequency bands respectively. The coefficients for each of the AR(2) component are listed in table 1.

Table 3.1: AR(2) Coefficients for each frequency band

Frequency band	Diagonal value of Φ_1	Diagonal value of Φ_2
Delta	1.998	-0.998
Theta	1.9	-0.998
Alpha	1.616	-0.998
Beta	-0.617	-0.998
Gamma	-1.616	-0.998

- We then generate the p-dimensional signals from k latent sources as follows:

$$X(t) = M\mathbf{S}(t) + \epsilon(t)$$

Each column of the noise term $\epsilon(t)$ follows $N(\mathbf{0_{p\times 1}}, \mathbf{I}_p)$. In the simulation study, we are interested in multiple factors that might affect the results, such as different fractions of channels with changes, the number of change points, different types of latent source, the change point locations and different block lengths. Figure 3.2 shows the latent source in theta band and alpha band. The sampling rate is 100 Hz. By using AR(2) model, we can constrain the power of sources to center at given frequency bands.

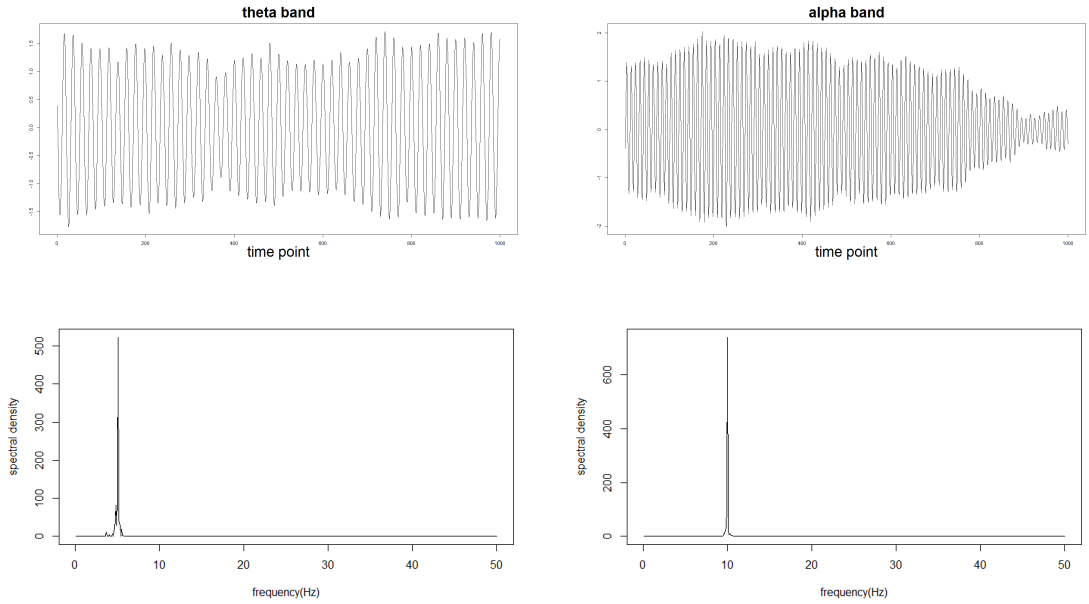

Figure 3.2: The series and spectra of the process in theta band and alpha band. Sampling rate: 100 Hz. (Top: series, bottom: spectra)

Summary of Simulation Settings

To provide sensitivity analysis on our simulation, we used a variaty of simulations settings, which are summarized as follows:

- The length of time series is $T = 1,000$ with 1 change point(at $t = 550$) or $T = 4,000$ with 3 change points(at $t = 980; 2,120; 3,050$).
- The number of channels with change points (out of total of p=128) is 16, 32, 64.
- The type of latent source is AR(2) or a mixed source of ARMA(1,1), MA(1) and AR(2).
- The first three spectral principal components that correspond to the first three largest eigenvalues from Spectral PCA.

3.3.2 Scenario I: Proportions of channels with a change point

First we evaluate the case where we have AR(2) latent source, 1 change point at $t = 550$ and $T = 1,000$. We assume that changes mainly in the gamma band (the 5-th column of the weight matrix) and add time lag with length 10 or 15 on $X(t)$ to create lead-lag relationship. Under such settings we expect the Spectral PCA method could better summarize the time series because of its ability to extract the latent source and capture the dynamics of the source with time lag. A total of $p = 128$ channels will be generated. The following numbers of channels with change points are considered: 16, 32, 64.

$$M_{10}^{(1)} = \begin{pmatrix} 0.1 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & 0.1 \\ & & \vdots & & \\ 0.1 & 0.1 & 0.1 & 0.1 & 0.1 \end{pmatrix} \quad M_{15}^{(1)} = \begin{pmatrix} 0.2 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0.2 & 0.1 & 0.1 & 0.1 & 0.1 \\ & & \vdots & & \\ 0.2 & 0.1 & 0.1 & 0.1 & 0.1 \end{pmatrix}$$

$$M_{10}^{(2)} = \begin{pmatrix} 0.1 & 0.1 & 0.1 & 0.1 & 0.9 \\ 0.1 & 0.1 & 0.1 & 0.1 & 0.9 \\ & & \vdots & & \end{pmatrix} \quad M_{15}^{(2)} = \begin{pmatrix} 0.3 & 0.1 & 0.1 & 0.1 & 0.9 \\ 0.3 & 0.1 & 0.1 & 0.1 & 0.9 \\ & & \vdots & & \end{pmatrix}$$

$$X(t) = \begin{cases} M_{10}^{(1)} S(t-10) + M_{15}^{(1)} S(t-15) + \epsilon(t) & (0 < t <= 550) \\ M_{10}^{(2)} S(t-10) + M_{15}^{(2)} S(t-15) + \epsilon(t) & (550 < t <= 1,000) \end{cases}$$

where each column of $\epsilon(t)$ follows $N(0_{k\times 1}, I_k)$.

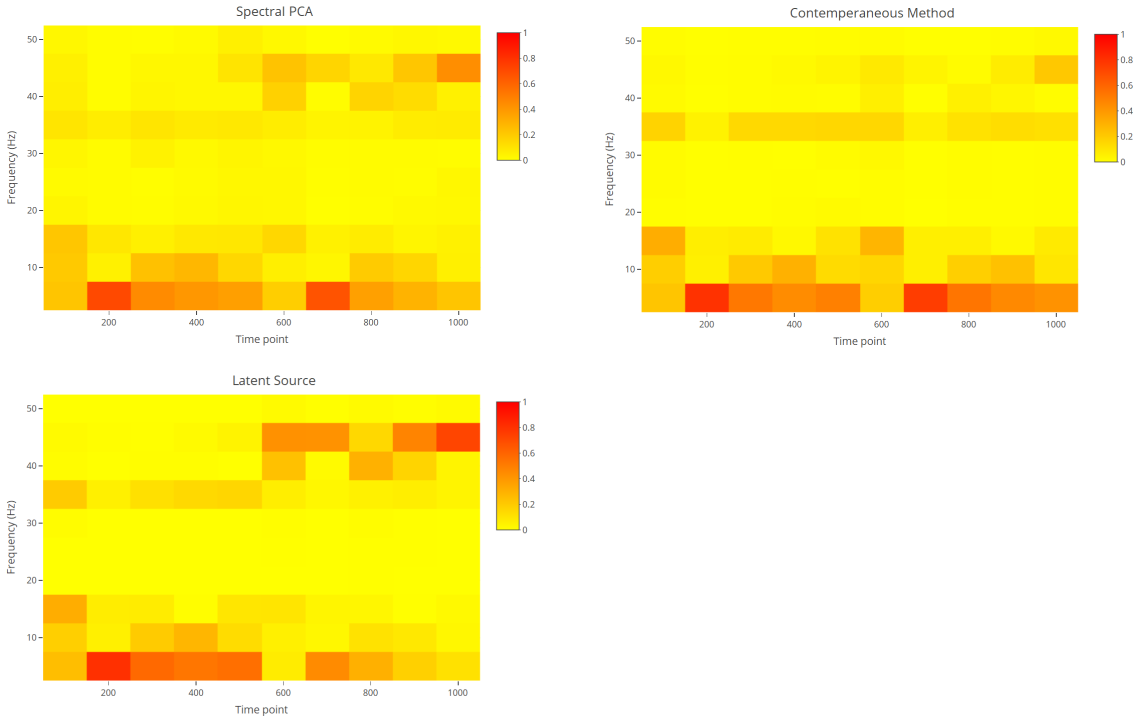

Figure 3.3: The Time-Frequency plots of the summarization series (top left: the first Spectral PC, top right: the first component of Contemporaneous Mixture, bottom: the latent source. Total time points $T = 1,000$, 16/128 channels with change point)

As the Time-Frequency plots shown in figure 3.3, Spectral PCA has higher weights in high frequencies after the change point and lower weights in other frequency bands. As a comparison, the contemporaneous method has higher weights in lower frequencies. This indicates that Spectral PCA gives a better summary than Contemporaneous Method.

Table 3.4 gives the simulation results. Among the three leading spectral PCs, the first PC performs the best. As the number of channels with change increases, the detection rate increases in all methods. Spec PC-CP and Contemporaneous Method have comparable performance when there are a large number of channels with change points. The advantage of our Spec PC-CP method is particularly substantial when the fraction of channels with change points is small (16 channels). Besides the detection rate, our method is more accurate in locating the change point, as qualified in mean absolute difference (MAD) between the estimated change points and true ones (Figure 3.4 and Table 3.4). The change points detected

by Spec PC-CP are closer to the true change points ($t = 550$) than contemporaneous method, especially when there is a small proportion of channels with change points. To confirm the robustness of our method against simulation methods, we also run simulations under Structural Break and Binary Segmentation settings. Table 3.2 and 3.3 show the results. In these settings there is one change point and the Spec PC-CP method gives comparable results in terms of detection rate and slightly higher mean absolute difference.

- Structural Break setting: We have a piecewise stationary time series model with a single change point at $t = 500$. At each time point t, $X(t)$ is a $k \times 1$ vector. $\epsilon(t) \sim N(0, 0.01I_k)$. We have $p = 10$ in this case.

$$X(t) = \begin{cases} 0.9X(t-1) + \epsilon(t) & (0 < t <= 500) \\ -0.9X(t-1) + \epsilon(t) & (500 < t <= 1,000) \end{cases}$$

Table 3.2: Summary of simulation results using Structural Break setting. The length of time series lengths is chosen as $T = 1,000$. There is one change point at $t = 500$.

	Detection rate	MAD
Spec PC CP	0.97	20.1
Contemporaneous Method	0.96	18.6
Structural Break	1	1.14
Cho's Method	0.93	2.05

- Binary Segmentation setting: We have a piecewise stationary time series model with a single change point at $t = 500$ with $\alpha \sim Uniform(0.5, 0.59)$ and $\beta \sim Uniform(-0.79, -0.5)$. At each time point t, $X(t)$ is a $k \times 1$ vector. $\epsilon(t) \sim N(0, 4I_k)$. We have $k = 100$ in this case.

$$X(t) = \begin{cases} \alpha X(t-1) + \epsilon(t) & (0 < t <= 500) \\ \beta X(t-1) + \epsilon(t) & (500 < t <= 1,000) \end{cases}$$

Table 3.3: Summary of simulation results under Cho's setting. The length of time series lengths is chosen as $T = 1,000$. There is one change point at $t = 500$.

	Detection rate	MAD
Spec PC CP	0.99	21.2
Contemporaneous Method	0.98	22.3
Structural Break	0.34	442.0
Cho's Method	0.91	51.5

3.3.3 Scenario II: Multiple Change Points

Next we conduct simulations with multiple change points. The length of time series lengths are chosen as $T = 4,000$, and the three change points are at $t = 980; 2,150; 3,020$.

$$X(t) = \begin{cases} M_{10}^{(1)} S(t-10) + M_{15}^{(1)} S(t-15) + \epsilon(t) & (0 < t <= 980) \\ M_{10}^{(2)} S(t-10) + M_{15}^{(2)} S(t-15) + \epsilon(t) & (981 < t <= 2,150) \\ M_{10}^{(1)} S(t-10) + M_{15}^{(1)} S(t-15) + \epsilon(t) & (2,151 < t <= 3,020) \\ M_{10}^{(2)} S(t-10) + M_{15}^{(2)} S(t-15) + \epsilon(t) & (3,021 < t <= 4,000) \end{cases}$$

where each column of $\epsilon(t)$ follows $N(0_{k\times 1}, I_k)$.

The performance of the four methods are summarized in Table 3.5. Again, the Spec PC-CP method performs the best. In particular, our method outperforms the other methods when there are low or moderate number of channels (16 and 32). When the true number of change point is one, we used detection rate as one metric to compare the performance of different methods. Since there are multiple change points in this setting, it is more reasonable to compare the detection proportion, which is the number of detected change points over total number of change points. The Structural Break Method has about 0.33 detection proportion and only detects 1 change point at the very end of the series in most cases. The Sparsified Binary Segmentation gives lower detection proportion around 0.4 with higher bias. Figure 3.5 shows the estimated locations for Spec PC-CP and contemporaneous method. Change points detected by our Spec PC-CP are closer to the true change points. Moreover,

most of the estimated change points are in a small neighborhood of the true locations $(t = 1,000; 2,100; 3,100)$.

3.3.4 Scenario III: Mixed sources

In this setting, we considered mixed latent sources with ARMA(1,1), MA(1) and AR(2) sources.

$$X(t) = \begin{cases} M_{10}^{(1)} S(t-10) + M_{15}^{(1)} S(t-15) + \epsilon(t) & (0 < t <= 550) \\ M_{10}^{(2)} S(t-10) + M_{15}^{(2)} S(t-15) + \epsilon(t) & (550 < t <= 1,000) \end{cases}$$

where each column of $\epsilon(t)$ follows $N(0_{k\times 1}, I_k)$. Here $S(t)$ is a mixed latent source with ARMA(1,1), MA(1) and AR(2). The series lengths are $T = 1,000$ and there is one change point at $t = 550$.

The results are summarized in Table 3.6 and Figure 3.6. The spec PC-CP method using the first or second spectral PC outperform contemporaneous and structural break method when there are less channels (16) with change points and gives comparable results when mthe number of channels with change points is moderate or large. Our method has detection rates mostly from 0.6 to 0.9 in this case. Although structural break method has moderate detection rates as well, it tends to overestimate the number of change points and has inferior performance in terms of the estimated locations. The Sparsified Binary Segmentation has inferior performance with around 0.3 detection rate. Besides, The locations of change point detected in Figure 3.6 of our method are mostly near the truth, which shows that Spec PC-CP also outperforms contemporaneous method under this setting. The results indicates that our method is robust with different type of sources.

Table 3.4: Summary of simulation results for the AR(2) model. Time series length is $T = 1,000$. There is one change point at $t = 550$.
Presented are the detection rates and the mean absolute distances (MADs) between the estimated and true change point.

		Detection Rate			MAD		
		16/128	32/128	64/128	16/128	32/128	64/128
1st component	spec PC CP	**0.92**	**0.98**	0.96	51.8	52.0	52.1
	Contemporaneous Method	0.72	0.92	0.97	54.7	53.3	50.9
2nd component	spec PC CP	**0.87**	0.90	0.91	61.2	56.2	52.2
	Contemporaneous Method	0.66	0.90	0.92	59.3	55.3	53.1
3rd component	spec PC CP	**0.75**	0.77	0.84	63.3	65.6	60.9
	Contemporaneous Method	0.65	0.81	0.92	56.4	51.4	53.4
	Structual Break	0.34	0.77	0.62	439.4	440.1	439.1
	Sparsified Binary Segmentation	0.22	0.29	0.44	39.6	30.4	26.4

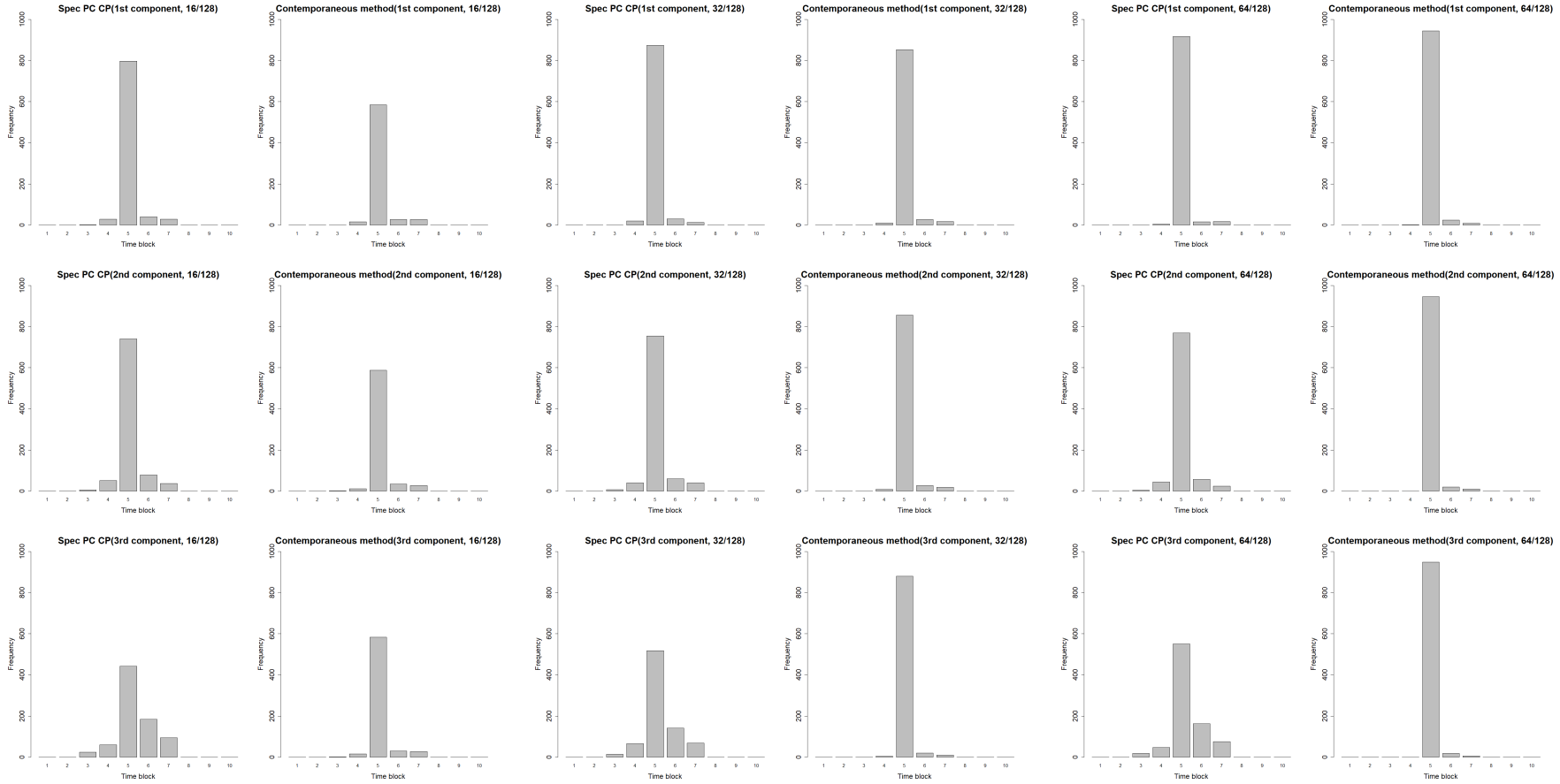

Figure 3.4: Proportion of being detected as change point(out of 1,000). From left to right: 16,32 and 64 channels with change point, respectively. From top to bottom: using the first, second, or third spectral PC.

Table 3.5: Summary of simulation results for multiple change point case. Time series length is $T = 4,000$. There is one change point at $t = 980; 2,150; 3,020$.
Presented are the detection rates and the mean absolute distances (MADs) between the estimated and true change point.

		Detection proportion			MAD		
		16/128	32/128	64/128	16/128	32/128	64/128
1st component	Spec PC CP	**0.71**	**0.84**	0.95	885.1	808.3	321.4
	Contemporaneous Method	0.46	0.73	0.96	1195.8	1076.7	443.5
2nd component	Spec PC CP	**0.63**	**0.69**	0.72	544.4	584.5	432.7
	Contemporaneous Method	0.45	0.63	0.72	610.0	598.1	525.7
3rd component	Spec PC CP	**0.56**	**0.61**	0.66	633.2	623.8	346.7
	Contemporaneous Method	0.45	0.56	0.64	732.4	706.3	413.2
	Structural Break	0.33	0.33	0.37	975.8	975.7	975.4
	Sparsified Binary Segmentation	0.43	0.42	0.43	1511.9	1576.0	1605.0

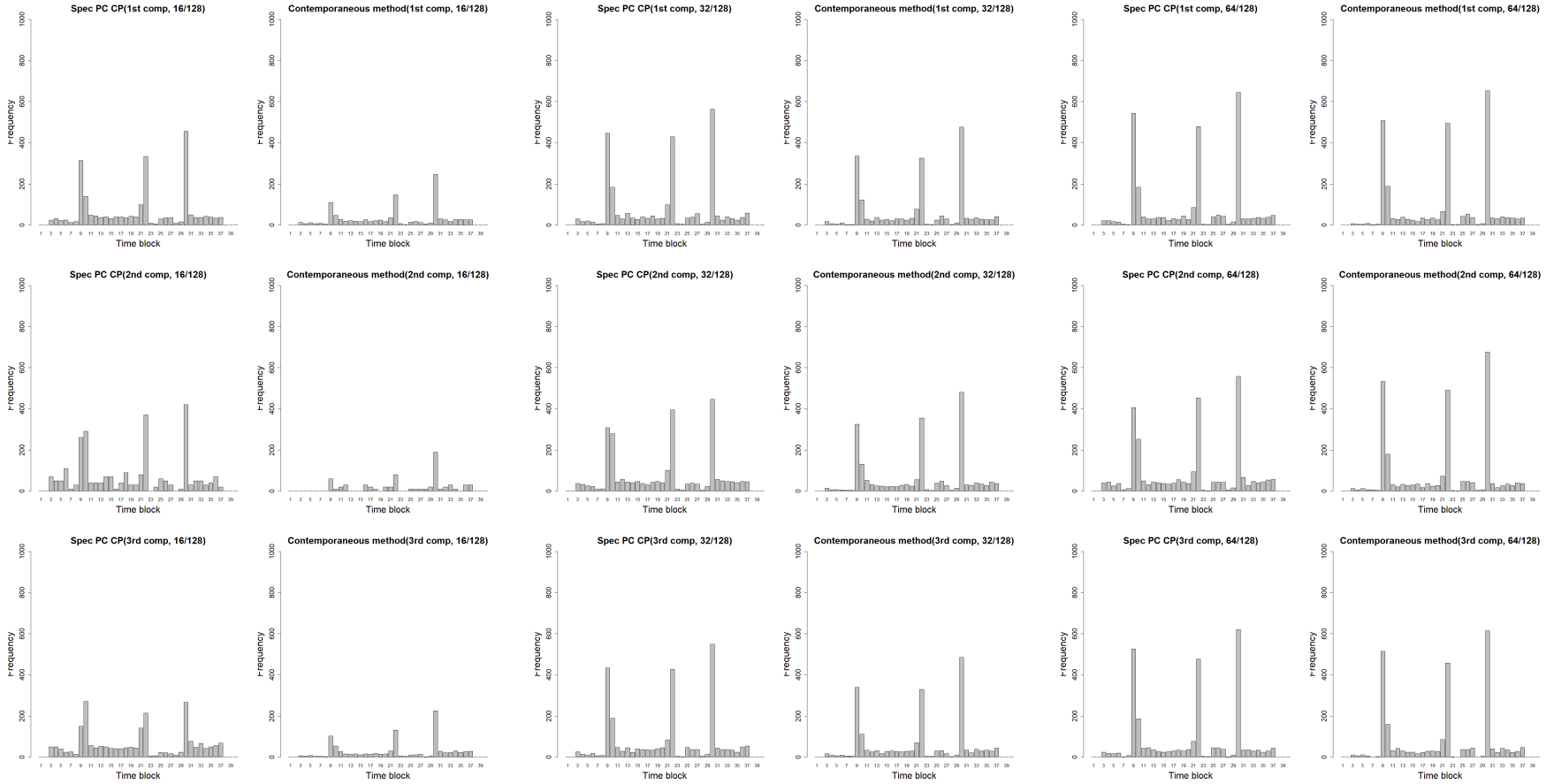

Figure 3.5: Proportion of being detected as change point(out of 1,000). From left to right: 16,32 and 64 channels with change point, respectively. From top to bottom: using the first, second, or third spectral PC.

Table 3.6: Summary of simulation results for the mixed source simulation method. Time series length is $T = 1,000$. There is one change point at $t = 550$.
Presented are the detection rates and the mean absolute distances (MADs) between the estimated and true change point.

