Title

A Graph-based Framework for Multiple Change-point Detection

Author

Zhang, Yuxuan

Publication Date

2023

Peer reviewed|Thesis/dissertation

A Graph-based Framework for Multiple Change-point Detection

By

YUXUAN ZHANG
DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Statistics

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

Hao Chen, Chair

Alexander Aue

Ethan Anderes

Committee in Charge

2023

To my parents.

Contents

A Graph-based Framework for Multiple Change-point Detection

Abstract

We study the problem of multiple change-point detection in high-dimensional data and non-Euclidean data with graph-based statistics. With the emergence of more complex data with multiple change-points, traditional change-point detection methods for low-dimensional data are not suitable anymore. We first propose a nonparametric multiple change-point detection framework using graph-based statistics. The framework is a two-step procedure. In the first step, we combine generalized edge count scan statistics with wild binary segmentation or seeded binary segmentation to search for a pool of candidate change-points. We then prune the candidate change-points through a novel goodness-of-fit statistic in the second step. Numerical studies show that this new framework outperforms existing methods under a wide range of settings. The resulting change-points can further be arranged hierarchically based on the goodness-of-fit statistic.

Next, to further improve the detection accuracy under frequent changes scenarios and pure mean or covariance changes scenarios, we incorporate max-type edge-count scan statistics in the first step. In the second step, a new goodness-of-fit statistic built on max-type two-sample test statistics with a stepwise algorithm is used for model selection.

Furthermore, we consider an important application of multiple change-point detection on Neuropixels data. Neuropixels is a new tool in neuroscience allowing the recording of brain neuronal activities in high resolution for a long period of time. The large size of Neuropixels data and its non-stationarity make it challenging for statistical analysis. We propose a nonparametric method for detecting multiple change-point for this type of data. Change-point analysis can be served as a preliminary step for further statistical modeling. The proposed method combines max-type edge count scan statistics and wild binary segmentation to search for change-points in parallel, greatly reducing the computation time required for long sequences. The method is demonstrated by an application to Neuropixels data recorded from an awake mouse in nine brain regions for 20 minutes.

Acknowledgments

First of all, I would like to dedicate my sincerest gratitude to my advisor, Prof. Hao Chen. During my time working with her, I have often marveled at her keen insight and profound opinions. Whenever our research got stuck in a bottleneck, she was always able to guide me with her superb statistical intuition and insights into the problem, so that our research could be carried out smoothly. I learned from her all the characteristics of a good scholar: wise, knowledgeable, hardworking, and willing to try. She is my role model and has always inspired me to do better research, to be a better scholar, and to strive for excellence and knowledge in my future academic work. At the same time, she has also been an easy-going and caring mentor, caring about my work and life so that I can focus more on my academic research. Without her guidance and continuous dedication, I would not be where I am today.

I would like to express my gratitude to the members of my qualification exam committee and dissertation committee: Prof. Ethan Anderes, Prof. Alexander Aue, Prof. Thomas Lee, Prof. Xiaodong Li, and Prof. Jie Peng. Thank you for your valuable comments and encouragement on my work. Furthermore, I would like to thank all the professors who have taught me in the department of statistics, including but not limited to Prof. Debashis Paul, Prof. Wolfgang Polonik, Prof. Miles Lopes, and Prof. Prabir Burman. It is you who have shown me the charm and the boundless possibilities of the field of statistics. I would like to extend my appreciation to Prof. Chih-Ling Tsai. Thank you for your guidance in demonstrating the qualities of a respectable researcher.

I am deeply grateful to have met Dr. Rui Pan as my teacher, friend, and collaborator. You have given me considerable guidance and advice in various aspects. Academically, you have taught me how to start research and write papers. Moreover, you have demonstrated how to apply the knowledge I've learned to analyze and get insight from practical data. More importantly, you have shown me through your actions that scholars should adhere to principles and serve as role models for students. I would also like to thank my friend and collaborator, Dr. Xuening Zhu, for your assistance. Discussing research questions with you has always been enjoyable and rewarding.

I would like to especially thank my good friends Jinchang Fan and Yongkai Qiu. Since our undergraduate days, whether it was sharing joy or shouldering pain, you have always been the first friends I think of and the ones I can rely on the most. On countless nights, we reminisced about

our golden days in Beijing, shared our present joys and worries, and looked forward to the bright future. You are the source of my happiness and the emotional support that has enabled me to successfully complete my doctoral studies. Even though we may be thousands of miles apart, I can genuinely feel your care. I believe that we will reunite in Beijing in the future, spending one five-year period after another together.

I would also like to thank my friends that I met at Davis: Zhixuan Shao, Xi Yang, and Tianke Li. Every moment spent with all of you has shaped the entirety of my cherished memories from my time in Davis. Because of you, my Ph.D. life can become more pleasant and colorful. Especially with Zhixuan, I have met many friends through you. We played keyboard, traveled, cooked delicious dishes, and shared numerous interesting memes and experiences. Thank you for making my life in Davis no longer lonely.

I would like to express my gratitude to my friends, Jiabao Feng and Xuefei Lei, for your constant confidence in me and your affirmation of my professional choice. And to Zhuoran Fang, thank you for your unwavering encouragement and assistance in aesthetics. During the most severe times of the pandemic, I often reminisce about the happy moments we spent together before. These memories of joy and the anticipation of reuniting kept me hopeful for the future, even in my loneliest moments during the pandemic.

Last, I would like to thank my parents for their endless love and support. I know you will always be my strongest backing and my warmest harbor. Thank you for always supporting me in chasing my academic dreams, encouraging me to try various possibilities in life, and providing me with great conditions that allowed all these wonderful things to happen.

CHAPTER 1

Introduction

1.1. Multiple Change-point Detection Problem and Literature Review

Change-point analysis is a long-established statistical topic and has received much attention in this century. In the era of big data, data are often of high dimension and complexity. For example, in bioinformatics, finding common DNA copy number variants in hundreds of samples from high-throughput sequencing data is of scientific interest (Jiang et al. 2015, Zhang et al. 2010). In astrophysics, experts discover the presence of galaxies using thousands of images obtained by integral field spectrograph (Enikeeva & Harchaoui 2019). There is also a need for finding abrupt changes in dynamic networks, such as email communication pattern changes and brain state transitions (Braun et al. 2021, Dong et al. 2020, Peel & Clauset 2015).

Consider a sequence of independent observations $\{\mathbf{y}_i : i = 1, \ldots, n\}$, indexed by time or some other meaningful orderings, such that

$$\mathbf{y}_i \sim F_j, \quad \tau_j + 1 \leq i \leq \tau_{j+1}, \quad j = 0, \ldots, m, \tag{1.1}$$

where $0 = \tau_0 < \tau_1 < \cdots < \tau_{m+1} = n$, and F_j's are arbitrary unknown probability measures, satisfying $F_j \neq F_{j+1}$. The parameters $\boldsymbol{\tau} = \{\tau_1, \ldots, \tau_m\}$ are the change-points of the process. Our goal is to estimate m and $\boldsymbol{\tau}$.

The earlier works of change-point detection focus on univariate data under parametric models. Cumulative sum statistics (CUSUM) is a widely used method for univariate data. It is equivalent to likelihood ratio statistics under Gaussian assumption. Later, more work focused on the scenario with multiple change-point for univariate data. Yao (1988) proposed a BIC-type statistic and showed its consistency for bounded number of change-points. Many greedy algorithms also play an important role in this field, including binary segmentation, circular binary segmentation, and wild binary segmentation (WBS) (Fryzlewicz 2014, Olshen et al. 2004, Vostrikova 1981).

Most existing works for multiple change-point detection with multivariate observations are based on parametric models. For example, Zhang et al. (2010) and Enikeeva & Harchaoui (2019) considered ℓ_2 aggregation of CUSUM. Cho & Fryzlewicz (2015) developed a truncated CUSUM combined with binary segmentation to tackle the sparsity in high dimensional data. Wang & Samworth (2016) studied a projected CUSUM procedure also under a sparse high dimensional setting. Lavielle & Teyssiere (2006) introduced a set of methods based on penalized Gaussian log-likelihood to detect changes in covariance structure. Wang et al. (2018) improved Pearson's Chi-squared test for multinomial data, and added a penalty term to allow for multiple change-point selection.

In recent years, more nonparametric methods have been developed to avoid model misspecification in parametric methods. For example, Matteson & James (2014) proposed E-Divisive that combined Euclidean-based divergence measure and divisive algorithm. Harchaoui et al. (2008) and Arlot et al. (2019) used kernel-based statistics to measure the discrepancy between segments (KCpA and KCP). Another framework for multivariate and non-Euclidean data is the graph-based method proposed by Chen & Zhang (2015). For the first time, it gives an *analytic p-value approximation* for a nonparametric framework that can be applied to data in an arbitrary dimension or non-Euclidean data, facilitating its application to large data sets. Chu & Chen (2019) improved the graph-based method by introducing new graph-based statistics that perform well under a wider range of alternatives. However, unlike E-Divisive and KCP, the existing graph-based methods focused on the single change-point and the changed interval alternatives.

1.2. Contribution of the Dissertation

My doctoral research aims to address these challenges with a non-parametric framework built on graph-based statistics and greedy algorithms. In Chapter 2, we develop a reliable way of finding multiple change-points utilizing graph-based statistics for modern data. In particular, we first adopt the idea of WBS (Fryzlewicz 2014) or seeded binary segmentation (SBS) (Kovács et al. 2020) to find a pool of candidate change-points. We then propose a pseudo-BIC criterion for change-point selection. Simulation shows that this new framework has superb performance compared to other

state-of-the-art methods under a variety of settings. The new approach is illustrated by analyzing a snippet of Neuropixels dataset where multiple types of changes are found.

Then, in Chapter 3, we further improve the framework by incorporating max-type graph-based statistics to better deal with more frequent changes. Max-type statistics are more sensitive to pure mean and covariance changes than generalized statistics (Chu & Chen 2019). In addition, max-type scan statistic has a more accurate p-value approximation, making it also more suitable for frequent changes. We present a goodness-of-fit statistic based on max-type two sample test statistics. The efficacy of this approach is demonstrated through various simulation experiments.

In Chapter 4, the main focus is on fast multiple change-point detection in long sequence, especially for Neuropixels data. Neuropixels is a complementary metal-oxide semi-conductor (CMOS) probe utilized for continuously recording neural activities in the brain (Jun et al. 2017). Neuropixels recordings can last for minutes to hours with hundreds or thousands of neurons. In addition, Neuropixels recordings are highly noisy and may involve a lot of distributional change. All these properties make the analysis of Neuropixels data challenging. However, there has been limited work in this area. In order to address these issues, we propose a new framework built on max-type edge-count statistics and a parallel WBS algorithm. The new algorithm detects change-points in parallel, greatly reducing the time complexity of the traditional WBS algorithm. We apply this new framework to real Neuropixels data recorded across nine brain regions in an awake mouse lasting for about 20 minutes.

Finally, we conclude the dissertation with summary of contributions and a discussion of future directions in Chapter 5.

CHAPTER 2

A Nonparametric Framework for Detecting Multiple Change-point in Modern Data

2.1. Notations and Graph-based Single Change-point Detection

We first introduce some notations. Consider the scenario of detecting a single change point on $\{\mathbf{y}_i : a \leq i \leq b\}$, i.e., testing the null hypothesis $H_0^{[a,b]} : \mathbf{y}_i \sim F_0,\ i = a, \ldots, b$ against the alternative $H_1^{[a,b]}$: exists $\ a \leq \tau < b,\ \mathbf{y}_i \sim F_0\ $ for $a \leq i \leq \tau$ and $\mathbf{y}_i \sim F_1$ otherwise. When the null hypothesis $H_0^{[a,b]}$ is true, the permutation null distribution that places $1/(b-a+1)!$ probability on each of the $(b-a+1)!$ permutations of $\{\mathbf{y}_i : a \leq i \leq b\}$ can be used as a surrogate for the true null distribution.

Let $G^{[a,b]}$ be the similarity graph on $\{\mathbf{y}_i : a \leq i \leq b\}$. The graph $G^{[a,b]}$ is an unweighted undirected acyclic graph within which edges are constructed based on a distance measure defined on the sample space up to a criterion. Some examples of $G^{[a,b]}$ include k-minimum spanning tree (k-MST) and k-nearest neighbor graph (k-NNG) (see Chen & Zhang (2015, 2013) for more discussions on choices of the similarity graph). We use $G^{[a,b]}$ to denote both the graph and the set of edges in the graph when its vertex set is implicitly obvious. Let $R_1^{[a,b]}(t)$ be the number of edges connecting observations within $[a, t]$, and $R_2^{[a,b]}(t)$ be the number of edges that connect observations within $[t+1, b]$. The generalized edge-count scan statistic proposed in Chu & Chen (2019) is defined as:

$$\max_{n_{le}^{[a,b]} \leq t \leq n_{ri}^{[a,b]}} S^{[a,b]}(t), \tag{2.1}$$

where

$$S^{[a,b]}(t) = \begin{bmatrix} R_1^{[a,b]}(t) - \mathbf{E}\left[R_1^{[a,b]}(t)\right] \\ R_2^{[a,b]}(t) - \mathbf{E}\left[R_2^{[a,b]}(t)\right] \end{bmatrix}^\top \left[\mathbf{\Sigma}^{[a,b]}(t)\right]^{-1} \begin{bmatrix} R_1^{[a,b]}(t) - \mathbf{E}\left[R_1^{[a,b]}(t)\right] \\ R_2^{[a,b]}(t) - \mathbf{E}\left[R_2^{[a,b]}(t)\right] \end{bmatrix},$$

with $\Sigma^{[a,b]}(t) = \mathbf{Var}\left[\left(R_1^{[a,b]}(t), R_2^{[a,b]}(t)\right)^\top\right]$. The expectation and variance are defined under the permutation null distribution. Here, $n_{le}^{[a,b]}$ and $n_{ri}^{[a,b]}$ are pre-specified endpoints for the scan. In the following, we use $\lceil a + 0.1(b-a+1)\rceil$ and $\lfloor b - 0.1(b-a+1)\rfloor$ as default choices for $n_{le}^{[a,b]}$ and $n_{ri}^{[a,b]}$, respectively, where $\lceil\cdot\rceil$ is the ceiling function, and $\lfloor\cdot\rfloor$ is the floor function. We will focus on the generalized edge-count statistic in this paper as it considers a useful pattern for high-dimensional data and works well for a wide range of alternatives (Chen & Friedman 2017). Chu & Chen (2019) also provided an analytic p-value approximation for the test statistics (2.1), and we denote it by $\hat{p}(\{\mathbf{y}_i : a \le i \le b\})$ in the following.

2.2. Step 1: Candidate Change-point Search

We adapt the idea of WBS to construct the pool of candidate change-points. The pseudocodes are provided in Algorithm 1 (G.WBS). Let α be the pre-specified significance level and MinLen be the minimum length of generated intervals. The number of randomly generated intervals is L.

Algorithm 1 Change-points search by graph-based WBS

function G.WBS(a, b, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
  if $b - a + 1 <$ MinLen **then**
    STOP
  end if
  if $L \ge (b - a - \text{MinLen} + 2)(b - a - \text{MinLen} + 3)/2$ **then**
    $L \leftarrow (b - a - \text{MinLen} + 2)(b - a - \text{MinLen} + 3)/2$
    Draw all intervals $[a_l, b_l] \subseteq [a, b]$, $l = 1, \ldots, L$, s.t. $b_l - a_l + 1 \ge$ MinLen
  else
    Draw random intervals $[a_l, b_l] \subseteq [a, b]$, $l = 1, \ldots, L$, s.t. $b_l - a_l + 1 \ge$ MinLen
    Add $[a_0, b_0] = [a, b]$ to the set of intervals
  end if
  $l' \leftarrow \text{argmin}_{l \in \{0,\ldots,L\}}\ \hat{p}(\{\mathbf{y}_i : a_l \le i \le b_l\})$
  $\hat{t} \leftarrow \text{argmax}_{n_{le}^{[a_{l'},b_{l'}]} \le t \le n_{ri}^{[a_{l'},b_{l'}]}}\ S^{[a_{l'},b_{l'}]}(t)$
  if $\hat{p}(\{\mathbf{y}_i : a_{l'} \le i \le b_{l'}\}) < \alpha$ (or $S^{[a_{l'},b_{l'}]}(\hat{t}) > \zeta_n$) **then**
    Add $\hat{t}$ to the set $\tilde{\boldsymbol{\tau}}$.
    G.WBS(a, $\hat{t}$, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
    G.WBS($\hat{t} + 1$, b, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
  else
    STOP
  end if
end function

The function G.WBS can be applied recursively to find candidate change-points. It starts by applying the generalized edge-count statistic to L randomly generated intervals. If the smallest p-value is less than the significance level α (or the largest statistic value is greater than pre-specified threshold ζ_n) , we add the corresponding detected change-point into $\tilde{\boldsymbol{\tau}}$ and continue by applying the function to the subsegments. All potential intervals will be scanned if the subsequence $\{\mathbf{y}_i : a \le i \le b\}$ is too short to draw L different intervals longer than MinLen. G.WBS combines advantages of WBS (Fryzlewicz 2014) and WBS2 (Fryzlewicz 2020) by choosing intervals adaptively. Intervals in traditional WBS are generally too long and are likely to span multiple change-points, causing power loss. The recursive drawing in G.WBS gradually narrows the search and generates shorter intervals which are more likely to cover single change-point.

SBS is also applicable for multiple change-point detection. Following the recommendation in Kovács et al. (2020), the collection of seeded intervals used in G.SBS is

$$\mathcal{I}_\gamma = \bigcup_{k=1}^{\lfloor \log_\gamma \frac{\text{MinLen}-1}{n}+1 \rfloor} \bigcup_{j=1}^{2\lceil (\frac{1}{\gamma})^{k-1} \rceil - 1} \left\{ \left[\lfloor (j-1)s_k \rfloor, \lceil (j-1)s_k + n\gamma^{k-1} \rceil \right] \right\},$$

where $s_k = n(1-\gamma^{k-1})/(2\lceil 1/\gamma^{k-1} \rceil - 2)$ and decay parameter $\gamma = \sqrt{0.5}$.

Algorithm 2 Change-points search by graph-based SBS

function G.SBS(a, b, $\tilde{\boldsymbol{\tau}}$, α, $\mathcal{I}_\gamma$, MinLen)
  if $b - a + 1 <$ MinLen **then**
    STOP
  end if
  $\mathcal{M}_{a,b} \leftarrow$ set of indices $l \in \mathcal{I}_\gamma$ such that $[a_l, b_l] \subseteq [a, b]$
  $\mathcal{M}_{a,b} \leftarrow \mathcal{M}_{a,b} \cup \{0\}$, where $[a_0, b_0] = [a, b]$
  $l' \leftarrow \text{argmin}_{l \in \mathcal{M}_{a,b}} \hat{p}(\{\mathbf{y}_i : a_l \le i \le b_l\})$
  $\hat{t} \leftarrow \text{argmax}_{n_{le}^{[a_{l'}, b_{l'}]} \le t \le n_{ri}^{[a_{l'}, b_{l'}]}} S^{[a_{l'}, b_{l'}]}(t)$
  if $\hat{p}(\{\mathbf{y}_i : a_{l'} \le i \le b_{l'}\}) < \alpha$ (or $S^{[a_{l'}, b_{l'}]}(\hat{t}) > \zeta_n$) **then**
    Add $\hat{t}$ to the set $\tilde{\boldsymbol{\tau}}$
    G.SBS(a, $\hat{t}$, $\tilde{\boldsymbol{\tau}}$, α, $\mathcal{I}_\gamma$, MinLen)
    G.SBS($\hat{t}+1$, b, $\tilde{\boldsymbol{\tau}}$, α, $\mathcal{I}_\gamma$, MinLen)
  else
    STOP
  end if
end function

In the first step, we aim to find all potential change-points, so it is tolerable to have some falsely detected change-points. Generally, larger values of α, γ, and L would bring in more candidate change-points, but also result in a longer computation time. Investigators can set those parameters according to their needs. The default value of L and α are 100 and 0.01, respectively. The value of MinLen affects the power of detecting frequent changes. The functions G.WBS and G.SBS could detect change-points that are at least MinLen/2 apart from each other. We set MinLen to be 10 as the default choice, which is usually enough even for cases with frequent changes.

We have no intention to compare the two methods in detail since they are both reliable in general. One may choose between them according to their needs and understandings. From our experience, when change-points are sparse and evenly distributed, G.SBS has similar power and faster speed compared with G.WBS. However, when there are more frequent change-points, G.WBS performs better, as it scans on more intervals when subsequences $\{\mathbf{y}_i : a \le i \le b\}$ are short.

Next we present the consistency of estimated change-points $\tilde{\boldsymbol{\tau}}$ returned by G.WBS. We will show that both the number and relative positions of detected change-points are consistent when $n \to \infty$ under certain conditions of the change-points. Throughout the paper, it is only assumed that F_j's are continuous multivariate distributions. Consider two sequences u_n and v_n. Write $u_n \lesssim v_n$ if there exists $c > 0$ and $n_0 \in \mathbb{N}^+$ not depending on n such that $u_n \le cv_n$, for all $n > n_0$. Also, if we have $v_n \lesssim u_n$ at the same time as $u_n \lesssim v_n$, then write $u_n \asymp v_n$ or $u_n = O(v_n)$. Besides, $u_n \prec v_n$ or $u_n = o(v_n)$ means that $\lim_{n\to\infty} u_n/v_n = 0$. Let $T^{[a_l,b_l]}(u) = \lim_{n\to\infty} S^{[a_l,b_l]}(nu)/(b_l - a_l)$, $\omega_j = \lim_{n\to\infty} \tau_j/n$, $\tilde{\omega}_j = \lim_{n\to\infty} \tilde{\tau}_j/n$, $\boldsymbol{\omega} = \{\omega_1, \ldots, \omega_m\}$, and $\tilde{\boldsymbol{\omega}} = \{\tilde{\omega}_1, \ldots, \tilde{\omega}_{|\tilde{\boldsymbol{\tau}}|}\}$, where $|\cdot|$ is the cardinality of a set.