		Detection Rate			MAD		
		16/128	32/128	64/128	16/128	32/128	64/128
1st component	Spectral PCA	**0.66**	0.82	0.90	63.6	52.0	54.0
	Contemporaneous Method	0.49	0.80	0.92	70.9	54.2	105.4
2nd component	Spectral PCA	**0.83**	0.88	0.91	57.8	56.6	56.7
	Contemporaneous Method	0.52	0.79	0.92	76.9	58.1	55.1
3rd component	Spectral PCA	**0.68**	0.72	0.91	61.8	61.0	55.6
	Contemporaneous Method	0.52	0.80	0.93	76.9	57.5	54.2
	Structual Break	0.69	0.63	0.23	439.7	439.7	439.1
	Sparsified Binary Segmentation	0.24	0.28	0.36	10.2	18.0	21.5

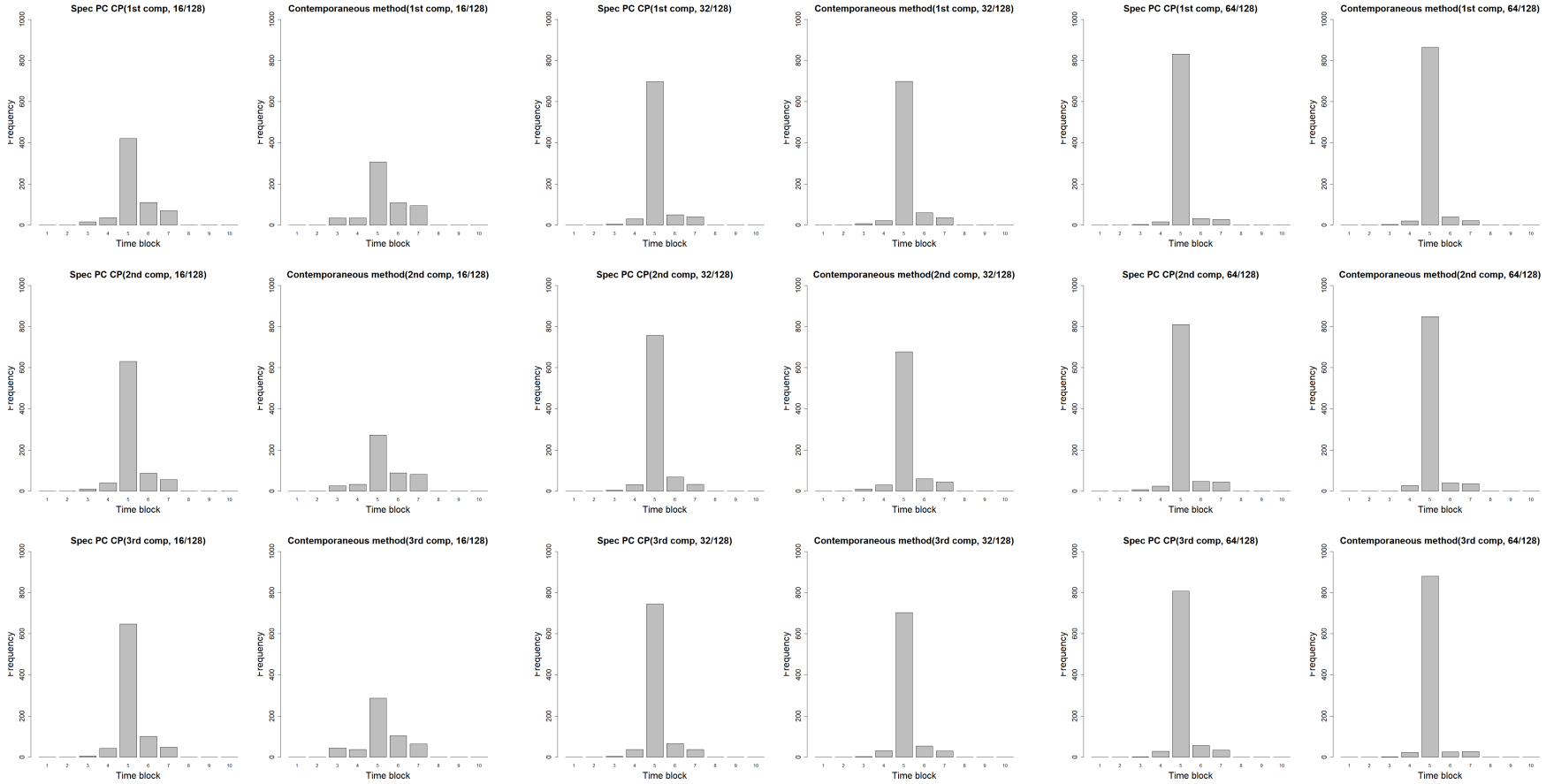

Figure 3.6: Proportion of being detected as change point(out of 1,000). From left to right: 16,32 and 64 channels with change point, respectively. From top to bottom: using the first, second, or third spectral PC.

3.4 Data Analysis

3.4.1 EEG Data

We applied our two-stage change point detection method to two data sets. The first is a seizure recording which captured brain activity of a subject monitored at the epilepsy center at the University of Michigan. The EEG data was sampled at 100 Hz and lasted for about 500 seconds with a total length of 50,000 (Schröder and Ombao, 2019). The data was recorded at 31 channels. The placement of the scalp electrodes is illustrated in Figure 3.7. The abbreviation correspond to different locations on the scalp. For example, Fp means frontal polar and C means central. Figure 3.8 shows the time series for six channels where the seizure happening time are similar among different channels.

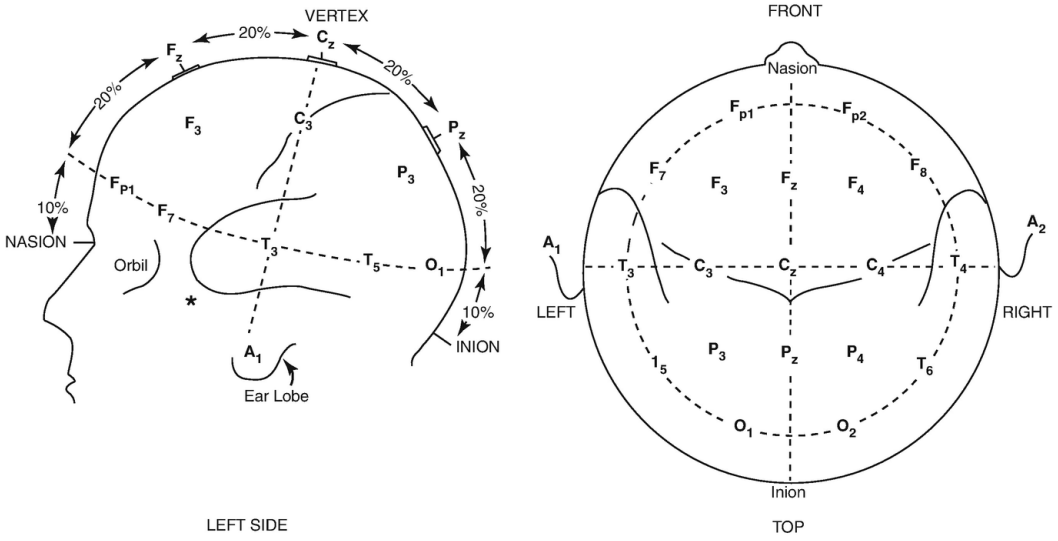

Figure 3.7: EEG scalp topography. According to the 10-20 system and two sphenoidal electrodes placed at the base of the temporal lobe.

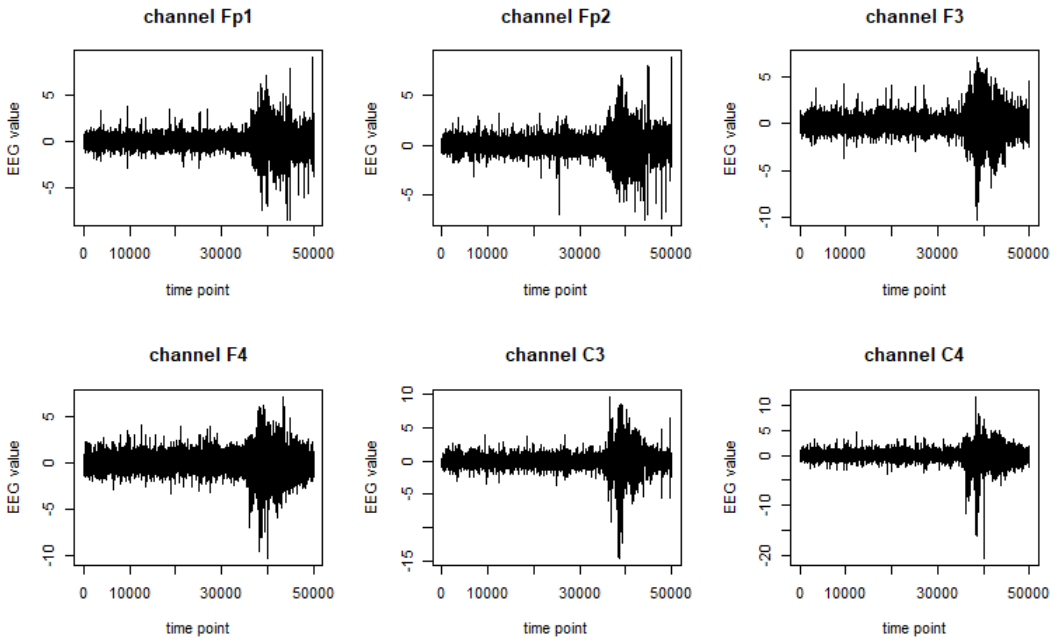

Figure 3.8: Six channels of the EEG seizure data (channel Fp1, Fp2, F3, F4, C3, C4). The data was sampled at 100 Hz and lasted for 500 seconds.

We first examine the time-frequency plots of the first three spectral PCs (Figure 3.9) where darker color indicates higher weights at certain frequency and time points. It has been known that the low-frequency energy is more varying during epileptic seizure (Schröder and Ombao, 2019). Therefore, it is of interest to use delta band to detect change points.

Figure 3.10 shows the detected change points using the first three spectral PCs. The red dotted lines denote the location of estimated change points in delta band. We have detected a few pre-ictal change points in all three components at the beginning of the recording. The first component captures multiple change points while the second and third component each detects one change point before seizure took place. The result suggests that it might be possible to build early warning systems on seizure. The other group of change points is identified right before and during seizure onset. All of the three components capture several changes around $t = 40,000$, which agree with visual inspection on the frequency-time plot and the findings of neurologists. This demonstrates that our method can detect epileptic

seizure well on this data.

To understand how many spectral PCs should be analyzed, we plotted the variance proportion plot (Figure 3.10), which shows that the first two components explain over 90% of the total variance. Thus, analyzing the first three principal component is adequate for this particular dataset.

We also conduct analysis using the other comparison methods. Figure 3.11 shows the results using Contemperaneous Method and it did not detect the pre-ictal change points using the first two components. In Figure 3.12 it can be seen that the Structural Break Method can only detect a change point at the end of the series and the Sparsified Binary Segmentation cannot detect the pre-ictal change point either.

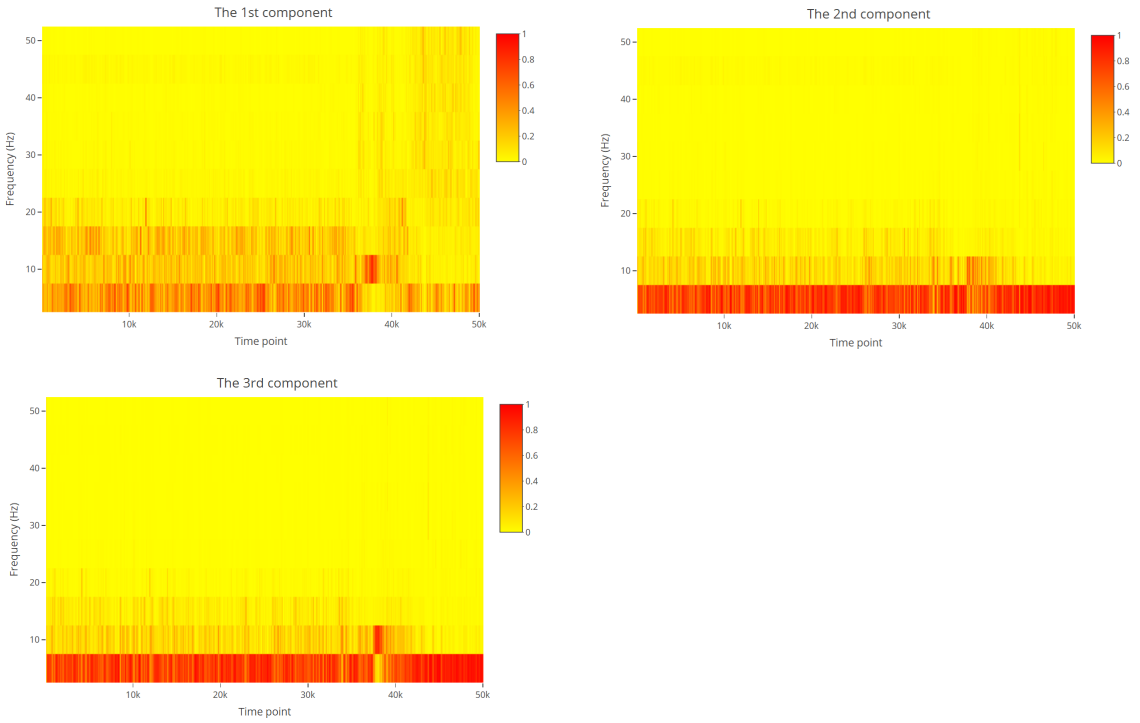

Figure 3.9: The time-frequency plot of the first three spectral PCs. The x-axis denotes the time; the y-axis denoted the frequency from 0 to 50 Hz.

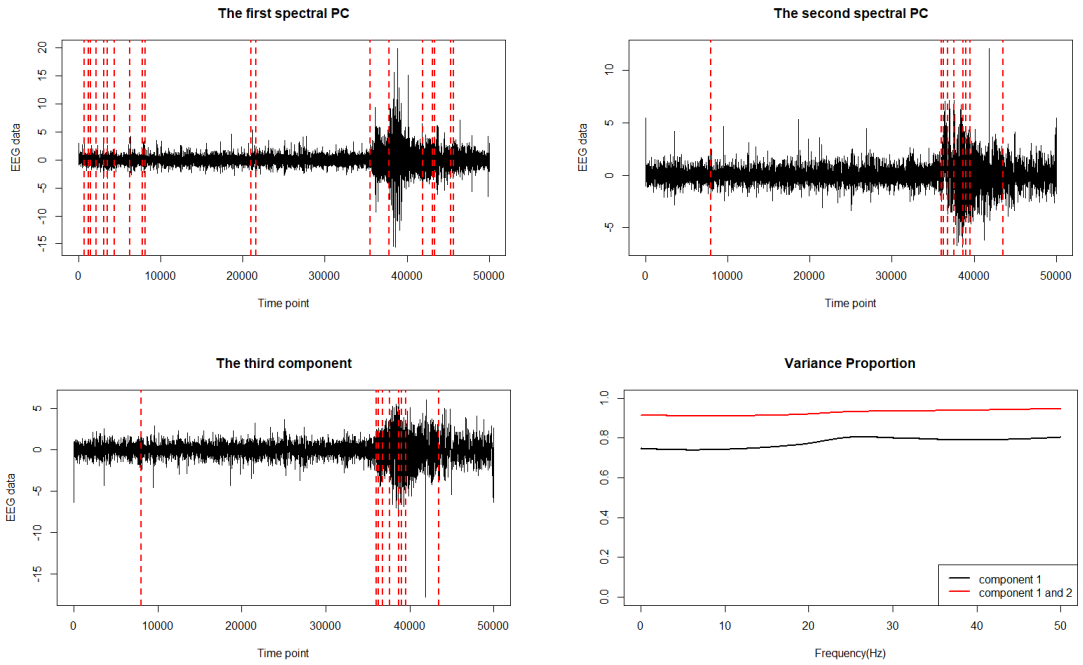

Figure 3.10: The estimated change point using the first, second or third spectral PC. The red dotted lines denote the locations of estimated change points. Bottom right: The variance explained by the first component only and first and second components together.

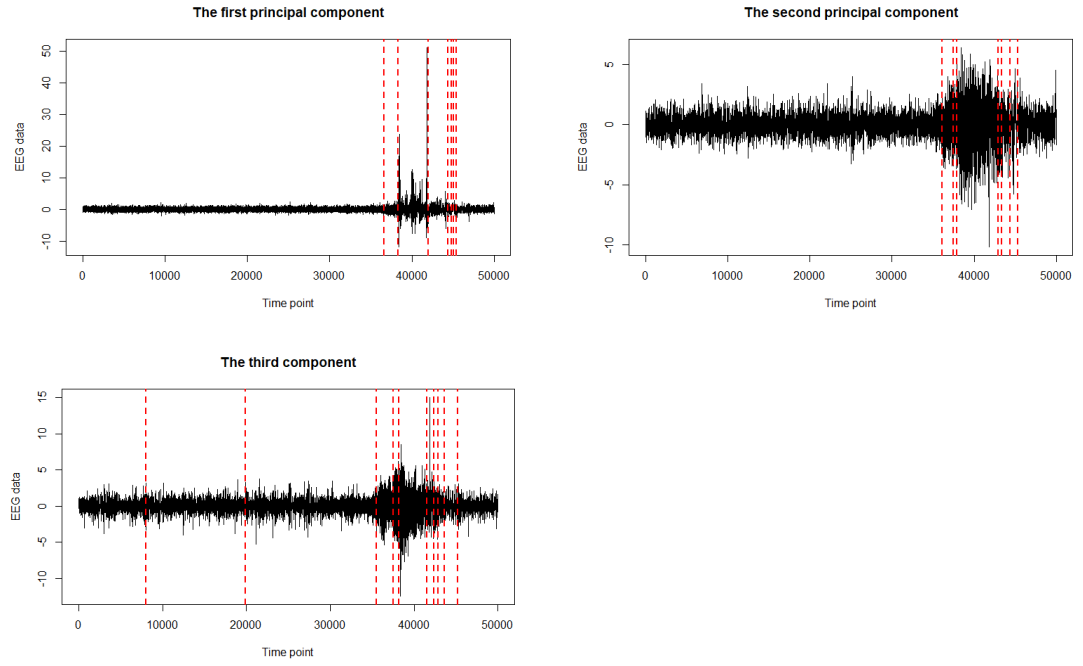

Figure 3.11: The estimated change point using the first, second or third component of Contemperaneous Method. The red dotted lines denote the location of estimated change points.

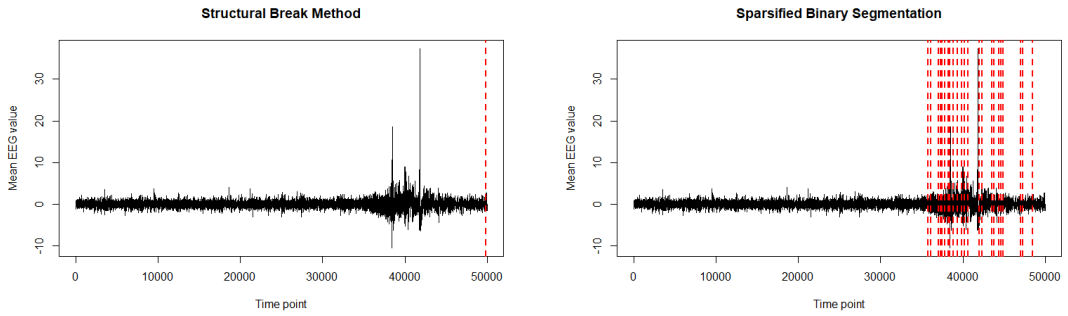

Figure 3.12: The mean EEG values across all the channels and estimated change point using Structural Break Method and Sparsified Binary Segmentation (Downsampling is performed by keeping only every 50th sample to apply the Structural Break Method). The red dotted lines denote the location of estimated change points.

3.4.2 Stock Data

We also applied our Spec PC-CP method to the log returns of the daily closing values of 9 representative $S\&P$ 100 stocks which has been analyzed in previous work (Barigozzi et al., 2018). The stock data was observed between 4 January 2000 and 10 August 2016 from Yahoo finance. The total length is 4,177 days. We analyzed the following 9 stocks: AXP (American Express), BAC (Bank of America), BK (The Bank of New York Mellon), C (Citigroup), COF (Capital One Financial), GS (Goldman Sachs), JPM (JPMorgan), MS (Morgan Stanley), WFC (Wells Fargo). Figure 3.13 shows the stock prices of Bank of America and JP Morgan.

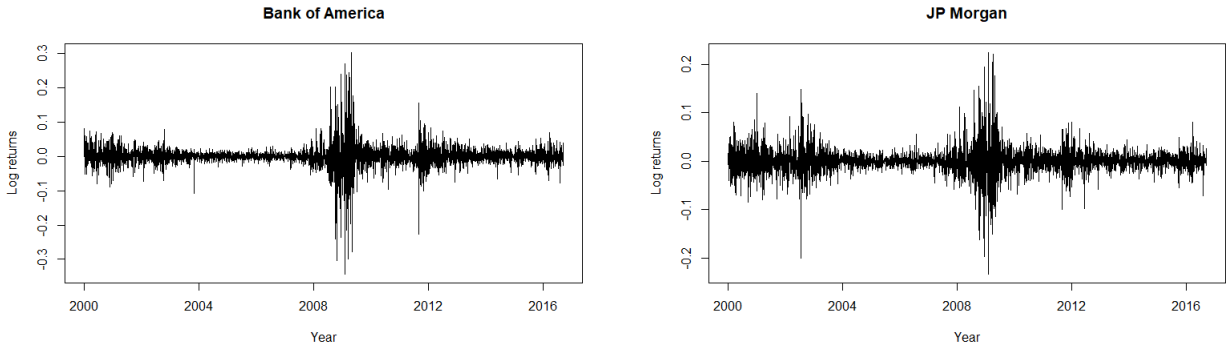

Figure 3.13: Daily log return values of the daily closing values of Bank of America and JPMorgan between January 2000 and August 2016.

We used the first two spectral principal components for change point detection. Figure 3.14 gives the frequency-time plots of two stocks and the first two spectral PCs. Changes appear in multiple frequency bands in this case. The detected change points are shown in Figure 3.15. Most of the change points are around events that might have impact on the financial market. For example, the financial crisis around 2008, the Greek and EU sovereign debt crisis in 2011 and 2015. Both the first and second spectral principal components are able to detect the change points in a neighbourhood of the above events. These results demonstrates that our method also work well for stock data. Figure 3.16 shows the results using the comparison methods. The Contemperaneous Method has comparable estimates while the other two methods fail to detect events such as the financial crisis in 2008 and the debt crisis in 2015.

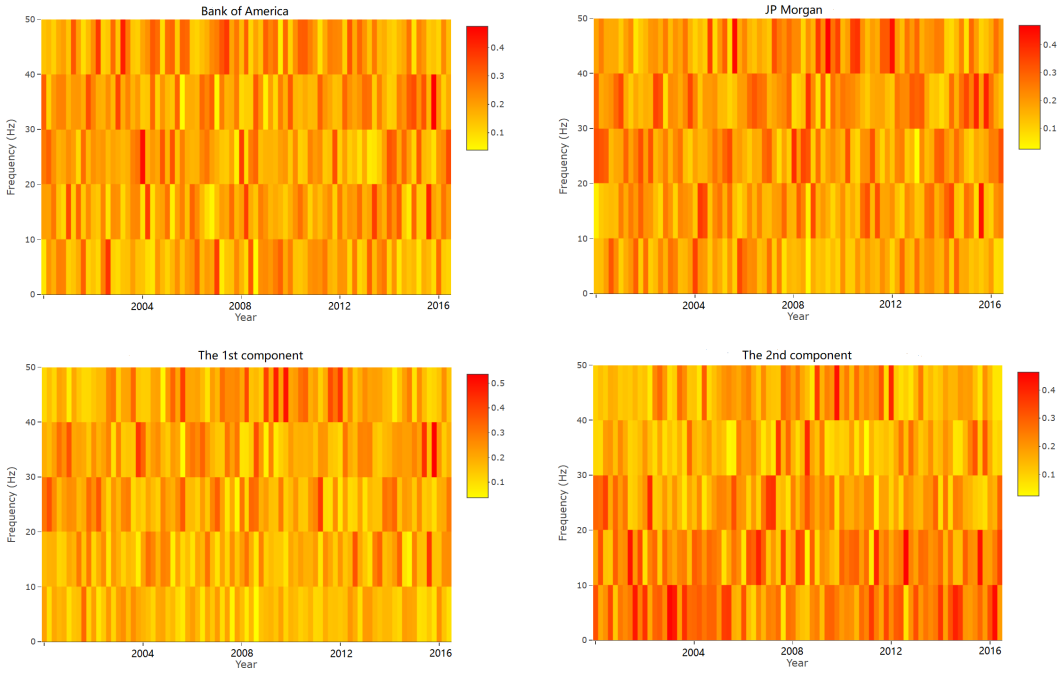

Figure 3.14: Time-frequency plots. Top: Bank of America and JP Morgan. Bottom: The first and second spectral PCs. The x-axis denotes the time; the y-axis denotes the frequency from 0 to 50 Hz.

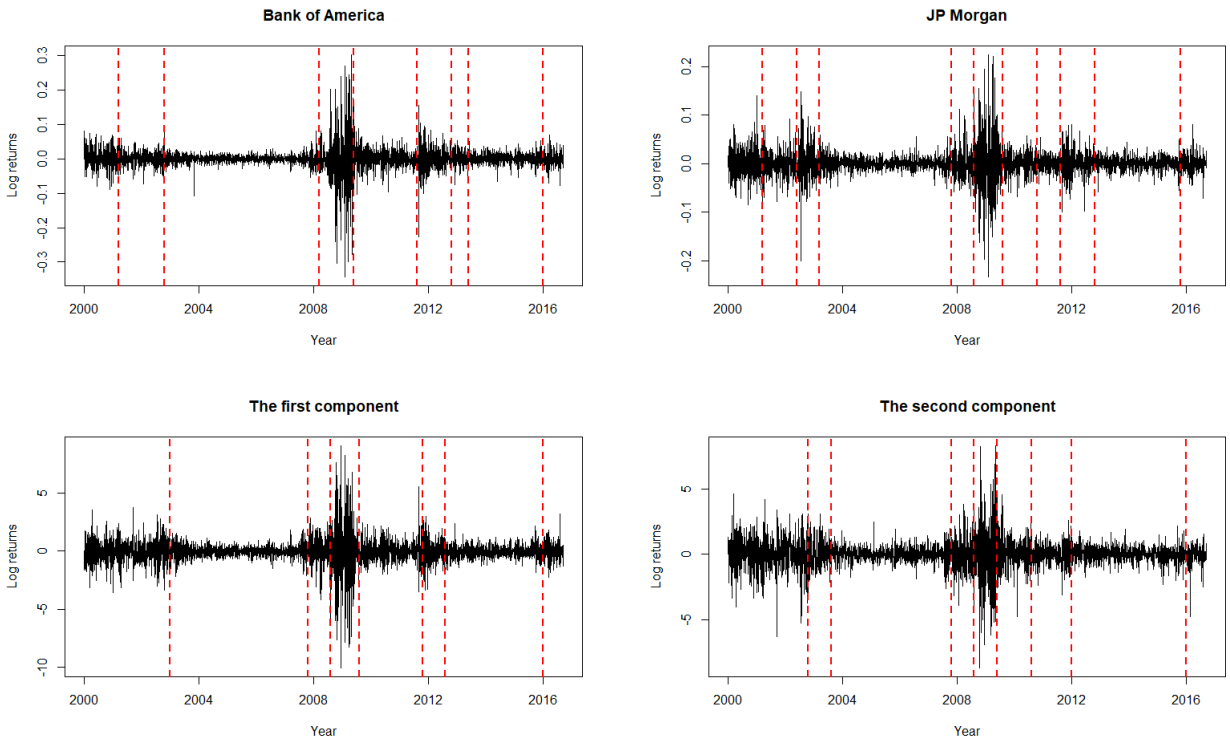

Figure 3.15: Top: The daily log returns of Bank of America and JP Morgan. Bottom: The first and second spectral PCs estimated from all the 9 stocks in dataset. The dotted lines denote the locations of estimated change points

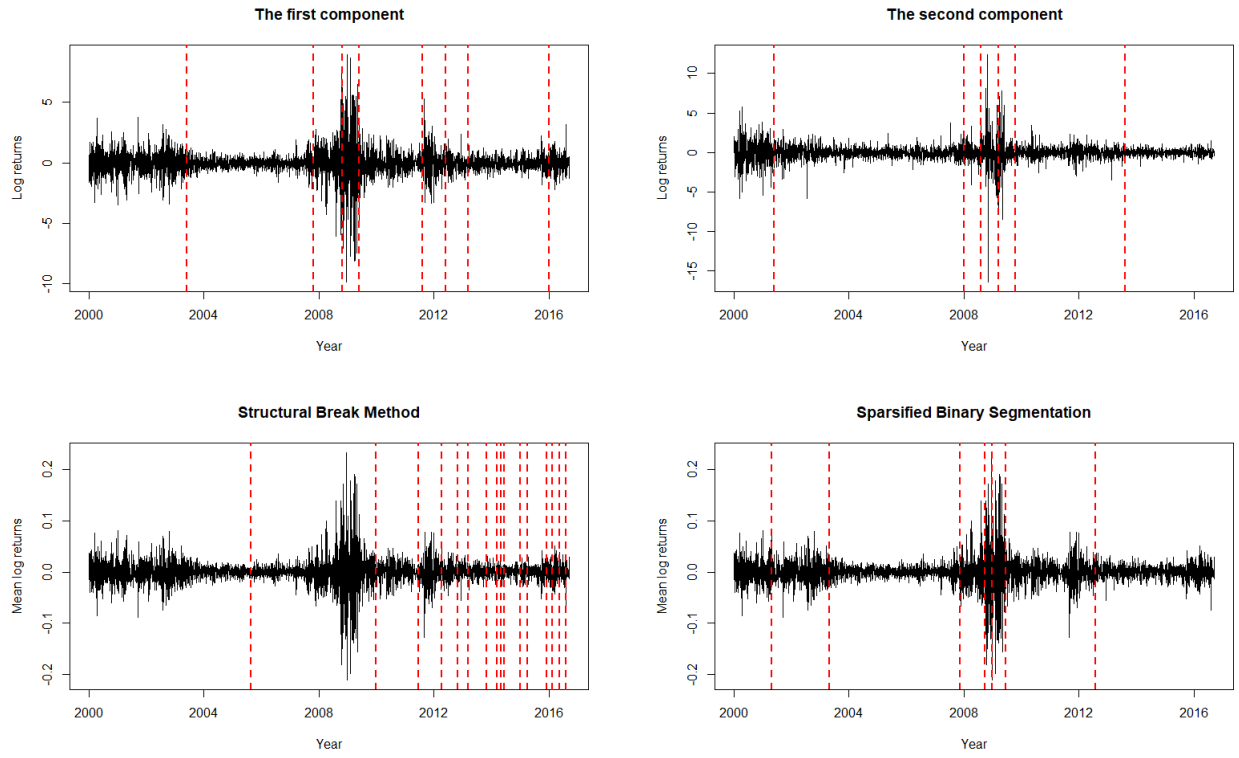

Figure 3.16: Top: The estimated change point using the first or second component of Contemperaneous Method. Bottom: The estimated change point using Structural Break Method and Sparsified Binary Segmentation. The red dotted lines denote the location of estimated change points.

Chapter 4

Time-varying ℓ_0 optimization for Spike Inference from Multi-Trial Calcium Recordings

4.1 Introduction

The problems presented in Chapter 3 are concerned with the changes of multivariate time series in the spectral domain. In this chapter, we will focus on a deconvolution problem in imaging analysis. Seemingly unrelated to change point detection, the deconvolution problem in estimating the times of action potentials from calcium fluorescence traces can be treated as a special change point problem due to the almost instantaneous rise and slow decay of a calcium transient in the presence of a neural spike. We first provide some background about using calcium recordings for neural activities.

In the past few years, calcium imaging has been increasingly adopted in neuroscience research since it allows simultaneous measurement of the activity of a large population of neurons

at the single-neuron resolution over weeks using optical imaging of living animals. For example, we recently conducted a longitudinal investigation of the neural ensemble dynamics of contextual discrimination by recording mice's calcium imaging in the hippocampus for about 60 days (Johnston et al., 2020) using appropriate genetically encoded calcium indicators (Tian et al., 2009) and miniature fluorescence miscroscopes (Ghosh et al., 2011). The technical advancements bring great flexibility to neuroscience research; they also create significant challenges to every aspect of data analysis - from storing the large amount of video recordings to downstream statistical modeling and inference (Pnevmatikakis, 2019). In longitudinal recordings of freely behaving animals, after motion correction and registration of detected neurons across multiple sessions (Giovannucci et al., 2019), fluorescence traces of individual neurons can be extracted, for example using independent component analysis (Mukamel et al., 2009) or nonnegative matrix factorization methods (Maruyama et al., 2014; Pnevmatikakis et al., 2016; Zhou et al., 2018). Strategies to reduce false positive rates of neuron detection have also been proposed, such as our recent work on imposing spatial constraints on footprints of neurons (Johnston et al., 2020). The extracted fluorescence traces of individual neurons allow for the statistical analysis of complex neural interactions at the single cell level. Fluorescence traces are often used for visualization, clustering neurons based on similar neural activity profiles, comparing neural activity levels between various experimental conditions, and studying neural encoding and decoding.

However, calcium fluorescence data provide a noisy proxy, rather than direct observations of neural activity. For many important questions, such as those involving the analysis of the precise timing of neural activity in response to stimuli, it is essential to estimate the underlying spike train from a noisy fluorescence trace. Several approaches have been developed including linear deconvolution (Yaksi and Friedrich, 2006) or nonnegative convolutions (Vogelstein et al., 2010). Fully Bayesian methods have also been developed to obtain posterior distributions of spike trains, thus allowing uncertainty quantification of spikes' estimates (Pnevmatikakis et al., 2013). Theis et al. (2016) proposed a supervised learning approach based on a probabilistic

relationship between fluorescence and spikes. This algorithm is trained on data where spike times are known. Research has shown that nonnegative deconvolution outperforms supervised algorithms and is more robust to the shape of calcium fluorescence responses (Pachitariu et al., 2018). One limitation of these methods is that these spike detection methods analyze only one trial at time; thus, shared information across trials is largely ignored. In a longitudinal study, neural activities are measured in multiple trials or sessions. In these settings, aggregating information across trials could increase the accuracy of spike detection.

There has been limited work on how to utilize the information from multiple trials to improve accuracy in spike detection. Among the few multi-trial methods we identified, Picardo et al. (2016) assumed that repeated trials share the same burst time but have trial-specific magnitude, baseline fluorescence, and noise level. Conceptually, integrating multiple trials should be beneficial if the trial-to-trial variation is mainly due to randomness. In reality, however, as pointed out in Deneux et al. (2016), the gain by naively combining trials may be limited - it is likely to bring improvement for some trials but might perform worse in others when the neural activity in some trials does not follow the marginal pattern across all trials. During a learning process or when adjusting to a new environment, neurons constantly reorganize and show plasticity, which leads to varying neural dynamics across trials.

In this paper, we develop a robust multi-trial spike inference method to address the challenges from longitudinal calcium imaging data. Our approach both integrates the commonality and accounts for evolving neural dynamics across trials. Efficiency is achieved by aggregating information from temporally adjacent trials whereas robustness is guaranteed by incorporating a time-varying firing rate function into a dynamic ℓ_0 penalization framework, rather than forcing shared parameters across trials.

4.2 Methods

In this section, we first propose a time-varying ℓ_0 penalized framework to analyze multi-trial calcium trace data. We then present details on how to implement it by alternating a firing rate estimation step and a spike detection step.

4.2.1 Multi-Trial time-varying ℓ_0 penalized auto-regressive model (MTV-PAR)

Let $y_r(t)$ be the fluorescence recorded at time point t of trial r, where $t = 1, ...T$, $r = 1, \cdots, R$. In practice, multiple pre-processing steps, including normalization, (Vogelstein et al., 2010), are typically implemented to obtain $y_r(t)$. The calcium fluorescence trace $y_t(t), t = 1, \cdots, T$ is often modeled using the following first-order auto-regressive model

$$\begin{aligned} &y_r(t) = c_r(t) + \epsilon_r(t), \quad \epsilon_r(t) \sim N(0, \sigma_r^2), \\ &c_r(t) = \gamma_r c_r(t-1) + s_r(t), \text{ s.b.t. } s_r(t) \geq 0 \end{aligned} \tag{4.1}$$

where $c_r(t)$ and $\epsilon_r(t)$ denote the underlying calcium concentration and noise at time t in the rth trial, respectively; $0 < \gamma_r(t) < 1$ is the decay parameter for a calcium transient, which depends on multiple factors such as sampling frame, cell types, and the kinetics of genetically encoded indicators. In this model, $s_r(t)$ denotes the change in calcium concentration between time $t-1$ and time t with $s_r(t) > 0$ indicates a spike occurs at time t and 0 means no spike. The main goal of spike detection to locate the time points with a positive $s_r(t)$.

Here, we propose to regularize the inference on spike detection by introducing a time-varying penalty function $\lambda_r(t)$ on the number of spikes, as the spikes tend to be sparse but not evenly

distributed over time:

$$\min_{c_r(1),\ldots,c_r(T)} \left\{ \frac{1}{2} \sum_{t=1}^{T} (y_r(t) - c_r(t))^2 + \sum_{t=2}^{T} \lambda_r(t) \mathbb{1}_{s_r(t)>0} \right\}. \tag{4.2}$$

We further assume that $\lambda_r(t)$ is a decreasing function of the instantaneous firing rate $f_r(t)$, which is assumed to be smooth over time t within trials and change slowly over trials r. Because the parameters in the AR(1) model are trial-specific but the firing rate functions across trials are assumed to change slowly to capture the evolving neural dynamics, efficiency and robustness are well balanced. In addition, our modeling approach allows the simultaneous estimation of spike trains and firing rate functions.

In the rest of this subsection, we present an iterative approach that (1) estimates the firing rate function, given the spike trains derived in the previous iterations, by using a local nonparametric method; (2) detects spike trains, given the current estimate of the firing rate function, by using a time-varying ℓ_0 penalization.

4.2.2 Firing rate estimation and time-varying penalty

Estimating the firing rate function is a crucial task in the analysis of spike train data (Cunningham et al., 2009). One commonly used descriptive approach involves building the peristimulus time histogram (PSTH) where the spike counts are averaged from multiple trials within each time bin (Gerstein and Kiang, 1960). Kernel methods are often applied to achieve smoothness (Cunningham et al., 2008). Bayesian methods have also been considered, e.g. by proposing Gaussian processes (Cunningham et al., 2008; Shahbaba et al., 2015) and Bayesian Adaptive Regression Splines (Behseta and Kass, 2005; DiMatteo et al., 2001; Kass et al., 2003, 2005; Olson et al., 2000).

The PSTH approach and the other smoothing methods assume that the underlying firing rate

function does not change over trials. The estimated firing rates are then compared between different groups to capture the association between firing rate and animal behavior (Jog et al., 1999; Wirth et al., 2003; Wise and Murray, 1999). However, recent evidence suggests to move beyond the independent and identically distributed trial assumption and regard both neural and behavioral dynamics as smooth and continuous (Huk et al., 2018; Ombao et al., 2018). Thus, it is essential to model the between-trial dynamics. To account for the between-trial dynamics, some authors have proposed a state-space framework (Czanner et al., 2008; Paninski et al., 2010). In the state-space framework, the spike train is characterized by a point process model (Brown, 2005; Brown et al., 2003; Daley and Vere-Jones, 2007; Kass et al., 2005) for the underlying fire rate. In this paper, to make it feasible to jointly estimate the spike trains and the firing rate function from the observed multi-trial fluorescence data, we consider instead a computationally less demanding two-dimensional Gaussian-boxcar kernel smoothing function $G(r,t)$, which is formulated as follows

$$G(r,t) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{t^2}{2\sigma^2}\right) I(|r| < B/2) \tag{4.3}$$

where σ denotes the within-trial kernel bandwidth in the Gaussian kernel, $I(\cdot)$ is the indicator function, and B is a bandwidth for the between-trial sliding windows in the boxcar kernel. Thus, our estimate of $f_r(t)$ is given by

$$\hat{f}_r(t) \propto \sum_{r',t'} G(r-r', t-t') p_{r'}(t'),$$

where $p_{r'}(t')$ is the estimated number of spikes per second in a small time bin centered at time t' in trial r'.

For PSTH based on spike train data, common choices of the Gaussian bandwidth are around 50-150 ms (Cunningham et al., 2009). When choosing the optimal bandwidth to smooth a spike train estimated from calcium fluorescnce trace data, one should also consider the

fluorescence transient kinetics. Ali and Kwan (2019) reviewed the kinetics of calcium transients with several fluorescent indicators. For example, they reported that GCaMP6f has a rise time of 42 ms and decay time of 142 ms while GCaMP6s has a rise time of 179 ms and decay time of 550 ms; a time-bin of 200 ms is suggested in one of their analyses. In our analysis, σ is chosen between 200ms and 400ms. For the box-car bandwidth to allow borrowing information across trials, we use a pre-selected window size B, which is mainly determined by to the number of available trials. As discussed in the Discussion section, data driven methods, such as the one used in (Fiecas and Ombao, 2016; Ombao et al., 2018), will be considered in future work.

In spike detection, sparsity has been enforced via penalization or introducing appropriate prior distributions. Existing approaches usually adopt a tuning parameter uniformly for the whole time series of a fluorescence trace. Within a trial, the firing rate right after each stimulus are expected to be higher than baseline. Thus, using a constant penalty may not be optimal. Our proposed non-constant penalization is inspired by prior work in the literature. For example, time-varying penalization was used for analyzing multivariate time series data (Fan et al., 2013; Yu et al., 2017). Zbonakova et al. (2016) studied the dynamics of the penalty term in a Lasso framework for the analysis of interdependences in the stock markets. Monti et al. (2017) used varying regularization for different edges in a Gaussian graphical model to study brain connectivity in fMRI data.

Here, we expand on those approaches and consider a decreasing function of the firing rate function for the time-varying penalty, motivated by the fact that spikes are expected to be less frequent in regions with low firing rates than in regions with high firing rates. Ideally, the penalty should be small in the locations with higher firing rates. Hence, we use a negative exponential function (Tang et al., 2010) to achieve adaptive regularization. Specifically, the

time-varying function is chosen as

$$\lambda_r(t) \propto e^{-a\hat{f}_r(t)}$$

where $\hat{f}_r(t)$ is the estimated firing rate (See Algorithm 2). Here the value a controls how much the penalty function should depend on the firing rate function. In particular, $a = 0$ reverts to a constant penalty. To avoid extremely large or small penalties, we scale the estimated firing rate function of each trial by dividing by its maximum value, so that it ranges between 0 and 1. This implies that the default value $a = 1$ in our analysis leads to mildly time-varying penalties - within a trial, the penalty at the highest firing rate is about 37% of the penalty at a firing rate of 0. In addition, to facilitate comparison with the case of a constant penalty, the following formula is used in each trial to scale the penalty function $\lambda_r(t)$ to have mean λ,

$$\lambda_r(t) = \lambda T \times \frac{\exp\left\{-a\frac{\hat{f}_r(t)}{\max(\hat{f}_r(t))}\right\}}{\sum_{t'=1}^{T}\exp\left\{-a\frac{\hat{f}_r(t')}{\max(\hat{f}_r(t'))}\right\}}. \tag{4.4}$$

4.2.3 Time-varying penalized ℓ_0 AR(1) model

In our proposed multi-trial time-varying ℓ_0 penalized auto-regressive model (MTV-PAR), a calcium fluorescence trace at the rth trial $y_r(t), t = 1, \cdots, T$, is modeled by a first-order auto-regressive model, as given in Equation (4.1). As previously stated, $c_r(t)$ denotes the underlying calcium concentration and a positive value of $s_r(t)$ implies that a spike occurs at time t.

Because spikes tend to be sparse, ideally, one should penalize the number of spikes by introducing an ℓ_0 penalty. However, ℓ_0 penalization is computationally intractable; therefore, existing methods often impose an ℓ_1 penalization (Friedrich and Paninski, 2016; Friedrich et al., 2017; Vogelstein et al., 2010). Recently, Jewell and Witten (2018) found that, for

spike detection, the use of an ℓ_0 penalty brings substantial improvements over an ℓ_1 penalty. They also showed that relaxing the nonnegative constraint of $s_r(t)$ has a negligible effect on the results but the corresponding ℓ_0 optimization problem is equivalent to a change point detection problem whose solution can be obtained efficiently using a dynamic programming algorithm. Use a similar strategy, we prove that the following time-varying penalization problem can also be solved by a dynamic programming algorithm:

$$\min_{c_r(1),\ldots,c_r(T)} \left\{ \frac{1}{2} \sum_{t=1}^{T} (y_r(t) - c_r(t))^2 + \sum_{t=2}^{T} \lambda_r(t) \mathbb{1}_{s_r(t)\neq 0} \right\}. \tag{4.5}$$

Specifically, we find that the time-varying ℓ_0 optimization problem is equivalent to the following change point problem whose solution can be efficiently identified using a dynamic programming algorithm. For the ease of presentation, we drop the trial index r and focus on the fluorescence trace of a cell at a given trial. The equivalent change point problem can be shown as:

$$\begin{aligned}
&\min_{c(1),\ldots,c(T)} \left\{ \frac{1}{2} \sum_{t=1}^{T} (y(t) - c(t))^2 + \sum_{t=2}^{T} \lambda(t) 1_{(c(t)-\gamma c(t-1)\neq 0)} \right\} \\
\Leftrightarrow &\min_{\substack{0=\tau_0<\tau_1<\ldots<\tau_k<\tau_{k+1}=T,k \\ c(1),\ldots,c(T)}} \left\{ \sum_{j=0}^{k} \frac{1}{2} \sum_{t=\tau_j+1}^{\tau_{j+1}} (y(t) - c(t))^2 + \sum_{t=2}^{T} \lambda(t) 1_{c(t)-\gamma c(t-1)\neq 0} \right\} \\
\Leftrightarrow &\min_{\substack{0=\tau_0<\tau_1<\ldots<\tau_k<\tau_{k+1}=T,k \\ c(1),\ldots,c(T)}} \left\{ \sum_{j=0}^{k} \frac{1}{2} \sum_{t=\tau_j+1}^{\tau_{j+1}} (y(t) - c(t))^2 + \sum_{j=0}^{k} \lambda(\tau_j) \right\} - \lambda(0) \\
\Leftrightarrow &\min_{0=\tau_0<\tau_1<\ldots<\tau_k<\tau_{k+1}=T,k} \left\{ \sum_{j=0}^{k} \frac{1}{2} \sum_{\substack{t=\tau_j+1\ldots\tau_{j+1} \\ (c(t)=\gamma c(t-1))}} (y(t) - c(t))^2 + \sum_{j=0}^{k} \lambda(\tau_j) \right\} \\
\Leftrightarrow &\min_{0=\tau_0<\tau_1<\ldots<\tau_k<\tau_{k+1}=T,k} \left\{ \sum_{j=0}^{k} \left[\mathcal{D}\left(y_{(\tau_j+1):\tau_{j+1}} \right) + \lambda(\tau_j) \right] \right\}
\end{aligned}$$

where $\tau_1, ...\tau_k$ are k change points, i.e., the points satisfying $c(t) - \gamma c(t-1) \neq 0$ and

$$\mathcal{D}(y_{a:b}) = \min_{\substack{c(a), c(t)=\gamma c(t-1) \\ t=a+1,\ldots,b}} \left\{ \frac{1}{2} \sum_{t=a}^{b} (y(t) - c(t))^2 \right\}, \tag{4.6}$$

which has the following closed-form solution:

$$\mathcal{D}(y_{a:b}) = \frac{1}{2} \sum_{t=a}^{b} \left(y(t) - \gamma^{t-a} \frac{\sum_{t=a}^{b} y(t)\gamma^{t-a}}{\sum_{t=a}^{b} \gamma^{2(t-a)}} \right)^2. \tag{4.7}$$

Note that the parameter γ, which measures the speed at which the calcium concentration decays, is not estimated. The value of γ is usually close to 1 (Vogelstein et al., 2010; Yaksi and Friedrich, 2006), as a somatic calcium transient caused by an action potential is often characterized by an almost instantaneous rise but a slow decay. For computational feasibility, rather than estimating γ iteratively, similar to (Friedrich et al., 2017; Pnevmatikakis et al., 2016), we estimate it using the auto-correlation at lag 1.

Finally, as shown below, the optimization problem (4.5) can be simplified as follows:

$$\begin{aligned}
&\min_{0=\tau_0<\tau_1<\cdots<\tau_k<\tau_{k+1}=t,k} \left\{ \sum_{j=0}^{k} \left[\mathcal{D}\left(y_{(\tau_j+1):\tau_{j+1}}\right) + \lambda(\tau_j) \right] \right\} \\
&\Leftrightarrow \min_{0=\tau_0<\tau_1<\cdots<\tau_k<\tau_{k+1}=t,k} \left\{ \sum_{j=0}^{k-1} \left[\mathcal{D}\left(y_{(\tau_j+1):\tau_{j+1}}\right) + \lambda(\tau_j) \right] + \mathcal{D}\left(y_{(\tau_k+1):\tau_{k+1}}\right) + \lambda(\tau_k) \right\} \\
&\Leftrightarrow \min_{0<\tau_k<\tau_{k+1}=t} \Big\{ \min_{0=\tau_0<\tau_1<\cdots<\tau_{k'}<\tau_{k'+1}=\tau_k,k'} \left\{ \sum_{j=0}^{k'} \left[\mathcal{D}\left(y_{(\tau_j+1):\tau_{j+1}}\right) + \lambda(\tau_j) \right] \right\} \\
&+ \mathcal{D}\left(y_{(\tau_k+1):\tau_{k+1}}\right) + \lambda(\tau_k) \Big\} \\
&\Leftrightarrow \min_{0\leq\tau<t} \left\{ F(\tau) + \mathcal{D}\left(y_{(\tau+1):t}\right) + \lambda(\tau) \right\}
\end{aligned}$$

Therefore, we can solve the problem by computing $F(t)$ recursively:

$$F(t) = \min_{0 \leq \tau < t} \left\{ F(\tau) + \mathcal{D}\left(y_{(\tau+1):t}\right) + \lambda(\tau) \right\}, \quad \text{for } t = 1, 2, \cdots, T. \tag{4.8}$$

The resulting algorithm has a time complexity $\mathcal{O}(n^2)$. This optimization problem can be solved in $\mathcal{O}(n)$ time (Algorithm 4) with a dynamic programming algorithm (Auger and Lawrence, 1989; Jackson et al., 2005; Jewell and Witten, 2018; Killick et al., 2012) .

Algorithm 4 Dynamic programming algorithm to detect spikes with a time-varying penalization function

Input: Time-varying penalty function $\lambda(t)$, single-trial fluorescence $y(t)$ with T time points.
Initialize: $F(0) = -\lambda, spikeset = \emptyset, \mathcal{E}_1 = \{0\}$
for $t = 1, 2, ...T$ **do**
    Calculate $F(t) = \min_{\tau \in \mathcal{E}_t} \left\{ F(\tau) + \mathcal{D}\left(y_{(\tau+1):t}\right) + \lambda(\tau) \right\}$
    Find $t^* = \underset{\tau \in \mathcal{E}_s}{\operatorname{argmin}} \{ F(\tau) + \mathcal{D}\left(y_{(\tau+1):t}\right) + \lambda(\tau) \}$
    Let $\mathcal{E}_{t+1} = \left\{ \tau \in \{\mathcal{E}_t \cup t\} : F(\tau) + \mathcal{D}\left(y_{(\tau+1):t}\right) < F(t) \right\}$
    if $t^* \notin spikeset$ **then**
        Add t^* to $spikeset$
    end if
end for
Return $Spikeset$

4.2.4 Algorithms

Thus far, we have presented our solutions to two problems separately. The first problem is to use multi-trial spike data to estimate the firing rate function $f_r(t)$ for $r = 1, \cdots, R$ and $t = 1, \cdots, T$ using a Gaussian-boxcar smoothing method. The second problem is to detect spikes from a single calcium fluorescence trace using a time-varying ℓ_0 penalized approach under the assumption that the time-varying penalty function $\lambda_r(t)$ is already known. In practice, neither firing rate functions nor spike locations are known. We therefore propose an Expectation-Maximization-type algorithm to jointly estimate firing rate and detect spike

locations. As shown in Algorithm 5, we alternate between the spike detection and firing rate estimation steps until the spike indicator function does not change anymore.

Algorithm 5 Simultaneous spike detection and firing rate estimation

Input: Multi-trial fluorescence data $y(t)^{R\times T}$ with R trials and T time points. Penalty term λ.
Initialize: Penalty function $\lambda_r(t) = \lambda$.
Set the binary spike indicator $x(r,t)^{R\times T} = 0$.
while the indicator matrix $x(r,t)^{R\times T}$ changes **do**
    for $r = 1, 2, ...R$ **do**
        Apply Algorithm 4 to detect spikes and let *spikeset* denote the times at which spikes were detected.
        Set $x(r, spikeset) = 1$.
    end for
    Estimate the firing rate $\hat{f}_r(t)$, for $r = 1, \cdots, R$ and $t = 1, \cdots, T$ using Gaussian-Sliding-Window kernel smoothing.
    Calculate the weight function

$$w_r(t) = e^{-a\frac{\hat{f}_r(t)}{\max(\hat{f}_r(t))}}$$

    Update the time-varying penalty function $\lambda_r(t) = \lambda * T * w_r(t) / \sum_{t'} w_r(t')$
end while

4.3 Simulation

4.3.1 Metrics for quantifying accuracy

We conducted a set of simulation studies to evaluate the performance of our MTV-PAR method. Among the many competing methods, we choose the ℓ_0 approach in Jewell and Witten (2018) because its performance has been shown better than competing methods using both simulated and benchmark data. When comparing estimated spike trains with the ground truth, we use the Victor-Purpura (VP) distance (Victor and Purpura, 1996, 1997), which has been commonly used for comparing spike trains. It is defined as the minimum total cost required to transform one spike train into another using the following three basic

operations:

- Insert a spike into a spike train. (Cost = 1)
- Delete a spike. (Cost = 1)
- Shift a spike by an interval Δt. (Cost = $q\Delta t$) A large q makes the distance more sensitive to fine timing differences. We use the default value $q = 1$.

Because MTV-PAR estimates the firing rate function together with the spikes, we also evaluate its performance on firing rate estimation. The approach in Jewell and Witten (2018) does not estimate firing rates; hence, for a fair comparison, we apply the same Gaussian-boxcar kernel smoothing in (4.3) to the spikes estimated using Jewell and Witten (2018) in order to estimate the firing rates. The accuracy of firing rate estimation is calibrated by the ℓ_2 norm (Adams et al., 2009) of the difference between an estimated firing rate function and the true function:

$$\|f(t) - \hat{f}(t)\|_2 = \left(\frac{1}{\int m(t)dt} \int |f(t) - \hat{f}(t)|^2 m(t)dt \right)^{1/2}$$

where $m(t)$ is the weight at t and its default value is a constant.

4.3.2 Simulation of spike trains from inhomogeneous Poisson processes

In this simulation setting, the trials are treated as repeated trials. In other words, the trials share the same underlying firing rate function and each trial is an independent realization of an inhomogeneous Poisson process. Several forms of firing rate functions have been considered in previous work. For example, Behseta and Kass (2005); Kass et al. (2003) assumed bell-shaped firing rate functions. Pachitariu et al. (2018) used a piecewise constant

stimulus rate. Firing rate functions with multiple peaks from exponential stimuli functions have also been considered (Reynaud-Bouret et al., 2014). In our simulation, we chose the following bi-modal firing rate function

$$f(t) = 0.01 + 0.19 * \sum_{t_0} \exp(-(t - t_0)^2/d^2) \quad (t = 1, 2, ...1000) \tag{4.9}$$

where $d = 150$ and the stimuli peaks are at $t_0 = 300$ and 700. Thus, $f(t)$ reaches its maximum value 0.2 (equivalent to 10 spikes per second) at time 300 and 700.