THEOREM 2.2.1. *Assume there are a fixed number of change-points $\{\tau_j\}_{j=1}^m$ and distributions $\{F_j\}_{j=0}^m$ that are continuous multivariate distributions with density functions f_j satisfying $f_j \ne f_{j+1}$ for all j's. The spacing between contiguous change-points $\tau_{j+1} - \tau_j \asymp n$. Assume the similarity graph is k-MST based on the Euclidean distance, where $k \asymp 1$. If $\zeta_n \asymp n^{1/2}$, and $L \succ 1$ in* G.WBS, *and for each generated interval $[a_l, b_l]$ in* G.WBS,

$$\sup_{u \in \left[\frac{n_{le}^{[a_l,b_l]}}{n}, \frac{n_{ri}^{[a_l,b_l]}}{n}\right]} \left| \frac{S^{[a_l,b_l]}(nu)}{b_l - a_l} - T^{[a_l,b_l]}(u) \right| \xrightarrow{p} 0, \tag{2.2}$$

then for all $\epsilon > 0$,

$$P\left(|\widetilde{\boldsymbol{\omega}}| = |\boldsymbol{\omega}|\right) \to 1,$$

$$P\left(\forall\, \omega_j \in \boldsymbol{\omega}, \exists\, \tilde{\omega}_{j'} \in \widetilde{\boldsymbol{\omega}}, |\tilde{\omega}_{j'} - \omega_j| < \epsilon\right) \to 1,$$

as $n \to \infty$.

The theorem is proved by a two step procedure. First we use the trick of extremum estimator to prove the singe change-point consistency. Next we generalize the consistency to multiple change-points scenario. The details of the proof is given in Supplement A.1.

REMARK 2.2.1. *In the proof of Theorem 2.2.1, we showed that there exists some function* $c(n)$*, when* $c(n) \prec \zeta_n \prec n$*, the consistency is also achieved. The* $c(n)$ *could be numerically simulated by*

$$\max_{l\in\{1\ldots,L\}} \max_{u\in\{\frac{1}{n},\ldots,1\}} \left[Z^*_{l,diff}(u)\right]^2 + \left[Z^*_{l,w}(u)\right]^2,$$

where $Z^*_{l,diff}(u)$ *and* $Z^*_{l,w}(u)$ *are independent Gaussian process defined in Theorem 4.1 and 4.3 in Chu and Chen (2019). Some simulation results are shown in Table 2.1. For general purpose of approximating* ζ_n *in* G.WBS*,* $\zeta_n \asymp n^{1/2}$ *would suffice. A moderate order of* ζ_n *can guarantee the consistency theoretically and has enough power empirically.*

TABLE 2.1. Simulated average $\max_{l\in\{1\ldots,L\}} \max_{u\in\{1/n,\ldots,1\}} [Z^*_{l,\text{diff}}(u)]^2 + [Z^*_{l,w}(u)]^2$ and corresponding standard deviations (in parenthesis) with 1000 replications.

		n				
		50	100	500	1000	3000
L	1	6.81 (2.75)	7.51 (2.83)	8.54 (2.93)	8.99 (2.86)	9.41 (3.07)
	50	15.41 (2.67)	16.18 (2.66)	17.51 (2.83)	17.88 (2.83)	18.75 (2.99)
	100	16.97 (2.77)	17.73 (2.78)	19.09 (2.85)	19.57 (3.04)	19.93 (2.84)
	200	18.46 (2.66)	19.18 (2.60)	20.52 (2.77)	20.93 (2.75)	21.32 (2.63)

REMARK 2.2.2. *Condition (2.2) requires uniform convergence of* $S^{[a_l,b_l]}(nu)/(b_l - a_l)$ *towards its limit. This is necessary to ensure the maximizer of* $S^{[a_l,b_l]}(nu)$ *is close to the maximizer of its limit. We checked the convergence through numerical studies in Supplement A.2, and the convergence is satisfactory when* $n > 1000$.

2.3. Step 2: Candidates Pruning with a Goodness-of-fit Statistic

To improve finite sample performance, especially to reduce false discoveries, we use a goodness-of-fit statistic for candidate change-points pruning.

Let $\tilde{\boldsymbol{\tau}} = \{\tilde{\tau}_1, \ldots, \tilde{\tau}_{\tilde{m}}\}$ denotes the set of cadidate change-points found in step 1, where $1 \leq \tilde{\tau}_1 < \cdots < \tilde{\tau}_{\tilde{m}} \leq n-1$, and $\tilde{m} = |\tilde{\boldsymbol{\tau}}|$. We define a set $\tilde{\boldsymbol{\eta}}$ on top of $\tilde{\boldsymbol{\tau}}$:

$$\tilde{\boldsymbol{\eta}} = \{\tilde{\eta}_0, \ldots, \tilde{\eta}_{\tilde{m}}\} = \begin{cases} \{0, n\} & \text{if } \tilde{m} = 1, \\ \{0, \lceil \frac{\tilde{\tau}_1 + \tilde{\tau}_2}{2} \rceil, \ldots, \lceil \frac{\tilde{\tau}_{\tilde{m}-1} + \tilde{\tau}_{\tilde{m}}}{2} \rceil, n\} & \text{if } \tilde{m} \geq 2. \end{cases}$$

We then define an adjacent sum goodness-of-fit statistic in the following way:

$$\mathrm{AS}(\tilde{\boldsymbol{\tau}}) = \sum_{j=1}^{\tilde{m}} S^{[\tilde{\eta}_{j-1}+1, \tilde{\eta}_j]}(\tilde{\tau}_j).$$

Each term in $\mathrm{AS}(\tilde{\boldsymbol{\tau}})$ is a local two-sample test statistic measuring credibility of a candidate change-point $\tilde{\tau}_j$. The subsample used in $S^{[\tilde{\eta}_{j-1}+1, \tilde{\eta}_j]}(\tilde{\tau}_j)$ starts from the middle point of $\tilde{\tau}_j$ and $\tilde{\tau}_{j-1}$ and ends at the middle point of $\tilde{\tau}_j$ and $\tilde{\tau}_{j+1}$. If $\tilde{\tau}_j$ is a true change-point, it will lead to a relatively large $S^{[\tilde{\eta}_{j-1}+1, \tilde{\eta}_j]}(\tilde{\tau}_j)$. While if $\tilde{\tau}_j$ is not a change-point, we would expect $S^{[\tilde{\eta}_{j-1}+1, \tilde{\eta}_j]}(\tilde{\tau}_j)$ to be relatively small.

We illustrate how $\mathrm{AS}(\tilde{\boldsymbol{\tau}})$ works through a toy example: a normally distributed sequence with $n = 400$ and $\boldsymbol{\tau} = \{90, 230, 320\}$. In a simulation run, candidate change-points derived from step 1 are $\tilde{\boldsymbol{\tau}}^4 = \{90, 229, 320, 377\}$, with a falsely detected change-point 377. Now, $\mathrm{AS}(\tilde{\boldsymbol{\tau}}^4) = 141.30$ and 4 local statistics are shown in Figure 2.1 (a). When $\tilde{\boldsymbol{\tau}}$ contains false discoveries, $S^{[\tilde{\eta}_{j-1}+1, \tilde{\eta}_j]}(\tilde{\tau}_j)$'s on falsely detected $\tilde{\tau}_j$'s are usually small, as they are calculated on homogeneous subsequences ($S^{[350,400]}(377)$ in Figure 2.1 (a)). In addition, the existence of falsely detected $\tilde{\tau}_j$'s decreases the values of $S^{[\tilde{\eta}_{j-2}+1, \tilde{\eta}_{j-1}]}(\tilde{\tau}_{j-1})$ ($S^{[276,349]}(320)$ in Figure 2.1 (a)) and $S^{[\tilde{\eta}_j+1, \tilde{\eta}_{j+1}]}(\tilde{\tau}_{j+1})$ as the observations in $[\tilde{\eta}_{j-1} + 1, \tilde{\eta}_j]$ are seized by $\tilde{\tau}_j$. When $\tilde{\boldsymbol{\tau}}$ is close to true change-points, $\mathrm{AS}(\tilde{\boldsymbol{\tau}})$ tends to be maximized (Figure 2.1 (b)). When $\tilde{\boldsymbol{\tau}}$ misses true change-points, $\mathrm{AS}(\tilde{\boldsymbol{\tau}})$ would lose the portions contributed by those left-out true change-points. Also, those left-out true change-points might affect the remaining $\tilde{\tau}_j$'s in $\tilde{\boldsymbol{\tau}}$ as some corresponding intervals could contain more than two

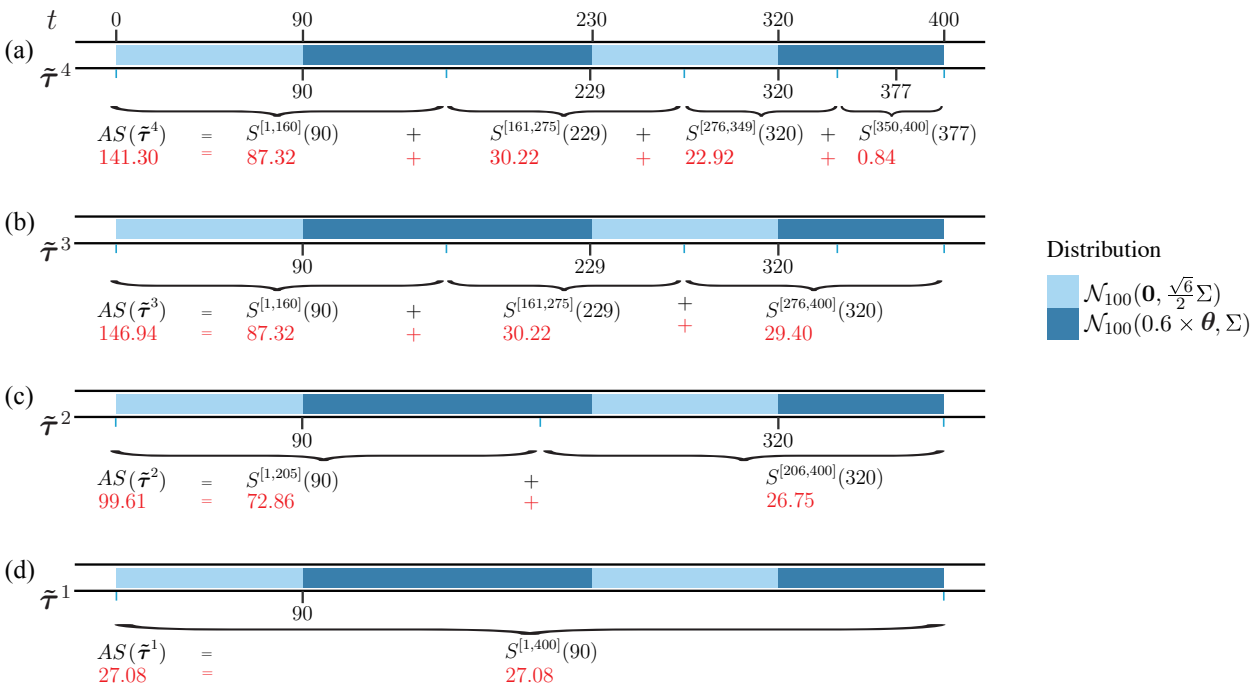

FIGURE 2.1. AS($\tilde{\boldsymbol{\tau}}$) on four possible change-points sets $\tilde{\boldsymbol{\tau}}$. Parameters of the distributions: $\boldsymbol{\theta}$ is a vector with the first 20 elements all ones and the rest zeros, and $\Sigma_{jk} = 0.3^{|j-k|}$ for $1 \leq j, k \leq 100$. In (a), $\tilde{\boldsymbol{\tau}}^4$ overfits the data. In (b), $\tilde{\boldsymbol{\tau}}^3$ is close to the true $\boldsymbol{\tau}$, and AS($\tilde{\boldsymbol{\tau}}$) is maximized. In (c) and (d), underestimated $\tilde{m}$ lead to small adjacent sum values.

segments, causing $S^{[\tilde{\eta}_{j-1}+1,\tilde{\eta}_j]}(\tilde{\tau}_j)$'s on true change-points to decrease. ($S^{[206,400]}(320)$ in Figure 2.1 (c)).

Under the null that there is no change-point in the entire sequence, AS($\tilde{\boldsymbol{\tau}}$) has good asymptotic properties. Let $G_i^{[a,b]}$ be a subgraph of $G^{[a,b]}$ containing all edges that connect to node $\mathbf{y}_i$, and $|G_i^{[a,b]}|$ be the degree of node $\mathbf{y}_i$ in $G^{[a,b]}$. Let $node_{G_i^{[a,b]}}$ be the set of nodes connected by $G_i^{[a,b]}$ excluding node i, $N_{sq}^{[a,b]}$ be the number of squares in the graph $G^{[a,b]}$, $\tilde{d}_i^{[a,b]} = \left|G_i^{[a,b]}\right| - 2|G^{[a,b]}|/(b-a+1)$, and $V_{G^{[a,b]}} = \sum_{i=a}^{b} \left|G_i^{[a,b]}\right|^2 - 4(|G^{[a,b]}|)^2/(b-a+1)$, where $|G^{[a,b]}|$ is the number of edges in $G^{[a,b]}$.

THEOREM 2.3.1. *Under the null hypothesis that there is no change-point in the sequence, for mutually disjoint intervals* $[a_j, b_j]$ *and* $t_j \in [a_j, b_j]$, $j = 1, \ldots, m$, *when*

$$\sum_{i=a_j}^{b_j} |G_i^{[a_j,b_j]}|^2 = o(|G^{[a_j,b_j]}|^{\frac{3}{2}}), \quad \sum_{i=a_j}^{b_j} |\tilde{d}_i^{[a_j,b_j]}|^3 = o(V^{\frac{3}{2}}_{G^{[a_j,b_j]}}),$$

$$\sum_{i=a_j}^{b_j} (\tilde{d}_i^{[a_j,b_j]})^3 = o(V_{G^{[a_j,b_j]}} \sqrt{|G^{[a_j,b_j]}|}), \quad N_{sq}^{[a_j,b_j]} = o(|G^{[a_j,b_j]}|^2),$$

$$\sum_{i=a_j}^{b_j} \sum_{k,l \in \ node \ G_i^{[a_j,b_j]}}^{k \neq l} \tilde{d}_k^{[a_j,b_j]} \tilde{d}_l^{[a_j,b_j]} = o\left(|G^{[a_j,b_j]}| V_{G^{[a_j,b_j]}}\right)$$

as $b_j - a_j \to \infty$, *and* $t_j/(b_j - a_j) \to u_j$, $0 < u_j < 1$ *holds for each* j *under the permutation null distribution where the observations in each interval* $[a_j, b_j]$ *are permuted within the interval, we have*

$$\sum_{j=1}^{m} S^{[a_j,b_j]}(t_j) \xrightarrow{d} \chi^2_{2m}.$$

Theorem 2.3.1 is a natural extension of Theorem 2.1 of Zhu & Chen (2021). The conditions on the graphs look complicated, but are not hard to satisfy. For example, for $k-$MST constructed on multivariate data, the conditions always hold for $k = O(n^\beta)$, $\beta < 0.5$ (Zhu & Chen 2021). These conditions are sufficient conditions. In practice, we found the conclusion holds for even denser graphs.

With the results in Theorem 2.3.1, we could adopt model selection techniques to prune candidate change-points. Let F_j's in (1.1) be univariate Gaussian distributions with unknown means and a known variance 1, and $\mathbb{M}_m$ be the Gaussian model with m change-points. Correspondingly, $\mathbb{M}_0$ is the Gaussian model with no change-point. It is obvious that (see for example Zhang (2005)):

$$2 \log \frac{\mathbf{P}(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \mathbb{M}_m)}{\mathbf{P}(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \mathbb{M}_0)} \sim \chi^2_m \tag{2.3}$$

under the universal null $\mathbb{M}_0$. The corresponding BIC in selecting change-points is

$$2 \log \frac{\mathbf{P}(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \mathbb{M}_m)}{\mathbf{P}(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \mathbb{M}_0)} - m \log n. \tag{2.4}$$

The null distribution of (2.3) and the asymptotic distribution of $\mathrm{AS}(\tilde{\boldsymbol{\tau}})$ are both chi-square, but with different degrees of freedom. Hence, we propose a pseudo-BIC for our framework:

$$\text{pseudo-BIC}(\tilde{\boldsymbol{\tau}}) = \mathrm{AS}(\tilde{\boldsymbol{\tau}}) - 2\tilde{m}\log n. \tag{2.5}$$

The penalty term used in (2.5) is two times that in (2.4), which is consistent with the degrees of freedom. We find the added penalty term could alleviate overfitting generally (See Figure 2.2 for an example).

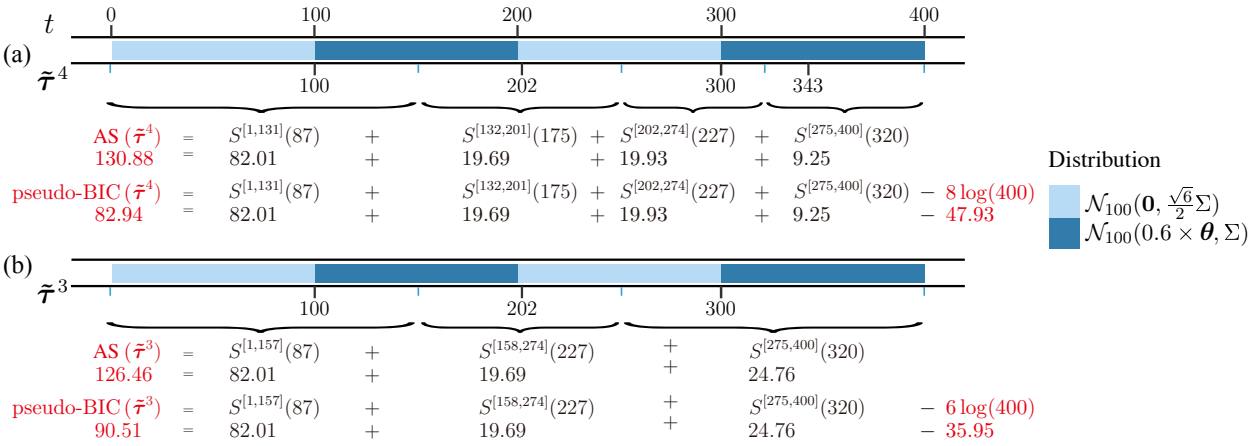

FIGURE 2.2. Comparison between $\mathrm{AS}(\tilde{\boldsymbol{\tau}})$ and pseudo-BIC$(\tilde{\boldsymbol{\tau}})$ on two possible change-points sets $\tilde{\boldsymbol{\tau}}^4$ and $\tilde{\boldsymbol{\tau}}^3$. Here, pseudo-BIC$(\tilde{\boldsymbol{\tau}}^3)$ is greater than pseudo-BIC$(\tilde{\boldsymbol{\tau}}^4)$, while $\mathrm{AS}(\tilde{\boldsymbol{\tau}}^3)$ is less than $\mathrm{AS}(\tilde{\boldsymbol{\tau}}^4)$.

After conducting extensive numerical studies, we notice a drawback of the adjacent sum in that each two-sample test statistic uses only half of the information of subsequences. This might cause power loss if homogeneous subsequences are short or signal is relatively weak. Especially, it might result in loss of true change-points. For example, in Figure 2.3, pseudo-BIC$(\tilde{\boldsymbol{\tau}}^2)$ is almost equal to pseudo-BIC$(\tilde{\boldsymbol{\tau}}^3)$ while $\tilde{\boldsymbol{\tau}}^2$ misses the true change-point 200. We hence adopt a more aggressive quantity by defining the following expanded adjacent sum statistic

$$\mathrm{eAS}(\tilde{\boldsymbol{\tau}}) = \sum_{j=1}^{\tilde{m}} S^{[\tilde{\tau}_{j-1}+1, \tilde{\tau}_{j+1}]}(\tilde{\tau}_j),$$

where $\tilde{\tau}_0 = 0$ and $\tilde{\tau}_{\tilde{m}+1} = n$. This expanded version uses two times the information in each summand compared to the non-overlapped $\mathrm{AS}(\tilde{\boldsymbol{\tau}})$. When $\tilde{\boldsymbol{\tau}}$ is close to the true change-points $\boldsymbol{\tau}$,

eAS($\tilde{\boldsymbol{\tau}}$) will be greater than AS($\tilde{\boldsymbol{\tau}}$) (Figure 2.3 (a2)). On the other hand, when $\tilde{\boldsymbol{\tau}}$ misses some true change-points, $S^{[\tilde{\tau}_{j-1}+1,\tilde{\tau}_{j+1}]}(\tilde{\tau}_j)$ is more likely to cross true change-points and results in a smaller value (Figure 2.3 (b2)). The corresponding pseudo-BIC is also updated to the expanded pseudo-BIC:

$$\text{ep-BIC}(\tilde{\boldsymbol{\tau}}) = \text{eAS}(\tilde{\boldsymbol{\tau}}) - c\tilde{m}\log n. \tag{2.6}$$

Due to the local dependency between $S^{[\tilde{\tau}_{j-1}+1,\tilde{\tau}_{j+1}]}(\tilde{\tau}_j)$'s resulted from overlapping regions, it is challenging to give an appropriate c analytically. We did some simulation studies (Supplement A.4) and found $c = 2$ is still a good choice, so this is set as the default value.