To generate spike trains for multiple trials under an inhomogeneous Poisson process with the firing rate function in 4.9, we followed the idea of Adams et al. (2009) (Algorithm 6). In each trial, the spikes were first randomly drawn from a homogeneous Poisson process. A thinning process was then applied to create a realization from the desired inhomogeneous Poisson process. After obtaining the simulated spike trains, we generated calcium fluorescence traces

Algorithm 6 Simulate Spike Trains from an Inhomogeneous Poisson Process (Modified from Algorithm 1 of Adams et al. (2009))

1: **Input** Length of data n, upper bound of rate intensity f^*, firing rate function $f(t)$
2: Draw the number of spikes: $S \sim Poisson(nf^*)$
3: Uniformly distribute the spikes: $\mathcal{E}_S = \{s_i\}_{i=1}^{S} \sim$ Uniform (n)
4: **for** $i = 1, 2, ...S$ **do**
5: **if** $uniform(0, 1) \geq \sigma\left(f(s_i)\right)$ **then**
6: exclude s_i from $\mathcal{E}_S$
7: **end if**
8: **end for**
9: Return the set of spikes $\mathcal{E}_S$

using the auto-regressive model (4.1). The following parameters were used in the simulations: $T = 1000$, $\gamma = 0.96$, $\sigma = 0.15$ and $R = 50$ trials in total. According to Vogelstein et al. (2010), the parameters above correspond to a sampling rate of 50 Hz and a length of 20 seconds per trial. For each simulation setting, 100 data sets were generated.

We implemented our MTV-PAR using Algorithm 2. When estimating $\lambda(t)$, we chose the Gaussian kernel bandwidth equal to 200 ms. Because the trials are assumed to have the

same underlying firing rate function, B=50 is used, which is equivalent to averaging the spike counts across trials to estimate the firing rate $f(t)$. As shown in the top panel of Figure 4.1, the firing rate functions estimated from MTV-PAR are closer to the true underlying function. Thus, adopting the proposed time-varying penalty led to an improvement in both spike detection (measured by mean VP distance, the lower panel of Figure 4.1) and firing rate estimation (measured by the ℓ_2 norm, the upper panel of Figure 4.1) across all the λ values considered. For the optimal λ based on VP, the reductions brought by the time-varying penalty in VP and ℓ_2 norm are 17.3% and 76.3%, respectively.

We also simulated data with a larger noise level ($\sigma = 0.3$), longer series ($T = 2,000$) and higher decay rate ($\gamma = 0.98$). In all scenarios, we observed improved accuracy from using the proposed MTV-PAR.

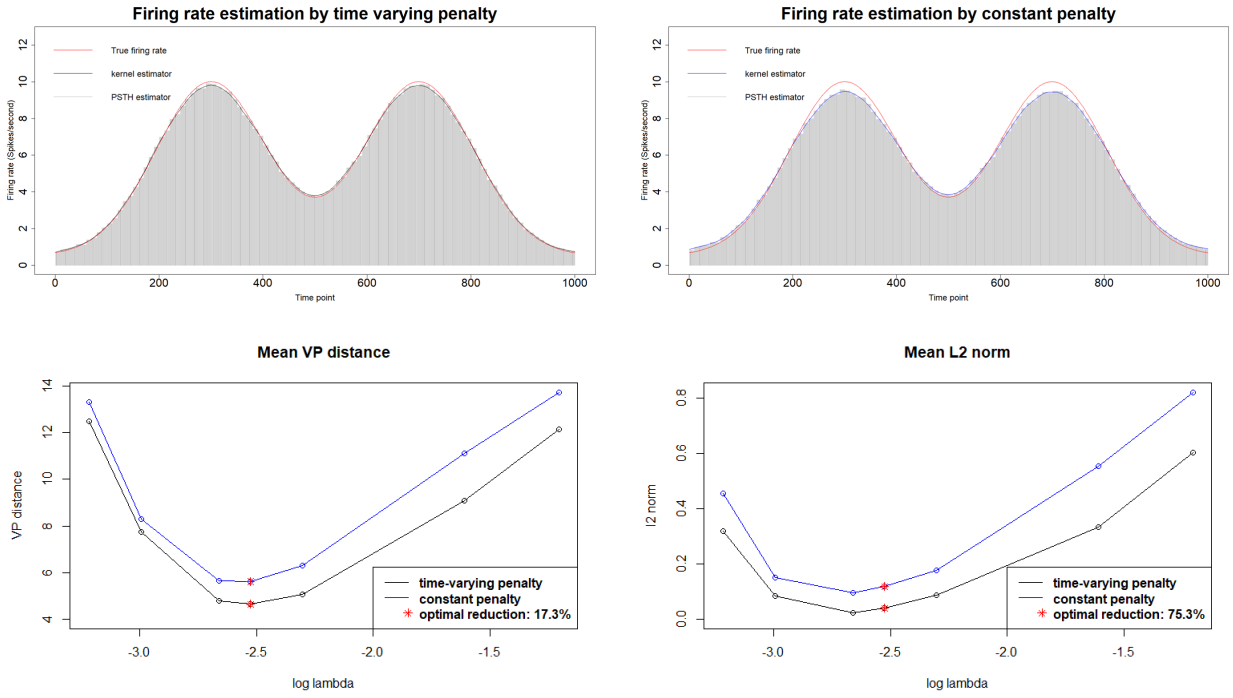

Figure 4.1: Top: firing rate estimates using MTV-PAR (time-varying) and constant penalty. Bottom: VP distance of spike trains and ℓ_2 norm of firing rates between the truth and estimators.

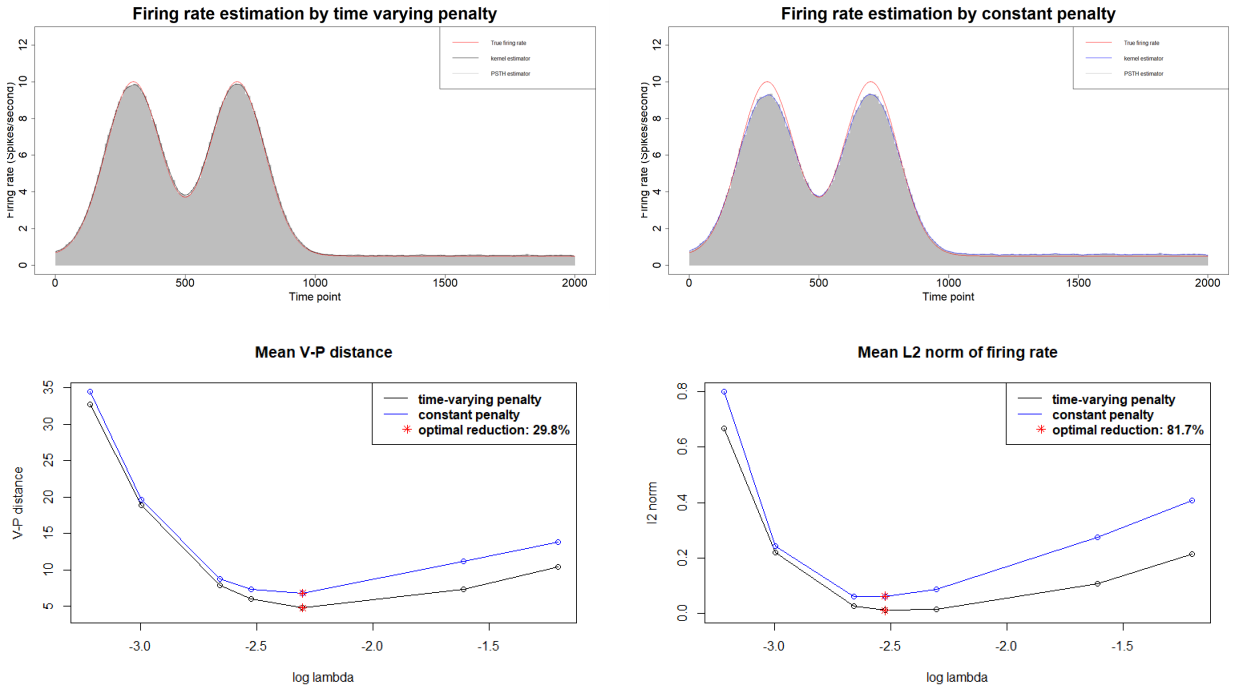

Figure 4.2: Top: firing rate estimates using MTV-PAR (time-varying) and constant penalty. Bottom: VP distance of spike trains and ℓ_2 norm of firing rates between the truth and estimators. (Series length $n = 2,000$)

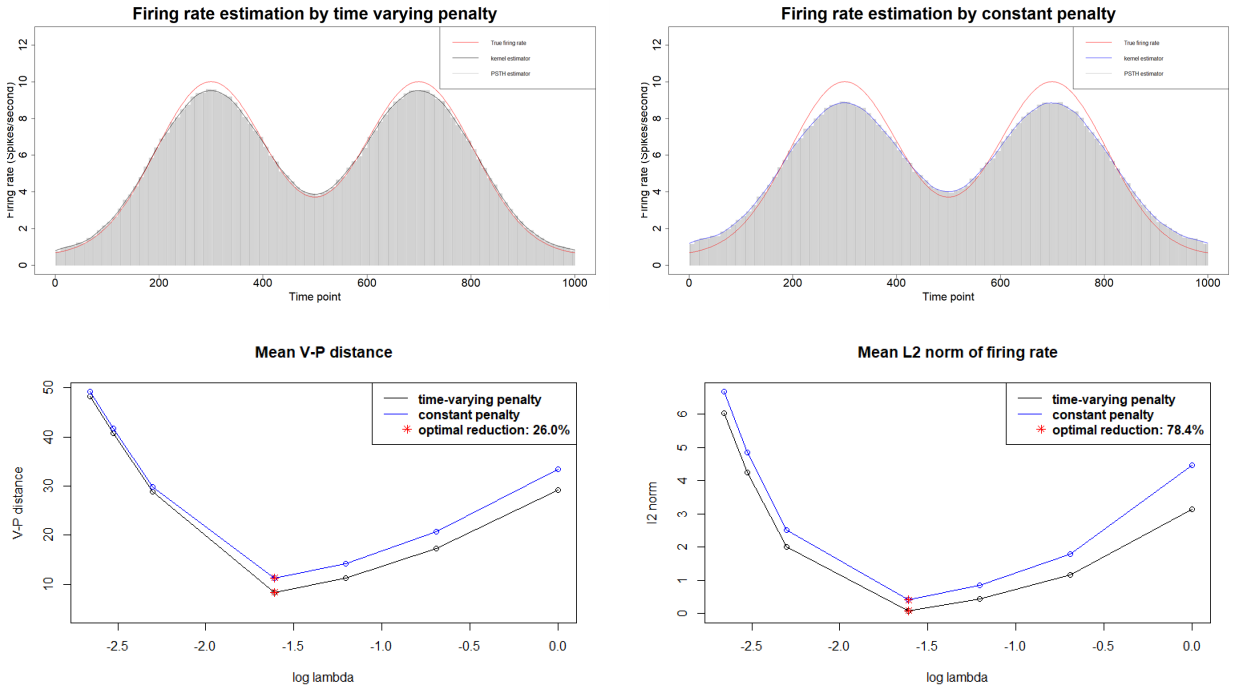

Figure 4.3: Top: firing rate estimates using MTV-PAR (time-varying) and constant penalty. Bottom: VP distance of spike trains and ℓ_2 norm of firing rates between the truth and estimators. (Standard deviation of noise: 0.3)

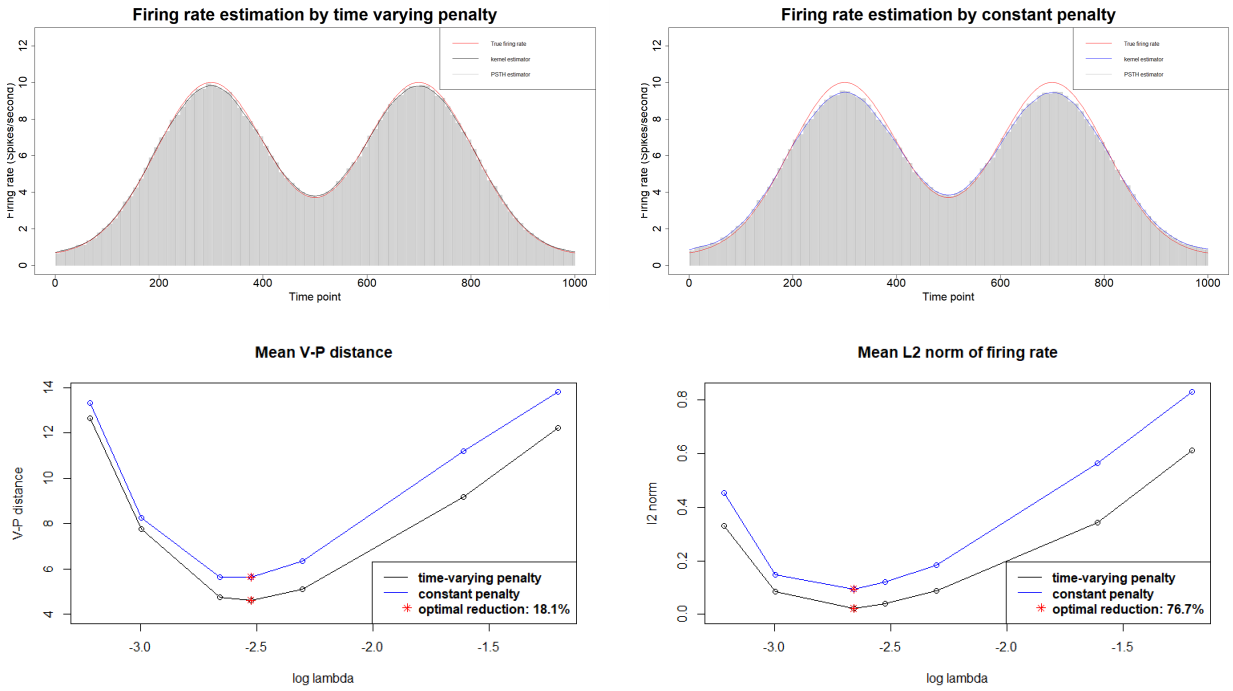

Figure 4.4: Top: firing rate estimates using MTV-PAR (time-varying) and constant penalty. Bottom: VP distance of spike trains and ℓ_2 norm of firing rates between the truth and estimators. (Decay rate: 0.98)

4.3.3 Simulation under a dynamic firing rate function

We also simulated data under dynamic firing rate functions across trials. Figure 4.5 (the upper left panel) shows a dynamic firing rate function with two peaks within each trial; across trials, the peak values first increase then decrease. The two-dimensional firing rate function at time point t and trial r is as follows:

$$f_r(t) = 0.01 + 0.19 \left(\exp\left\{ -\frac{(t-300)^2}{150^2} \right\} + \exp\left\{ -\frac{(t-700)^2}{150^2} \right\} \right) * \exp\left\{ -\frac{(r-R/2)^2}{1000} \right\} \tag{4.10}$$

where $t = 1, 2, ..., T$ and $r = 1, 2, ..., R$.

Figure 4.5 summarizes the simulation results with $T = 1,000$, $\gamma = 0.96$, $\sigma = 0.15$ and $R = 50$.

The sampling rate is 50 Hz and the length is 20 seconds per trial. The maximum of the firing rate is 0.2 (equivalent to 10 spikes per second). The results are based on 100 simulations. In firing rate estimation, Gaussian kernel smoothing is applied with a kernel bandwidth of 200 ms within trials and a window length of 10 across trials. Similar to the simulation in Section 3.2, MTV-PAR performs much better than using a constant penalty in estimating spikes and firing rate. Specifically, at the optimal λ (based on minimal VP distance), the reduction brought by MTV-PAR in VP distance for spike detection is 11.5% and the reduction of ℓ_2 error of firing rate estimation is 42.1%.

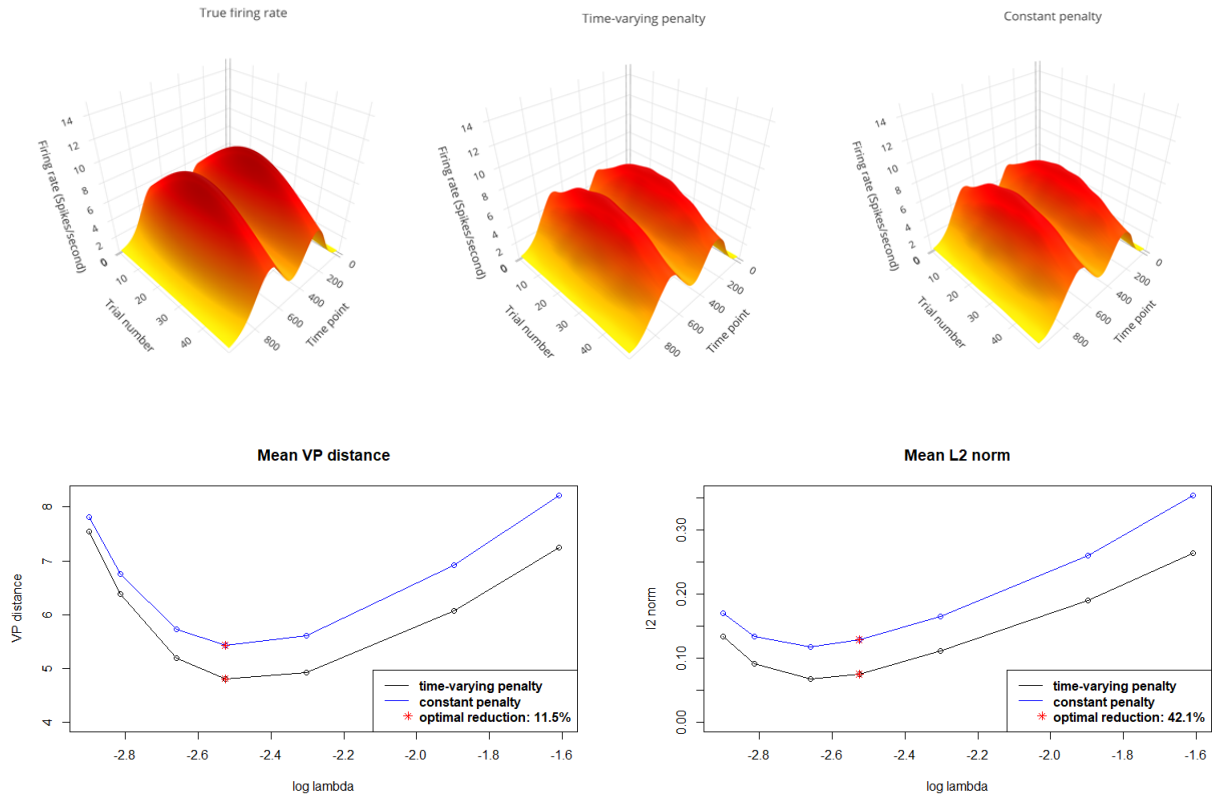

Figure 4.5: Top Left: True firing rates; Top Middle: Estimated firing rates using MTV-PAR; Top Right: Estimated firing rates using constant penalty. Bottom: VP distance of spike trains and L2 norm of firing rates between the truth and estimators.

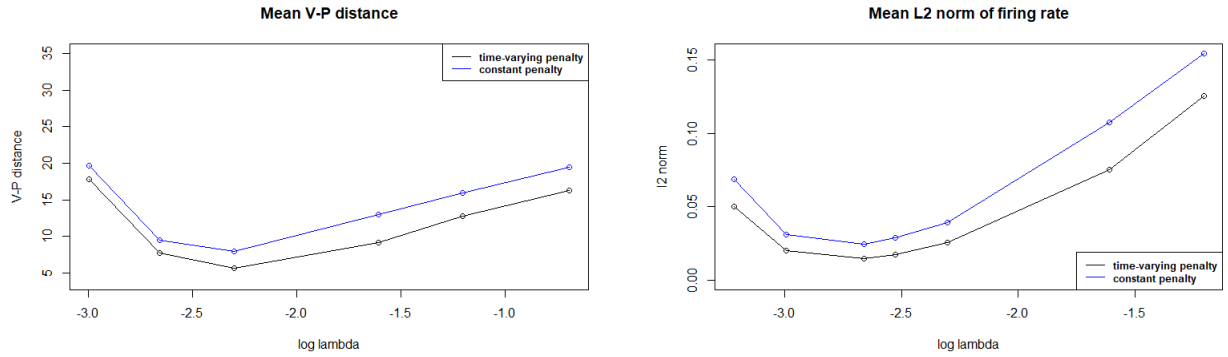

Figure 4.6: Top: firing rate estimates using MTV-PAR (time-varying) and constant penalty. Bottom: VP distance of spike trains and ℓ_2 norm of firing rates between the truth and estimators. (Series length $n = 2,000$)

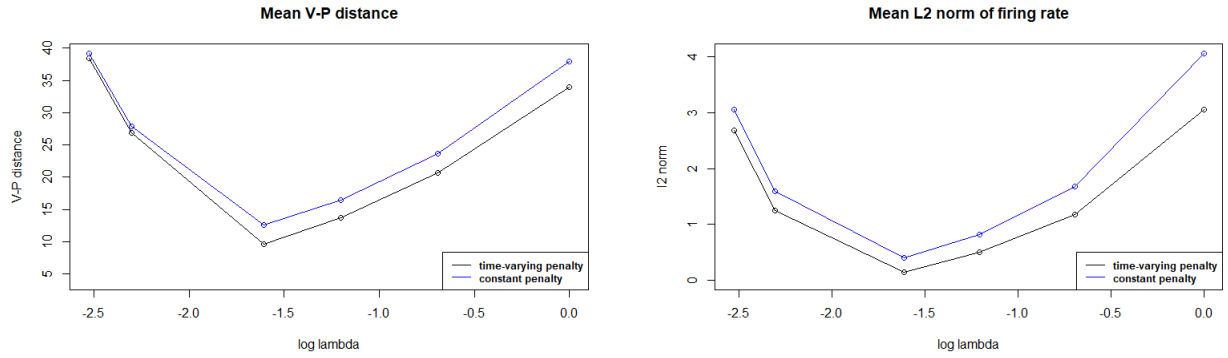

Figure 4.7: Top: firing rate estimates using MTV-PAR (time-varying) and constant penalty. Bottom: VP distance of spike trains and ℓ_2 norm of firing rates between the truth and estimators. (Standard deviation of noise: 0.3)

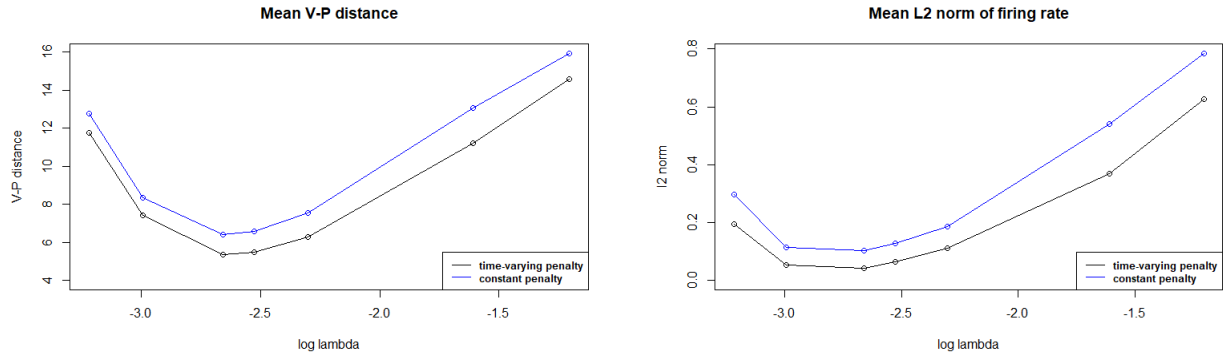

Figure 4.8: Top: firing rate estimates using MTV-PAR (time-varying) and constant penalty. Bottom: VP distance of spike trains and ℓ_2 norm of firing rates between the truth and estimators. (Decay rate: 0.98)

4.4 Application to Calcium Imaging Data

We now pursue the scientific investigations on the neural activities in two longitudinal mice studies in which both calcium imaging data and behavioral data were recorded. In this first data set, calcium imaging data were collected within a few days after the participating mice had been trained for a discrimination task; thus, it is reasonable to treat trials with same behavior outcome as repeated trials. As a comparison, in the second study, calcium recording started in the first learning trial and lasted for a few weeks; therefore, neural dynamics are expected as a result of the learning process and neural plasticity. For this reason, incorporating neuronal dynamics in firing rate estimation is likely to improve spike detection and firing rate estimation.

4.4.1 Mouse task data I

The activity of neurons (labelled with GCaMP6s) from the mouse's anterior lateral motor cortex (ALM) was recorded with two-photon calcium imaging during a head-fixed whisker-based discrimination task (Li et al., 2015). In the experiment, mice were supposed to discriminate the pole locations using their whiskers and report the perceived pole position by

licking. Each trial is composed of three epochs: sample epoch (mice presented with a vertical pole), delay epoch (the pole was removed), response epoch (mice cued to give a response). If a mouse licked the correct lick port, it was rewarded with liquid.

The data we present here is from one mouse with 73 trials from multiple days. Each trial contained more than 100 data points at 15 Hz. The fluorescence data were obtained after typical pre-processing procedures such as correction for neuropil contamination and $\Delta F/F_0$ transformation where F_0 is the baseline averaged fluorescence within a 0.5s period right before the start of each trial. There were four possible behavioral outcomes in the experiment: correct/incorrect lick left/right. Because most of the trials were either "correct lick left" or "correct lick right", we combined the two incorrect groups as a single group. Figure 4.9 shows the calcium fluorescence traces of a pyramidal tract neuron in 73 trials, including 31 trials of "correct lick left", 21 trials of "correct lick right", and 21 trials of "incorrect lick" .

One interesting question is whether the neuron responded differently for different outcomes. Therefore, we conducted a stratified analysis for each outcome. In the firing rate estimation, we use a Gaussian kernel smoothing with bandwidth 200 ms within trials. Since the mouse has been well trained before calcium imaging recording, firing rates across trials under the same outcome group are relatively stable, which was confirmed by available 2D visualization (data not shown). Thus, it is sensible to combine all the trials within an outcome type when estimating firing rate.

As indicated in Figure 4.9, in "correct lick left" trials, the neuron fired right after the cue time (when the mouse was cued to make decision); however, there was almost no neural activity under the other two outcome groups. The estimated firing rate function also confirmed this difference. It is known that the ALM brain region of mice is involved in planning directed licking (Guo et al., 2014). The estimated spikes and firing rate functions provide convincing evidences that this neuron is likely to show neural selectivity and play a critical role in the cognitive process of making the correct decision of licking the left pole.