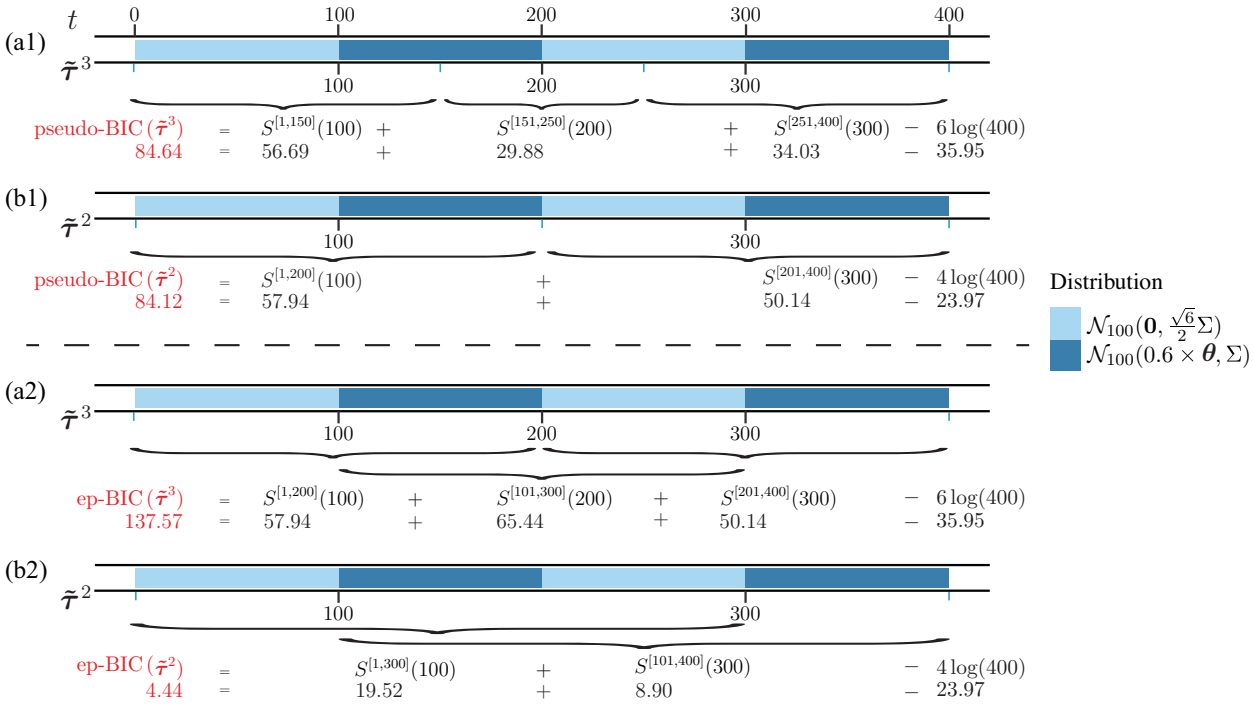

FIGURE 2.3. Comparison between pseudo-BIC($\tilde{\boldsymbol{\tau}}$) and ep-BIC($\tilde{\boldsymbol{\tau}}$). By using more information in each generalized edge-count statistic, ep-BIC is more likely to choose the correct model.

Given the set $\tilde{\boldsymbol{\tau}}$ of candidate change-points and the goodness-of-fit statistic ep-BIC($\tilde{\boldsymbol{\tau}}$), the pruning of the change-points becomes a model selection problem. We use backward elimination to fastly get the final set of pruned change-points $\hat{\boldsymbol{\tau}}$ as shown in Algorithm 5 (G.BE). G.BE returns a sequence of ep-BIC($\tilde{\boldsymbol{\tau}}^l$), and the $\tilde{\boldsymbol{\tau}}^l$ with the largest ep-BIC value is chosen as $\hat{\boldsymbol{\tau}}$. Note that G.BE

stops until there is no change-point. Together with the change-point dendrogram proposed in the following (Section 2.5), the sequence of ep-BIC($\tilde{\boldsymbol{\tau}}^l$) provides investigators an ordered list of the change-points.

REMARK 2.3.1. *Given the nature of model selection, one may consider all subset approach, i.e., evaluating all possible subsets of $\tilde{\boldsymbol{\tau}}$ and choose the one with the best fit. This can be easily done when $|\tilde{\boldsymbol{\tau}}|$ is small. However, all subset approach can be computationally inhibitive in real applications where hundreds or thousands of change-points exist.*

Algorithm 3 Backward elimination with ep-BIC

function G.BE($\tilde{\boldsymbol{\tau}}$)
  $l \leftarrow \tilde{m}$
  $\tilde{\boldsymbol{\tau}}^l \leftarrow \tilde{\boldsymbol{\tau}}$
  while $|\tilde{\boldsymbol{\tau}}^l| \geq 1$ **do**
    $\mathbf{T}^l$ = collection of change-points set $\tilde{\boldsymbol{\tau}}^l \backslash \{\tilde{\tau}_j^l\}$, where $\tilde{\tau}_j^l \in \tilde{\boldsymbol{\tau}}^l$, $j = 1, \ldots, l$
    $\tilde{\boldsymbol{\tau}}^{l-1} \leftarrow \text{argmax}_{\boldsymbol{t} \in \mathbf{T}^l} \text{ep-BIC}(\boldsymbol{t})$
    $l \leftarrow l - 1$
  end while
  $\hat{m} \leftarrow \text{argmax}_{0 \leq l \leq \tilde{m}} \text{ep-BIC}(\tilde{\boldsymbol{\tau}}^l)$
  $\hat{\boldsymbol{\tau}} \leftarrow \tilde{\boldsymbol{\tau}}^{\hat{m}}$
  return $\hat{\boldsymbol{\tau}}$
end function

2.4. Graph Choice

Now we explore the choices of similarity graphs used in the two steps and their impact. From earlier works on graph-based tests (Chen et al. 2018, Chen & Friedman 2017, Chen & Zhang 2013, Friedman & Rafsky 1979), k-MST is a recommended choice. However, the choice of k is unsettled. Chen & Friedman (2017), Chen & Zhang (2015), Chu & Chen (2019), Friedman & Rafsky (1979) showed that, for $k = O(1)$, larger k's are preferred. Zhu & Chen (2021) further showed that k of a higher order than $O(1)$ could even result in a higher power. We next discuss the choice of k for the two steps separately.

In step 1, power is the main factor. By design, it can have some false discoveries. Without any prior knowledge about a sequence, it is intuitive to choose k depending on n. A k-MST built on $G^{[a_l, b_l]}$ contains $k(b_l - a_l)$ edges, while the information contained in the distance matrix is of order

$O((b_l - a_l)^2)$. We consider $k = (b_l - a_l)^\lambda$, $0 < \lambda < 1$, and compare the detection power for different λ's under various simulation settings. Specifically, i.i.d. sequences with a change-point $\tau = n/2$ were generated from three distribution pairs. The $\lfloor n^\lambda \rfloor$-MST was constructed based on the Euclidean distance. If $\hat{p}(\{\mathbf{y}_i : 1 \le i \le n\}) < 0.01$ and $\text{argmax}_t S^{[1,n]}(t) \in [\tau - 0.05n, \tau + 0.05n]$, we deem it a successful detection. Detection power is defined as the proportion of successful detections. From Figure 2.4, when $\lambda = 0.5$, graph-based method shows adequate detection power, though the optimal λ varies between 0.3 to 0.7. Therefore, $\min(30, \lfloor\sqrt{b_l - a_l}\rfloor)$-MST is used as the default similarity graph in Algorithm 1 and 2. The upper bound 30 is set merely for computational consideration for very long sequences.

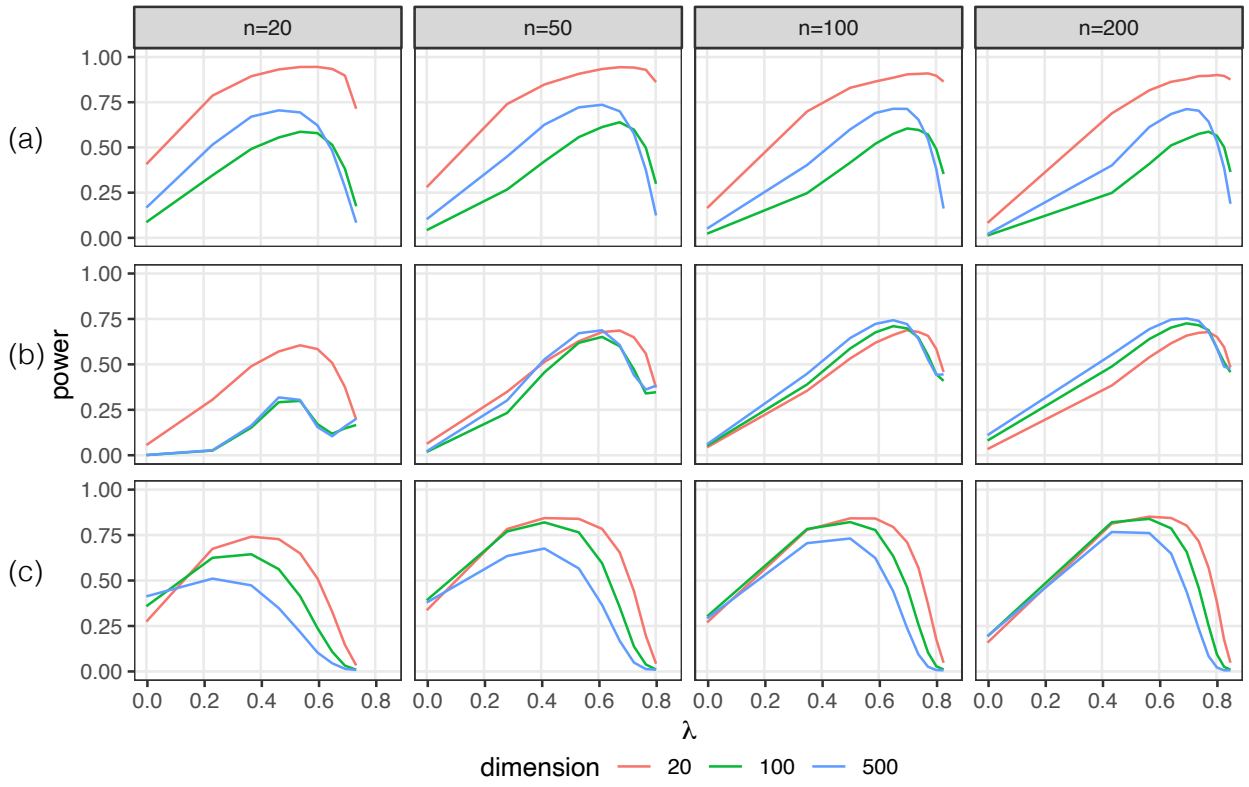

FIGURE 2.4. Estimated detection power (from 5,000 replicates) for singe change-point detection based on the $\lfloor n^\lambda \rfloor$-MST. Three distribution pairs are (a): $(\mathcal{N}_d(\mathbf{0}, \Sigma), \mathcal{N}_d(\frac{20}{\sqrt{dn}}\mathbf{1}, \Sigma))$, (b): $(\mathcal{N}_d(\mathbf{0}, \Sigma), \mathcal{N}_d(\frac{15}{\sqrt{dn}}\mathbf{1}, (1 + \frac{3}{2\sqrt{n}})\Sigma))$, and (c): $(t_{3,d}(\mathbf{0}, I), t_{3,d}(\frac{12}{\sqrt{n}\log d}\mathbf{1}, I))$, where $\Sigma_{jk} = 0.3^{|j-k|}$ and I is the identity matrix. Here, specific alternatives are chosen so that the detection power is moderate to be comparable across different λ's.

For step 2, however, comparability of ep-BIC($\tilde{\boldsymbol{\tau}}$) on different $\tilde{\boldsymbol{\tau}}$'s is more essential. Setting k to be a function of $\tilde{\tau}_{j+1} - \tilde{\tau}_{j-1}$ might lead to incomparable $S^{[\tilde{\tau}_{j-1}+1,\tilde{\tau}_{j+1}]}(\tilde{\tau}_j)$'s between long and short subsequences. Even when a signal is strong for a short subsequence, a small k may yield small $S^{[\tilde{\tau}_{j-1}+1,\tilde{\tau}_{j+1}]}(\tilde{\tau}_j)$. On the contrary, for a long sequence with a weak signal, a large k may yield large statistic values, making ep-BIC prefer over-simplified models. To avoid this imbalance while keeping fine test performance, a constant $k-$MST, like the 5-MST, is preferred. Given that 5-MST can not be built on very short intervals, we set the default choice to be $\min(5, \lfloor\sqrt{\tilde{\tau}_{j+1} - \tilde{\tau}_{j-1}}\rfloor)$-MST for step 2.

2.5. Result Visualization

Given estimated change-points $\hat{\boldsymbol{\tau}}$, remaining questions are whether these change-points are of scientific interest and what the relationship among those subsegments is. We provide a visualization tool to explore the hierarchical structure of detected change-points naturally resulted from G.BE.

In each step of G.BE, a suspicious change-point is removed, which is equivalent to merging two neighboring subsequences. So we build a change-point dendrogram with the height evaluated by negative ep-BIC (Figure 2.5). The bottom of this dendrogram is $\hat{\boldsymbol{\tau}}$ and the top is the last merged change-point.

The tree structure of a change-point dendrogram depicts the relationship between estimated change-points and their relative importance. If, for example, removing a change-point results in minimal change in height, that change-point should be considered less important or even doubtful. In contrary, a change-point that leads to a considerable ep-BIC lose is locally more important. Change-points close to the root of the dendrogram are usually globally important. These change-points are removed at the end of backward elimination, which shows their importance in maintaining a high ep-BIC. In other words, these change-points cuts the inhomogeneous sequence into roughly homogeneous segments in a best effort with a fixed (small) number of change-points.

One advantage of this hierarchical representation is that cutting the tree at a certain height gives a partitioning clustering at a corresponding level. This provides researchers with the freedom of choosing different scales of study. Often when dealing with complex data, it's more important to grasp key changes than all of them. If so, one can cut the tree close to the top to get the

top-level structure of the data. This is especially helpful when the data sequence is long and full of change-points.

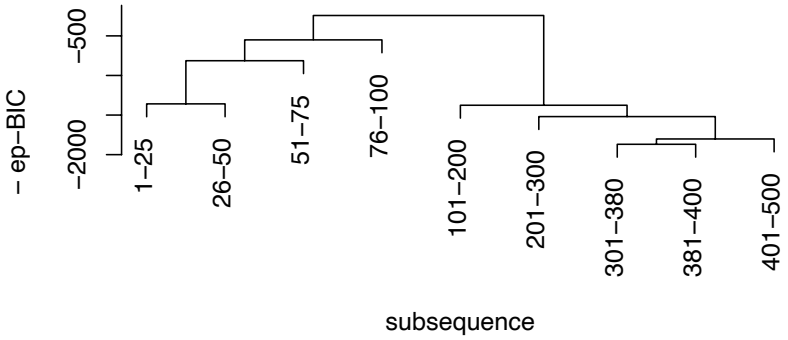

FIGURE 2.5. A change-point dendrogram constructed on a simulated dataset containing 7 change-points, $\boldsymbol{\tau} = \{25, 50, 75, 100, 200, 300, 400\}$. Among those detected change-points, 380 is a falsely detected one. The dendrogram shows that 380 is suspicious as adding 380 increases little in ep-BIC.

2.6. Performance Analysis

In this section, we examine the performance of the proposed approach against two state-of-the-art nonparametric multivariate multiple change-point detection methods: E-Divisive (Matteson & James 2014) and KCP (Arlot et al. 2019) implemented by the R package `ecp` (James & Matteson 2015). Throughout the simulation, we use the default choice of parameters and similarity graphs for the proposed approach. For the E-Divisive approach, we set the minimum cluster size to $\lfloor \min(\tau_{j+1} - \tau_j)/2 \rfloor$ and all other parameters to be default. For KCP, the maximum number of change-point is set to $2m$.

Three methods are tested under five settings, with dimensions $d = 20, 50, 100, 500$, and 1000 for Setting 1-4. The number of truly and falsely detected change-points are reported in Table 2.2 and 2.3, respectively. Location and scale parameters (δ, σ) are chosen for each value of d so that most methods have moderate power to be comparable. Define $\Sigma_{jk} = 0.3^{|j-k|}$, $\boldsymbol{\theta}$ be a $d-$length vector with the first $d/5$ entries equal to 1 and all others equal to 0, and $[\![a, b]\!]$ be the set of integers between a and b. The following models are used to generate the data.

- Setting 1: $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, \Sigma)$ if $i \in [\![1, 50]\!] \cup [\![101, 150]\!] \cup [\![201, 250]\!]$; $\mathbf{y}_i \sim \mathcal{N}_d(\delta\boldsymbol{\theta}, \sigma\Sigma)$ if $i \in [\![51, 100]\!] \cup [\![151, 200]\!] \cup [\![251, 300]\!]$.

- Setting 2: $\mathbf{y}_i \sim t_{3,d}(\mathbf{0}, I)$ if $i \in [\![1, 40]\!] \cup [\![91, 145]\!] \cup [\![191, 255]\!]$; $\mathbf{y}_i \sim t_{3,d}(\delta\boldsymbol{\theta}, \Sigma)$ if $i \in [\![41, 90]\!] \cup [\![146, 190]\!] \cup [\![256, 300]\!]$.
- Setting 3: $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, I)$ if $i \in [\![1, 55]\!] \cup [\![91, 140]\!] \cup [\![196, 255]\!]$; $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, \sigma\Sigma)$ if $i \in [\![56, 90]\!] \cup [\![141, 195]\!] \cup [\![256, 300]\!]$.
- Setting 4: $\mathbf{y}_i \sim t_{3,d}(\mathbf{0}, I)$ if $i \in [\![1, 50]\!]$; $\mathbf{y}_i \sim \mathcal{N}_d(7\boldsymbol{\theta}/\log(d), \Sigma)$ if $i \in [\![51, 65]\!]$; $\mathbf{y}_i \sim \mathcal{N}_d(4\boldsymbol{\theta}/\log(d), \Sigma)$ if $i \in [\![111, 135]\!]$; $\mathbf{y}_i \sim \mathrm{Exp}_d(1) - 1$ if $i \in [\![136, 180]\!]$; $\mathbf{y}_i \sim \mathcal{N}_d(3\boldsymbol{\theta}/\log(d), I)$ if $i \in [\![181, 230]\!]$.
- Setting 5: A sequence of $n = 240$ random networks are generated from the configuration model. To be specific, all nodes in a random graph have degree 2 if $i \in [\![1, 30]\!] \cup [\![71, 115]\!] \cup [\![151, 205]\!]$, otherwise the first 4 nodes have degree 4 and the others have degree 2. Let $\mathbf{y}_i$

TABLE 2.2. Average number of detected true change-points based on 1,000 replicates and corresponding standard deviations (in parenthesis). The largest value under each setting is in bold.

Setting 1	d	20	50	100	500	1000
	(δ, σ)	$(0.6, 1.85)$	$(0.45, 1.75)$	$(0.37, 1.55)$	$(0.1, 1.4)$	$(0.05, 1.35)$
	New (G.WBS)	3.70 (0.99)	**4.29** (0.79)	**4.39** (0.73)	**4.91** (0.34)	**4.97** (0.19)
	New (G.SBS)	**3.75** (0.96)	4.14 (0.86)	4.21 (0.80)	4.74 (0.50)	4.91 (0.28)
	E-Divisive	2.81 (1.35)	3.83 (1.01)	3.88 (1.02)	3.66 (1.17)	4.00 (1.00)
	KCP	3.01 (1.83)	4.03 (1.46)	3.99 (1.43)	3.80 (1.75)	3.90 (1.71)
Setting 2	d	20	50	100	500	1000
	δ	1.1	0.85	0.76	0.64	0.6
	New (G.WBS)	**4.05** (0.88)	**4.25** (0.80)	4.33 (0.80)	4.38 (0.81)	4.12 (0.93)
	New (G.SBS)	**4.05** (0.87)	4.22 (0.83)	**4.36** (0.78)	**4.59** (0.61)	**4.54** (0.67)
	E-Divisive	3.86 (1.00)	3.43 (1.54)	2.78 (1.97)	0.82 (1.58)	0.41 (1.10)
	KCP	0.79 (1.53)	0.66 (1.28)	0.57 (1.08)	0.63 (0.98)	0.50 (0.89)
Setting 3	d	20	50	100	500	1000
	σ	1.9	1.65	1.45	1.2	1.15
	New (G.WBS)	**3.58** (1.00)	**3.99** (0.93)	**4.04** (0.92)	**4.22** (0.84)	**4.27** (0.82)
	New (G.SBS)	3.56 (1.04)	3.96 (0.93)	3.90 (0.94)	4.09 (0.87)	4.14 (0.87)
	E-Divisive	0.64 (0.93)	0.81 (1.07)	0.27 (0.54)	0.10 (0.35)	0.04 (0.20)
	KCP	3.03 (1.59)	2.65 (1.85)	1.29 (1.57)	1.34 (0.98)	1.00 (0.81)
Setting 4	d	20	50	100	500	1000
	New (G.WBS)	**4.66** (0.54)	**4.55** (0.65)	**4.57** (0.62)	**4.70** (0.59)	**4.54** (0.83)
	New (G.SBS)	4.54 (0.62)	4.41 (0.74)	4.41 (0.73)	4.54 (0.67)	4.38 (0.86)
	E-Divisive	4.61 (0.62)	4.26 (0.88)	3.83 (1.02)	2.74 (0.87)	2.37 (0.86)
	KCP	3.06 (2.08)	1.89 (1.85)	1.87 (1.70)	1.35 (1.19)	1.23 (1.10)
Setting 5	number of nodes	20	30	50	75	100
	New (G.WBS)	4.84 (0.40)	4.90 (0.32)	4.93 (0.26)	4.93 (0.25)	4.95 (0.22)
	New (G.SBS)	**4.87** (0.35)	**4.92** (0.28)	**4.95** (0.22)	**4.96** (0.18)	**4.96** (0.21)
	E-Divisive	4.78 (0.45)	3.73 (1.39)	0.87 (1.25)	0.23 (0.60)	0.10 (0.35)
	KCP	3.27 (2.21)	4.14 (0.88)	2.78 (1.21)	1.92 (1.11)	1.56 (0.95)

be the vectorized adjacency matrix of the i-th network so that E-Divisive and KCP are directly applicable, while our method does not need such embedding.

TABLE 2.3. Average number of falsely detected change-points based on 1,000 replicates and corresponding standard deviations (in parenthesis). The smallest value under each setting is in bold.

Setting 1	d	20	50	100	500	1000
	(δ, σ)	$(0.6, 1.85)$	$(0.45, 1.75)$	$(0.37, 1.55)$	$(0.1, 1.4)$	$(0.05, 1.35)$
	New (G.WBS)	1.80 (1.25)	1.46 (1.14)	1.41 (1.08)	0.99 (0.98)	0.90 (0.97)
	New (G.SBS)	**1.39** (1.04)	**1.07** (0.97)	**1.09** (0.94)	**0.72** (0.83)	**0.43** (0.61)
	E-Divisive	1.52 (1.10)	1.14 (0.97)	1.11 (0.98)	1.15 (1.01)	0.96 (0.93)
	KCP	2.74 (2.73)	3.82 (2.54)	5.00 (1.83)	4.62 (2.09)	4.63 (2.00)
Setting 2	d	20	50	100	500	1000
	δ	1.1	0.85	0.76	0.64	0.6
	New (G.WBS)	1.02 (0.96)	0.82 (0.87)	0.75 (0.93)	0.99 (1.40)	1.64 (2.07)
	New (G.SBS)	**0.98** (0.90)	**0.80** (0.84)	0.66 (0.79)	0.50 (0.73)	0.73 (1.16)
	E-Divisive	1.11 (0.94)	0.92 (0.95)	**0.58** (0.77)	**0.27** (0.62)	**0.23** (0.59)
	KCP	1.39 (2.69)	2.00 (3.37)	2.36 (3.72)	3.59 (4.23)	3.53 (4.33)
Setting 3	d	20	50	100	500	1000
	σ	1.9	1.65	1.45	1.2	1.15
	New (G.WBS)	1.93 (1.23)	1.71 (1.18)	1.64 (1.20)	1.61 (1.19)	1.51 (1.13)
	New (G.SBS)	1.60 (1.07)	1.30 (1.03)	1.34 (1.04)	1.21 (0.98)	1.18 (0.98)
	E-Divisive	**0.70** (0.96)	**0.82** (1.04)	**0.45** (0.78)	**0.23** (0.57)	**0.15** (0.46)
	KCP	5.08 (2.67)	4.49 (2.97)	3.55 (3.76)	8.57 (1.00)	8.90 (0.84)
Setting 4	d	20	50	100	500	1000
	New (G.WBS)	**0.36** (0.57)	**0.46** (0.68)	**0.48** (0.68)	1.10 (1.10)	1.70 (1.35)
	New (G.SBS)	0.46 (0.64)	0.59 (0.74)	0.62 (0.75)	**0.80** (0.91)	**1.16** (1.02)
	E-Divisive	1.38 (0.58)	1.34 (0.59)	1.33 (0.57)	1.40 (0.63)	1.47 (0.65)
	KCP	2.25 (2.07)	1.78 (1.78)	1.99 (2.06)	2.69 (2.98)	2.88 (3.12)
Setting 5	number of nodes	20	30	50	75	100
	New (G.WBS)	1.09 (0.95)	1.11 (0.98)	0.94 (0.90)	0.79 (0.85)	0.69 (0.79)
	New (G.SBS)	0.77 (0.83)	**0.76** (0.79)	0.64 (0.73)	0.52 (0.67)	0.52 (0.66)
	E-Divisive	**0.25** (0.52)	0.85 (0.90)	**0.55** (0.90)	**0.27** (0.60)	**0.16** (0.46)
	KCP	3.67 (2.48)	5.86 (0.88)	7.21 (1.21)	8.06 (1.11)	8.42 (0.95)

From Table 2.2 and 2.3, we see that the new method performs the best among these nonparametric methods under most simulation settings – its power is higher than the other two methods, and its false discoveries are on the lower end. For cases where E-Divisive has a lower false discovery than the new method, the power of E-Divisive is very low. Among the two implementations of the new method, G.SBS-based version has similar power and marginally better FDR compared to the G.WBS-based version. E-Divisive and KCP show comparable power under normal settings or low dimensions, but their performance quickly fail under high dimension, covariance matrix change or non-Gaussian data. For KCP, it can have much more falsely detected change-points than correctly

detected ones (e.g. Setting 3, especially at $d = 500$ and 1000). These results show effectiveness and robustness of the new method compared to E-Divisive and KCP.

2.7. Real Data Analysis

We illustrate the new approach on the Neuropixels data that record the activity of neurons in the brain of an awake mouse during spontaneous behaviors (Steinmetz et al. 2019). The original data recorded the position and times of neural firings through eight Neuropixels probes. For illustration, we use the spike data for $d = 176$ neurons in caudate-putamen during the first three minutes. Probes detected spikes in a small area of the brain that may cover more than one neuron. Here, we call this area neuron for simplicity. The three minutes recording was discretized into $n = 5400$ intervals of 1/30 second. Then $\mathbf{y}_{ij}$ represents the number of spikes recorded during time interval i for neuron j. Given the lack of parametric model for such complex neural data, our proposed approach would be an appropriate choice for initial analysis. We use both G.WBS-based and G.SBS-based versions to analyze the sequence, with $\alpha = 0.001$ to control local type I error and $L = 200$ to ensure enough coverage for long sequences. The detected change-points are plotted by dendrograms in Figure 2.6 and 2.7.

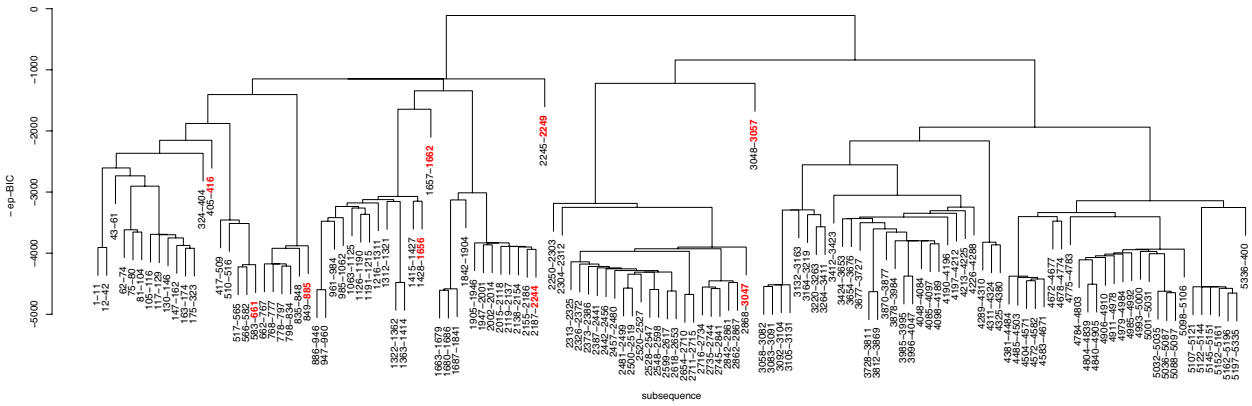

FIGURE 2.6. The change-point dendrogram of the Neuropixels recordings found by the G.WBS-based approach. For better visualization, the height of nodes are set to be at least the height of their children. The change-points in $\tilde{\boldsymbol{\tau}}^9$ are in bold.

For the G.WBS-based version, in step 1, there are 258 candidate change-points detected, and 131 of them are kept after step 2, indicating frequent pattern changes in neural activities. For the G.SBS-based version, the two numbers are 181 and 98, respectively. Among those final 98

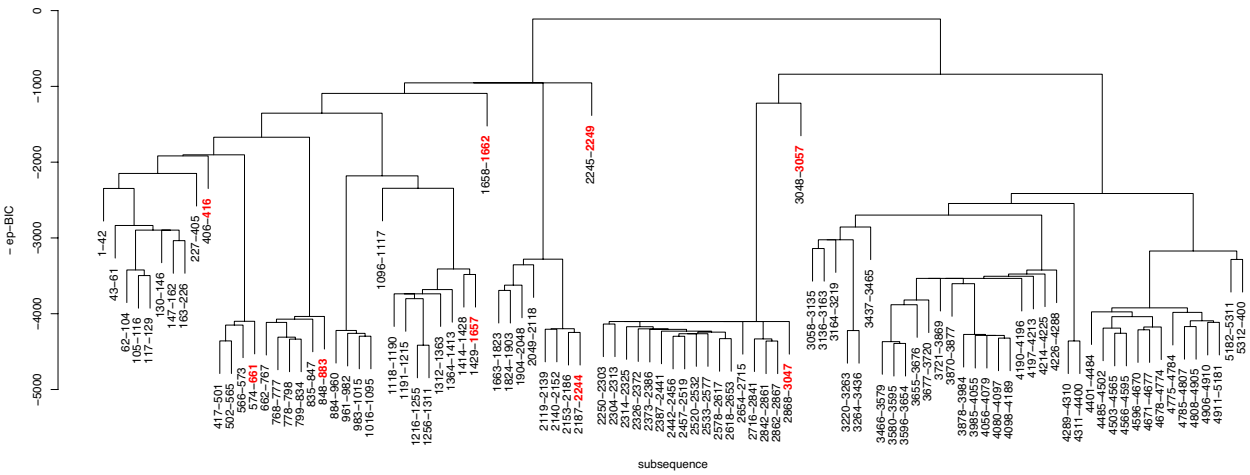

FIGURE 2.7. The change-point dendrogram of the Neuropixels recordings found by the G.SBS-based approach.

change-points in the G.SBS-based version, 73 of them are within 2 observations from the final change-points found by the G.WBS-based version. The top level structure of the two versions are also very similar. The top 9 change-points in the G.WBS-based and G.SBS-based dendrograms are {416, 661, 885, 1656, 1662, 2244, 2249, 3047, 3057} and {416, 661, 883, 1657, 1662, 2244, 2249, 3047, 3057}, respectively (They are in red in Figure 2.6 and 2.7).

Figure 2.8 plots some typical change-point patterns that might be of scientific interests. We call the first pattern *single hyperactive neuron*, where a neuron suddenly becomes hyperactive for a short time interval. The hyperactivity can happen and disappear quickly and unexpectedly. An example is the 78th neuron for $t = 2245$ to 2249 (Figure 2.8 (a)). The second interesting pattern is *overall intensity change*. This pattern is commonly seen in the data, like Figure 2.8 (b), where most neurons are more (or less) active after the detected change-point. After an overall intensity change, the status can last for a long time until the next change-point. The third one is *correlation pattern change*. This sometimes can happen together with overall intensity change, but sometimes not. In Figure 2.8 (c), the overall intensity barely changes after the change-point. If we use the overall number of spikes and perform the Mann-Whitney test, the p-value is 0.777. Nonetheless, if we calculate the correlation matrix of the five most active neurons before and after the change-point, we see that several neurons become more strongly correlated after the change-point (Figure 2.9).

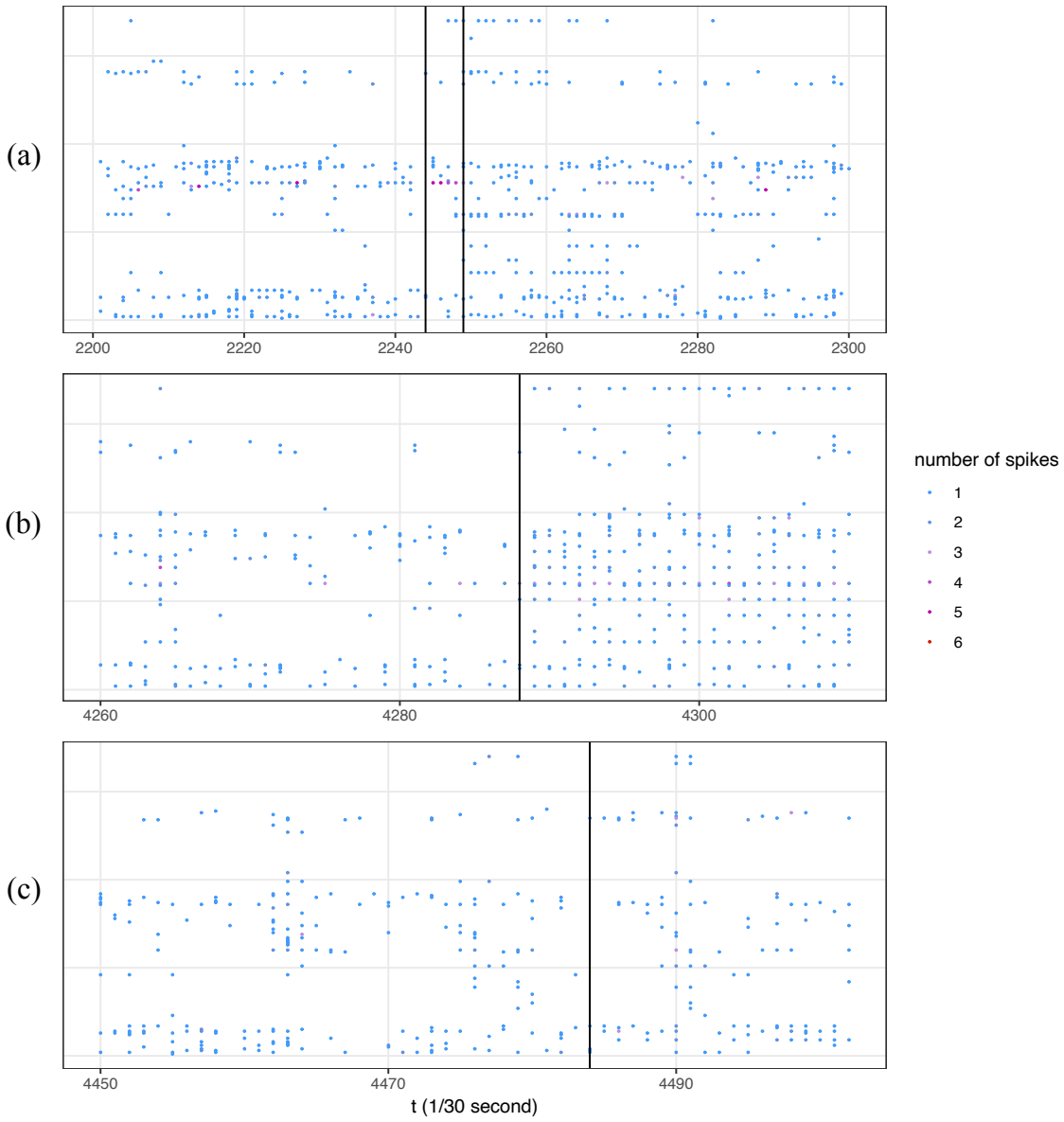

FIGURE 2.8. Neuropixel recordings (each row corresponds to one neuron) and detected change-points. Vertical lines indicate positions of detected change-points. In (a), the 78th neuron is hyperactive for $t = 2245, \ldots, 2249$. In (b), most neurons are more active after $t = 4288$. In (c), the covariance pattern changes after $t = 4484$.

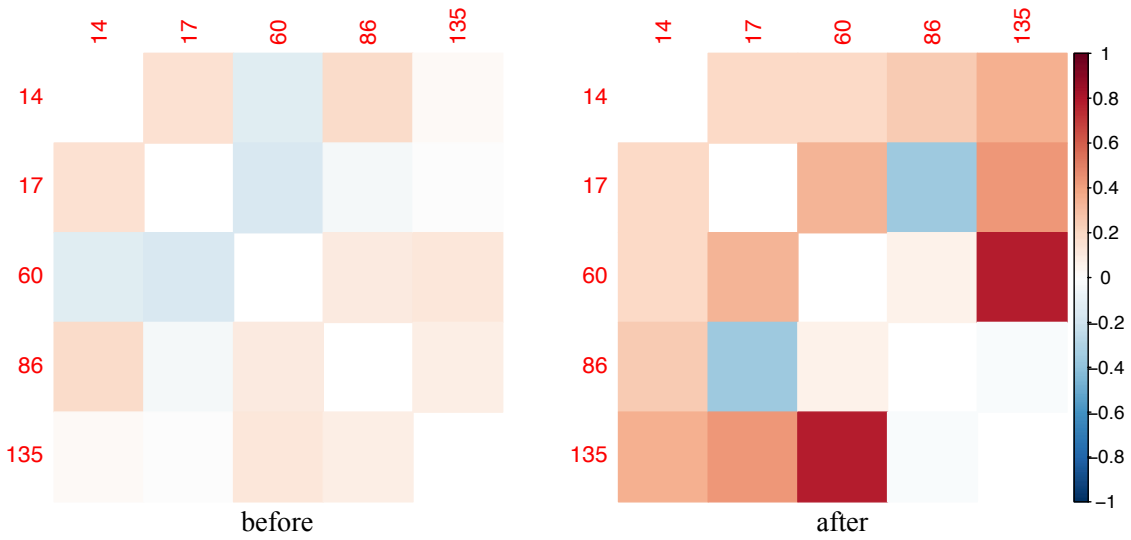

FIGURE 2.9. Correlation matrix of the five most active neurons before and after the change-point 4484. Indices of those neurons are also presented. The left panel shows the correlation matrix from $t = 4450$ to 4484, and the right panel shows that from $t = 4485$ to 4502.

CHAPTER 3

An Improved Framework Dealing with More Frequent Changes

3.1. Limitation on Generalized Edge-count Scan Statistics

Generalized edge-count scan statistics play a central role in the framework proposed in Chapter 2. It provide a powerful and reliable way to find change-point locally. However, generalized edge-count scan statistics have some drawbacks that could potentially harm the detection power and detection accuracy. In the following, we elaborate these limitations and explain the reason of causing them.

Generalized edge-count scan statistics could be written in the following form:

$$S^{[a,b]}(t) = \left[Z_w^{[a,b]}(t)\right]^2 + \left[Z_{\text{diff}}^{[a,b]}(t)\right]^2,$$

where $Z_w^{[a,b]}(t)$ is weighted edge-count scan statistic and $Z_{\text{diff}}^{[a,b]}(t)$ is difference edge-count scan statistic. Both two are graph-based edge-count scan statistics. $Z_w^{[a,b]}(t)$ is sensitive to mean changes and $Z_{\text{diff}}^{[a,b]}(t)$ is sensitive to covariance changes. In the following, we use $S(t)$, $Z_w(t)$, and $Z_{\text{diff}}(t)$ to represent these statistics when the detection interval is not specified. For more detail about these statistics, we refer to Chen et al. (2018), Chu & Chen (2019)

One of the most important steps in Algorithm 1 and Algorithm 2 is giving approximated p-value of generalized edge-count scan statistics on subintervals. The p-value approximation uses the method of Chu & Chen (2019). They proved that under permutation null distribution and some mild conditions, $Z_{\text{diff}}([nu])$ and $Z_w([nu])$ converge to independent Gaussian processes in finite dimensional distributions $\{Z_{\text{diff}}^*(u) : 0 < u < 1\}$ and $\{Z_w^*(u) : 0 < u < 1\}$, where $[x]$ is used to denote the largest integer that is no larger than x. Then, they approximate the tail probabilities by Woodroofe's method (Woodroofe 1976, 1978). However, as pointed out by Chu & Chen (2019), analytical approximations deviate if the minimum window length decreases because the convergence to normal process becomes slow. The skewness of weighted edge-count scan statistics $Z_w(t)$

and difference edge-count scan statistics $Z_{\text{diff}}(t)$ depend on the relative position of the change-point, affecting the analytic $p-$value approximation. This problem is even more severe under high dimension.

In this Chapter, we incorporate max-type edge-count scan statistics and max-type two-sample test statistics in the WBS and backward elimination framework. This generalization could further improve the detection accuracy when there are more frequent changes or the alternatives are mostly pure mean and covariance changes.

3.2. Max-type Edge-count Scan Statistics

Consider the task of detecting single change-point on $\{\mathbf{y}_i : a \leq i \leq b\}$ using max-type edge-count scan statistics. In the following, we use the same notation as in Chapter 2. The weighted and difference edge-count statistic are

$$R_w^{[a,b]}(t) = \frac{t-1}{n-2} R_1^{[a,b]}(t) + \frac{n-t-1}{n-2} R_2^{[a,b]}(t),$$
$$R_{\text{diff}}^{[a,b]}(t) = R_1^{[a,b]}(t) - R_2^{[a,b]}(t).$$

The standardized versions of the two are

$$Z_w^{[a,b]}(t) = \frac{R_w^{[a,b]}(t) - \mathbf{E}[R_w^{[a,b]}(t)]}{\sqrt{\mathbf{Var}[R_w^{[a,b]}(t)]}},$$
$$Z_{\text{diff}}^{[a,b]}(t) = \frac{R_{\text{diff}}^{[a,b]}(t) - \mathbf{E}[R_{\text{diff}}^{[a,b]}(t)]}{\sqrt{\mathbf{Var}[R_{\text{diff}}^{[a,b]}(t)]}}.$$

Max-type edge-count scan statistic is defined as:

$$\max_{n_{le}^{[a,b]} \leq t \leq n_{ri}^{[a,b]}} M^{[a,b]}(t), \tag{3.1}$$

where

$$M^{[a,b]}(t) = \max(|Z_{\text{diff}}^{[a,b]}(t)|, Z_w^{[a,b]}(t)). \tag{3.2}$$

We set $n_{le}^{[a,b]}$ and $n_{ri}^{[a,b]}$ as pre-specified endpoints, whose default settings are $\lceil a + 0.1(b-a+1) \rceil$ and $\lfloor b - 0.1(b-a+1) \rfloor$, where $\lceil \cdot \rceil$ is the ceiling function, and $\lfloor \cdot \rfloor$ is the floor function.

By design, $Z_w^{[a,b]}(t)$ is sensitive to mean vector change and $Z_{\text{diff}}^{[a,b]}(t)$ is sensitive to covariance matrix change especially under high dimension. Since $M(t)$ takes the maximum value of both $Z_w^{[a,b]}(t)$ and $Z_{\text{diff}}^{[a,b]}(t)$, it exhibits high sensitivity to both types of changes. An advantageous feature of graph-based edge-count scan statistics is their ability to effectively control the type-I error rate, making it a powerful tool for high-dimensional change-point detection (Chu & Chen 2019). In the following, we denote the p-value of $\max_{n_{le}^{[a,b]} \le t \le n_{ri}^{[a,b]}} M^{[a,b]}(t)$ by $\hat{p}_M^{[a,b]}$.

Max-type statistics can provide more accurate p-value approximation by skewness correction. Chu & Chen (2019) adopt a skewness correction approach similar to Chen & Zhang (2015). The degree of correction applied varies depending on the level of skewness at value of t. By incorporating skewness corrected $\hat{p}_M^{[a,b]}$, WBS algorithm could detect candidate change-points more accurately than generalized edge-count scan statistics.

3.3. Change-point Detection and Selection with Max-type Statistics

The new framework we proposed consists of a two-step procedure. First, we use a WBS with max-type edge-count scan statistics to search for candidate change-points. Subsequently, a goodness-of-fit statistic based on the maximum-type statistic is introduced. It is utilized for change-point selection.

Let α be the pre-specified significance level and MinLen be the minimum length of generated intervals, and L as the number of randomly generated intervals. The function M.WBS is defined in Algorithm 4.

M.WBS outputs a pool of candidate change-points $\tilde{\boldsymbol{\tau}} = \{\tilde{\tau}_1, \ldots, \tilde{\tau}_{\tilde{m}}\}$. To further improve the detection accuracy and understand their inner relationship, a new goodness-of-fit statistic using max-type statistics is desired. However, different from pseudo-BIC (2.5) and ep-BIC (2.6) that are justified by theoretical derivation, max-type two-sample test statistic does not follow chi-squared distribution due to its unique maximum structure.