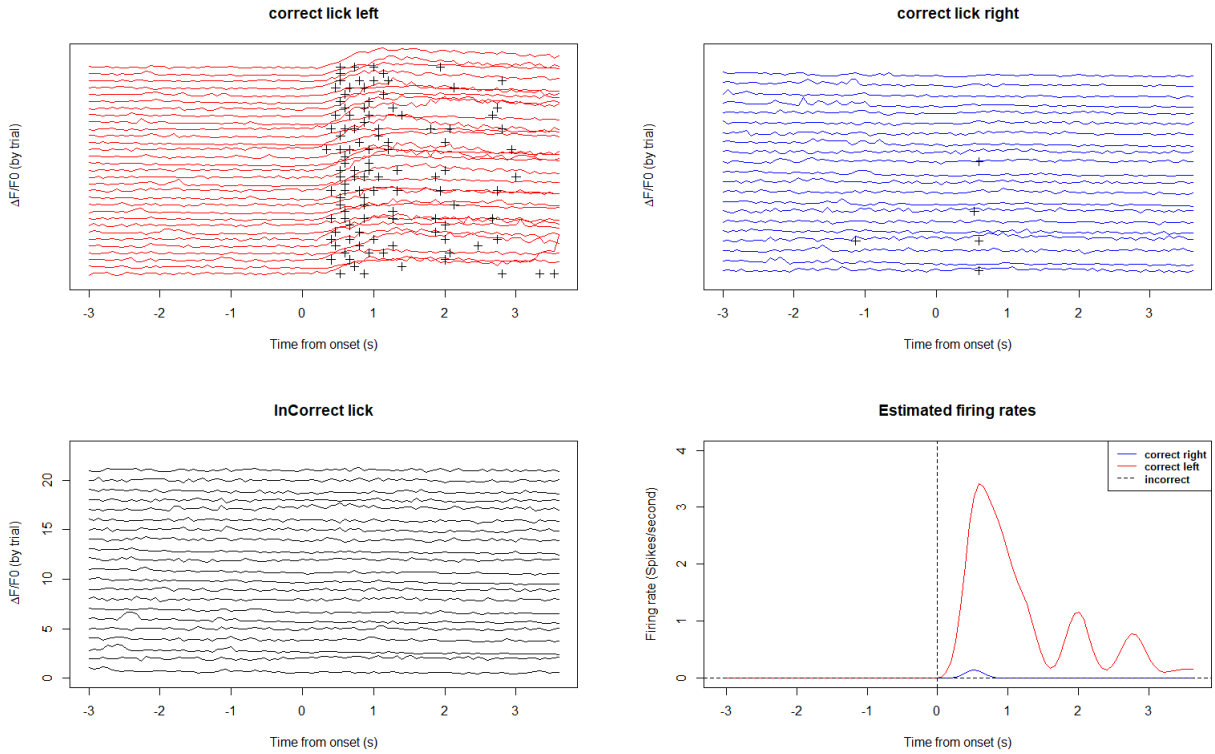

Figure 4.9: An example cell from the water lick experiment (Li et al., 2015). Top Left: The calcium fluorescence trace under correctly lick left outcome (black dots: estimated spikes). Top Right: The calcium fluorescence trace under correctly lick right outcome. Bottom Left: The calcium fluorescence trace under incorrect lick outcomes. Bottom Right: Estimated firing rate functions of under different conditions. Vertical dashed line: the start of response epoch.

4.4.2 Mouse task data II

The second data set is from our study of long-term contextual discrimination experiment, in which mice were trained to recognize two contexts via fear conditioning (foot shock) (Johnston et al., 2020). The research goal was to understand the behavior-associated hippocampal neural ensemble dynamics at single-cell resolution. Viral injections were administered into mice brain to introduce a genetically encoded calcium indicator (GCaMP6f). Fluorescence signals from hippocampal CA1 excitatory neurons were then optically recorded using one photon head-mounted miniscopes from freely moving mice (Figure 4.10, upper left). There were four stages in the experiment: habituation (mice freely exploring environment), learning

(learning to freeze in a stimulus context with foot shock), extinction (no foot shock), and relearning (stimulus reinstated).

In the mouse we analyzed, the activity of 141 neurons were recorded for several weeks. We chose the 21 foot shock sessions with the first 11 sessions in the learning stage and the remaining 10 sessions in the relearning stage. In each shock session, a foot shock was administered and the fluorescence trace was recorded at 15 Hz for 2 minutes. Figure 4.10 (upper right panel) shows an example neuron. The calcium fluorescence traces from different trials were temporally aligned by the start of shock time. For the firing rate estimation, we apply the Gaussian-boxcar kernel smoothing with a bandwidth of 400 ms for within trials and a window length of 5 for across trials. The estimated spikes (Figure 4.10, upper right) and firing rate functions (Figure 4.10, lower left) suggest that the neuron is more synchronous to the stimulus in the relearning stage than in the learning stage, which may reflect the evolving learning-related neuronal dynamics.

It is worth noting that treating the trials as independent realizations of the same underlying process will lead to undesired results. As shown in Figure 4.10 (lower right panel), when assuming the trials as samples from the same underlying distribution, the estimated peak firing time is misleading. Although the stratified estimates from the two stages showed that neuronal firing took place sooner and was more frequent in the relearning stage than in the learning stage in response to the foot shock stimulus, the results based on stratified analysis were not able to completely characterize the intrinsic evolution of neural firing during the cognitive learning process.

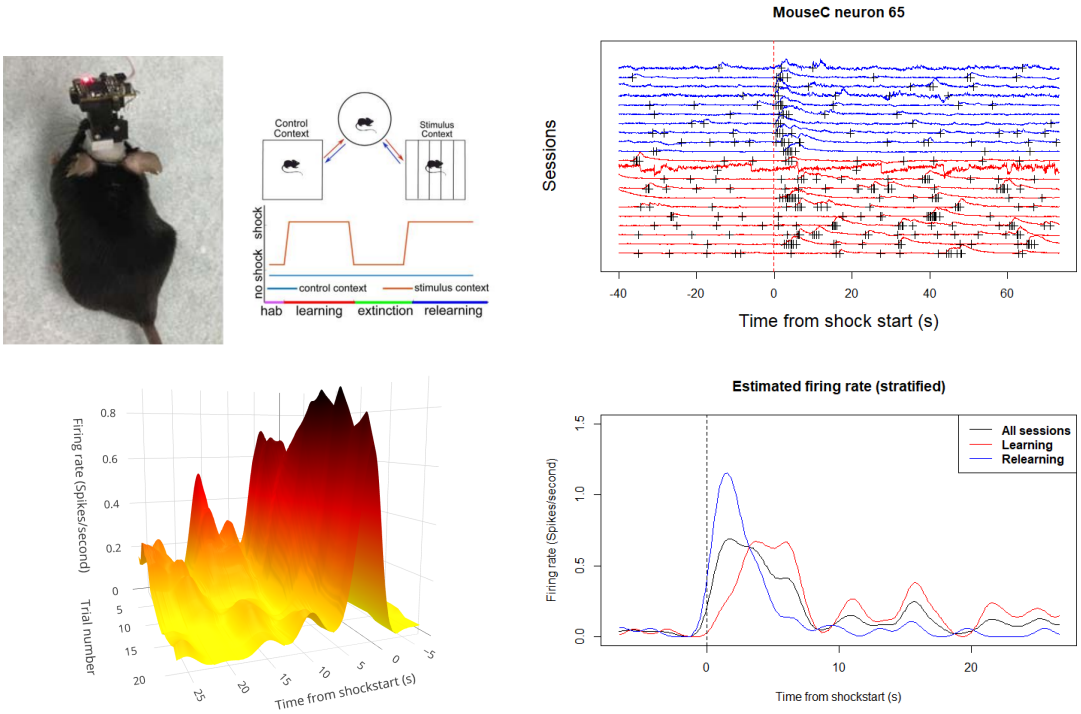

Figure 4.10: An example neuron from the foot shock study. Top Left: Calcium imaging of mouse hippocampus using a miniaturized scope and the four stages of the experimental design: habituation, learning, extinction, and relearning (Johnston et al., 2020). Top Right: Calcium fluorescence traces and estimated spikes of a sample neuron. The red and blue traces were obtained during the learning and relearning stages, respectively. The long dashed vertical line denotes the time when a foot shock was applied and the short black vertical lines are the time locations at which spikes were detected. Bottom Left: the estimated firing rate function using MTV-PAR. Bottom Right: the estimated firing rate function using stratified analysis. Vertical dashed line: the start time of foot shock.

Chapter 5

To Deconvolve, or Not to Deconvolve: Inferences of Neuronal Activities using Calcium Imaging Data

5.1 Introduction

In Chapter 4, we focus on spike deconvolution problem. In this Chapter we will follow up this topic and the goal is to assess the performance of using calcium trace and estimated spikes for common analyses: clustering, PCA and decoding. We start with providing background on using calcium trace and the comparison to electrophysiology data, which leads us to better understand calcium trace results and compare with estimated spikes.

With the technical advances in multiple fields (Chen et al., 2013; Dombeck et al., 2010; Ghosh et al., 2011; Grienberger and Konnerth, 2012; Yang and Yuste, 2017), calcium imaging has been increasingly adopted as a supplement or substitute to the traditional electrophysiological methods for measuring neuronal firing activities. Compared to electrophysiological methods,

imaging methods offer flexibility in a number of ways such as better spatial resolutions, longer follow up time, ability to study not only awake but also freely moving animals, and the larger number of neurons that can be simultaneously measured. One trade-off for the greater flexibility is the requirement of more sophisticated pre-possessing, as the measurement of neuronal activities from calcium imaging is indirect, and its estimate of calcium concentration is complicated by several factors such as measurement noises and contamination of signals from non-neuronal cells Johnston et al. (2020). As a consequence, each calcium trace is only a proxy of the underlying spiking activities with a reduced signal-to-noise ratio and temporal resolution. Thus, recent comparisons of electrophysiology and calcium imaging data are timely to guide us on how to interpret the results obtained from calcium imaging data. Using matched neuron populations and experimental conditions, Wei et al. (2020) reported both consistent and divergent results between electrophysiology and calcium data on temporal dynamics (within each trial), trial-type selectivity, sources of variances, and population decoding.

Despite the continued improvement in the quality of calcium imaging due to state of the art optical imaging devices and techniques, sensitive genetically encoded indicators, and pre-processing methods, extracting the underlying spike activities that would otherwise be accurately measured by electrophysiology data is still one major challenge in analyzing calcium imaging data in neuroscience research. The measured fluorescence intensity of calcium concentration is characterized by a rapid rise but slow decay after the occurrence of an action potential. As a result, numerous spike deconvolution methods have been proposed, from the simple thresholding with 2-3 standard deviations away from the calcium trace baseline to formal and fully Bayesian models (Jewell and Witten, 2018; Pachitariu et al., 2018; Pnevmatikakis, 2019; Pnevmatikakis et al., 2016; Vogelstein et al., 2010; Yaksi and Friedrich, 2006), including two of our recent work on multi-trial data (Johnston et al., 2020; Shen et al., 2021). Comparisons between estimated spike data and the true spike data from simulations or benchmark data usually showed that deconvolution leads to satisfactory results; on the other

hand, the estimated spikes might be inconsistent across methods, which produces different estimates of firing rate, number of tuned neurons, and distribution of estimated firing rates (Evans et al., 2019).

Some research problems, such as investigating temporal coding (Abbott, 1994; Abeles et al., 1993; Theunissen and Miller, 1995) and the estimation of instantaneous firing rates, requires the precise timing of action potentials. In addressing many other scientific questions, however, either calcium traces or estimated spike data can be used. For visualization purposes, both heat maps of calcium traces and raster plots of estimated spike trains are commonly presented. Cluster analysis is another example. It has been conducted to group neurons or trials with similar temporal patterns using spike train data (Adler et al., 2012; Humphries, 2011). Due to the increased availability of calcium recordings, in recent work, calcium trace data have also been directly used for clustering (Barbera et al., 2016; Dombeck et al., 2009; Ozden et al., 2008). Cluster analysis based on imaging data suggest that spatially compact neural clusters exist in awake mouse motor cortex at not only macrocircuitry but also microcircuitry levels (Dombeck et al., 2009). These clusters may represent cell assemblies that are stable over days; their dynamics often represent unique behavioral states and carry useful encoding information for behaviors (Barbera et al., 2016). When calcium imaging data are recorded, one can conduct cluster analysis using either calcium traces or the Deconvolved spike data (Romano et al., 2017). In this situation, one natural question is should one conduct cluster analysis on calcium traces or the Deconvolved data?

A related analysis is principal component analysis (PCA). Compared to cluster analysis, which groups observations to homogeneous clusters, PCA aims to extract component (features) that can keep as much as possible the variation in the original data. In neuroscience, PCA is frequently performed for dimension reduction and data visualization. Quantifying the dimensionality defined based on the corresponding eigenvalues may also shed light on the dynamics of the underlying neural circuits of various tasks and stimuli (Gao et al., 2017). PCA

analysis can be conducted to both electrically recorded spikes trains and optimally recorded calcium imaging data (Cunningham and Byron, 2014). One advantage of calcium recording over electrophysiology recording is that calcium imaging allows a large number of neurons to be recorded simultaneously. A popular method to compactly visualize the neuronal dynamics is to plot PCA trajectories over time using the first two or three components (Churchland et al., 2012; Cunningham and Byron, 2014). PCA is more straightforward for calcium traces, which are continuous values. Methods aiming to identify components using spike train data often filtered such as Gaussian kernel (Churchland et al., 2012) and other choices (Paiva et al., 2010). When the components of the true spike activities are of interest, PCA using calcium trace data might not be desirable, as a substantial source of the variation might be from the noises, rather than the signals, in the calcium trace data. In this situation, deconvolution might be helpful to recover the true underlying components.

Another important analysis in neuroscience is decoding, which refers to finding the mapping from neural activities to either external stimuli such as presented images and experimental types or animal behavioral outcomes such as movements, speed, and positions, and decision making. An interesting question is whether and how information is represented in an ensemble of neurons. Thus, population decoding methods have been widely adopted to investigate of the joint activities of a group of neurons using multiple spike trains (Brown et al., 2004). For example, Yang and Masmanidis (2020) analyzed simultaneously measured spike train data from two brain regions to compare their population decoding of choices in a two-alternative choice task. With the increased popularity of using calcium imaging to measure a large population of neurons, more and more population decoding analyses are conducted using calcium trace data. In most published work, the calcium trace data were first Deconvolved to spike train data before they were used to population decoding. A recent study (Wei et al., 2020) found that, as expected, the decoding accuracy of the Deconvolved spike data from calcium traces was much lower than the eletrophysiology data; but counter-intuitively, calcium trace data have higher predictability than their Deconvolved spike train data and

the less noisy eletrophysiology data. A possible explanation suggested by Wei et al. (2020) is, due to the slow decay rate of the observed calcium transients, the calcium trace data provides prediction based on integrating effects rather than instantaneous decoding. This is evidenced by the improved predictability of the electrophysiology data when a filter of 1 second was applied. It is unknown whether it is beneficial to conduct deconvolution when "integrating effects" are taken consideration to make a fair comparison between calcium traces and their Deconvolved spike trains.

In this paper, our goal is to examine the necessity of spike estimation for calcium trace data in three widely used methods, which are cluster analysis, PCA analysis, and population decoding.

5.2 Cluster Analysis

K-means is perhaps the easiest to understand but also the most widely used clustering method. The idea is to allocate a neuron to the cluster with the nearest centroid. Its improved versions have also been used in clustering either neurons or trials. For example, the meta k-means, which is a consensus clustering methods that aggregates clustering results from multiple runs of k-means, has been developed to increase the clustering stability of neurons (Ozden et al., 2008). To account for clustering uncertainty when clustering trials, Bezdek (2013); Dunn (1973) introduced fuzzy k-means, a method that produces probabilities of cluster assignments to each trial. When comparing the impact of calcium trace and estimated spike data on cluster analysis, we choose the fuzzy k-means to take the inherent uncertainty in clustering into consideration.

5.2.1 Fuzzy k-means

In the standard K-means algorithm, each data point is assigned to one cluster. The fuzzy k-means algorithm (FKM) was first developed in Dunn (1973) and improved in (Bezdek, 2013; Bezdek et al., 1984). It assigns each data point a probability of belonging to each cluster instead. Suppose we have the data $Y = \{y_1, y_2, \ldots, y_N\} \subset \mathbb{R}^p$. FKM is based on minimizing the objective function:

$$J_f(U, C) = \sum_{i=1}^{N} \sum_{j=1}^{K} u_{ij}^f \left\| y_i - c_j \right\|^2$$

where U is a fuzzy partition of the N data points y_i with K cluster centers $C = \{c_1, ...c_k\} \subset \mathbb{R}^p$. The numbers u_{ij} are the degree of membership of data point i to cluster j (the fuzzy partition) and $f > 1$ is the fuzziness factor. For most data, $1.5 \leq f < 3$ gives good results (Bezdek et al., 1984). In the simulation data used in Fellous et al. (2004), f is chosen as 2 and it gives robust results. The membership matrix satisfies $\sum_{j=1}^{K} u_{ij} = 1$. The fuzzy partition and the cluster centers were computed iteratively. At each iteration, the membership matrix and cluster centers were updated as:

$$u_{ij} = \frac{1}{\sum_{k=1}^{K} \left(\frac{d_{ij}}{d_{ik}} \right)^{\frac{2}{f-1}}}$$

and

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^f y_i}{\sum_{i=1}^{N} u_{ij}^f}$$

where

$$d_{ij}^2 = \left\| y_i - c_j \right\|^2$$

The iteration stopped when the norm of two consecutive membership matrix $\left\| U^{new} - U^{old} \right\|^2$ was smaller than a termination criterion ϵ.

5.2.2 Results based on simulated data

We first compare the clustering accuracy based on calcium trace and the Deconvolved spike data using a simulation study, where the true underlying cluster structure of the neurons are unknown. The spike train data we analyze here is a subset of the spike trains simulated by Fellous et al. (2004). Each simulated data set consists of three clusters, which are treated as three neuron clusters here. The 35 neurons (spike trains) within each cluster share 4-6 spikes with the spike times uniformly distributed between the time interval (0-1 second). The following three ways are used to add noises at various levels.

- 15% spike events are dropped randomly.
- X extra random spikes are added to each train.
- All the spike times were jittered by a value drawn from a normal distribution with mean 0 and standard deviation J million second (ms).

We consider the following seven noise levels, from well separated clusters to very "fuzzy" cluster memberships.

noise level	2	3	4	5	6	7	8
X	2	3	4	8	11	15	20
G	1	3	5	10	15	20	30

Table 5.1: The noise levels in the simulated spike trains from Fellous et al. (2004). X is the extra number spikes added per spike train; G is the standard deviation when jittering the spike times. The labels of the noise levels from Fellous et al. (2004) are used.

The raster plots for Figure 5.1 shows a set of spikes trains with low noise (left panel: level 2) and a set with high noise (right panel: level 8), respectively. In Fellous et al. (2004), 30 data sets were generated for each of the seven noise levels to account for variations in data simulations. For each of the $30 \times 7 = 210$ data sets, we generate 100 sets of fluorescence traces, where each fluorescence trace is generated for a given spike train using an AR(1)

model (Vogelstein et al., 2010). In the AR(1) model, the calcium fluorescence for a neuron at a single trial $y(t), t = 1, \cdots, T$ is modeled using the first-order auto-regressive model

$$y(t) = c(t) + \epsilon(t), \quad \epsilon(t) \sim N(0, \sigma^2),$$
$$c(t) = \gamma c(t-1) + s(t),$$

where $c(t)$ denotes the underlying true calcium concentration, $s(t)$ represents the change in calcium concentration between time points $t-1$ and t with $s(t) > 0$ indicates a spike at time t and $s(t) = 0$ otherwise, γ is the decay rate of calcium transients, and σ^2 denote the variance of the noise in measuring calcium concentration. In our simulations, the parameters we choose are as follows: the rate of decay $\gamma = 0.96$, the magnitude of each spike $s(t) = 1$ for any $s(t) > 0$, and two levels of Gaussian noise $\sigma = 0.1$ or $\sigma = 0.3$. To estimate spikes from simulated calcium data, we use the ℓ_0 penalized approach of Jewell and Witten (2018) for each calcium trace.

We perform cluster analysis using the fuzzy k-means (Fellous et al., 2004) with the assumption of three clusters to all the three types of data, namely, the calcium traces, the estimated spike trains from calcium traces, and the true spike trains. Their clustering accuracy is then quantified using two metrics - the rand index (Rand, 1971) and the normalized mutual information $I(A, B)$ (Danon et al., 2005) between the estimated and true cluster memberships. Both metrics measure the consistency between two categorical variables, with the maximum 1 indicating perfect agreement. Not surprising, as presented in Figure 5.2, the true spike data have the best clustering performance at all noise levels. Importantly, estimated spikes perform better than calcium trace at all noise levels.

In practice, the true number of clusters is unknown. To assess whether the results are sensitive to the choice of number of clusters, we also calculate the rand indices and mutual information for four and five clusters (Fig 5.3, 5.4). The results lead to the same conclusion, i.e., estimate spike data perform uniformly better than calcium trace on clustering.

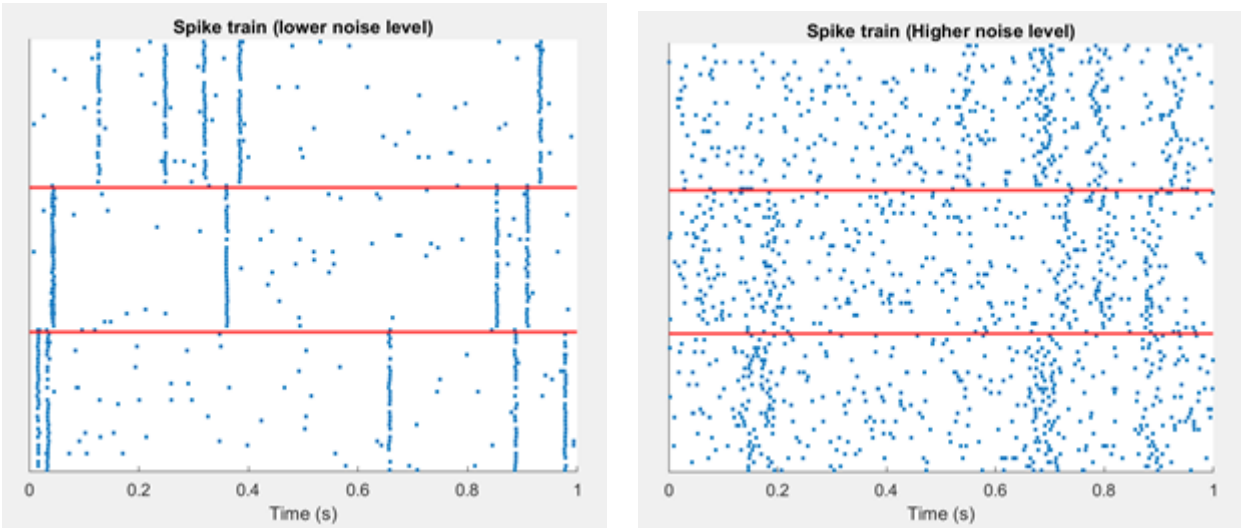

Figure 5.1: Rasterplot of a simulated spike data with $G = 3$ groups (separated by red lines) and 35 spike trains in each cluster. Left: with noise $X = 3$ extra spikes and $J = 3$ ms jitter. Right: with noise $X = 11$ extra spikes and $J = 20$ ms jitter.

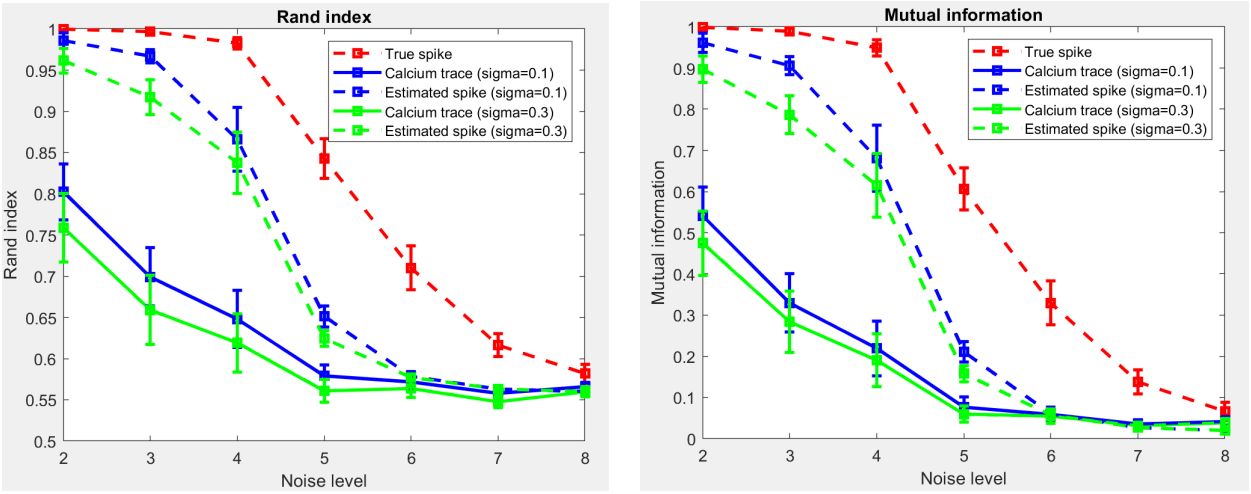

Figure 5.2: Clustering results for simulated data over seven noise levels (using three clusters). Presented are means and 95% confidence intervals over 30 data sets in each simulation setting. For the true spike data, each data point is averaged over 30 data sets; for data involving simulated calcium traces, at each noise level, clustering results from the 100 simulations for each of the 30 data sets are averaged first, and the results of the 30 data sets are then used to compute means and standard errors. Left: rand index. Right: mutual index.

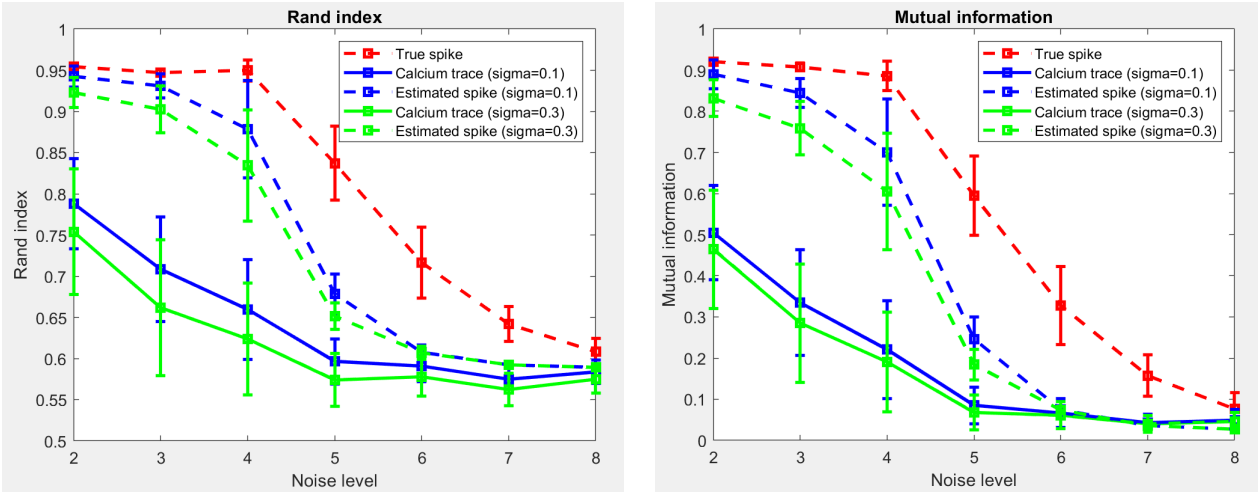

Figure 5.3: Clustering results for simulated data over seven noise levels (using four clusters). Presented are means and 95% confidence intervals over 30 data sets in each simulation setting. For the true spike data, each data point is averaged over 30 data sets; for data involving simulated calcium traces, at each noise level, clustering results from the 100 simulations for each of the 30 data sets are averaged first, and the results of the 30 data sets are then used to compute means and standard errors. Left: rand index. Right: mutual index.

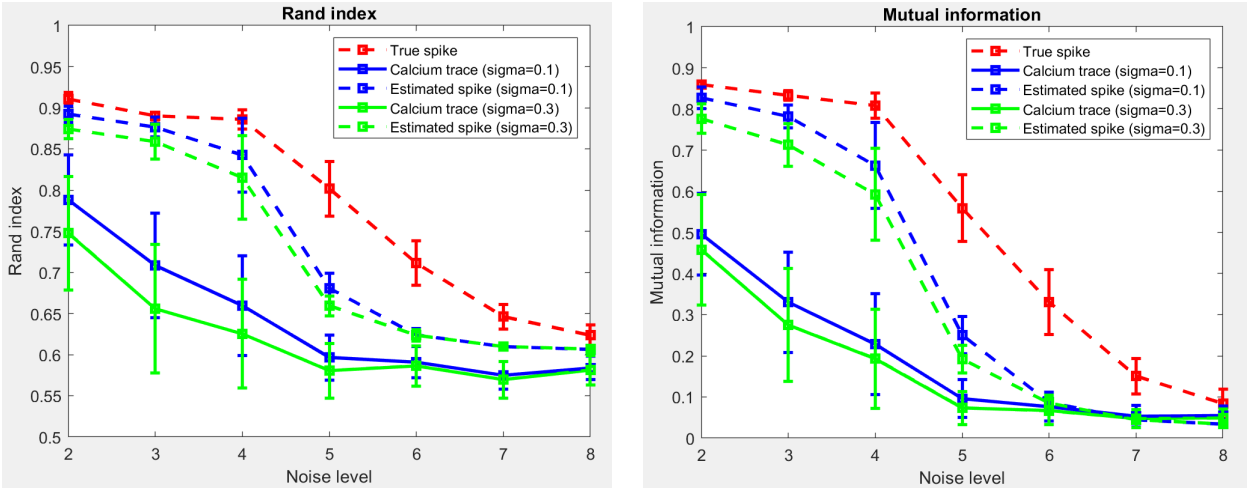

Figure 5.4: Clustering results for simulated data over seven noise levels (using five clusters). Presented are means and 95% confidence intervals over 30 data sets in each simulation setting. For the true spike data, each data point is averaged over 30 data sets; for data involving simulated calcium traces, at each noise level, clustering results from the 100 simulations for each of the 30 data sets are averaged first, and the results of the 30 data sets are then used to compute means and standard errors. Left: rand index. Right: mutual index.

5.2.3 A delayed response task

Next, we compare calcium traces and estimated spikes on clustering using calcium imaging data from two mice task studies. This application falls into the situation of no ground truth, as neither the true number of clusters nor the clustering membership is unknown a priori. This is a common situation, as benchmark data is not always available. Similar to Pachitariu et al. (2018), who used concordance of between repeated trials estimation as a metric for spike detection, we will use rand index to examine the between-trial consistency for estimated clusters.

In this multi-trial experiment, mice were first trained to discriminate pole location using their whiskers and reported the perceived pole position by licking (Li et al., 2015). Their neuronal activities in the left anterolateral motor cortex region were then measured using two-photon calcium imaging using GCaMP6s. Each trial consisted of three epochs: sample epoch (mice presented with a vertical pole), delay epoch (the pole was removed), and response epoch

(mice cued to give a response). Here we clustered the 67 neurons of a mouse that made 31 "correct lick left" trials and 21 "correct lick right" trials. Specifically, for each trial, we clustered the 67 calcium traces or the estimated spike trains into four clusters using the fuzzy k-means (Bezdek et al., 1984; Fellous et al., 2004). The rand index between every pair of two trials was calculated to assess the trial-to-trial consistencies.

Figure 5.5 shows that clusters based on estimated spike data are more consistent between trials than clusters based on calcium trace data. Similar conclusions are obtained when the neurons are clustered to three or five clusters (Figure 5.6, 5.7). Because the mouse was trained before the calcium recording, individual neurons were likely have been tuned and this type of consistency across trials is expected. Therefore, the higher rand index using spike data over calcium data might indicate better performance in cluster analysis.

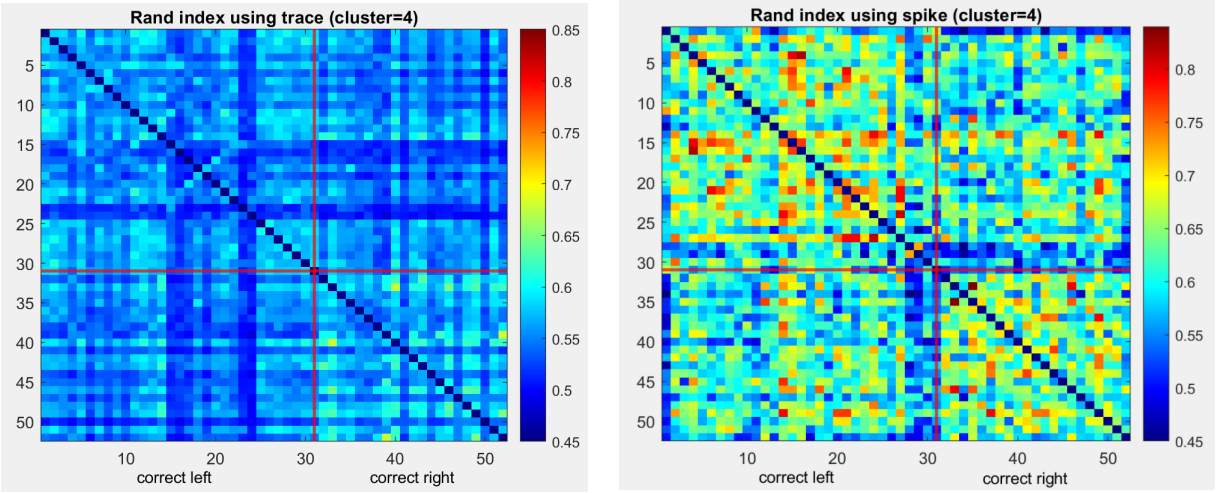

Figure 5.5: Rand index in water lick data of a mouse in one session (using four clusters) (Left: using calcium trace. Right: using estimated spike data). Each element in the matrix is the Rand index for the 2 trials of the corresponding row and column. First 31 trials: correct left trial. Last 21 trials: correct right trial.

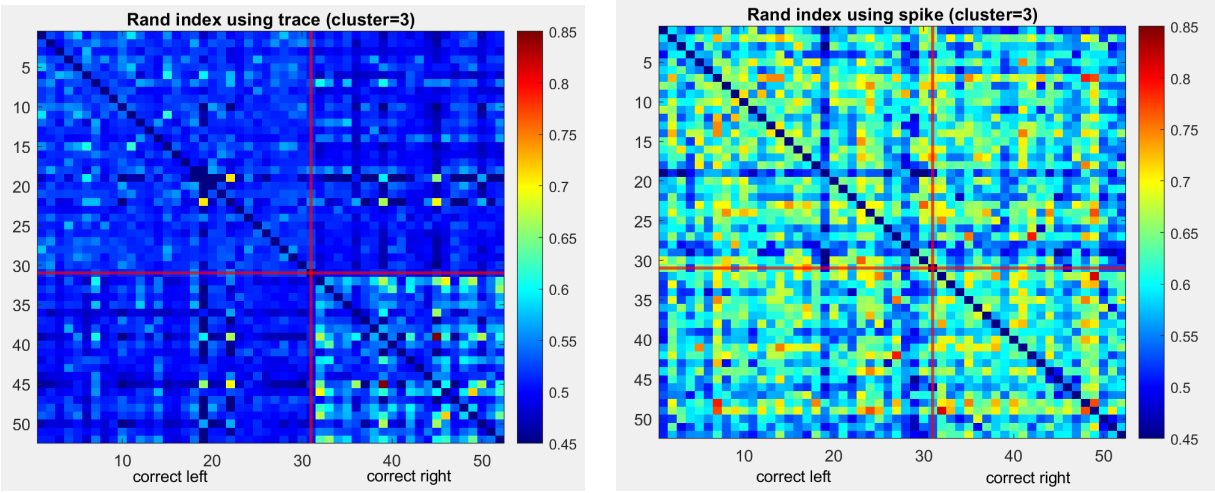

Figure 5.6: Rand index in water lick data of a mouse in one session (using three clusters) (Left: using calcium trace. Right: using estimated spike data). Each element in the matrix is the Rand index for the 2 trials of the corresponding row and column. First 31 trials: correct left trial. Last 21 trials: correct right trial.

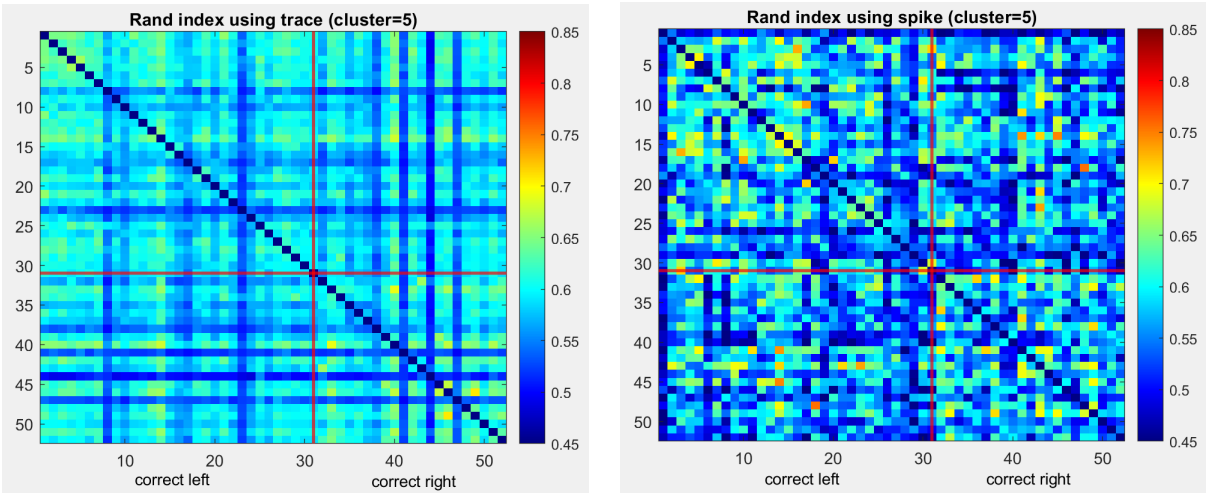

Figure 5.7: Rand index in water lick data of a mouse in one session (using five clusters) (Left: using calcium trace. Right: using estimated spike data). Each element in the matrix is the Rand index for the 2 trials of the corresponding row and column. First 31 trials: correct left trial. Last 21 trials: correct right trial.

5.2.4 A Fear-based contextual discrimination experiment

In one of our previous studies (Johnston et al., 2020), mice were trained to recognize two contexts via fear conditioning (foot shock). Meanwhile, fluorescent miniature microscopes (Ghosh et al., 2011) were used to track cell populations in contextual discrimination experiments in mice's hippocampus using a genetically encoded calcium indicator (GCaMP6f) for up to 60 days. The whole experiment included four stages: habituation (mice freely exploring environment), learning (learning to freeze in a stimulus context with foot shock), extinction (no foot shock), and relearning (stimulus reinstated).

We analyze a mouse whose 141 neurons were measured in 21 trials, with the first 11 trials in the learning stage and the last 10 trials in the relearning stage. For each trial, we clustered the neurons into four clusters using the fuzzy k-means (Fellous et al., 2004) using the calcium traces or deconvolved spike trains. The rand index based on the clustering results were computed for each pair of trials. The comparison between calcium traces and estimated spike data showed again that estimated spike trains leads to more trial-to-trial agreement in neuron clusters (Figure 5.8).

The conclusion is similar when the neurons were clustered to three clusters (Figure 5.9) and five clusters (Figure 5.10). Due to the lack of ground truth in neuron clusters, cautions should be taken when relating results to potential misspecifications of cluster numbers. We leave further investigations to future research.

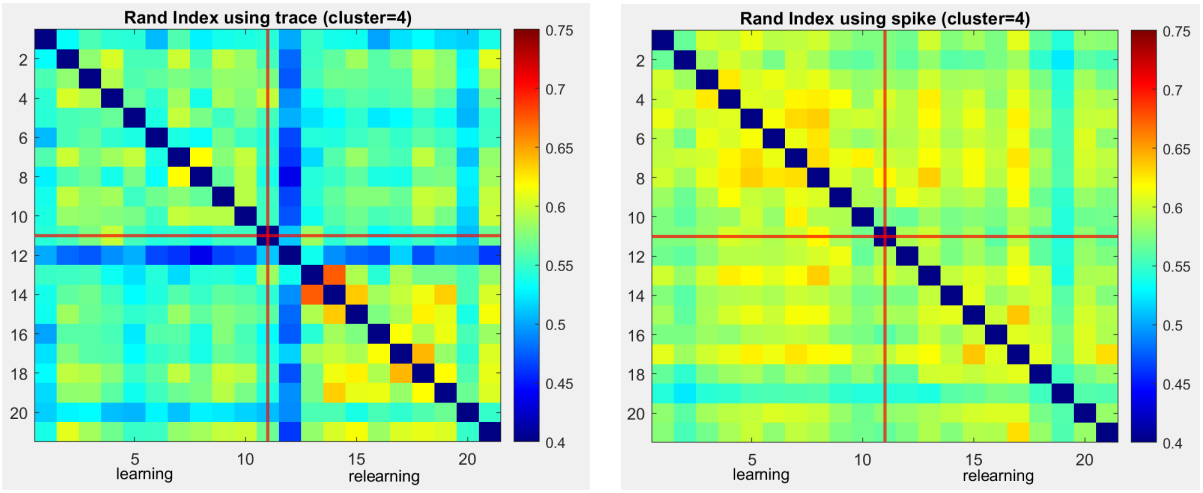

Figure 5.8: Rand index in fear conditioning data of a mouse in one shock session (using four clusters). (Left: using calcium trace. Right: using estimated spike data). Each element in the matrix is the Rand index for the 2 trials of the corresponding row and column. First 11 sessions: learning session. Last 10 sessions: relearning session.

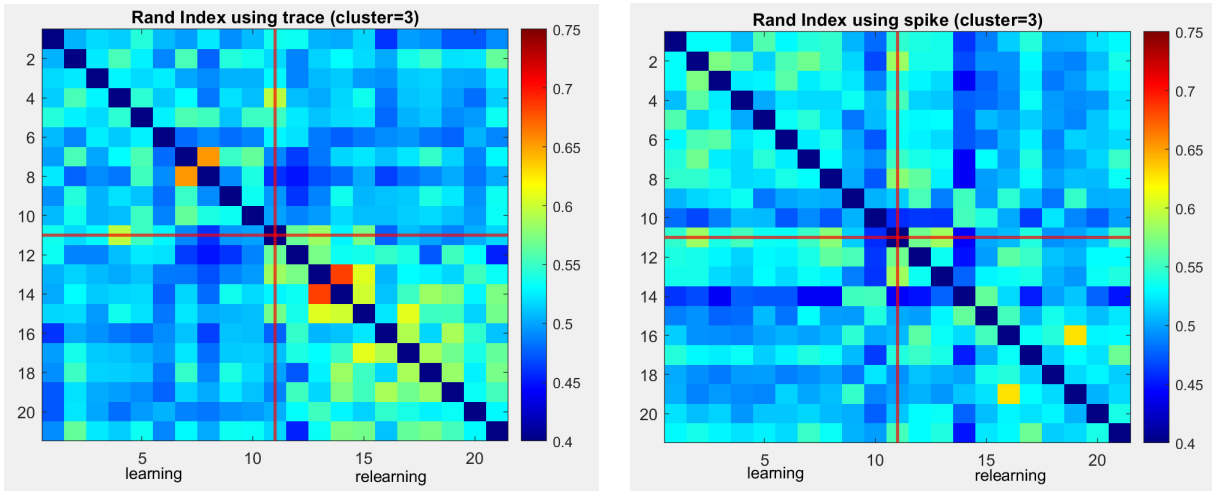

Figure 5.9: Rand index in fear conditioning data of a mouse in one shock session (using three clusters). (Left: using calcium trace. Right: using estimated spike data). Each element in the matrix is the Rand index for the 2 trials of the corresponding row and column. First 11 sessions: learning session. Last 10 sessions: relearning session.

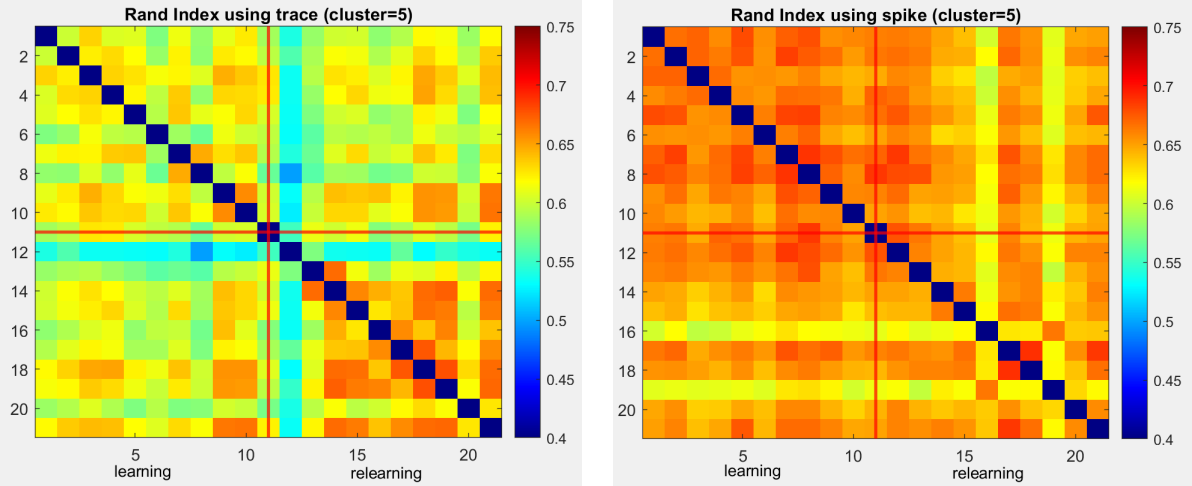

Figure 5.10: Rand index in fear conditioning data of a mouse in one shock session (using five clusters). (Left: using calcium trace. Right: using estimated spike data). Each element in the matrix is the Rand index for the 2 trials of the corresponding row and column. First 11 sessions: learning session. Last 10 sessions: relearning session.

5.3 Principal Component Analysis (PCA)

PCA is one of the most common methods in analyzing large scale neuron data (Chapin and Nicolelis, 1999; Churchland et al., 2012; Cunningham and Byron, 2014; Harvey et al., 2012; Kobak et al., 2016). In PCA analysis of neuron data, PCA extracts principal components that are linear combinations of individual neurons. For example, in Bekolay et al. (2014), PCA was used to capture major patterns of firing within the neuronal populations in prefrontal cortex. They looked in to the first two principal components to analyze the firing patterns in motor cortex by looking at example neurons and the histogram of the loadings of neurons onto the PCs.

Because the baseline firing rate is usually less than 1 spike/s, spike train data for most neurons are sparse. Thus, the conventional PCA method is not reliable. One way to handle sparsity is to consider kernels that are appropriate for sparse data. In the following section we introduce the Spike train PCA method.

5.3.1 Spike train PCA

Since spike trains are binary time series and the information is also often represented using the ordered spike times, the kernel function between two spike trains should applicable to two sets, rather than two vectors. (Carnell and Richardson, 2005) suggested compare all the spike times of one spike train to all the spike times of the other spike train, i.e.,

$$K\left(s_i, s_j\right) = \sum_{m=1}^{N_i} \sum_{n=1}^{N_j} \kappa\left(t_m^i, t_n^j\right)$$

where κ is a symmetric, shift-invariant, and positive definite kernel. In this article, we use the Gaussian kernel $\kappa(t_m^i, t_n^j) = \exp\{-(t_m^i - t_n^j)^2/\tau\}$.

The above spike train inner product has been adopted by other researchers (Carnell and Richardson, 2005; Paiva et al., 2009; Schrauwen and Van Campenhout, 2007). Recall that the components in the conventional PCA are based on the sample variance-covariance or correlation matrix, which is computed using centered data. To conduct kernel PCA using the spike train inner product, one also needs a similar centering process (Paiva et al., 2010), which can be shown equivalent to conduct the double-centering. Specifically, let $\mathbf{P}$ denote the inner product matrix of the spike trains, where $\mathbf{P}_{ij} = K(s_i, s_j)$. Then the double centered inner product is $\tilde{\mathbf{P}}$ is

$$\tilde{\mathbf{P}} = (I - \frac{\mathbf{1}_N \mathbf{1}_N^T}{N})\mathbf{P}(I - \frac{\mathbf{1}_N \mathbf{1}_N^T}{N})$$

where $\mathbf{1}_N$ is the $N \times 1$ vector with all ones. After obtaining the centered inner product matrix, eigen decomposition is performed on $\tilde{\mathbf{P}}$ to compute the eigenvectors and corresponding eigenvalues.

5.3.2 Results based on simulated Data

We generate spike trains using the method described in Paiva et al. (2010) where the spike trains are 1s long and have mean spike rate 20 spikes per second. The inter-spike interval was gamma distributed with shape parameter $\theta = 0.5$ and 3 (See Figure 5.11). For each generated spike train, a calcium fluorescence trace is simulated from an auto-regressive model (Vogelstein et al., 2010) with decay rate $\gamma = 0.96$, the magnitude of each spike $s(t) = 1$ and the standard deviation of noise $\sigma = 0.1$ and 0.5.

We first compare the estimated components in explaining the variances of the true spike trains of neurons. The simulation result is averaged over 100 simulated data sets. The cumulative proportions of explained variances are shown in the upper panel of Figure 5.12. Note that the curves based on the true spike trains (red) are very similar to those based on the estimated spike trains (blue) are quite different from those based on the calcium traces (black). The results are not surprising, as calcium traces are the noisy and integrated measurements of the true underlying spike activities. The PCA analysis using calcium traces also indicates that it requires a much large number of components to reach a certain percentage of variance. For example, when $\sigma = 0.1$, it requires less than five components for the true or estimate spike train but over 15 components for the calcium traces.

A summary metric based on the eigenvalues is the effective dimension of an embedding defined in Victor and Purpura (1997), which is also noted as the dimensionality (Gao et al., 2017). The effective dimension index E is defined as $E = \frac{(\sum \lambda_i)^2}{\sum \lambda_i^2}$ where λ_i is the i-th eigenvalue of the covariance matrix of calcium traces/true spike trains/estimated spike trains between neurons. The result in the lower panel of Figure 5.12 agrees with the explained variance and shows that the dimensionality calculated from calcium traces can be quite different from those the true and estimated spike trains.

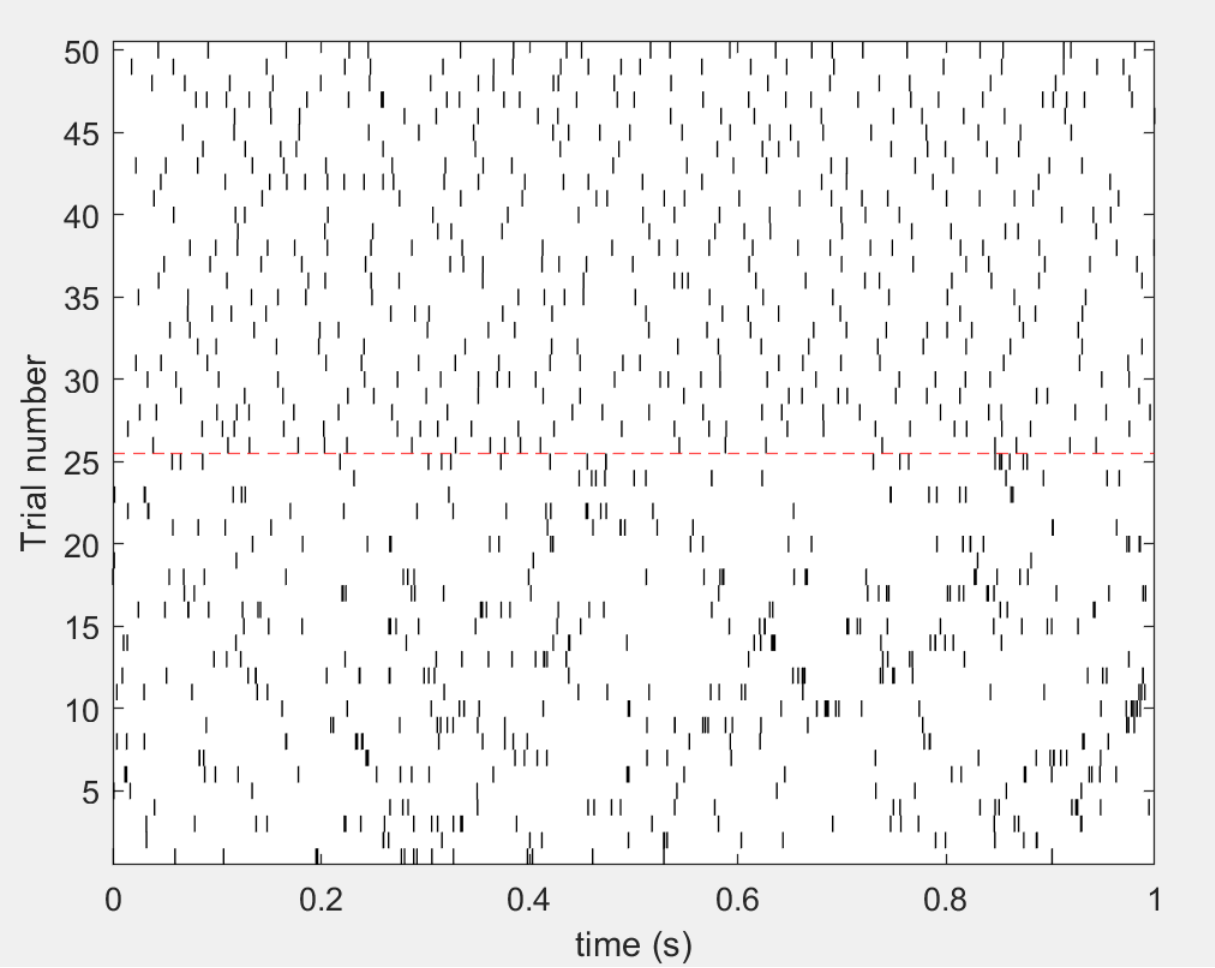

Figure 5.11: The raster plot of 50 simulated spike trains (1 second with rate 20 spikes/second). Top half: 25 spike trains with shape parameter of inter-spike interval $\theta = 0.5$. Bottom half: 25 spike trains with shape parameter of inter-spike interval $\theta = 3$.