We mimic extended pseudo BIC (2.6) and proposed a max-type version goodness-of-fit statistic called max-type extended pseudo BIC:

$$\text{mep-BIC}(\tilde{\boldsymbol{\tau}}) = \sum_{j=1}^{\tilde{m}} \left[M^{[\tilde{\tau}_{j-1}+1, \tilde{\tau}_{j+1}]}(\tilde{\tau}_j) \right]^2 - c\tilde{m} \log n. \tag{3.3}$$

Algorithm 4 Max-type graph-based WBS

function M.WBS(a, b, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
  if $b - a + 1 <$ MinLen **then**
    STOP
  end if
  if $L \geq (b - a - \text{MinLen} + 2)(b - a - \text{MinLen} + 3)/2$ **then**
    $L \leftarrow (b - a - \text{MinLen} + 2)(b - a - \text{MinLen} + 3)/2$
    Draw all intervals $[a_l, b_l] \subseteq [a, b]$, $l = 1, \ldots, L$, s.t. $b_l - a_l + 1 \geq \text{MinLen}$
  else
    Randomly draw intervals $[a_l, b_l] \subseteq [a, b]$, $l = 1, \ldots, L$, s.t. $b_l - a_l + 1 \geq \text{MinLen}$
    Add $[a_0, b_0] = [a, b]$ to the set of intervals
  end if
  $l' \leftarrow \text{argmin}_{l \in \{0, \ldots, l\}} \ \hat{p}_M^{[a_l, b_l]}$
  $\hat{t} \leftarrow \text{argmax}_{n_{le}^{[a_{l'}, b_{l'}]} \leq t \leq n_{ri}^{[a_{l'}, b_{l'}]}} \ M^{[a_{l'}, b_{l'}]}(t)$
  if $\hat{p}_M^{[a_l', b_l']} < \alpha$ **then**
    Add $\hat{t}$ to the set $\tilde{\boldsymbol{\tau}}$.
    M.WBS(a, $\hat{t}$, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
    M.WBS($\hat{t} + 1$, b, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
  else
    STOP
  end if
end function

For notation simplicity, we let $\tilde{\tau}_0 = 0$ and $\tilde{\tau}_{\tilde{m}+1} = n$. Similar to the searching step, max-type edge-count statistic is used. Each two-sample test statistic is squared in (3.3). Recall that $[Z_w^{[a,b]}(t)]^2$ and $[Z_{\text{diff}}^{[a,b]}(t)]^2$ are independently χ_1^2 distributed. By taking the square, the order of each squared max-type statistic in (3.3) is between χ_1^2 and χ_2^2. For this reason, mep-BIC share a similar intrinsic structure with ep-BIC. We determine the penalty term using a data driven method later. mep-BIC favors pure mean or covariance change-points than ep-BIC.

The mep-BIC achieves a favorable trade-off between mitigating overfitting and underfitting. In the situation of overfitting, mep-BIC($\tilde{\boldsymbol{\tau}}$) is relatively small under the effect of false change-points. Values of $M^{[\tilde{\tau}_{j-1}+1, \tilde{\tau}_{j+1}]}(\tilde{\tau}_j)$'s calculated on homogeneous subsequences are relatively small due to the nature of two-sample test statistics. If those falsely detected change-points are removed, $M^{[\tilde{\tau}_{j-1}+1, \tilde{\tau}_{j+1}]}(\tilde{\tau}_j)$'s calculated on the genuine change-points will increase by a significant margin, surpassing the value of the discarded ones. In the case of underfitting, a single $M^{[\tilde{\tau}_{j-1}+1, \tilde{\tau}_{j+1}]}(\tilde{\tau}_j)$ can cross multiple homogenous subsequences, leading to small statistic value.

We used a data-driven way to determine an appropriate choice of the penalty parameter c. Numbers of average true discoveries and false discoveries are reported under 4 different settings:

(1) $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, I)$ if $i \in [1, 30] \cup [61, 90] \cup [121, 150]$; $\mathbf{y}_i \sim \mathcal{N}_d(\frac{6}{5\log(d)}\mathbf{1}, I)$ if $i \in [31, 60] \cup [91, 120] \cup [151, 180]$.

(2) $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, I)$ if $i \in [1, 30] \cup [61, 90] \cup [121, 150]$; $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, (1 + \frac{7}{4\sqrt{d}})I)$ if $i \in [31, 60] \cup [91, 120] \cup [151, 180]$.

(3) $\mathbf{y}_i \sim t_{3,d}(\mathbf{0}, \Sigma)$ if $i \in [1, 50] \cup [101, 150] \cup [201, 250]$; $\mathbf{y}_i \sim t_{3,d}(\frac{7}{4\log(d)}\mathbf{1}, \Sigma)$ if $i \in [51, 100] \cup [151, 200] \cup [251, 300]$, where $\Sigma_{jk} = 0.5^{|j-k|}$

(4) $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, \Sigma)$ if $i \in [1, 40] \cup [81, 120] \cup [161, 200]$; $\mathbf{y}_i \sim \mathcal{N}_d(\frac{6}{5\log(d)}\mathbf{1}, (1 + \frac{3}{2\sqrt{d}})I)$ if $i \in [41, 80] \cup [121, 160] \cup [201, 240]$, where $\Sigma_{jk} = 0.5^{|j-k|}$.

Any $\tilde{\tau}_j$ that is within a range of two observations from a true change-point is considered a true discovery, while those outside of that range is considered a false discovery. The result is shown in Figure 3.1. It is conceivable that power decreases dramatically after $c = 2$ to 4, while false positive rate decreases steadily over four different settings. We also notice that the power is relatively stable when c is small, showing the robustness of the method. Taking into account the above points, $c = 2$ is adopted as the default choice for mep-BIC.

Next, we employ a backward elimination algorithm with mep-BIC to conduct model selection (Algorithm 5). In each step of M.BE, one candidate change-point is removed until there is J

Algorithm 5 Backward elimination with mep-BIC

procedure M.BE($\tilde{\boldsymbol{\tau}}$, J)
  $l \leftarrow \tilde{m}$
  $\tilde{\boldsymbol{\tau}}^l \leftarrow \tilde{\boldsymbol{\tau}}$
  while $|\tilde{\boldsymbol{\tau}}^l| \geq J$ **do**
    $\mathbf{T}^l :=$ collection of change-points set $\tilde{\boldsymbol{\tau}}^l \backslash \{\tilde{\tau}_j^l\}$, where $\tilde{\tau}_j^l \in \tilde{\boldsymbol{\tau}}^l$, $j = 1, \ldots, l$
    $\tilde{\boldsymbol{\tau}}^{l-1} \leftarrow \text{argmax}_{\boldsymbol{t} \in \mathbf{T}^l}$mep-BIC($\boldsymbol{t}$)
    $l \leftarrow l - 1$
  end while
  $\hat{m} \leftarrow \text{argmax}_l$mep-BIC($\tilde{\boldsymbol{\tau}}^l$)
  $\hat{\boldsymbol{\tau}} \leftarrow \tilde{\boldsymbol{\tau}}^{\hat{m}}$
  return $\hat{\boldsymbol{\tau}}$
end procedure

change-points left. The final estimated change-points $\hat{\boldsymbol{\tau}}$ are those give the largest mep-BIC value.

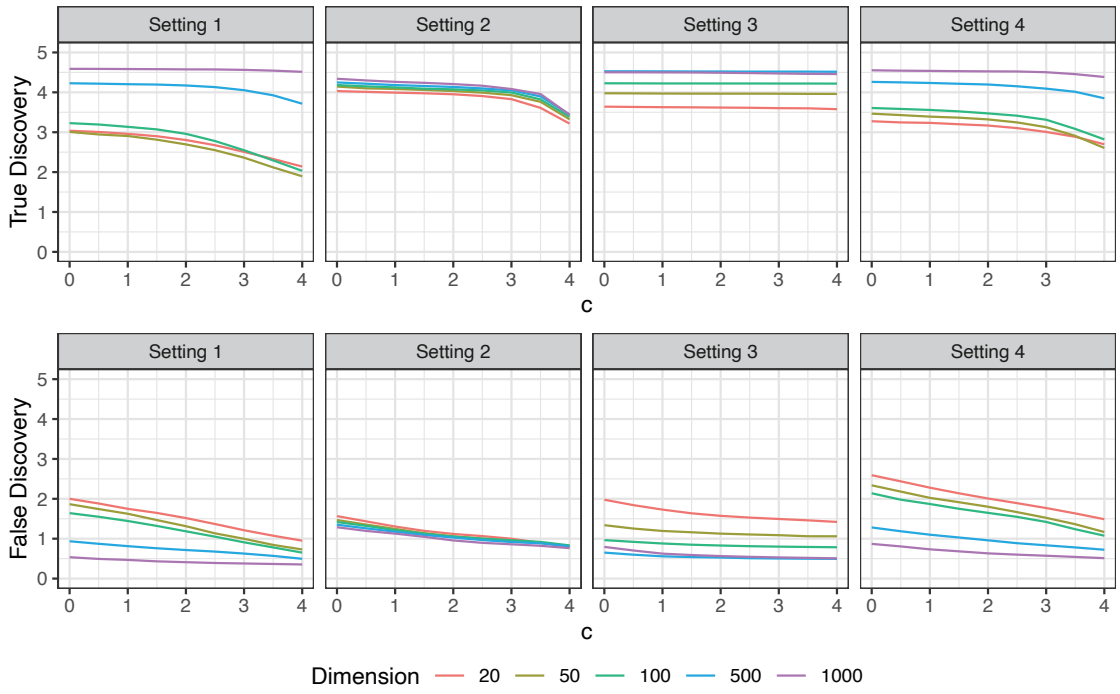

FIGURE 3.1. Average number of true and false discoveries over different penalty choice c under four settings.

Ideally, J should be set to 1 to enlarge the searching space. But if prior information is available, J can be set to a larger number to save computational resources. Change-point dendrogram can serve as a tool to understand the hierarchical structure of homogeneous subsequences. In the process of M.BE, neighboring change-points are merged. In combination with mep-BIC, these tools are ideal ingredients for constructing a dendrogram when $J = 1$. In a dendrogram, change-points or subsequences close to the root are considered more important in maintaining the high-level structure of the data. Researchers may segment the dendrogram by selecting an appropriate height, resulting in a set of change-points at any desired resolution.

3.4. Numerical Studies

In this section we use simulated data to assess the performance of proposed method. Specifically, in each simulation run, we apply M.WBS to the generated data and pass the first-step result $\tilde{\boldsymbol{\tau}}$ to M.BE and get the final estimated result $\hat{\boldsymbol{\tau}}$. We have configured the algorithm by setting the following parameters: $\alpha = 0.01$ to regulate the local significance level, $L = 100$ to provide adequate

coverage of the data, MinLen = 10 to ensure detection of frequent change-points, and $J = 1$ to enlarge searching space. Through the incorporation of various distributions and change-point structures, it is anticipated that the efficacy of the proposed methodology in handling complex data will be demonstrated. Let $\boldsymbol{\theta}$ be a $d-$length vector with the first $d/5$ entries equal to 1 and all others equal to 0, and $\boldsymbol{\theta_1}$ be a $d-$length vector with $d-1$ entries equal to 0 and 1 entry equal to 1. The proposed method is tested under the following four settings:

(5) $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, I)$ if $(i \bmod 50) \in [1, 15] \cup [21, 35]$; $\mathbf{y}_i \sim \mathcal{N}_d(\delta\boldsymbol{\theta}, \Sigma)$ if $(i \bmod 50) \in [16, 20]$; $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, \sigma^2\Sigma)$ if $(i \bmod 50) \in [36, 49] \cup \{0\}$ where $i = 1, \ldots, 500$, and $\Sigma_{jk} = 0.2^{|j-k|}$.

(6) $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, \Sigma)$ if $(i \bmod 45) \in [1, 40]$; $\mathbf{y}_i \sim \mathcal{N}_d(\delta\boldsymbol{\theta_1}, \Sigma)$ if $(i \bmod 45) \in [41, 44] \cup \{0\}$ where $i = 1, \ldots, 450$, and $\Sigma_{jk} = 0.2^{|j-k|}$.

(7) $\mathbf{y}_i \sim t_{1,d}(\mathbf{0}, I)$ if $i \in [1, 40] \cup [91, 145] \cup [191, 255] \cup [300, 340] \cup [391, 445] \cup [491, 555]$; $\mathbf{y}_i \sim t_{1,d}(\delta\boldsymbol{\theta}, \Sigma)$ otherwise, where $i = 1, \ldots, 600$, and $\Sigma_{jk} = 0.3^{|j-k|}$.

The number of true discoveries and false discoveries are reported in Table 3.1 and 3.2. If a detected change-point is within 1 observations of a true change-point in Setting 5 and 6, and 2 observations in Setting 7, then it is counted as a true discovery. Otherwise it is counted as a false discovery. In addition, we compare the new framework with the method proposed in Chapter 2, which is used as a baseline to better understand the new framework.

TABLE 3.1. Number of detected true change-points and corresponding standard deviations (in parenthesis) with 100 replications. The largest value under each setting is in bold.

Setting 5	d	20	50	100	500	1000
	δ	2.20	1.75	1.40	0.90	0.75
	σ	4.55	4.24	3.93	3.61	3.52
	New	**35.34** (3.39)	**36.17** (3.40)	**35.65** (4.26)	**35.60** (4.09)	**35.02** (5.66)
	Chapter 2	34.12 (3.67)	35.24 (3.29)	35.31 (3.92)	34.84 (4.13)	34.27 (5.57)
Setting 6	d	20	50	100	500	1000
	δ	3.85	4.55	5.95	8.05	10.50
	New	13.01 (4.25)	**12.41** (3.94)	**15.82** (2.81)	**12.80** (4.35)	**16.71** (2.60)
	Chapter 2	**13.47** (4.68)	12.02 (4.66)	15.35 (4.29)	12.08 (4.97)	15.97 (4.74)
Setting 7	d	20	50	100	500	1000
	δ	1.1	0.85	0.76	0.64	0.60
	New	**7.44** (1.58)	**7.32** (1.58)	**7.45** (1.78)	**8.08** (1.61)	**7.78** (1.56)
	Chapter 2	7.38 (1.74)	7.10 (2.02)	6.95 (1.92)	7.72 (1.76)	7.49 (1.61)

TABLE 3.2. Number of detected false change-points and corresponding standard deviations (in parenthesis) with 100 replications. The smallest value under each setting is in bold.

Setting 5	d	20	50	100	500	1000
	δ	2.20	1.75	1.40	0.90	0.75
	σ	4.55	4.24	3.93	3.61	3.52
	New	**1.16** (1.13)	**0.59** (0.73)	**0.77** (0.79)	**0.61** (0.72)	**0.49** (0.70)
	Chapter 2	2.46 (1.75)	1.19 (1.23)	1.12 (1.12)	1.01 (1.05)	0.91 (1.04)
Setting 6	d	20	50	100	500	1000
	δ	3.85	4.55	5.95	8.05	10.50
	New	**1.60** (1.51)	**1.42** (1.36)	**1.00** (1.15)	**0.97** (1.14)	**0.66** (0.92)
	Chapter 2	1.83 (1.80)	1.56 (1.47)	1.04 (1.17)	1.00 (1.05)	0.75 (0.91)
Setting 7	d	20	50	100	500	1000
	δ	1.1	0.85	0.76	0.64	0.60
	New	**3.39** (1.64)	**3.15** (1.49)	**3.34** (1.77)	**2.46** (1.51)	**2.30** (1.30)
	Chapter 2	3.86 (1.92)	3.28 (1.62)	3.63 (1.92)	2.90 (1.72)	2.57 (1.52)

It is conceivable that the new framework outperforms in terms of both power and false discovery under most settings. Setting 5 mimics two scenarios. First, the case where a small portion of dimensions has a mean change during a short time interval. Second, the case where the covariance matrix changes in a short interval. In Setting 6, the mean vector is even sparser. In Setting 7, we test its performance in pure mean changes under heavy tail distributions. The new framework shows better performance in both power and false discovery rate. This can be explained by the skewness correction of max-type scan statistics. When the scan interval is short, max-type scan statistics have more accurate estimated p-values, leading to better performance.

CHAPTER 4

A Parallel Computation Approach and an Application to Neuropixels Data

4.1. Introduction

Nowadays, electrophysiological methods are widely adopted in neuroscience to reveal the dynamics of neural processing across time scales (Chen et al. 2017). The key to comprehending how the brain represents, transforms, and communicates information is found in high-resolution neural recordings from scattered nodes of the brain network (Lewis et al. 2015). Starting from using insulated metal microelectrodes with single recording sites in 1950s, electrophysiological techniques to record neuronal activity in vivo has been dramatically improved by the use of CMOS fabrication (Steinmetz et al. 2021, 2018). Modern electrophysiological tools can allow accurate recordings with single neuron spatial precision and single spike resolution. In addition, they make population recording possible, allowing high-quality recordings of a large group of neurons distributed across different brain regions over long time scale.

Neuropixels is a CMOS-based silicon probe developed by Jun et al. (2017). There are 384 recording channels in each probe, each of which can be programmed to address 960 complimentary CMOS sites. Neuropixels were able to produce isolated spiking signals from hundreds of neurons in small rodents by densely sampling the signals (Gardner et al. 2019, Sauerbrei et al. 2020). Using multiple shanks, Neuropixels can record spikes from hundreds or even thousands of neurons across multiple brain regions in vivo (Stringer et al. 2019).

The vast data produced by Neuropixels present a challenge to statistical analysis. A typical dataset records the activity of hundreds of individual neurons over several minutes to hours. The data acquisition rate for each probe is around 1 gigabyte per minutes. Moreover, neural signals are generally noisy and non-stationary. The patterns of neural activity exhibit high temporal variability, with frequent changes occurring over time. All these factors makes modeling Neuropixels data

parametrically unfeasible theoretically and computationally. Dividing long Neuropixels sequences into stationary subsequences can be served as a preliminary procedure for statistical analysis. The division procedure is commonly referred to as change-point detection. Change-point detection contributes to the Neuropixels data analysis from two perspectives. First, detected change-points themselves can be considered valuable targets for research, as they could indicates a sudden change in neural activity pattern. Second, detected change-points split the data into homogeneous subsequences, which opens up possibilities for future statistical modeling.

However, this field has received little attention. The work of Chen et al. (2019) is one of the few attempts to address this problem. It involves finding an initial change-point set by graph-based statistics (Chu & Chen 2019) and binary segmentation (Vostrikova 1981), followed by refining the results through a three-step iterative process. Though graph-based statistics is suitable for such high-dimensional noisy data, the use of binary segmentation and a lengthy revision procedure can cause severe power loss. Another graph-based method proposed by Zhang & Chen (2021) combined generalized edge-count statistics and Wild Binary Segmentation (WBS) (Fryzlewicz 2014). This WBS-style method generates hundreds of random intervals along the sequence, and scan for change-points on each interval. The most significant one is chosen as the first change-point. Then the process recursed until no new change-point being found. After that, they prune candidate change-points using a goodness-of-fit statistic. Although their method shows satisfactory power and false discovery rate, it is only practical for analyzing short data.

In this work, we propose a comprehensive framework to detect and understand change-point in Neuropixels data. The new framework represents a significant improvement over the previous method: G.WBS and G.BE proposed in Chapter 2. We innovatively parallelized WBS algorithm on graph-based scan statistics, greatly reducing its computation time and overcoming the limitation of being used only on smaller data. In addition, we leveraged max-type edge-count scan statistics (Chu & Chen 2019) to further improve its detection accuracy. In the pruning part, mep-BIC built on max-type two-sample test statistic is used. To understand the structure of detected change-points, we utilized change-point dendrogram to represent the relative importance of change-points.

The organization of this Chapter is as follows. Some background information in neuroscience is described in Section 4.2. The details of the framework are illustrated in Section 4.3. In Section 4.4, the new framework is applied to a Neuropixels data collected from an alive mouse.

4.2. More Background in Neuroscience

For neurons, the communication of information between them relies heavily on action potentials, also known as spikes. For most neurons, a typical signaling process begins with the generation of an action potential by the neuron, which conducts a bioelectrical signal to the neuron terminal. The neurosynapse at the neuron terminal receives the electrical signal and releases neurotransmitters through the vesicle, thus converting the electrical signal into a chemical signal and transmitting it to neighboring neurons, completing the transmission of information. The mammalian cerebral cortex contains millions or even billions of neurons. The cerebral cortex can be divided into several regions, and different regions are responsible for different functions and coordinate with each other to accomplish a variety of activities. Exploring the relationship between neuronal activity in the brain and external stimuli and spontaneous activity has been one of the enduring research topics in the field of neuroscience.

Neuropixels is one of the state-of-the-art probe platform allowing recording neuronal activity of large neuronal population across multiple brain regions (Lewis et al. 2015). The application of Neuropixel has been extended to a diverse range of species, encompassing mice (Bennett et al. 2019, Evans et al. 2018, Park et al. 2022, Stringer et al. 2019), rats (Krupic et al. 2018), and ferrets (Gaucher et al. 2020). It is a custom implementation of a 200mm wafer scale 130nm CMOS silicon on insulator technology with aluminum back-end of line (Dutta et al. 2019). Before record extracellularly from neuronal population, the head of the experimental animal are usually fixed to reduce the drift of probes. Then multiple probes are slowly placed into the brain of the experimental animal and keep recording for hours. Once the recording is complete, the data needs to be processed through spike sorting and probe localization. Spike sorting aims to identifying single spike from the recording and attributes it to individual neuron. There are many mature algorithms could finish the process effectively, including KiloSort and MountainSort (Chung et al. 2017, Pachitariu et al. 2016). The regions of the brain where neurons were recorded can be identified during the process

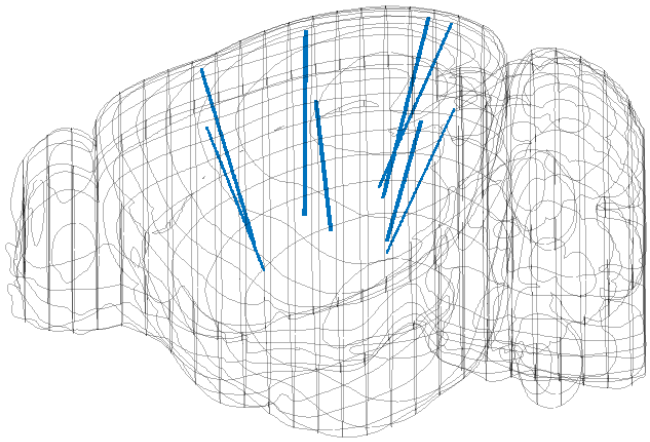

FIGURE 4.1. Neuropixels probe locations of recordings in a mouse (Stringer et al. 2019).

of probe localization. Once can either identify each structure along the recording track or using 3D atlases of the mouse brain (Johnson et al. 2010).

4.3. A Parallel Graph-based Multiple Change-point Analysis Framework

The new framework we proposed consists of a two-step procedure. First, we use a parallel WBS with max-type edge-count scan statistics to search for candidate change-points. Subsequently, mep-BIC is introduced for performing change-point selection. Figure 4.2 shows the pipeline of analysis. The first three steps include preprocessing of Neuropixels data. More information can be found in Section 4.2. Our new framework focuses on the last three steps related to change-point analysis.

Define K as the number of folds used to separate the sequence, α as the pre-specified significance level and MinLen be the minimum length of generated intervals, and L as the number of randomly generated intervals. The functions PARAGRAPHWBS is defined in Algorithm 6.

The searching procedure begins with evenly dividing the whole sequence into K folds. Usually we choose K such that each fold contains a hundreds of observations. Next, for each fold, apply M.WBS onto it. In each fold, M.WBS randomly generates L intervals and use max-type edge count scan statistics to search for candidate change-points. Among those L change-points, we keep the most significant one if its p-value is smaller than the significance level α. After that, most change-points should be found by M.WBS. Some change-points close to the boundaries of folds might be

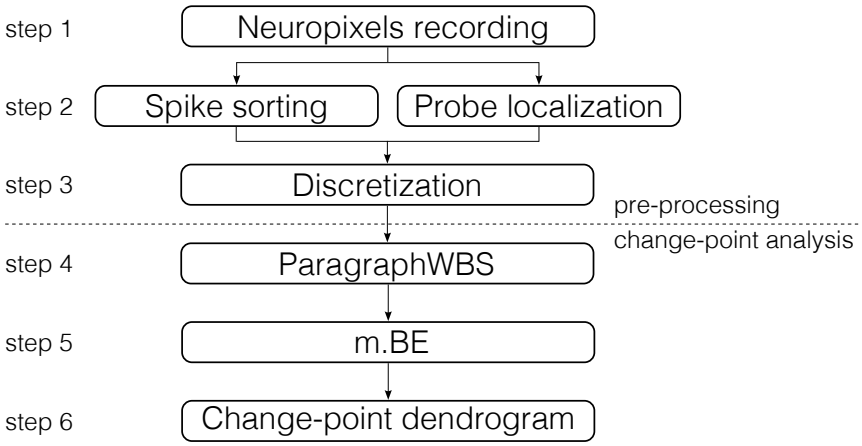

FIGURE 4.2. Process of analyzing Neuropixels data from data collection to change-point analysis.

Algorithm 6 parallel Change-point search by max-type graph-based statistic

procedure PARAGRAPHWBS(K, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
  for each fold $k = 1, \ldots, K$ **do in parallel**
    M.WBS($\lfloor (k-1)n/K \rfloor + 1$, $\lceil kn/K \rceil$, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
  end for
  for each fold $k = 1, \ldots, K-1$ **do**
    $a_k \leftarrow \text{argmin}_{\tilde{\tau}_j \in \tilde{\boldsymbol{\tau}}, \tilde{\tau}_j < \lceil nk/K \rceil} (\lceil nk/K \rceil - \tilde{\tau}_j)$
    $b_k \leftarrow \text{argmin}_{\tilde{\tau}_j \in \tilde{\boldsymbol{\tau}}, \tilde{\tau}_j > \lceil nk/K \rceil} (\tilde{\tau}_j - \lceil nk/K \rceil)$
    M.WBS($a_k + 1$, b_k, $\tilde{\boldsymbol{\tau}}$, α, L, MinLen)
  end for
end procedure

missed. So in the second loop, PARAGRAPHWBS search near each boundary of folds, making sure all parts of the data are scanned. After PARAGRAPHWBS gives a pool of change-points, M.BE can be used to further select the change-points and explore the relationship between them.

By dividing a long sequence into several short subsequences, PARAGRAPHWBS achieves high speed, high efficiency and high accuracy. The time complexity of edge-count statistics is mostly determined by finding distance matrix and similarity graph, which are $O((b-a)^2)$ and $O(k(b-a)^2 \log(b-a))$ with Prim's Algorithm. Shortening the scan region of M.WBS to 1/10 or even 1/100 of its original length will greatly speed up the computation. In addition, this strategy can make the loop easily parallelizable, further accelerate the algorithm. The idea behind WBS type algorithm is to scan over single change-point by generating a large number of random intervals. In

Neuropixels data, change-points are densely distributed (Zhang & Chen 2021). Scanning over long sequences is time-inefficient. Meanwhile, estimated change-points and their corresponding p-values are questionable as signal can be hidden and overwhelmed by overlong sequences with multiple change-points. In contrast, introducing folds into ParagraphWBS increased the probability that generated intervals cover only single change-point. This enhancement leads to improved efficiency and estimation accuracy of scan statistics.

Compared with the original g.WBS, the new algorithm uses max-type scan statistics in place of generalized edge-count scan statistics. One of prominent features of Neuropixels data is densely distributed change-points. Neighboring change-points are sometimes only a few observations apart; thus, scan statistics must have enough power when scanning over short intervals. The primary change-points detected in Neuropixels data pertain to mean vector change, with a small portion attributed to covariance pattern change (Zhang & Chen 2021). Max-type edge-count scan statistics are more sensitive to pure mean vector or covariance matrix changes than generalized edge-count scan statistics. Max-type edge-count scan statistic confers an additional benefit of accurate $p-$value estimation with skewness correction, which is especially advantageous in the context of scanning over short intervals. All these benefits makes max-type edge-count scan statistic a better choice for Neuropixels data analysis. On the other hand, if generalized edge-count scan statistic is preferred in the real data analysis for specific reasons, ParagraphWBS can be readily adapted to utilize generalized statistics.

4.4. Change-point Analysis of Spontaneous Neural Activity in Mice Using Neuropixels Recordings

Sensory cortex refers to the part of the cerebral cortex that process sensory information including visual cortex, auditory cortex, olfactory cortex, and more. The neurons in these cortices exhibit significant activity when receiving external stimuli. Even without external stimuli, the brain generates structured patterns of activities. The spontaneous activities have been inferred to be associated with recapitulation of sensory experience, behavioral and cognitive states, and ongoing behavior (Berkes et al. 2011, Schneider et al. 2014, Stringer et al. 2019). To study spontaneous activities in population level, we use Neuropixels recording in the brain of a mouse

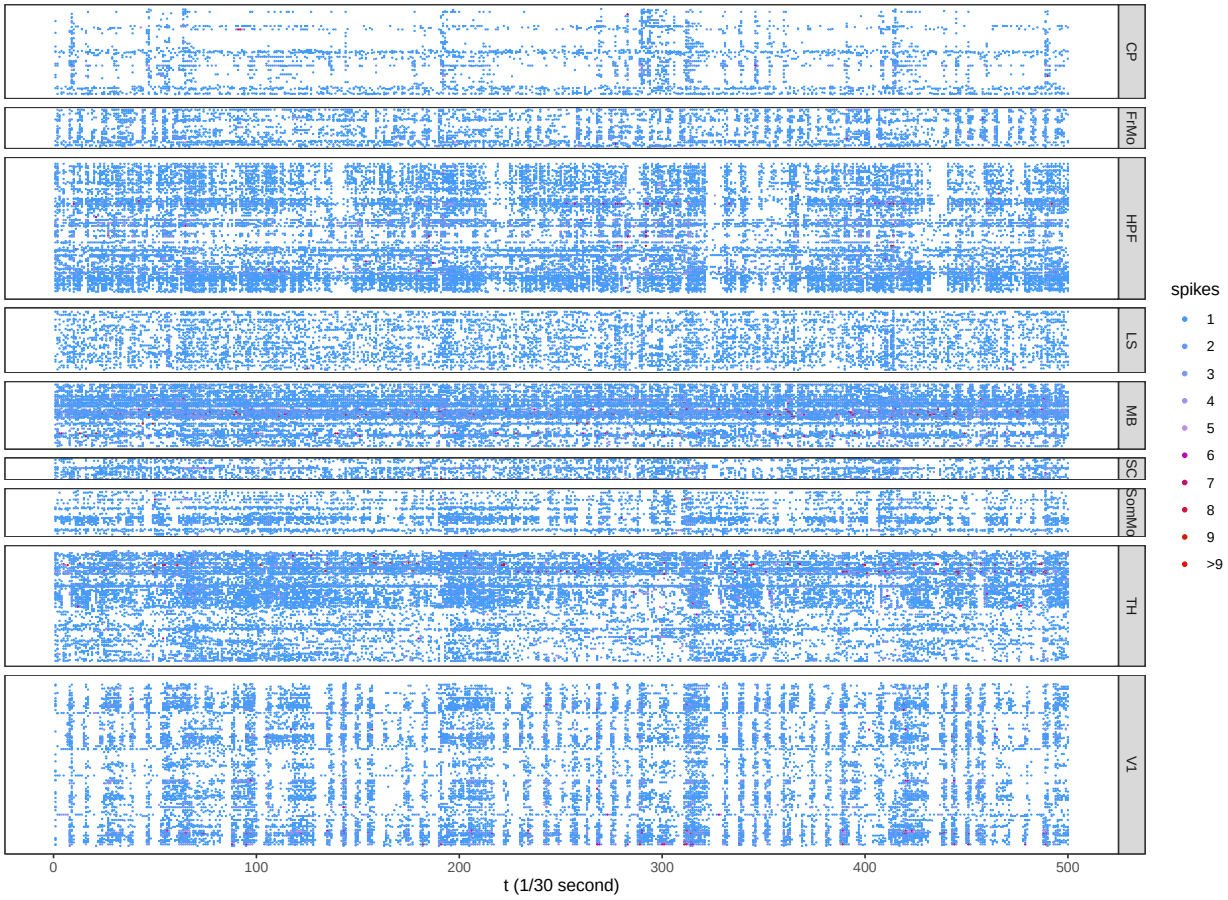

FIGURE 4.3. First 500 observations of Neuropixels data in nine brain cortical areas.

(`https://janelia.figshare.com/articles/dataset/Eight-probe_Neuropixels_recordings_during_spontaneous_behaviors/7739750`). The original data used eight Neuropixels probes to record activity across different cortex. During the data acquisition process, the mouse were awake and free to rotate a wheel. After spike sorting and probe localization, 1462 units were used from 9 different cortical areas, including Caudate putamen (CP, 176 units), Frontal motor (FrMo, 78 units), Hippocampus (HPF, 265 units), Lateral Septum (LS, 122 units), Midbrarin (MB, 127 units), Superior colliculus (SC, 42 units), Somatomotor (SomMo, 91 units), Thalamus (TH, 227 units), and V1 (334 units). Subsequently, the recordings were discretized into 1/30-second intervals, and to enhance data quality, the initial 1,000 and last 1,053 observations were excluded from the analysis. The cleaned dataset have $n = 37,000$ observations, and $y_{i,j}$ denotes the number of spikes recorded for neuron j during time interval i. A snippet of the data is shown in Figure 4.3.

TABLE 4.1. Number of change-points returned by PARAGRAPHWBS and M.BE in each step.

Region	CP	FrMo	HPF	LS	MB	SC	SomMo	TH	V1
Step 1	1490	1499	1598	919	1274	1135	1717	1645	1998
Step 2	636	335	495	235	206	362	299	504	611

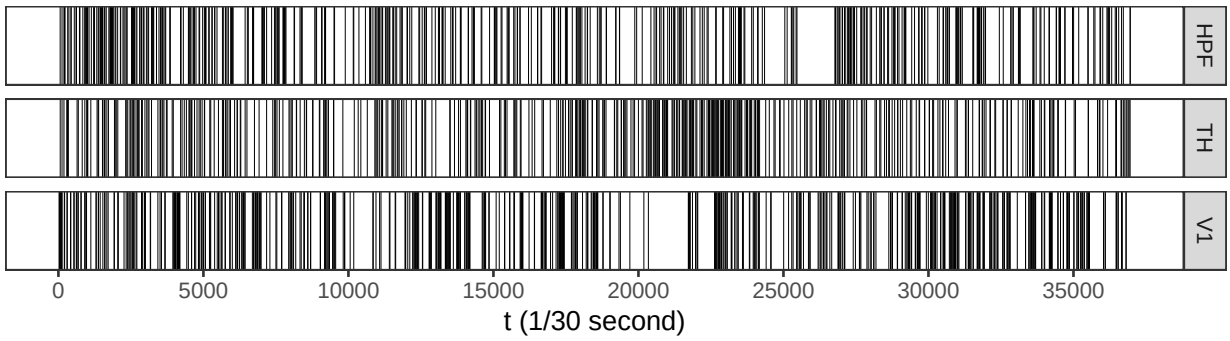

FIGURE 4.4. Detected change-points $\hat{\boldsymbol{\tau}}$ in three cortical area with most units. Each vertical line represents a detected change-point.

Using PARAGRAPHWBS and M.BE, we analyze the change-point structure of the mouse Neuropixels data. To be specific, we first applied PARAGRAPHWBS to the data in each cortical area, with $K = 100$, $\alpha = 0.0001$, $L = 200$, MinLen=10, and 20 CPU cores. On average, it takes only 10 minutes to scan each cortical area. PARAGRAPHWBS detected thousands of change-points, offering a high level of granularity for Neuropixels analysis. Often the case, researchers are interested in understanding the high-level structure of the data. We pass each $\tilde{\boldsymbol{\tau}}$ to M.BE with $J = 200$ and penalty parameter $c = 2$. The average time consumed for each area is 34 minutes. The numbers of detected and selected change-points in two steps are listed in Table 4.1. Detected change-points $\hat{\boldsymbol{\tau}}$ of three cortical areas with most units are plotted in Figure 4.4. If a coarser granularity is needed, J can be set to 1 to get the full hierarchical structure of change-points, which requires more computational time. Another choice is to continuously track the value of mep-BIC($\tilde{\boldsymbol{\tau}}$) as the algorithm progresses. One can manually stop M.BE if mep-BIC($\tilde{\boldsymbol{\tau}}$) keep decreasing for a long period of time. Using V1 as an instance, we set $J = 1$ to study its full hierarchical structure. The full trajectory of mep-BIC and a truncated change-point dendrogram with 30 change-points are shown in Figure 4.5 (A) and (D). One may choose an appropriate resolution of change-points based on the trajectory of mep-BIC. Except for choosing the global maximum of mep-BIC, some local maximum with a

few change-points are also good candidates to choose from. A relatively high mep-BIC can still be reached with a few change-points, showing their importance in maintaining the high-level change-point structure of the data. In addition to the dendrogram, the order in which change-poins are removed in M.BE reflects their importance in the data. The last two change-points removed from the data are 20,343 and 28,121. The two change-points are plotted in observation-wise mean and standard deviation plot (Figure 4.5 (B) and (C)). We can clearly see the data become more stable during that subsequence.

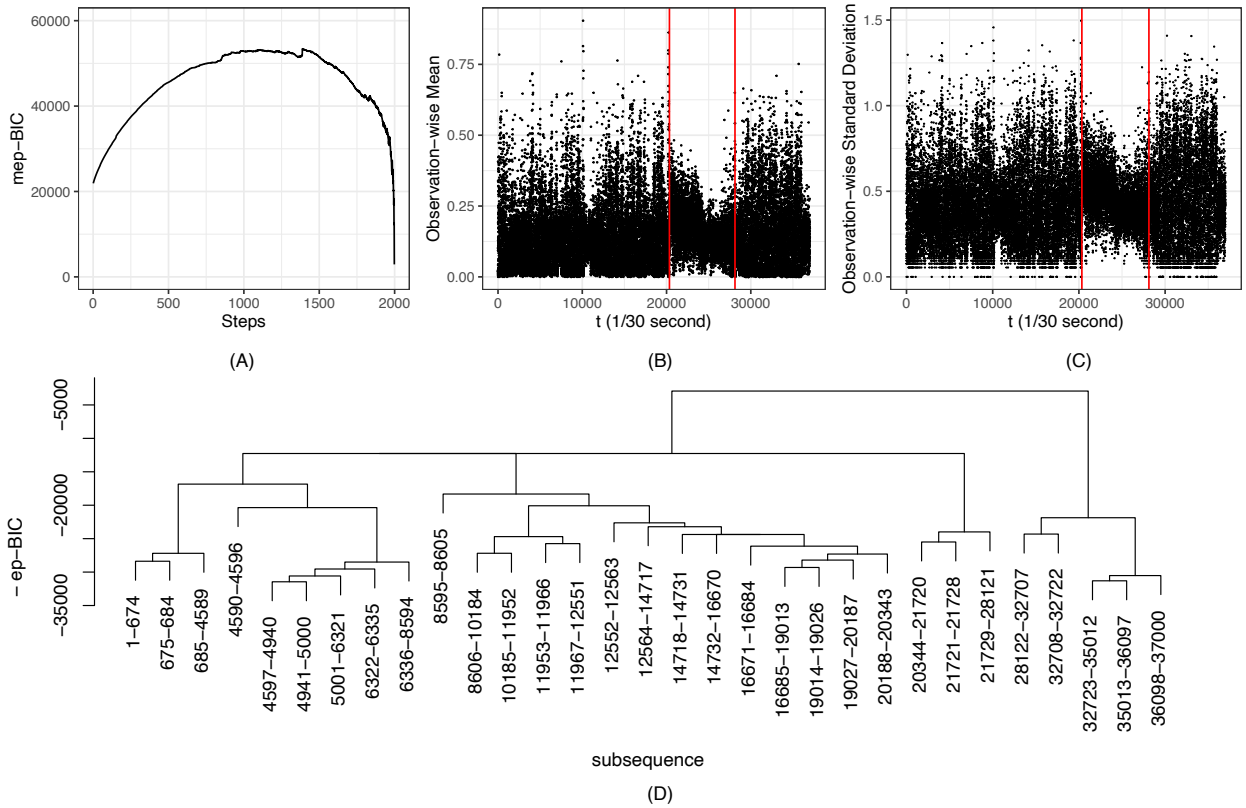

FIGURE 4.5. (A): Value of mep-BIC during the process of M.BE in the cortical area V1. (B, C): Mean and standard deviation along each observation of the Neuropixels recording of cortical area V1. The last two change-points to be removed in M.BE (20,343 and 28,121) are marked with red vertical lines. (D): Change-point dendrogram of V1 with the last 30 change-points to be removed in M.BE.

The use of max-type statistics in the framework shows its advantage in detecting more frequent changes over generalized statistics. We applied PARAGRAPHWBS again with G.WBS and G.BE to the Neuropixels data, and compared their respective results. We observed that several important

change-points identified by the max-type-based method are not found using the generalized-based method. Change-points $t = 9,338$ and 9,343 in the LS region exemplify this (Figure 4.6 (A)). During this short time interval, a single neuron becomes highly active. Also, we found that the majority of neurons exhibit activity at $t = 9,335$, but rapidly become inactive between $t = 9,336$ and $t = 9,337$. This intriguing observation is only discerned by the max-type-based method. Another example is change-point $t = 19,488$ in the region TH (Figure 4.6 (B)). The boxplot of observation-wise sum before and after the detected change-point is shown in Figure 4.6 (C). It could be observed in both plots that neurons are more active after the change-point.

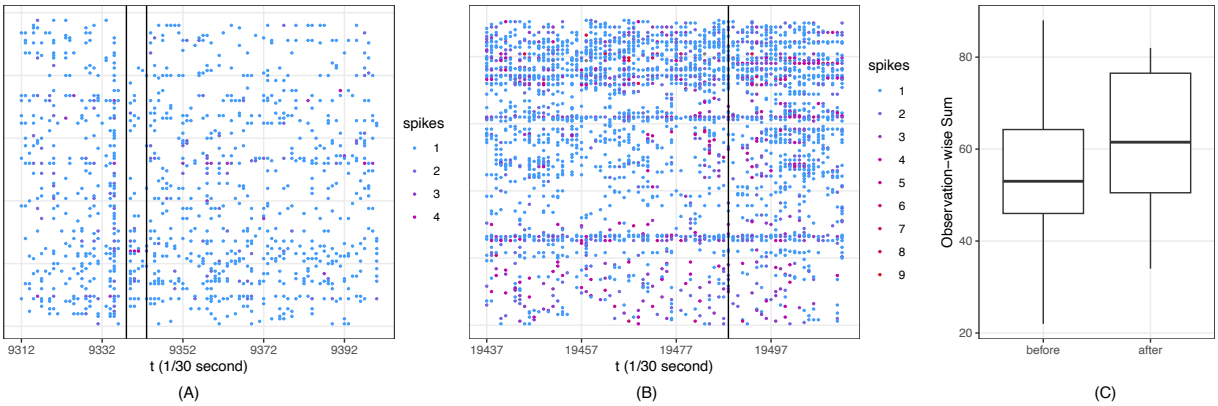

FIGURE 4.6. (A): A snippet of Neuropixels recording in the region LS. Two detected change-points 9,338 and 9,343 are marked. (B): A snippet of Neuropixels recording in the region TH. The detected change-point 19,488 is marked. (C): Boxplot of sum of spikes along each observation of the Neuropixels recording of cortical area LS before and after the change-point $t = 19,488$.

CHAPTER 5

Conclusion

In this dissertation, we propose a graph-based framework for multiple change-point detection for high-dimensional and non-Euclidean data. In Chapter 2, we build a two step method using generalized edge-count scan statistics and greedy algorithms. In addition, to prune candidate change-points, a new goodness-of-fit statistics is used with change-point dendrogram. Then, we incorporate max-type edge-count statistics in Chapter 3 to further improve its power under frequent changes scenario. Finally, a parallel computation approach is utilized to analyze long series in Chapter 4 with special attention to Neuropixels data.

To conclude this article, we would like to discuss some interesting topics for future research. When there are some prior information, it could be that some other graph-based methods are more suitable. For example, the weighted edge-count test would be preferred if one is only interested in location alternatives. The arguments in this work can be extended to the weighted edge-count test (Chen et al. 2018). Let $Z_w^{[a,b]}(t)$ be the weighted edge-count scan statistic for $a \leq i \leq b$. Given the fact that $Z_w^{[a,b]}(t)^2 \xrightarrow{d} \chi_1^2$ under some regularity conditions, the corresponding expanded pseudo-BIC may be defined as $\sum_{j=1}^{\tilde{m}} Z_w^{[\tilde{\tau}_{j-1}+1,\tilde{\tau}_{j+1}]}(\tilde{\tau}_j)^2 - \tilde{m}\log n$. These goodness-of-fit statistics and detection algorithm may further be generalized to other nonparametric statistics. How to generalize them to other statistics in a uniform framework is our next goal. Constructing distance matrix and MST demands a large amount of memory resources. Further reducing both memory and CPU usage is our next topic. Lastly, we plan to generalize the framework to kernel-based statistics to bring more flexibility to researchers.