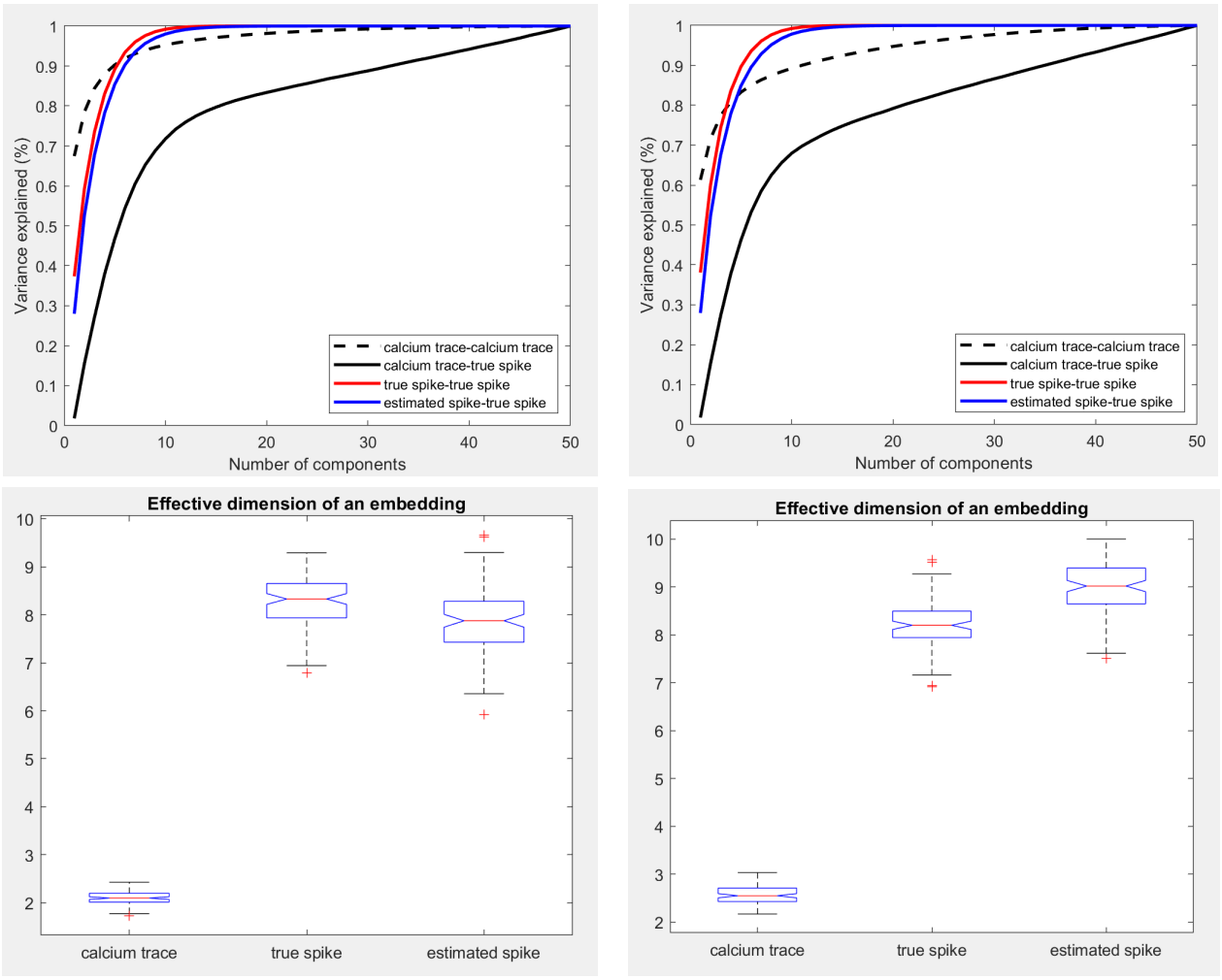

Figure 5.12: Cumulative variance explained by different methods: true spike trains (red), estimated spike trains (blue), and calcium traces (black), . Each curve is averaged over 100 simulated data sets. Left: noise of calcium trace $\sigma = 0.1$. Right: noise of calcium trace $\sigma = 0.5$. Bottom left: Boxplot of the effective dimensions of calcium trace/true spike data/estimated spike data, noise of calcium trace $\sigma = 0.1$. Each plot contains the result for 100 simulated data sets. Bottom right: effective dimension of the calcium trace/true spike data/estimated spike data, noise of calcium trace $\sigma = 0.5$. Each plot contains the result for 100 simulated data sets.

5.4 Population Decoding Analysis

Our goal in this section is to investigate whether deconvolution is helpful in population decoding. We follow a common practice that builds a predictive model using the neural activities at each time bin to predict an variable of interest (Berens et al., 2012; Meyers, 2013; Wei et al., 2020).

5.4.1 Data "pre-processing"

As reported in Wei et al. (2020), electrophysiology and estimated spike data are less informative in population decoding at each time bin. This is somewhat surprising but not totally unexpected. Two possible explanations are the integration effect of the calcium trace data (Wei et al., 2020) and the sparsity of spike data in each time window. One support of these explanations is that the one-second filtered electrophysiology data reached higher peak decoding accuracy than the original analysis of electrophysiology data with a bin size 67ms. In fact, temporal filtering is a common practice in population decoding and choosing the appropriate filtering encourages data integration temporally, which might lead to improved decoding accuracy Yates et al. (2020). Here we consider two strategies to take neural firing history into consideration by applying "pre-processing" to the sparse count data. The first method is to use cumulative counts, which is defined as the total number of spikes up to a time t. The second method, which is labeled with "spike history", uses the matrix of the time series (up to time t) from a neuron population as input in decoding.

5.4.2 Data types

Note that the spike detection framework based on AR(1) model in 4.1 not only produces estimated timestamps of spike events but also provides estimates of spike magnitudes, denoted

by $s(t)$. The spike magnitude $s(t)$ was originally motivated by the number of spikes in a small time bin Vogelstein et al. (2010). Although the interpretation of a positive $s(t)$ is not clear, it might still provide useful quantitative information related to spike counts and firing rates and therefore useful for population decoding. Another quantity estimated by the AR(1) model is the calcium concentration $c(t)$ in the AR(1) model. The estimated $c(t)$ can be considered as a denoised version of the observed noisy calcium trace. Interestingly, $c(t)$ can also be viewed as a filtered version of the underlying spike train. As spike train filters are often applied in various spike train analysis in neuroscience research, it is reasonable to use the estimated $c(t)$ as a candidate for population decoding. Thus, we will focus on the following four types of data: the raw calcium traces, the estimated spike trains, the estimated changes $s(t)$, and the estimated/denoised trace $c(t)$.

5.4.3 Prediction methods

Taken together, the data types and ways of pre-processing create various combinations of decoding, as shown in Table 5.2. Note that in simulated data the "spike train" category includes true and estimated spike data whereas in a real study it includes spike train data from electrophysiology measurements and estimated spike train from calcium trace data.

	calcium trace	estimated trace $c(t)$	spike train	$s(t)$
original	✓	✓	✓	✓
cumulative			✓	✓
history			✓	✓

Table 5.2: Population decoding methods. The "cumulative" and "history" method for trace data are excluded due to lack of justification.

In decoding with trace and cumulative count, we apply linear support vector machine (SVM) (Cortes and Vapnik, 1995) with 5-fold cross validation on the training data. To assess the dependency of results on predictive models, we also used Fisher's linear discrimant analysis. Because they give similar results in the scenarios we consider, only results from SVM will

be reported. In decoding with spike history, the input predictor is a matrix with time and neuron dimension. Traditional method such as SVM reshapes the matrices into vectors, causing a loss of structural information. Here we apply a sparse support matrix machine method (SSMM) (Zheng et al., 2018), which is a regularized binary matrix classifier that uses a ℓ_1 penalty to ensure sparsity and a nuclear norm penalty to encourage low rank of the coefficient matrix. Given a set of samples $\{\mathbf{X}_i, y_i\}_{i=1}^n, \mathbf{X}_i \in \mathbb{R}^{p\times q}$ is the i_{th} predictor matrix and $y_i \in \{1, -1\}$ is its corresponding label. The proposed SSMM (Zheng et al., 2018) is based on support vector machine (Cortes and Vapnik, 1995) and it combines a hinge loss with a new regularization on the regression matrix $\mathbf{W}$. The objective function of SSMM method is presented as:

$$\arg\min_{\mathbf{W},b} \left\{ \lambda\|\mathbf{W}\|_1 + \tau\|\mathbf{W}\|_* + \sum_{i=1}^{n} \left\{1 - y_i \left[\text{tr}\left(\mathbf{W}^T\mathbf{X}_i\right) + b\right]\right\}_+ \right\}$$

where $\mathbf{X}_i, \mathbf{W} \in \mathbb{R}^{p\times q}$. It incorporates the hinge loss, ℓ_1 norm $\|\mathbf{W}\|_1$ and nuclear norm $\|\mathbf{W}\|_*$ on regression matrix $\mathbf{W}$ for matrix classification. The ℓ_1 norm controls the sparsity of $\mathbf{W}$ and the nuclear norm encourages $\mathbf{W}$ to be low-rank. An efficient algorithm to solve the optimization problem is presented in Zheng et al. (2018), where they smooth the loss function by using generalized smooth hinge loss with Lipschitz-continuous gradient. The computational cost is $\mathcal{O}\left(n^2pq\right)$.

For the tuning parameters τ and λ, it has been reported that non-zero values often give better results than zero (Zheng et al., 2018). We tried different combinations of the two parameters and compare the decoding accuracy on simulated and real data with all time points (results omitted).

5.4.4 Decoding accuracy for the water lick data

We evaluate the usefulness of deconvolution on population coding using a subset of the data published in Wei et al. (2020). Their primary goal was to compare spike and imaging data for measuring neuronal activities. They recorded neural activities in anterior lateral motor cortex using electrophysiology signals and calcium imaging. Although the electrophysiology and calcium imaging recordings were separate, neuron populations of matched depth were measured from the same delayed response task. Each 5 second long recording was composed of three epochs: sample epoch (mice presented with a vertical pole), delay epoch (the pole was removed), response epoch (mice cued to give a response). The duration of sample and delay epoch was 2.6 s. In electrophysiology, the sample epoch was 1.3 s while in calcium imaging was 1.2 s. Here we analyzed the calcium imaging data for 1,493 neurons from 4 mice with adeno-associated virus expressing GCaMP6s and electrophysiology data of 720 neurons from 19 mice. To match the temporal resolution of calcium trace data, the spike train data from electrophysiology recordings were temporally subsampled. Note that the data we downloaded from *https://doi.org/10.6084/m9.figshare.12786296.v1* has been pre-processed. We also followed several other strategies used in Wei et al. (2020), such as resampling trials to cope with the limited number of trials, resampling the same number of neurons when comparing different data types, using cross validation to choose tuning parameters (60% data for training and 40% data for testing). The full description of the experiments and data processing is available from Wei et al. (2020) and further backgrounds about the experimental design can be found in other articles of the group (Li et al., 2015; Wei et al., 2019).

Figure 5.13 visualizes the accuracy, which is defined as the proportion of correct predictions in testing data, of the prediction methods in Table 5.2. The top panels replicated the difference between the decodability of the calcium trace and the electrophysiology recordings in Wei et al. (2020). Although electrophysiology data showed earlier latency (defined as the time reaching %70 decoding accuracy), surprisingly, it had lower decoding accuracy than calcium

trace data, which are a noisy measurement of neural activities. This difference might be due to the integration effect of the calcium trace data, as a calcium transient is characterized by a rapid rise but slow decay. This hypothesis is supported by the improved decoding accuracy of the 1-second filtered electrophysiology data in most time bins. However, the 1-second filtered electrophysiology data lost the early latency. As a comparison, electrophysiology history, a new method considered here, outperforms the calcium trace data in all time bins. Thus, when used appropriately, the less noisy spike train data from electrophysiology recordings do have higher population decodability.

The lower panel of Figure 5.13 focused on the effects of deconvoluting calcium traces. Consistent with Wei et al. (2020), the estimated spike data has lower decodability than both calcium traces and electrophysiology data. The magnitude of changes, i.e., $s(t)$ has a similar accuracy rate. However, their cumulative and history have much higher accuracy, with their cumulative outperforming the calcium trace data. It is worthy pointing out that the estimated calcium concentrations ("estimated calcium trace"), which is a denoised version of the calcium traces, also demonstrates higher decoding accuracy than the original calcium traces. These results indicate that the decodability of a neuron population can be improved by appropriate analysis of the Deconvolved data.

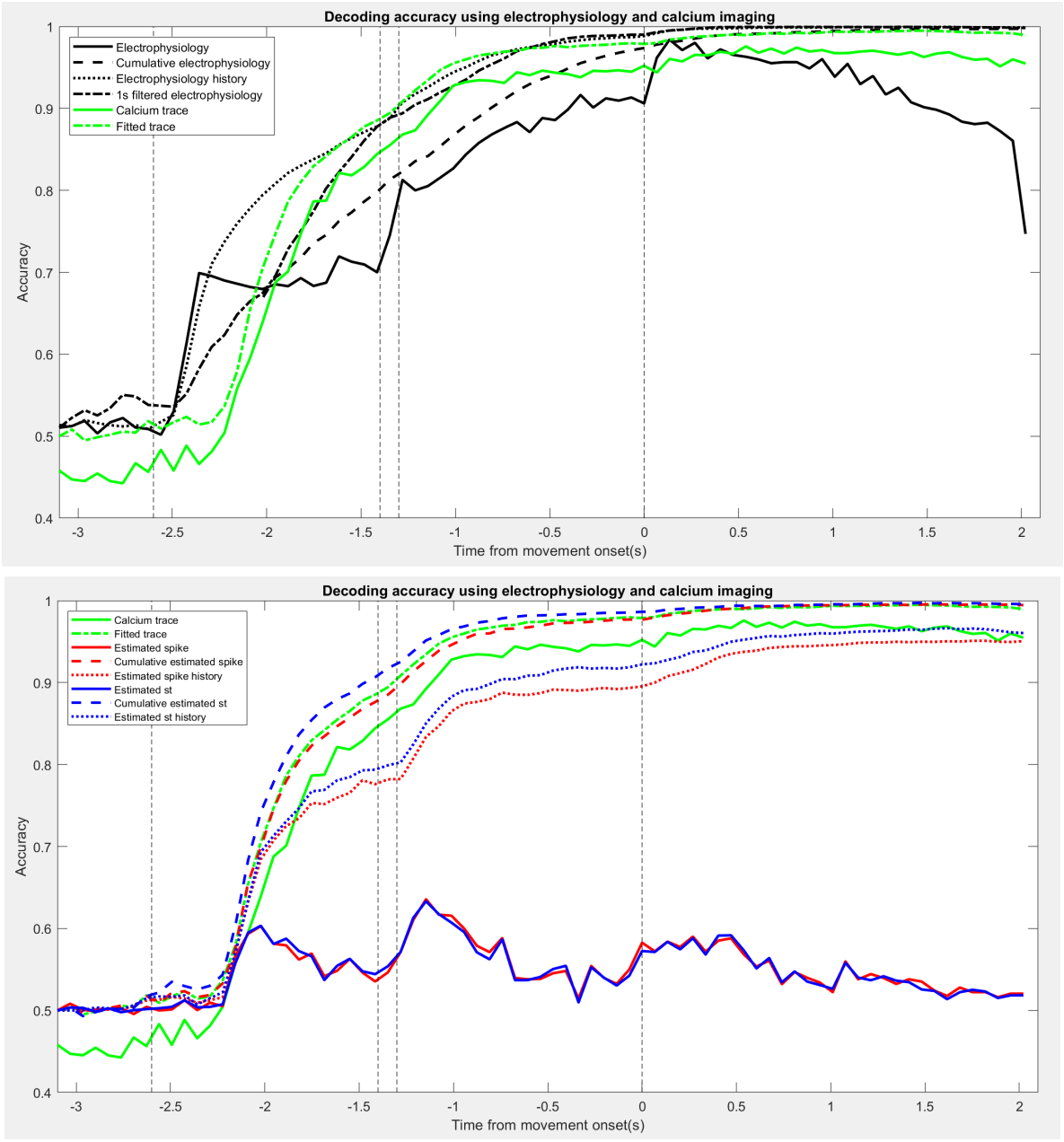

Figure 5.13: Decoding results in water lick data. Top: decoding accuracy using electrophysiology and calcium imaging data. Black lines: electrophysiology. Green lines: calcium trace. vertical dashed lines from left to right: the start time of sample epoch, delay epoch of calcium imaging data, delay epoch of electrophysiology, and response epoch. Bottom: decoding accuracy using calcium imaging data and spikes. Green lines: calcium trace. Red lines: Estimated spikes from calcium trace. Blue lines: Estimated spike magnitude from calcium trace. Decoding results are averaged over 100 subsamples. In each subsample, 50 neurons are randomly selected and 500 trials are sampled from each neuron. vertical dashed lines from left to right: the start time of sample epoch, delay epoch and response epoch.

Chapter 6

Conclusions and Future Directions

In this dissertation, we discuss on change point detection and its applications in neuroscience research.

In Chapter 3, we proposed the Spec PC-CP (Spectral Principal Component Change Point detection) method, a new change-point detection approach on multivariate time series. The approach consists of two stages: using Spectral PCA to obtain a low-dimensional time series (stage 1) and then applying a binary segmentation algorithm to detect change points (stage 2). The simulation results show that our method outperforms competing methods in capturing the lead-lag relationship between time series. We applied the method to the EEG seizure data. Change points were identified at the beginning and around seizure onset. It suggests possibility to build early warning systems of seizure and gives more precise diagnosis, as changes are often not obvious to be observed through visual inspection. We also analyzed the stock data with daily log returns of nine representative stocks in $S\&P$ 100. Change points were detected around events that have impact on the financial market. It demonstrates that our method is efficient in analyzing financial data, which could lead to insights in understanding the fluctuation of financial market.

There are several potential future directions. First, in offline detection, we detect the changes that have already occurred. It would be helpful to consider forecasting time series to further improve our analysis. For example, Fryzlewicz (2005) modeled and forecast financial log-return series in the Locally Stationary Wavelet (LSW) framework. Other forecasting methods such as Autoregressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM) also have been adopted in modern time series analysis (Siami-Namini et al., 2018). We could combine Spectral PCA and forecasting methods to predict future events such as seizure and financial crisis. Besides, in current detection method, we only got the point estimates of change points. It lacks the ability to quantify the uncertainty. We could consider Bayesian methods. For example, Nam et al. (2015) used locally stationary wavelet (LSW) framework combined with hidden Markov model to quantify the uncertainty of auto-covariance changes. Moreover, bootstrap can also be used to make inference on change point estimates. However, naive bootstrap does not take the order and structure of time series into consideration. People have developed block bootstrap methods (Carlstein et al., 1986; Kunsch, 1989) to keep the structure within each block. Hušková and Kirch (2008) has further discussed constructing confidence intervals based on the resampled time series.

In Chapter 4, we switched to another field: an imaging analysis where spike deconvolution can also be treated as a change point problem. We proposed MTV-PAR, a time-varying ℓ_0 penalized method to simultaneously conduct spike detection and firing rate estimation from longitudinal calcium fluorescence trace data. In MTV-PAR, each iteration consists of two steps: the spike detection step using time-varying ℓ_0 regularization based on current estimate of firing rate functions and the firing rate estimation step based on currently detected spikes. Efficiency is achieved by aggregating information from temporally adjacent trials whereas robustness is guaranteed by incorporating a time-varying firing rate function into a dynamic ℓ_0 penalization framework, rather than forcing shared parameters across trials.

In our local nonparametric estimation for firing rate functions, for computational ease, we

chose the window size based on experience and the total number of available trials. The window size and other tuning parameters are best calibrated using benchmark data. In the absence of benchmark data, data driven methods might be helpful. Cunningham et al. (2007) proposed to use a Gaussian process prior and an iterative gradient optimization algorithm to optimally choose hyperparameters. Fiecas and Ombao (2016) proposed to use the window size that minimizes the generalized cross-validated deviance when investigating the evolution of dynamic interactions (coherence) between brain regions. Adopting similar strategies is likely to gain further insights and efficiency when analyzing calcium imaging data. Besides, in our current work, we focus on single neuron firing rate estimation. As many behaviors are driven by multiple neurons, incorporating dependency and analyze synchrony between neurons is also sensible. We could follow the Bayesian framework in Shahbaba et al. (2014) to model the neuron dependencies. Moreover, modeling the neural dynamics still remains challenges. In the current analysis, we did not take inference of firing rate across trials into account. Other directions also can be considered as correlating neural dynamics with behavioral data.

Like other existing methods, we aim to estimate the time of a neural spike or a burst of spike events. Bounded by the temporal resolution of calcium imaging data, when there are multiple spikes within the same time frame, we are not able to zoom in to estimate the times of individual spikes. In several work, the change in fluorescence intensities due to spike events at time t, i.e., $s(t)$, was initially motivated to model spike counts (Friedrich and Paninski, 2016; Friedrich et al., 2017; Vogelstein et al., 2010); following this line of thought, it is reasonable to assume that $s(t)$ is positively correlated with spike counts for re-scaled fluorescence trace. In other work (Jewell and Witten, 2018; Pnevmatikakis et al., 2013), the sign of $s(t)$ was the focus. The unknown and nonlinear relationship between spike counts and fluorescence transient measured based upon genetically encoded indicators (Lütcke et al., 2013; Rose et al., 2014; Vogelstein et al., 2010) complicates the accurate interpretation and modeling of fluorescence transient data. We could also consider Bayesian methods to quantify the uncertainty of the magnitude of $s(t)$. In D'Angelo et al. (2021), a Bayesian mixture model is

developed to estimate spike activity under different experiment conditions. We could extend our analysis by setting a spike-and-slab prior on the magnitude.

Finally, in Chapter 5 we discuss a question following Chapter 4. We compared calcium traces and the estimated spike data in three analyses: cluster analysis, PCA, and decoding. Using simulated data, we showed that deconvolution leads to better estimates of neural clusters and principal components than do calcium traces. In our analysis of a real study, the estimated spike trains produced neural clusters with higher trial-to-trial concordance and, when analyzed in appropriate ways, they also give more accurate results in population decoding.

For computational efficiency, we chose a recently published ℓ_0 penalized method (Jewell and Witten, 2018) to estimate spike trains from calcium traces. Because different deconvolution methods have varying accuracy in spike estimation it is expected that our comparison depends on the choice of the choice of deconvolution method, as reported in (Evans et al., 2019). On the other hand, given the consistent observations in our study and previous studies (Evans et al., 2019; Wei et al., 2020), it is reasonable to assume that the difference between choosing calcium trace and estimated spike data is mainly due to the nature of the data, such as quantitative vs binary, continuous vs sparse, and delayed/integrated vs instantaneous.

The results also likely vary from the specific analysis methods chosen, such as choice of clustering methods, choice of kernel in the kernel PCA for spike data, and decoding methods. For example, when using the cumulative spike data for population decoding, we use the whole time course no later than the current time window. In a recent work Yates et al. (2020) showed that population accuracy varied across window size. In Park et al. (2014); Tu et al. (2020), a temporal filter is applied to a calcium transient feature to encourage integrating neuronal activities to improve decoding accuracy. While adopting sophisticated decoding algorithms might improve the prediction accuracy for both calcium traces and the deconvolved spike data, we expect that the main conclusion remains.

Bibliography

L. Abbott. Decoding neuronal firing and modelling neural networks. *Quarterly reviews of biophysics*, 27:291–291, 1994.

M. Abeles, H. Bergman, E. Margalit, and E. Vaadia. Spatiotemporal firing patterns in the frontal cortex of behaving monkeys. *Journal of neurophysiology*, 70(4):1629–1638, 1993.

S. Adak. Time-dependent spectral analysis of nonstationary time series. *Journal of the American Statistical Association*, 93(444):1488–1501, 1998.

R. P. Adams, I. Murray, and D. J. MacKay. Tractable nonparametric bayesian inference in poisson processes with gaussian process intensities. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 9–16, 2009.

A. Adler, S. Katabi, I. Finkes, Z. Israel, Y. Prut, and H. Bergman. Temporal convergence of dynamic cell assemblies in the striato-pallidal network. *Journal of Neuroscience*, 32(7): 2473–2484, 2012.

G. Alarcon, C. Binnie, R. Elwes, and C. Polkey. Power spectrum and intracranial eeg patterns at seizure onset in partial epilepsy. *Electroencephalography and clinical neurophysiology*, 94 (5):326–337, 1995.

F. Ali and A. C. Kwan. Interpreting in vivo calcium signals from neuronal cell bodies, axons, and dendrites: a review. *Neurophotonics*, 7(1):011402, 2019.

A. Aue, S. Hörmann, L. Horváth, M. Reimherr, et al. Break detection in the covariance structure of multivariate time series models. *The Annals of Statistics*, 37(6B):4046–4087, 2009a.

A. Aue, L. Horváth, and M. L. Reimherr. Delay times of sequential procedures for multiple time series regression models. *Journal of Econometrics*, 149(2):174–190, 2009b.

I. E. Auger and C. E. Lawrence. Algorithms for the optimal identification of segment neighborhoods. *Bulletin of mathematical biology*, 51(1):39–54, 1989.

J. Bai. Least squares estimation of a shift in linear processes. *Journal of Time Series Analysis*, 15(5):453–472, 1994.

G. Barbera, B. Liang, L. Zhang, C. R. Gerfen, E. Culurciello, R. Chen, Y. Li, and D.-T. Lin. Spatially compact neural clusters in the dorsal striatum encode locomotion relevant information. *Neuron*, 92(1):202–213, 2016.

M. Barigozzi, H. Cho, and P. Fryzlewicz. Simultaneous multiple change-point and factor analysis for high-dimensional time series. *Journal of Econometrics*, 206(1):187–225, 2018.

M. Basseville, I. V. Nikiforov, et al. *Detection of abrupt changes: theory and application*, volume 104. prentice Hall Englewood Cliffs, 1993.

S. Behseta and R. E. Kass. Testing equality of two functions using bars. *Statistics in medicine*, 24(22):3523–3534, 2005.

T. Bekolay, M. Laubach, and C. Eliasmith. A spiking neural integrator model of the adaptive control of action by the medial prefrontal cortex. *Journal of Neuroscience*, 34(5):1892–1902, 2014.

P. Berens, A. S. Ecker, R. J. Cotton, W. J. Ma, M. Bethge, and A. S. Tolias. A fast and simple population code for orientation in primate v1. *Journal of Neuroscience*, 32(31): 10618–10626, 2012.

J. C. Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Springer Science & Business Media, 2013.

J. C. Bezdek, R. Ehrlich, and W. Full. Fcm: The fuzzy c-means clustering algorithm. *Computers & geosciences*, 10(2-3):191–203, 1984.