APPENDIX A

Appendix for Chapter 2

A.1. Proof of Theorem 2.2.1

Define

$$
\begin{aligned}
T(u) &:= \lim_{n\to\infty} \frac{S^{[1,n]}(nu)}{n}, \\
\hat{u} &:= \operatorname*{argmax}_{u\in\left[\frac{n_{le}^{[1,n]}}{n}, \frac{n_{ri}^{[1,n]}}{n}\right]} \frac{S^{[1,n]}(nu)}{n}, \\
\bar{\boldsymbol{\omega}} &:= \operatorname*{argmax}_{u\in\left[\frac{n_{le}^{[1,n]}}{n}, \frac{n_{ri}^{[1,n]}}{n}\right]} T(u), \\
T^{[a_l,b_l]}(u) &:= \lim_{n\to\infty} \frac{S^{[a_l,b_l]}(nu)}{b_l - a_l}, \\
\hat{u}^{[a_l,b_l]} &:= \operatorname*{argmax}_{u\in\left[\frac{n_{le}^{[a_l,n]}}{n}, \frac{n_{ri}^{[a_l,b_l]}}{n}\right]} \frac{S^{[a_l,b_l]}(nu)}{b_l - a_l}, \\
\bar{\boldsymbol{\omega}}^{[a_l,b_l]} &:= \operatorname*{argmax}_{u\in\left[\frac{n_{le}^{[a_l,b_l]}}{n}, \frac{n_{ri}^{[a_l,b_l]}}{n}\right]} T^{[a_l,b_l]}(u).
\end{aligned}
$$

Recall that $\boldsymbol{\omega} := \{\omega_1, \ldots, \omega_m\}$, and $\widetilde{\boldsymbol{\omega}} := \{\tilde{\tau}_1/n, \tilde{\tau}_2/n, \ldots\}$. Then define the proportion p_j for each multivariate distribution $f_j(x)$ between ω_j and ω_{j+1} as $p_j = \omega_{j+1} - \omega_j$.

Let $T_j(\Delta) = \lim_{n\to\infty} \frac{S^{[1,n]}((\omega_{j+1}-\Delta)n)}{n}$, where

$$
\Delta \in \begin{cases} [0, p_0) & \text{when } j = 0, \\ [0, p_j] & \text{when } 1 \le j \le m-1, \\ (0, p_m] & \text{when } j = m. \end{cases}
$$

Follow Chen & Friedman (2017), Henze & Penrose (1999), for k-MST, we have

$$T_j(\Delta) = \frac{[\delta_{1,j}(\Delta) - \delta_{2,j}(\Delta)]^2}{(\omega_{j+1} - \Delta)(1 - \omega_{j+1} + \Delta)\mathrm{Var}(D_{d,k})} + \frac{[(1 - \omega_{j+1} + \Delta)\delta_{1,j}(\Delta) + (\omega_{j+1} - \Delta)\delta_{2,j}(\Delta)]^2}{(\omega_{j+1} - \Delta)^2(1 - \omega_{j+1} + \Delta)^2 k}, \tag{A.1}$$

where

$$\begin{aligned}
\delta_{1,j}(\Delta) :&= \lim_{n\to\infty} \frac{R_1^{[1,n]}((\omega_{j+1} - \Delta)n) - \mathbf{E}\left[R_1^{[1,n]}((\omega_{j+1} - \Delta)n)\right]}{n} \\
&= k \int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx - k(\omega_{j+1} - \Delta)^2, \\
\delta_{2,j}(\Delta) :&= \lim_{n\to\infty} \frac{R_2^{[1,n]}((\omega_{j+1} - \Delta)n) - \mathbf{E}\left[R_2^{[1,n]}((\omega_{j+1} - \Delta)n)\right]}{n} \\
&= k \int \frac{\left[\Delta f_l(x) + \sum_{l=j+1}^{m} p_l f_l(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx - k(1 - \omega_{j+1} + \Delta)^2,
\end{aligned}$$

and $D_{d,k}$ is the degree of vertex at the origin in the k-MST on a homogeneous Poisson process on R^d of rate 1, with a point added at the origin. Specially,

$$\begin{aligned}
\delta_{1,0}(\Delta) &= k \int \frac{[(p_0 - \Delta) f_0(x)]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx - k(\omega_1 - \Delta)^2, \\
\delta_{2,m}(\Delta) &= k \int \frac{[\Delta f_m(x)]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx - k\Delta^2.
\end{aligned}$$

Define

$$A_{1,j} = \left\{ \sum_{l=0}^{j-1} p_l f_l(x) \neq f_j(x) \sum_{l=0}^{j-1} p_l \right\}$$

for $j \neq 0$ and

$$A_{2,j} = \left\{ \sum_{l=j+1}^{m} p_l f_l(x) \neq f_j(x) \sum_{l=j+1}^{m} p_l \right\}$$

for $j \neq m$.

We can prove the following seven lemmas (The detailed proofs are provided in Section A.3.).

LEMMA A.1.1. *$\delta_{1,j}(\Delta) = \delta_{2,j}(\Delta)$ for all $j = 0, \ldots, m$.*

LEMMA A.1.2. *For $j = 0, \ldots, m$,*

$$(1 - \omega_{j+1} + \Delta)\delta_{1,j}(\Delta) + (\omega_{j+1} - \Delta)\delta_{2,j}(\Delta) \geq 0. \tag{A.2}$$

For $j = 1, \ldots, m-1$, the equality of (A.2) hold, when

$$(1 - \omega_{j+1} + \Delta)(\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x)) = (\omega_{j+1} - \Delta)(\Delta f_j(x) + \sum_{l=j+1}^{m} p_l f_l(x)).$$

For $j = 0$, the equality of (A.2) holds when $(1 - p_0) f_0(x) = \sum_{l=1}^{m} p_l f_l(x)$. For $j = m$, the equality of (A.2) holds when $\omega_m f_m(x) = \sum_{l=0}^{m-1} p_l f_l(x)$.

LEMMA A.1.3. *$\forall \Delta \in [0, p_j]$, $T_j(\Delta) = 0$ only when $A_{1,j}^c \cap A_{2,j}^c$ holds. Specially, for $j = 0$, $T_0(\Delta) = 0$ when $(1 - p_0) f_0(x) = \sum_{l=1}^{m} p_l f_l(x)$; for $j = m$, $T_m(\Delta) = 0$ when $\omega_m f_m(x) = \sum_{l=0}^{m-1} p_l f_l(x)$.*

LEMMA A.1.4. (1) *When $A_{1,j} \cap A_{2,j}$ holds, $T_j(\Delta)$ takes its maximum at $\Delta = 0$ or $\Delta = p_j$.*

(2) *When $A_{1,j} \cap A_{2,j}^c$ holds, $T_j(\Delta)$ takes its maximum at $\Delta = p_j$.*

(3) *When $A_{1,j}^c \cap A_{2,j}$ holds, $T_j(\Delta)$ takes its maximum at $\Delta = 0$.*

(4) *When $A_{1,j}^c \cap A_{2,j}^c$ holds, $T_j(\Delta)$ is a constant function on $\Delta \in [0, p_j]$.*

Specially, when $j = 0$ and $A_{2,0}^c$ holds, $T_0(\Delta)$ is a constant function on $\Delta \in [0, p_0)$, otherwise $T_0(\Delta)$ takes its maximum at $\Delta = 0$. When $j = m$ and $A_{1,m}^c$ holds, $T_m(\Delta)$ is a constant function on $\Delta \in (0, p_m]$, otherwise $T_m(\Delta)$ takes its maximum at $\Delta = p_m$.

LEMMA A.1.5. *If $f_j \neq f_{j+1}$ for all j, there exists ω_j, $j \in \{1, \ldots, m\}$ such that $\omega_j \in \partial argmax_{u \in (0,1)} T(u)$, where ∂ represents the boundary of a set.*

LEMMA A.1.6. *When $A_{1,j}^c \cap A_{2,j}^c$ holds, $T_{j-1}(\Delta)$ is strictly increasing on $\Delta \in [0, p_{j-1}]$, and $T_{j+1}(\Delta)$ is strictly decreasing on $\Delta \in [0, p_{j+1}]$.*

LEMMA A.1.7. *Assume*

$$\sup_{u \in \left[\frac{n_{le}^{[1,n]}}{n}, \frac{n_{ri}^{[1,n]}}{n}\right]} \left| \frac{S^{[1,n]}(nu)}{n} - T(u) \right| \xrightarrow{p} 0, \tag{A.3}$$

and $|\bar{\boldsymbol{\omega}}|$ *is finite, then*

$$P\left(\exists\, \omega_j \in \bar{\boldsymbol{\omega}}, |\hat{u} - \omega_j| < \epsilon\right) \to 1,$$

$\forall\, \epsilon > 0$ *as* $n \to \infty$.

With these Lemmas, we next prove Theorem 2.2.1.

When there are true change-points τ_j between $[a, b]$ in the process of G.WBS, for any generated interval $[a_l, b_l]$, there are two possibilities:

Case 1. *Interval* $[n_{le}^{[a_l,b_l]}, n_{ri}^{[a_l,b_l]}]$ *contains at least one true change-points.*

Case 2. *Interval* $[n_{le}^{[a_l,b_l]}, n_{ri}^{[a_l,b_l]}]$ *contains no true change-point.*

We first consider Case 1, for $\tau_j \in [n_{le}^{[a_l,b_l]}, n_{ri}^{[a_l,b_l]}]$, the limiting relative position $\lim_{n\to\infty} \frac{\tau_j - a_l}{b_l - a_l}$ are fixed, as $\lim_{n\to\infty} \frac{a_l}{n}$, $\lim_{n\to\infty} \frac{b_l}{n}$, and $\lim_{n\to\infty} \frac{\tau_j}{n}$ are fixed. Notice that $\forall (u_1, u_2) \subseteq \left[\frac{n_{le}^{[a_l,b_l]}}{n}, \frac{n_{ri}^{[a_l,b_l]}}{n}\right]$, the probability that $\lim_{n\to\infty} \frac{S^{[a_l,b_l]}(nu)}{b_l - a_l}$ is a constant function on (u_1, u_2) is 0. The condition in Lemma A.1.4 for constant function happens with 0 probability in the limiting regime, since $\frac{a_l}{n}$ and $\frac{b_l}{n}$ are uniformly generated between $[\frac{a}{n}, \frac{b}{n}]$. Even if the limit is a constant function, by Lemma A.1.3 and Lemma A.1.6, we know the constant is 0 and the corresponding interval (u_1, u_2) is a local minimizer. Therefore, this extreme case will not interfere the detection process.

By Lemma A.1.7, $\forall \epsilon > 0$,

$$P\left(\exists\, \omega_j \in \bar{\boldsymbol{\omega}}^{[a_l,b_l]}, |\hat{u}^{[a_l,b_l]} - \omega_j| < \epsilon\right) \to 1.$$

Thus, a true change-point is detected by $\hat{u}^{[a_l,b_l]}$. Next, we study the order of the statistic. Under assumption (2.2),

$$\frac{S^{[a_l,b_l]}(n\hat{u}^{[a_l,b_l]})}{b_l - a_l} \xrightarrow{p} T^{[a_l,b_l]}(\hat{u}^{[a_l,b_l]}).$$

By continuous mapping theorem, $T^{[a_l,b_l]}(\hat{u}^{[a_l,b_l]}) \xrightarrow{p} T^{[a_l,b_l]}(\omega_j)$, where $\omega_j \in \bar{\boldsymbol{\omega}}^{[a_l,b_l]}$. Then,

$$\frac{S^{[a_l,b_l]}(n\hat{u}^{[a_l,b_l]})}{b_l - a_l} \xrightarrow{p} T^{[a_l,b_l]}(\omega_j).$$

$\forall \epsilon > 0$,

$$P\left(\left|S^{[a_l,b_l]}(n\hat{u}^{[a_l,b_l]}) - (b_l - a_l)T^{[a_l,b_l]}(\omega_j)\right| > (b_l - a_l)\epsilon\right) \to 0,$$
$$P\left(S^{[a_l,b_l]}(n\hat{u}^{[a_l,b_l]}) < (b_l - a_l)T^{[a_l,b_l]}(\omega_j) - (b_l - a_l)\epsilon\right) \to 0.$$

$T^{[a_l,b_l]}(\omega_j)$ only depends on limiting relative position of change-points $\lim_{n\to\infty} \frac{\omega_j}{b_l - a_l}$ and distribution functions. Notice that $b_l - a_l \asymp n$. When $\zeta_n \prec n$,

$$P\left(S^{[a_l,b_l]}(n\hat{u}^{[a_l,b_l]}) > \zeta_n\right) \to 1,$$

as $n \to \infty$.

Next, we consider Case 2. In this case, the conditions of Theorem 4.1 in Chu & Chen (2019) are satisfied. For n given L and $\zeta_n \asymp \sqrt{n}$, the probability of any generated interval falsely detects a change-point,

$$\begin{aligned}
&\lim_{n\to\infty} P(\exists l = 1, \ldots, L, S^{[a_l,b_l]}(n\hat{u}^{[a_l,b_l]}) > \zeta_n) \\
&= \lim_{n\to\infty} P(\exists l = 1, \ldots, L, \sup_u \left[Z^*_{l,\text{diff}}(u)\right]^2 + \left[Z^*_{l,w}(u)\right]^2 > \zeta_n) \\
&\leq L \lim_{n\to\infty} P(\sup_u \left[Z^*_{\text{diff}}(u)\right]^2 + \sup_u \left[Z^*_w(u)\right]^2 > \zeta_n) \\
&\leq 2L \lim_{n\to\infty} P(\sup_u Z^*_{\text{diff}}(u) > (\frac{\zeta_n}{2})^{1/2}) + 2L \lim_{n\to\infty} P(\sup_u Z^*_w(u) > (\frac{\zeta_n}{2})^{1/2}), \qquad \text{(A.4)}
\end{aligned}$$

where Z^*_{diff} and Z^*_w are independent Gaussian process defined in Theorem 4.1 and 4.3 in Chu & Chen (2019).

Let $\mathcal{D}_{\text{diff}}$ and $\mathcal{D}_w$ represent Dudley's Integral of $Z^*_{\text{diff}}(u)$ and $Z^*_w(u)$ defined on their corresponding metric space. By the definition of Dudley's integral, it is easy to see that $\mathcal{D}_{\text{diff}}$ and $\mathcal{D}_w$ are two constants not depending on n. Then, when $\zeta_n = \max\{\mathcal{D}_{\text{diff}}, \mathcal{D}_w\}^2\sqrt{n}$,

$$\lim_{n\to\infty} (A.4) \lesssim 4L \lim_{n\to\infty} \exp\left(\frac{-\sqrt{n}}{4}\right) = 0.$$

Next we study how many intervals are necessary for G.WBS. When there exists at least one true change-points within $[a, b]$, at least one interval belonging to Case 1 is necessary. Define the

event $D_L^{[a,b]} = \left\{\forall\, l = 1\ldots, L, \forall \tau_j \in (a,b), \tau_j \notin [n_{le}^{[a_l,b_l]}, n_{ri}^{[a_l,b_l]}]\right\}$.

$$\begin{aligned} P(D_L^{[a,b]}) &= \prod_{l=1}^{L} P\left(\left\{\forall \tau_j \in (a,b), \tau_j \notin [n_{le}^{[a_l,b_l]}, n_{ri}^{[a_l,b_l]}]\right\}\right) \\ &= \left[P\left(\left\{\forall \tau_j \in (a,b), \tau_j \notin [n_{le}^{[a_1,b_1]}, n_{ri}^{[a_1,b_1]}]\right\}\right)\right]^L. \end{aligned}$$

If the default choice of $n_{le}^{[a_l,b_l]}$ and $n_{ri}^{[a_l,b_l]}$ are used, $P\left(\left\{\forall \tau_j \in (a,b), \tau_j \notin [n_{le}^{[a_1,b_1]}, n_{ri}^{[a_1,b_1]}]\right\}\right) < 1$. When $L \succ 1$, $P(D_L^{[a,b]}) \to 0$ as $n \to \infty$. Specially, if $n_{le}^{[a_1,b_1]} = a_l$ and $n_{ri}^{[a_1,b_1]} = b_l$ then

$$P(D_L^{[a,b]}) = \left[\frac{\sum_{t_j \in T^{[a_l,b_l]} \setminus \{t_1\}} (t_j - t_{j-1})^2}{(b-a)^2}\right]^L,$$

where $T^{[a_l,b_l]} = \{t_1, \ldots\} = \{a_l, b_l\} \bigcup (\boldsymbol{\omega} \bigcap [a_l, b_l])$ with $t_1 < t_2 < \ldots$.

A.2. Checking the Convergence Assumption in Theorem 2.2.1

Here, we check the convergence of $\frac{S^{[a_l,b_l]}(nu)}{b_l - a_l}$ towards its limit in Theorem 2.2.1. Since $T^{[a_l,b_l]}(u)$ usually does not have explicit formula, in each simulation run, we independently generate $K = 10$ $\frac{S^{[1,n]}(t)}{n}$ sequences ($\frac{S_k^{[1,n]}(nu)}{n}$, $k = 1, \ldots, K$), and measure $\max_{t,k} \left|\frac{S_k^{[1,n]}(t)}{n} - \frac{1}{K}\sum_{k'=1}^{K} \frac{S_{k'}^{[1,n]}(t)}{n}\right|$. This serves as an alternative way to measure the rate of uniform convergence of scan statistic. Setting 6-9 are used for illustrating:

- Setting 6: $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, I)$ if $i \in [\![1, n/6]\!] \cup [\![n/3+1, n/2]\!] \cup [\![2n/3+1, 5n/6]\!]$; $\mathbf{y}_i \sim \mathcal{N}_d(\frac{5}{4\log(d)}\mathbf{1}, I)$ otherwise.
- Setting 7: $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, I)$ if $i \in [\![1, n/6]\!] \cup [\![n/3+1, n/2]\!] \cup [\![2n/3+1, 5n/6]\!]$; $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, (1 + \frac{2}{\sqrt{d}})I)$ otherwise.
- Setting 8: $\mathbf{y}_i \sim t_{5,d}(\mathbf{0}, \Sigma)$ if $i \in [\![1, n/6]\!] \cup [\![n/3+1, n/2]\!] \cup [\![2n/3+1, 5n/6]\!]$; $\mathbf{y}_i \sim t_{5,d}(\frac{7}{4\log(d)}\mathbf{1}, \Sigma)$ otherwise, where $\Sigma_{jk} = 0.5^{|j-k|}$.
- Setting 9: $\mathbf{y}_i \sim \mathcal{N}_d(\mathbf{0}, \Sigma)$ if $i \in [\![1, n/6]\!] \cup [\![n/3+1, n/2]\!] \cup [\![2n/3+1, 5n/6]\!]$; $\mathbf{y}_i \sim \mathcal{N}_d(\frac{1}{\log(d)}\mathbf{1}, I)$ otherwise, where $\Sigma_{jk} = 0.5^{|j-k|}$.

We increase n from 60 to 6000, and the results are shown in Table A.1. It is conceivable from Table A.1 that under most cases, the uniform convergence empirically holds. When the original distribution is multivariate t distributed and dimension is high, the convergence is a bit slower.

TABLE A.1. Average maximum divergence of 100 replicates of $\frac{S^{[1,n]}(t)}{n}$.