N. A. Blondin and D. M. Greer. Neurologic prognosis in cardiac arrest patients treated with therapeutic hypothermia. *The neurologist*, 17(5):241–248, 2011.

D. R. Brillinger. A frequency approach to the techniques of principal components, factor analysis and canonical variates in the case of stationary time series. In *Invited Paper, Royal Statistical Society Conference, Cardiff Wales.(Available at http://stat-www. berkeley. edu/users/brill/papers. html)*, 1964.

D. R. Brillinger. Some aspects of modern population mathematics. *Canadian Journal of Statistics*, 9(2):173–194, 1981.

E. N. Brown. Theory of point processes for neural systems. In *Les Houches*, volume 80, pages 691–727. Elsevier, 2005.

E. N. Brown, R. Barbieri, U. T. Eden, and L. M. Frank. Likelihood methods for neural spike train data analysis. *Computational neuroscience: A comprehensive approach*, pages 253–286, 2003.

E. N. Brown, R. E. Kass, and P. P. Mitra. Multiple neural spike train data analysis: state-of-the-art and future challenges. *Nature neuroscience*, 7(5):456–461, 2004.

E. Carlstein et al. The use of subseries values for estimating the variance of a general statistic from a stationary sequence. *The annals of statistics*, 14(3):1171–1179, 1986.

A. Carnell and D. Richardson. linear algebra for time series of spikes. In *ESANN*, pages 363–368, 2005.

S. Chakraborti, P. Van der Laan, and S. Bakir. Nonparametric control charts: an overview and some results. *Journal of Quality Technology*, 33(3):304–315, 2001.

J. K. Chapin and M. A. Nicolelis. Principal component analysis of neuronal ensemble activity reveals multidimensional somatosensory representations. *Journal of neuroscience methods*, 94(1):121–140, 1999.

J. Chen and A. K. Gupta. *Parametric statistical change point analysis: with applications to genetics, medicine, and finance*. Springer Science & Business Media, 2011.

T.-W. Chen, T. J. Wardill, Y. Sun, S. R. Pulver, S. L. Renninger, A. Baohan, E. R. Schreiter, R. A. Kerr, M. B. Orger, V. Jayaraman, et al. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, 499(7458):295–300, 2013.

Y. Chen, W. K. Härdle, and U. Pigorsch. Localized realized volatility modeling. *Journal of the American Statistical Association*, 105(492):1376–1393, 2010.

L. Chisci, A. Mavino, G. Perferi, M. Sciandrone, C. Anile, G. Colicchio, and F. Fuggetta. Real-time epileptic seizure prediction using ar models and support vector machines. *IEEE Transactions on Biomedical Engineering*, 57(5):1124–1132, 2010.

H. Cho and P. Fryzlewicz. Multiscale and multilevel technique for consistent segmentation of nonstationary time series. *Statistica Sinica*, pages 207–229, 2012.

H. Cho and P. Fryzlewicz. Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pages 475–507, 2015.

T. Choudhry. Stock market volatility and the crash of 1987: evidence from six emerging markets. *Journal of International Money and Finance*, 15(6):969–981, 1996.

M. M. Churchland, J. P. Cunningham, M. T. Kaufman, J. D. Foster, P. Nuyujukian, S. I. Ryu, and K. V. Shenoy. Neural population dynamics during reaching. *Nature*, 487(7405): 51–56, 2012.

C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

J. P. Cunningham and M. Y. Byron. Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509, 2014.

J. P. Cunningham, B. M. Yu, K. V. Shenoy, and M. Sahani. Inferring neural firing rates from spike trains using gaussian processes. *Advances in neural information processing systems*, 20:329–336, 2007.

J. P. Cunningham, M. Y. Byron, K. V. Shenoy, and M. Sahani. Inferring neural firing rates from spike trains using gaussian processes. In *Advances in neural information processing systems*, pages 329–336, 2008.

J. P. Cunningham, V. Gilja, S. I. Ryu, and K. V. Shenoy. Methods for estimating neural firing rates, and their application to brain–machine interfaces. *Neural Networks*, 22(9): 1235–1246, 2009.

G. Czanner, U. T. Eden, S. Wirth, M. Yanike, W. A. Suzuki, and E. N. Brown. Analysis of between-trial and within-trial neural spiking dynamics. *Journal of neurophysiology*, 99(5): 2672–2693, 2008.

D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes: volume II: general theory and structure*. Springer Science & Business Media, 2007.

L. D'Angelo, A. Canale, Z. Yu, and M. Guindani. Bayesian nonparametric analysis for the detection of spikes in noisy calcium imaging data. *arXiv preprint arXiv:2102.09403*, 2021.

L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(09):P09008, 2005.

R. A. Davis, T. C. M. Lee, and G. A. Rodriguez-Yam. Structural break estimation for nonstationary time series models. *Journal of the American Statistical Association*, 101 (473):223–239, 2006.

T. Deneux, A. Kaszas, G. Szalay, G. Katona, T. Lakner, A. Grinvald, B. Rózsa, and I. Vanzetta. Accurate spike estimation from noisy calcium signals for ultrafast three-dimensional imaging of large neuronal populations in vivo. *Nature communications*, 7(1): 1–17, 2016.

I. DiMatteo, C. R. Genovese, and R. E. Kass. Bayesian curve-fitting with free-knot splines. *Biometrika*, 88(4):1055–1071, 2001.

D. A. Dombeck, M. S. Graziano, and D. W. Tank. Functional clustering of neurons in motor cortex determined by cellular resolution imaging in awake behaving mice. *Journal of Neuroscience*, 29(44):13751–13760, 2009.

D. A. Dombeck, C. D. Harvey, L. Tian, L. L. Looger, and D. W. Tank. Functional imaging of hippocampal place cells at cellular resolution during virtual navigation. *Nature neuroscience*, 13(11):1433–1440, 2010.

M. Dooley and M. Hutchison. Transmission of the us subprime crisis to emerging markets: Evidence on the decoupling–recoupling hypothesis. *Journal of International Money and Finance*, 28(8):1331–1349, 2009.

J. C. Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. 1973.

M. Evans, R. D. Petersen, and M. D. Humphries. On the use of calcium deconvolution algorithms in practical contexts. *bioRxiv*, page 871137, 2019.

Y. Fan, W. K. Härdle, W. Wang, and L. Zhu. Composite quantile regression for the single-index model. 2013.

J.-M. Fellous, P. H. Tiesinga, P. J. Thomas, and T. J. Sejnowski. Discovering spike patterns in neuronal responses. *Journal of Neuroscience*, 24(12):2989–3001, 2004.

M. Fiecas and H. Ombao. Modeling the evolution of dynamic brain processes during an associative learning experiment. *Journal of the American Statistical Association*, 111(516): 1440–1453, 2016.

J. Friedrich and L. Paninski. Fast active set methods for online spike inference from calcium imaging. In *Advances In Neural Information Processing Systems*, pages 1984–1992, 2016.

J. Friedrich, P. Zhou, and L. Paninski. Fast online deconvolution of calcium imaging data. *PLoS computational biology*, 13(3):e1005423, 2017.

P. Fryzlewicz. Modelling and forecasting financial log-returns as locally stationary wavelet processes. *Journal of Applied Statistics*, 32(5):503, 2005.

P. Fryzlewicz and S. S. Rao. Multiple-change-point detection for auto-regressive conditional heteroscedastic processes. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pages 903–924, 2014.

P. Fryzlewicz et al. Wild binary segmentation for multiple change-point detection. *The Annals of Statistics*, 42(6):2243–2281, 2014.

P. Gao, E. Trautmann, B. Yu, G. Santhanam, S. Ryu, K. Shenoy, and S. Ganguli. A theory of multineuronal dimensionality, dynamics and measurement. *BioRxiv*, page 214262, 2017.

X. Gao, B. Shahbaba, N. Fortin, and H. Ombao. Evolutionary state-space model and its application to time-frequency analysis of local field potentials. *arXiv preprint arXiv:1610.07271*, 2016.

G. L. Gerstein and N. Y.-S. Kiang. An approach to the quantitative analysis of electrophysiological data from single neurons. *Biophysical Journal*, 1(1):15, 1960.

K. K. Ghosh, L. D. Burns, E. D. Cocker, A. Nimmerjahn, Y. Ziv, A. El Gamal, and M. J. Schnitzer. Miniaturized integration of a fluorescence microscope. *Nature methods*, 8(10): 871, 2011.

A. Giovannucci, J. Friedrich, P. Gunn, J. Kalfon, B. L. Brown, S. A. Koay, J. Taxidis, F. Najafi, J. L. Gauthier, P. Zhou, et al. Caiman an open source tool for scalable calcium imaging data analysis. *Elife*, 8:e38173, 2019.

C. Grienberger and A. Konnerth. Imaging calcium in neurons. *Neuron*, 73(5):862–885, 2012.

Z. V. Guo, N. Li, D. Huber, E. Ophir, D. Gutnisky, J. T. Ting, G. Feng, and K. Svoboda. Flow of cortical activity underlying a tactile decision in mice. *Neuron*, 81(1):179–194, 2014.

P. Haccou, E. Meelis, and S. Van De Geer. The likelihood ratio test for the change point problem for exponentially distributed random variables. *Stochastic processes and their applications*, 27:121–139, 1987.

C. D. Harvey, P. Coen, and D. W. Tank. Choice-specific sequences in parietal cortex during a virtual-navigation decision task. *Nature*, 484(7392):62–68, 2012.

D. V. Hinkley. Inference about the change-point in a sequence of random variables. 1970.

L. Horváth and M. Hušková. Change-point detection in panel data. *Journal of Time Series Analysis*, 33(4):631–648, 2012.

A. Huk, K. Bonnen, and B. J. He. Beyond trial-based paradigms: Continuous behavior, ongoing neural activity, and natural stimuli. *Journal of Neuroscience*, 38(35):7551–7558, 2018.

M. D. Humphries. Spike-train communities: finding groups of similar spike trains. *Journal of Neuroscience*, 31(6):2321–2336, 2011.

M. Hušková and C. Kirch. Bootstrapping confidence intervals for the change-point of time series. *Journal of Time Series Analysis*, 29(6):947–972, 2008.

C. Inclan and G. C. Tiao. Use of cumulative sums of squares for retrospective detection of changes of variance. *Journal of the American Statistical Association*, 89(427):913–923, 1994.

B. Jackson, J. D. Scargle, D. Barnes, S. Arabhi, A. Alt, P. Gioumousis, E. Gwin, P. Sangtrakulcharoen, L. Tan, and T. T. Tsai. An algorithm for optimal partitioning of data on an interval. *IEEE Signal Processing Letters*, 12(2):105–108, 2005.

S. Jewell and D. Witten. Exact spike train inference via l0 optimization. *The annals of applied statistics*, 12(4):2457, 2018.

M. S. Jog, Y. Kubota, C. I. Connolly, V. Hillegaart, and A. M. Graybiel. Building neural representations of habits. *Science*, 286(5445):1745–1749, 1999.

K. G. Johnston, S. F. Grieco, Z. Yu, S. Jin, T. Shen, R. Crary, J. Guzowski, T. Holmes, Q. C. Nie, and X. Xu. Robust population single neuronal calcium signal extraction using scout allows for longitudinal analysis of behavior-associated neural ensemble dynamics. *bioRxiv*, 2020.

A. Kaplan, J. Röschke, B. Darkhovsky, and J. Fell. Macrostructural eeg characterization based on nonparametric change point segmentation: application to sleep analysis. *Journal of neuroscience methods*, 106(1):81–90, 2001.

R. E. Kass, V. Ventura, and C. Cai. Statistical smoothing of neuronal data. *Network-Computation in Neural Systems*, 14(1):5–16, 2003.

R. E. Kass, V. Ventura, and E. N. Brown. Statistical issues in the analysis of neuronal data. *Journal of neurophysiology*, 94(1):8–25, 2005.

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

H.-J. Kim and D. Siegmund. The likelihood ratio test for a change-point in simple linear regression. *Biometrika*, 76(3):409–423, 1989.

C. Kirch, B. Muhsal, and H. Ombao. Detection of changes in multivariate time series with application to eeg data. *Journal of the American Statistical Association*, 110(511): 1197–1216, 2015.

D. Kobak, W. Brendel, C. Constantinidis, C. E. Feierstein, A. Kepecs, Z. F. Mainen, X.-L. Qi, R. Romo, N. Uchida, and C. K. Machens. Demixed principal component analysis of neural population data. *Elife*, 5:e10989, 2016.

P. Kokoszka and R. Leipus. Testing for parameter changes in arch models. *Lithuanian Mathematical Journal*, 39(2):182–195, 1999.

H. R. Kunsch. The jackknife and the bootstrap for general stationary observations. *The annals of Statistics*, pages 1217–1241, 1989.

T. L. Lai. Sequential analysis: some classical problems and new challenges. *Statistica Sinica*, pages 303–351, 2001.

B. Levin and J. Kline. The cusum test of homogeneity with an application in spontaneous abortion epidemiology. *Statistics in Medicine*, 4(4):469–488, 1985.

N. Li, T.-W. Chen, Z. V. Guo, C. R. Gerfen, and K. Svoboda. A motor cortex circuit for motor planning and movement. *Nature*, 519(7541):51–56, 2015.

H. Lütcke, F. Gerhard, F. Zenke, W. Gerstner, and F. Helmchen. Inference of neuronal network spike dynamics and topology from calcium imaging data. *Frontiers in neural circuits*, 7:201, 2013.

R. Maruyama, K. Maeda, H. Moroda, I. Kato, M. Inoue, H. Miyakawa, and T. Aonishi. Detecting cells using non-negative matrix factorization on calcium imaging data. *Neural Networks*, 55:11–19, 2014.

E. Meyers. The neural decoding toolbox. *Frontiers in neuroinformatics*, 7:8, 2013.

R. P. Monti, C. Anagnostopoulos, G. Montana, et al. Learning population and subject-specific brain connectivity networks via mixed neighborhood selection. *The Annals of Applied Statistics*, 11(4):2142–2164, 2017.

F. Mormann, R. G. Andrzejak, C. E. Elger, and K. Lehnertz. Seizure prediction: the long and winding road. *Brain*, 130(2):314–333, 2006.

E. A. Mukamel, A. Nimmerjahn, and M. J. Schnitzer. Automated analysis of cellular signals from large-scale calcium imaging data. *Neuron*, 63(6):747–760, 2009.

C. F. Nam, J. A. Aston, I. A. Eckley, and R. Killick. The uncertainty of storm season changes: Quantifying the uncertainty of autocovariance changepoints. *Technometrics*, 57(2):194–206, 2015.

A. B. Olshen, E. Venkatraman, R. Lucito, and M. Wigler. Circular binary segmentation for the analysis of array-based dna copy number data. *Biostatistics*, 5(4):557–572, 2004.

C. R. Olson, S. N. Gettner, V. Ventura, R. Carta, and R. E. Kass. Neuronal activity in macaque supplementary eye field during planning of saccades in response to pattern and spatial cues. *Journal of Neurophysiology*, 84(3):1369–1384, 2000.

H. Ombao and M.-h. R. Ho. Time-dependent frequency domain principal components analysis of multichannel non-stationary signals. *Computational statistics & data analysis*, 50(9): 2339–2360, 2006.

H. Ombao, R. Von Sachs, and W. Guo. Slex analysis of multivariate nonstationary time series. *Journal of the American Statistical Association*, 100(470):519–531, 2005.

H. Ombao, M. Fiecas, C.-M. Ting, and Y. F. Low. Statistical models for brain signals with properties that evolve across trials. *NeuroImage*, 180:609–618, 2018.

I. Ozden, H. M. Lee, M. R. Sullivan, and S. S.-H. Wang. Identification and clustering of event patterns from in vivo multiphoton optical recordings of neuronal ensembles. *Journal of neurophysiology*, 100(1):495–503, 2008.

M. Pachitariu, C. Stringer, and K. D. Harris. Robustness of spike deconvolution for neuronal calcium imaging. *Journal of Neuroscience*, 38(37):7976–7985, 2018.

E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954.

A. R. Paiva, I. Park, and J. C. Principe. A reproducing kernel hilbert space framework for spike train signal processing. *Neural computation*, 21(2):424–449, 2009.

A. R. Paiva, I. Park, and J. C. Principe. Inner products for representation and learning in the spike train domain. In *Statistical signal processing for neuroscience and neurotechnology*, pages 265–309. Elsevier, 2010.

L. Paninski, Y. Ahmadian, D. G. Ferreira, S. Koyama, K. R. Rad, M. Vidne, J. Vogelstein, and W. Wu. A new look at state-space models for neural data. *Journal of computational neuroscience*, 29(1-2):107–126, 2010.

I. M. Park, M. L. Meister, A. C. Huk, and J. W. Pillow. Encoding and decoding in parietal cortex during sensorimotor decision-making. *Nature neuroscience*, 17(10):1395–1403, 2014.

M. A. Picardo, J. Merel, K. A. Katlowitz, D. Vallentin, D. E. Okobi, S. E. Benezra, R. C. Clary, E. A. Pnevmatikakis, L. Paninski, and M. A. Long. Population-level representation of a temporal sequence underlying song production in the zebra finch. *Neuron*, 90(4): 866–876, 2016.

E. A. Pnevmatikakis. Analysis pipelines for calcium imaging data. *Current opinion in neurobiology*, 55:15–21, 2019.

E. A. Pnevmatikakis, J. Merel, A. Pakman, and L. Paninski. Bayesian spike inference from calcium imaging data. In *2013 Asilomar Conference on Signals, Systems and Computers*, pages 349–353. IEEE, 2013.

E. A. Pnevmatikakis, D. Soudry, Y. Gao, T. A. Machado, J. Merel, D. Pfau, T. Reardon, Y. Mu, C. Lacefield, W. Yang, et al. Simultaneous denoising, deconvolution, and demixing of calcium imaging data. *Neuron*, 89(2):285–299, 2016.

W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

P. Reynaud-Bouret, V. Rivoirard, F. Grammont, and C. Tuleau-Malot. Goodness-of-fit tests and nonparametric adaptive estimation for spike train analysis. *The Journal of Mathematical Neuroscience*, 4(1):3, 2014.

G. Rigaill. A pruned dynamic programming algorithm to recover the best segmentations with 1 to k_max change-points. *Journal de la Société Française de Statistique*, 156(4):180–205, 2015.

J. Rissanen. *Stochastic complexity in statistical inquiry*, volume 15. World Scientific, 1989.

M. W. Robbins, C. M. Gallagher, and R. B. Lund. A general regression changepoint test for time series data. *Journal of the American Statistical Association*, 111(514):670–683, 2016.

S. A. Romano, V. Pérez-Schuster, A. Jouary, J. Boulanger-Weill, A. Candeo, T. Pietri, and G. Sumbre. An integrated calcium imaging processing toolbox for the analysis of neuronal population dynamics. *PLoS computational biology*, 13(6):e1005526, 2017.

T. Rose, P. M. Goltstein, R. Portugues, and O. Griesbeck. Putting a finishing touch on gecis. *Frontiers in molecular neuroscience*, 7:88, 2014.

A. Safikhani and A. Shojaie. Structural break detection in high-dimensional non-stationary var models. *arXiv preprint arXiv:1708.02736*, 2017.

M. Scheffer, J. Bascompte, W. A. Brock, V. Brovkin, S. R. Carpenter, V. Dakos, H. Held, E. H. Van Nes, M. Rietkerk, and G. Sugihara. Early-warning signals for critical transitions. *Nature*, 461(7260):53, 2009.

S. E. Schmitt, K. Pargeon, E. S. Frechette, L. J. Hirsch, J. Dalmau, and D. Friedman. Extreme delta brush: a unique eeg pattern in adults with anti-nmda receptor encephalitis. *Neurology*, 79(11):1094–1100, 2012.

B. Schrauwen and J. Van Campenhout. Linking non-binned spike train kernels to several existing spike train metrics. *Neurocomputing*, 70(7-9):1247–1253, 2007.

A. L. Schröder and H. Ombao. Fresped: Frequency-specific change-point detection in epileptic seizure multi-channel eeg data. *Journal of the American Statistical Association*, 114(525): 115–128, 2019.

A. J. Scott and M. Knott. A cluster analysis method for grouping means in the analysis of variance. *Biometrics*, pages 507–512, 1974.

B. Shahbaba, B. Zhou, S. Lan, H. Ombao, D. Moorman, and S. Behseta. A semiparametric bayesian model for detecting synchrony among multiple neurons. *Neural computation*, 26 (9):2025–2051, 2014.

B. Shahbaba, S. Behseta, and A. Vandenberg-Rodes. Neuronal spike train analysis using gaussian process models. In *Nonparametric Bayesian Inference in Biostatistics*, pages 271–285. Springer, 2015.

T. Shen, K. Johnston, G. Lur, M. Guindani, H. Ombao, and Z. Yu. Time-varying $\backslash\ell_0$ optimization for spike inference from multi-trial calcium recordings. *arXiv preprint arXiv:2103.03818*, 2021.

R. H. Shumway and D. S. Stoffer. *Time series analysis and its applications: with R examples*. Springer, 2017.

S. Siami-Namini, N. Tavakoli, and A. S. Namin. A comparison of arima and lstm in forecasting time series. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1394–1401. IEEE, 2018.

D. S. Stoffer, H. C. Ombao, and D. E. Tyler. Local spectral envelope: an approach using dyadic tree-based adaptive segmentation. *Annals of the Institute of Statistical Mathematics*, 54(1):201–223, 2002.

L. Tang, G. Hamarneh, and R. Abugharbieh. Reliability-driven, spatially-adaptive regularization for deformable registration. In *International Workshop on Biomedical Image Registration*, pages 173–185. Springer, 2010.

J. Terrien, G. Germain, C. Marque, and B. Karlsson. Bivariate piecewise stationary segmentation; improved pre-treatment for synchronization measures used on non-stationary biological signals. *Medical Engineering & Physics*, 35(8):1188–1196, 2013.

L. Theis, P. Berens, E. Froudarakis, J. Reimer, M. R. Rosón, T. Baden, T. Euler, A. S. Tolias, and M. Bethge. Benchmarking spike rate inference in population calcium imaging. *Neuron*, 90(3):471–482, 2016.

F. Theunissen and J. P. Miller. Temporal encoding in nervous systems: a rigorous definition. *Journal of computational neuroscience*, 2(2):149–162, 1995.

L. Tian, S. A. Hires, T. Mao, D. Huber, M. E. Chiappe, S. H. Chalasani, L. Petreanu, J. Akerboom, S. A. McKinney, E. R. Schreiter, et al. Imaging neural activity in worms, flies and mice with improved gcamp calcium indicators. *Nature methods*, 6(12):875–881, 2009.

M. Tu, R. Zhao, A. Adler, W.-B. Gan, and Z. S. Chen. Efficient position decoding methods based on fluorescence calcium imaging in the mouse hippocampus. *Neural Computation*, 32(6):1144–1167, 2020.

E. Venkatraman and A. B. Olshen. A faster circular binary segmentation algorithm for the analysis of array cgh data. *Bioinformatics*, 23(6):657–663, 2007.

E. S. Venkatraman. Consistency results in multiple change-point problems. 1993.

J. D. Victor and K. P. Purpura. Nature and precision of temporal coding in visual cortex: a metric-space analysis. *Journal of neurophysiology*, 76(2):1310–1326, 1996.

J. D. Victor and K. P. Purpura. Metric-space analysis of spike trains: theory, algorithms and application. *Network: computation in neural systems*, 8(2):127–164, 1997.

J. T. Vogelstein, A. M. Packer, T. A. Machado, T. Sippy, B. Babadi, R. Yuste, and L. Paninski. Fast nonnegative deconvolution for spike train inference from population calcium imaging. *Journal of neurophysiology*, 104(6):3691–3704, 2010.

L. Y. Vostrikova. Detecting "disorder" in multidimensional random processes. In *Doklady Akademii Nauk*, volume 259, pages 270–274. Russian Academy of Sciences, 1981.

Y. Wang, C.-M. Ting, X. Gao, and H. Ombao. Exploratory analysis of brain signals through low dimensional embedding. In *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 997–1002. IEEE, 2019.

Z. Wei, H. Inagaki, N. Li, K. Svoboda, and S. Druckmann. An orderly single-trial organization of population dynamics in premotor cortex predicts behavioral variability. *Nature communications*, 10(1):1–14, 2019.

Z. Wei, B.-J. Lin, T.-W. Chen, K. Daie, K. Svoboda, and S. Druckmann. A comparison of neuronal population dynamics measured with calcium imaging and electrophysiology. *PLoS computational biology*, 16(9):e1008198, 2020.

S. S. Wilks. The large-sample distribution of the likelihood ratio for testing composite hypotheses. *The annals of mathematical statistics*, 9(1):60–62, 1938.

S. Wirth, M. Yanike, L. M. Frank, A. C. Smith, E. N. Brown, and W. A. Suzuki. Single neurons in the monkey hippocampus and learning of new associations. *Science*, 300(5625): 1578–1581, 2003.

S. P. Wise and E. A. Murray. Role of the hippocampal system in conditional motor learning: mapping antecedents to action. *Hippocampus*, 9(2):101–117, 1999.

K. Worsley. The power of likelihood ratio and cumulative sum tests for a change in a binomial probability. *Biometrika*, 70(2):455–464, 1983.

K. Worsley. Confidence regions and tests for a change-point in a sequence of exponential family random variables. *Biometrika*, 73(1):91–104, 1986.

E. Yaksi and R. W. Friedrich. Reconstruction of firing rate changes across neuronal populations by temporally deconvolved ca 2+ imaging. *Nature methods*, 3(5):377–383, 2006.

L. Yang and S. C. Masmanidis. Differential encoding of action selection by orbitofrontal and striatal population dynamics. *Journal of Neurophysiology*, 124(2):634–644, 2020.

W. Yang and R. Yuste. In vivo imaging of neural activity. *Nature methods*, 14(4):349–359, 2017.

Q. Yao. Tests for change-points with epidemic alternatives. *Biometrika*, 80(1):179–191, 1993.

Y.-C. Yao and R. A. Davis. The asymptotic behavior of the likelihood ratio statistic for testing a shift in mean in a sequence of independent normal variates. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 339–353, 1986.

J. L. Yates, L. N. Katz, A. J. Levi, J. W. Pillow, and A. C. Huk. A simple linear readout of mt supports motion direction-discrimination performance. *Journal of neurophysiology*, 123 (2):682–694, 2020.

L. Yu, W. K. Härdle, L. Borke, and T. Benschop. Frm: a financial risk meter based on penalizing tail events occurrence. 2017.

L. Zbonakova, W. K. Härdle, and W. Wang. Time varying quantile lasso. 2016.

Q. Zheng, F. Zhu, J. Qin, B. Chen, and P.-A. Heng. Sparse support matrix machine. *Pattern Recognition*, 76:715–726, 2018.

P. Zhou, S. L. Resendez, J. Rodriguez-Romaguera, J. C. Jimenez, S. Q. Neufeld, A. Giovannucci, J. Friedrich, E. A. Pnevmatikakis, G. D. Stuber, R. Hen, et al. Efficient and accurate extraction of in vivo calcium signals from microendoscopic video data. *Elife*, 7: e28728, 2018.