Setting	d \ n	60	300	1500	6000
Setting 6	20	0.238	0.106	0.041	0.017
	100	0.230	0.103	0.054	0.026
	300	0.229	0.109	0.051	0.028
Setting 7	20	0.251	0.124	0.045	0.022
	100	0.250	0.114	0.043	0.017
	300	0.245	0.105	0.039	0.018
Setting 8	20	0.270	0.126	0.045	0.020
	100	0.257	0.145	0.070	0.040
	300	0.247	0.188	0.128	0.096
Setting 9	20	0.243	0.106	0.038	0.021
	100	0.230	0.105	0.048	0.024
	300	0.233	0.104	0.049	0.025

A.3. Proofs of Lemmas

PROOF OF LEMMA A.1.1. When $j \notin \{0, m\}$,

$$\begin{aligned}
\delta_{1,j}(\Delta) - \delta_{2,j}(\Delta) =& k \int \frac{[\sum_{l=0}^{m} p_l f_l(x)] \left[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - 2\Delta) f_j(x) - \sum_{l=j+1}^{m} p_l f_l(x)\right]}{\sum_{l=0}^{m} p_l f_l(x)} \, dx \\
& - k(2\omega_{j+1} - 2\Delta - 1) \\
=& k \int \sum_{l=0}^{j-1} p_l f_l(x) + (p_j - 2\Delta) f_j(x) - \sum_{l=j+1}^{m} p_l f_l(x) \, dx - k(2\omega_{j+1} - 2\Delta - 1) \\
=& 0.
\end{aligned}$$

The case when $j \in \{0, m\}$ can be proved similarly. □

PROOF OF LEMMA A.1.2.

$$\begin{aligned}
\delta_{1,j}(\Delta) =& k \int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x)\right]^2 - \left(\sum_{l=0}^{m} p_l f_l(x)\right)(\omega_{j+1} - \Delta)\left[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x)\right]}{\sum_{l=0}^{m} p_l f_l(x)} \, dx \\
=& k \int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x)\right]\left[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x) - \left(\sum_{l=0}^{m} p_l f_l(x)\right)(\omega_{j+1} - \Delta)\right]}{\sum_{l=0}^{m} p_l f_l(x)} \, dx \\
=& k \int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x)\right]\left(\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x)\right)(1 - \omega_{j+1} + \Delta)}{\sum_{l=0}^{m} p_l f_l(x)} \, dx \\
& - k \int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j - \Delta) f_j(x)\right]\left(\Delta f_j(x) + \sum_{l=j+1}^{m} p_l f_l(x)\right)(\omega_{j+1} - \Delta)}{\sum_{l=0}^{m} p_l f_l(x)} \, dx
\end{aligned}$$

$$
\begin{aligned}
\delta_{2,j}(\Delta) =& k \int \frac{\left[\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x)\right]^2 - \left(\sum_{l=0}^m p_l f_l(x)\right)(1-\omega_{j+1}+\Delta)\left[\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x)\right]}{\sum_{l=0}^m p_l f_l(x)} dx \\
=& k \int \frac{\left[\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x)\right]\left[\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x) - \left(\sum_{l=0}^m p_j f_l(x)\right)(1-\omega_{j+1}+\Delta)\right]}{\sum_{l=0}^m p_l f_l(x)} dx \\
=& k \int \frac{\left[\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x)\right]\left[\left(\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x)(\omega_{j+1}-\Delta)\right)\right]}{\sum_{l=0}^m p_l f_l(x)} dx \\
& - k \int \frac{\left[\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x)\right]\left(\sum_{l=0}^{j-1} p_l f_l(x) + (p_j-\Delta) f_j(x)\right)(1-\omega_{j+1}+\Delta)}{\sum_{l=0}^m p_l f_l(x)} dx
\end{aligned}
$$

$$
\begin{aligned}
\text{(A.5)} \quad & (1-\omega_{j+1}+\Delta)\,\delta_{1,j}(\Delta) + (\omega_{j+1}-\Delta)\delta_{2,j}(\Delta) \\
=& k \int \frac{\left[(\omega_{j+1}-\Delta)\left(\Delta f_j + \sum_{l=j+1}^m p_l f_l\right) - (1-\omega_{j+1}+\Delta)\left(\sum_{l=0}^{j-1} p_l f_l + (p_j-\Delta) f_j\right)\right]^2}{\sum_{l=0}^m p_l f_l(x)} dx \\
\text{(A.6)} \quad \geq& 0
\end{aligned}
$$

When $(1-\omega_{j+1}+\Delta)[\sum_{l=0}^{j-1} p_l f_l(x) + (p_j-\Delta) f_j(x)] = (\omega_{j+1}-\Delta)[\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x)]$, $(A.5) = 0$. The case when $j \in \{0, m\}$ can be proved similarly. □

Proof of Lemma A.1.3. When $A_{1,j}^c$ holds,

$$
\begin{aligned}
\sum_{l=0}^{j-1} p_l f_l(x) + (p_j-\Delta) f_j(x) &= \left(\sum_{l=0}^{j-1} p_l\right) f_j(x) + (p_j-\Delta) f_j(x) \\
&= (\omega_{j+1}-\Delta) f_j(x).
\end{aligned}
$$

When $A_{2,j}^c$ holds,

$$
\begin{aligned}
\Delta f_j(x) + \sum_{l=j+1}^m p_l f_l(x) &= \Delta f_j(x) + \left(\sum_{l=j+1}^m p_l\right) f_j(x) \\
&= (1-\omega_{j+1}+\Delta) f_j(x).
\end{aligned}
$$

By Lemma A.1.2, $\delta_{1,j}(\Delta) = \delta_{2,j}(\Delta) = 0$. Thus, $T_j(\Delta) = 0$. □

Proof of Lemma A.1.4. By Lemma A.1.1,

$$T_j(\Delta) = \frac{1}{k}\left[\frac{\delta_{1,j}(\Delta)}{(\omega_{j+1}-\Delta)(1-\omega_{j+1}+\Delta)}\right]^2.$$

By Lemma A.1.2, $\delta_{1,j} \geq 0$. The monotonicity of $T_j(\Delta)$ is the same as that of

$$\begin{aligned} T_j^{1/2}(\Delta) &= \frac{\delta_{1,j}(\Delta)}{(\omega_{j+1}-\Delta)(1-\omega_{j+1}+\Delta)\sqrt{k}} \\ &= \frac{\sqrt{k}\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x)+(p_j-\Delta)f_j(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)}\,dx - \sqrt{k}(\omega_{j+1}-\Delta)^2}{(\omega_{j+1}-\Delta)(1-\omega_{j+1}+\Delta)}. \end{aligned}$$

Here we do a transformation for easier calculation. Let $\omega_{j+1}-\Delta = 1-\theta$, where $\theta \in [1-\omega_{j+1}, 1-\omega_j] \subseteq (0,1)$ when $1 \leq j \leq m-1$.

Accordingly, we have $p_j - \Delta = 1-\theta-\omega_j$. Then,

$$\begin{aligned} \frac{T_j^{1/2}(\theta)}{\sqrt{k}} =& \frac{\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x)+(1-\theta-\omega_j)f_j(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)}\,dx - (1-\theta)^2}{\theta(1-\theta)} \\ =& \frac{\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x)+(1-\omega_j)f_j(x)\right]^2+\theta^2 f_j(x)^2-2\theta f_j(x)\left[\sum_{l=0}^{j-1} p_l f_l(x)+(1-\omega_j)f_j(x)\right]}{\sum_{l=0}^{m} p_l f_l(x)}\,dx}{\theta(1-\theta)} \\ & - \frac{\int \frac{\left[\sum_{l=0}^{m} p_l f_l(x)\right]\left[\left(\sum_{l=0}^{j-1} p_l f_l(x)+(1-\omega_j)f_j(x)\right)+\theta^2 f_j(x)-2\theta\left(\sum_{l=0}^{j-1} p_l f_l(x)+(1-\omega_j)f_j(x)\right)\right]}{\sum_{l=0}^{m} p_l f_l(x)}\,dx}{\theta(1-\theta)} \\ =& \frac{\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x)+(1-\omega_j)f_j(x)\right]\left[(1-\omega_{j+1})f_j(x)-\sum_{l=j+1}^{m} p_l f_l(x)\right]}{\sum_{l=0}^{m} p_l f_l(x)}\,dx}{\theta(1-\theta)} \\ & + \theta^2 \frac{\int \frac{f_j(x)^2-f_j(x)\left[\sum_{l=0}^{m} p_l f_l(x)\right]}{\sum_{l=0}^{m} p_l f_l(x)}\,dx}{\theta(1-\theta)} \\ & + 2\theta \frac{\int \frac{\left[\sum_{l=0}^{m} p_l f_l(x)-f_j(x)\right]\left[\sum_{l=0}^{j-1} p_l f_l(x)+(1-\omega_j)f_j(x)\right]}{\sum_{l=0}^{m} p_l f_l(x)}\,dx}{\theta(1-\theta)}. \end{aligned} \tag{A.7}$$

Take the derivative of the above quantity (A.7) w.r.t. θ, we have

$$\begin{aligned}\frac{dT_j^{1/2}(\theta)}{\sqrt{k}\,d\theta} =&\theta^2\frac{\int \frac{f_j(x)\left[f_j(x)-\sum_{l=0}^m p_l f_l(x)\right]}{\sum_{l=0}^m p_l f_l(x)}\,dx + 2\int \frac{\left[\sum_{l=0}^m p_l f_l(x)-f_j(x)\right]\left[\sum_{l=0}^{j-1} p_l f_l(x)+(1-\omega_j)f_j(x)\right]}{\sum_{l=0}^m p_l f_l(x)}\,dx}{\theta^2(1-\theta)^2}\\ &+(2\theta-1)\frac{\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x)+(1-\omega_j)f_j(x)\right]\left[(1-\omega_{j+1})f_j(x)-\sum_{l=j+1}^m p_l f_l(x)\right]}{\sum_{l=0}^m p_l f_l(x)}\,dx}{\theta^2(1-\theta)^2}. \qquad \text{(A.8)}\end{aligned}$$

Notice that $\theta^2(1-\theta)^2 > 0$ for all $\theta \in (0,1)$. Also notice that the numerator of (A.8) has the structure of

$$A\theta^2 + B(2\theta-1).$$

When $A, B \neq 0$ and $A+B \neq 0$, $\lim_{\theta\to 0^+}(A.8) = \text{sign}(-B)\cdot\infty$ and $\lim_{\theta\to 1^-}(A.8) = \text{sign}(A+B)\cdot\infty$. Next we show that $A+B \geq 0$ and $B \geq 0$.

$$\begin{aligned}A+B =& \int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) + (1-\omega_j)f_j(x) - f_j(x)\right]\left[\sum_{l=0}^m p_l f_l(x) - f_j(x)\right]}{\sum_{l=0}^m p_l f_l(x)}\,dx\\ &+\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) + (1-\omega_j)f_j(x)\right]\left[\sum_{l=0}^{j} p_l f_l(x) - \omega_{j+1} f_j(x)\right]}{\sum_{l=0}^m p_l f_l(x)}\,dx\\ =&\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) - \omega_j f_j(x)\right]\left[\sum_{l=0}^m p_l f_l(x) - f_j(x)\right]}{\sum_{l=0}^m p_l f_l(x)}\,dx\\ &+\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) + (1-\omega_j)f_j(x)\right]\left[\sum_{l=0}^{j-1} p_l f_l(x) - \omega_j f_j(x)\right]}{\sum_{l=0}^m p_l f_l(x)}\,dx\\ =&\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) - \omega_j f_j(x)\right]\left[\sum_{l=0}^m p_l f_l(x) + \sum_{l=0}^{j-1} p_l f_l(x) - \omega_j f_j(x)\right]}{\sum_{l=0}^m p_l f_l(x)}\,dx\\ =&\int \frac{\left[\sum_{l=0}^{j-1} p_l f_l(x) - \omega_j f_j(x)\right]^2}{\sum_{l=0}^m p_l f_l(x)}\,dx\\ \geq& 0\end{aligned}$$

The equality holds when $\sum_{l=0}^{j-1} p_l f_l(x) = \omega_j f_j(x)$.

$$\begin{aligned} B =& \int \frac{[\sum_{l=0}^{m} p_l f_l(x)] \left[(1-\omega_{j+1}) f_j(x) - \sum_{l=j+1}^{m} p_l f_l(x)\right]}{\sum_{l=0}^{m} p_l f_l(x)} \, dx \\ &+ \int \frac{\left[(1-\omega_{j+1}) f_j(x) - \sum_{l=j+1}^{m} p_l f_l(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)} \, dx \\ =& \int \frac{\left[(1-\omega_{j+1}) f_j(x) - \sum_{l=j+1}^{m} p_l f_l(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)} \, dx \\ \geq& 0 \end{aligned}$$

The equality holds when $(1-\omega_{j+1}) f_j(x) = \sum_{l=j+1}^{m} p_l f_l(x)$.

When $A + B > 0$ and $B > 0$, $\frac{dT_j^{1/2}(\theta)}{d\theta}$ is negative when θ is close to 0 and positive when θ is close to 1. Considering the quadratic structure of the numerator of (A.8), it indicates that $\frac{dT_j^{1/2}(\theta)}{d\theta}$ has only one root in $(0,1)$. There are 3 possibilities of monotonicity of $T_j^{1/2}(\theta)$ according to the position of ω_j and root of $\frac{dT_j^{1/2}(\theta)}{d\theta}$.

1. $T_j^{1/2}(\theta)$ is strictly decreasing on $(1-\omega_{j+1}, 1-\omega_j)$, i.e., the root of $\frac{dT_j^{1/2}(\theta)}{d\theta}$ is greater than $1-\omega_j$.

2. $T_j^{1/2}(\theta)$ is strictly increasing on $(1-\omega_{j+1}, 1-\omega_j)$, i.e., the root of $\frac{dT_j^{1/2}(\theta)}{d\theta}$ is smaller than $1-\omega_{j+1}$.

3. $T_j^{1/2}(\theta)$ first strictly decreases and then strictly increases on $(1-\omega_{j+1}, 1-\omega_j)$, i.e., the root of $\frac{dT_j^{1/2}(\theta)}{d\theta}$ lies between $(1-\omega_{j+1}, 1-\omega_j)$.

No matter which case happens, $T_j^{1/2}(\theta)$ take its maximum at $1-\omega_{j+1}$ or $1-\omega_j$. Or if we use $T_j(\Delta)$, $T_j(\Delta)$ takes its maximum at $\Delta = 0$ or p_j.

When $A = 0$ and $B > 0$, the numerator is not quadratic. But $\frac{dT_j^{1/2}(\theta)}{d\theta}$ is still negative when θ is close to 0 and positive when θ is close to 1. This is similar to the previous scenario.

When $A + B > 0$ and $B = 0$, $\frac{dT_j^{1/2}(\theta)}{d\theta}$ is always positive on $(0,1)$. Therefore $T_j(\Delta)$ takes its maximum at $\Delta = p_j$.

When $A + B = 0$ and $B > 0$, the numerator of (A.8) becomes $-B\theta^2 + B(2\theta - 1) = -B(1-\theta)^2$. So (A.8) becomes $-B/\theta^2$. Therefore, $\frac{dT_j^{1/2}(\theta)}{d\theta}$ is always negative and $T_j(\Delta)$ take its maximum at $\Delta = 0$.

When $A+B=0$ and $B=0$, $T_j(\Delta)$ is a constant function $\forall\,\Delta\in(0,p)$.

Next we discuss the corner case when $j=0$, i.e., $f_j(x)$ is the first distribution in the whole sequence. We will show that $T_0(\Delta)$ takes its maximum at $\Delta=0$ or it could be a constant function. The corresponding $T_0^{1/2}(\theta)$ is defined as:

$$T_0^{1/2}(\theta)=\frac{\sqrt{k}\int\frac{[(1-\theta)f_0(x)]^2}{\sum_{l=0}^m p_l f_l(x)}\,dx-\sqrt{k}(1-\theta)^2}{\theta(1-\theta)},$$

where $\theta\in[1-\omega_1,1)$. The derivative is

$$\begin{aligned}\frac{1}{\sqrt{k}}\frac{dT_0^{1/2}(\theta)}{d\theta}=&\theta^2\frac{\int\frac{\left[\sum_{k=1}^m p_k f_k(x)-(1-\omega_1)f_0(x)\right]f_0(x)}{\sum_{l=0}^m p_l f_l(x)}\,dx}{\theta^2(1-\theta)^2}\\ &-(2\theta-1)\frac{\int\frac{\left[\sum_{k=1}^m p_k f_k(x)-(1-\omega_1)f_0(x)\right]f_0(x)}{\sum_{l=0}^m p_l f_l(x)}\,dx}{\theta^2(1-\theta)^2}.\end{aligned}\tag{A.9}$$

Notice that the numerator of (A.9) has the structure of $A\theta^2+B(2\theta-1)$ with $A+B=0$. The numerator of (A.9) becomes $-B\theta^2+B(2\theta-1)=-B(1-\theta)^2$. So (A.9) becomes $-B/\theta^2$. It has been shown that $B\geq 0$. Therefore, $\frac{dT_0^{1/2}(\theta)}{d\theta}$ is always negative and $T_0(\Delta)$ take its maximum at $\Delta=0$. Therefore $T_0^{1/2}(\theta)$ is strictly decreasing on $[1-\omega_1,1)$, and $T_0(\Delta)$ takes its maximum at $\Delta=0$. Also, $B=0$ when $\sum_{l=1}^m p_l f_l(x)=(1-\omega_1)f_0(x)$, implying $T_0(\Delta)$ is a constant function when $\Delta\in[0,p_1)$.

Similarly, consider the other corner case that $j=m$, i.e., there is no other distributions on the right side of $f_m(x)$. This is symmetric to $j=0$ scenario. $T_m^{1/2}(\theta)$ is strictly increasing on $(0,p_m]$ when $A_{1,m}$ holds. Then, $T_m(\Delta)$ takes its maximum at $\Delta=p_m$. When $A_{1,m}^c$ holds, $T_m(\Delta)$ is a constant function. □

Proof of Lemma A.1.5. According to Lemma A.1.4, when u is between two adjacent change-points, the monotonicity of $T(u)$ is restricted to 4 possibilities: strictly increasing, strictly decreasing, strictly decreasing then increasing, degenerating to a constant function. Therefore, any $u\in(\omega_j,\omega_{j+1})$ cannot belongs to $\partial\text{argmax}_{u\in(0,1)}\,T(u)$. Next, we rule out the possibilities that $\partial\text{argmax}_{u\in(0,1)}\,T(u)=\{0,1\},\{0\}$, or $\{1\}$.

If $\partial\text{argmax}_{u\in(0,1)}\,T(u)=\{0\}$, it indicates $T(u)$ strictly decreases when $u\in(0,\omega_1)$, which contradicts with Lemma A.1.4. Similarly, $\partial\text{argmax}_{u\in(0,1)}\,T(u)=\{1\}$ can also be ruled out.

If $\partial \text{argmax}_{u\in(0,1)}\, T(u) = \{0,1\}$, it indicates that $T(u)$ is nonincreasing on $(0,\omega_1)$ and nondecreasing on $(\omega_m,1)$. Also, $\lim_{u\to 0} T(u) = \lim_{u\to 1} T(u)$. By Lemma A.1.4, $T(u)$ must be a constant function on $(0,\omega_1)$ and $(\omega_m,1)$. Furthermore, if $T(u)$ is not a constant function on $(0,1)$, there must exist another $\omega_j \in \partial \text{argmax}_{u\in(0,1)}\, T(u)$ other than 0 and 1. Therefore, $T(u)$ is a constant function on $u \in (0,1)$. By Lemma A.1.4, when $u \in (\omega_1,\omega_2)$, $T(u)$ is a constant function indicates $f_0 = f_1$, which is contradictory. □

Proof of Lemma A.1.6. In this proof, we use the same notation in Lemma A.1.4. Let

$$\frac{1}{\sqrt{k}}\frac{dT_{j-1}^{1/2}(\theta)}{d\theta} = \frac{A\theta^2 + B(2\theta-1)}{\theta^2(1-\theta)^2},$$

where

$$A + B = \int \frac{\left[\sum_{l=0}^{j-2} p_l f_l(x) - \omega_{j-1} f_{j-1}(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx,$$
$$B = \int \frac{\left[(1-\omega_j) f_{j-1}(x) - \sum_{l=j}^{m} p_l f_l(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx.$$

Furthermore,

$$\begin{aligned} B &= \int \frac{[(1-\omega_j) f_{j-1}(x) - p_j f_j(x) - (1-\omega_{j+1}) f_j(x)]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx \\ &= \int \frac{\left[(1-\omega_j) f_{j-1}(x) - \frac{(1-\omega_j)}{\omega_j}\sum_{l=0}^{j-1} p_l f_l(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx \\ &= \int \frac{\left[\frac{1-\omega_j}{\omega_j}\sum_{l=0}^{j-2} p_l f_l(x) - \frac{(1-\omega_j)\omega_{j-1}}{\omega_j} f_{j-1}(x)\right]^2}{\sum_{l=0}^{m} p_l f_l(x)}\, dx \\ &= \left(\frac{1-\omega_j}{\omega_j}\right)^2 (A+B). \end{aligned}$$

The derivative $\frac{dT_{j-1}^{1/2}(\theta)}{d\theta}$ can be represented in the following way:

$$\frac{1}{\sqrt{k}}\frac{dT_{j-1}^{1/2}(\theta)}{d\theta} = \frac{\left[\frac{2\omega_j - 1}{(1-\omega_j)^2}\right] B\theta^2 + 2B\theta - B}{\theta^2(1-\theta)^2}. \tag{A.10}$$

Notice that $1-\omega_j$ is a root of the derivative, which can be seen easily by plugging $\theta = 1-\omega_j$ in (A.10). Recall that $\frac{dT_{j-1}^{1/2}(\theta)}{d\theta}$ only has one root on $\theta \in (0,1)$, then $T_{j-1}(\theta)$ is strictly increasing on $[1-\omega_j, 1-\omega_{j-1}]$.

The case of $T_{j+1}(\theta)$ can be proved similarly. □

Proof of Lemma A.1.7. By Lemma A.1.5, if $f_j \neq f_{j+1}$, there exists ω_j, $j \in \{1,\dots,m\}$ such that $\omega_j \in \text{argmax}_{u\in(0,1)} T(u)$, since $T(u)$ is continuous. If $\bar{\boldsymbol{\omega}}$ only contains finite numbers of u, then $\bar{\boldsymbol{\omega}}$ must consists of $\omega_j \in \left[\frac{n_{le}^{[1,n]}}{n}, \frac{n_{ri}^{[1,n]}}{n}\right]$. Let $V_\epsilon(\omega_j) := \{u : |u-\omega_j| \le \epsilon\}$. Observe that

$$\bigcap_{\omega_j\in\bar{\boldsymbol{\omega}}} \{|\hat{u}-\omega_j| > \epsilon\} \subseteq \bigcap_{\omega_j\in\bar{\boldsymbol{\omega}}} \left\{ \sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} \left(\frac{S^{[1,n]}(nu)}{n} - \frac{S^{[1,n]}(n\omega_j)}{n} \right) \ge 0 \right\}.$$

We can rewrite the supremum on the right side as

$$\begin{aligned}
&\sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} \left(\frac{S^{[1,n]}(nu)}{n} - \frac{S^{[1,n]}(n\omega_j)}{n} \right) \\
=& \sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} \frac{S^{[1,n]}(nu)}{n} - \sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} T(u) \\
&- \left[\frac{S^{[1,n]}(n\omega_j)}{n} - T(\omega_j) \right] + \left[\sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} T(u) - T(\omega_j) \right].
\end{aligned}$$

By mimicking the proof of Theorem 5.2.1 in Chen & Friedman (2017), it is not hard to see that when f_j's are continuous multivariate distributions, if the graph is a $k-$MST, $k = O(1)$, based on the Euclidean distance, $\frac{S^{[1,n]}(n\omega_j)}{n} \xrightarrow{a.s.} T(\omega_j)$. Notice that

$$\begin{aligned}
&\left\{ \sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} \frac{S^{[1,n]}(nu)}{n} - \sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} T(u) > \epsilon \right\} \\
&\subseteq \left\{ \sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} \left| \frac{S^{[1,n]}(nu)}{n} - T(u) \right| > \epsilon \right\}.
\end{aligned}$$

By (A.3), the probability of the above event converge to 0. Finally, $\sup_{u\in\bigcap_{\omega_k\in\bar{\boldsymbol{\omega}}} V_\epsilon(\omega_k)^c} T(u) - T(\omega_j) < 0$ when $\omega_j \in \bar{\boldsymbol{\omega}}$, as u is bounded and the shape of $T(u)$ is determined by lemma A.1.4. □

A.4. Choice of c

We here check the choice of c in the expanded pseudo BIC ep-BIC$(\tilde{\boldsymbol{\tau}}) = eAS(\tilde{\boldsymbol{\tau}}) - c\tilde{m}\log n$ numerically. We use the G.WBS-based version and all the other parameters are set at their default values. We use the same simulation Settings 6-9 defined in Section A.1 with n set to be 120. Each simulation settings is repeated 1000 times, with dimensions $d = 20, 50, 100, 500$ and 1000. The number of truly and falsely detected change-points are plotted in Figure A.1. A true change-point τ_j is deemed to be detected if an estimated change-point exists within 2 observations of it. The number of falsely detected change-points is defined as the number of estimated change-points minus the number of truly detected change-points.

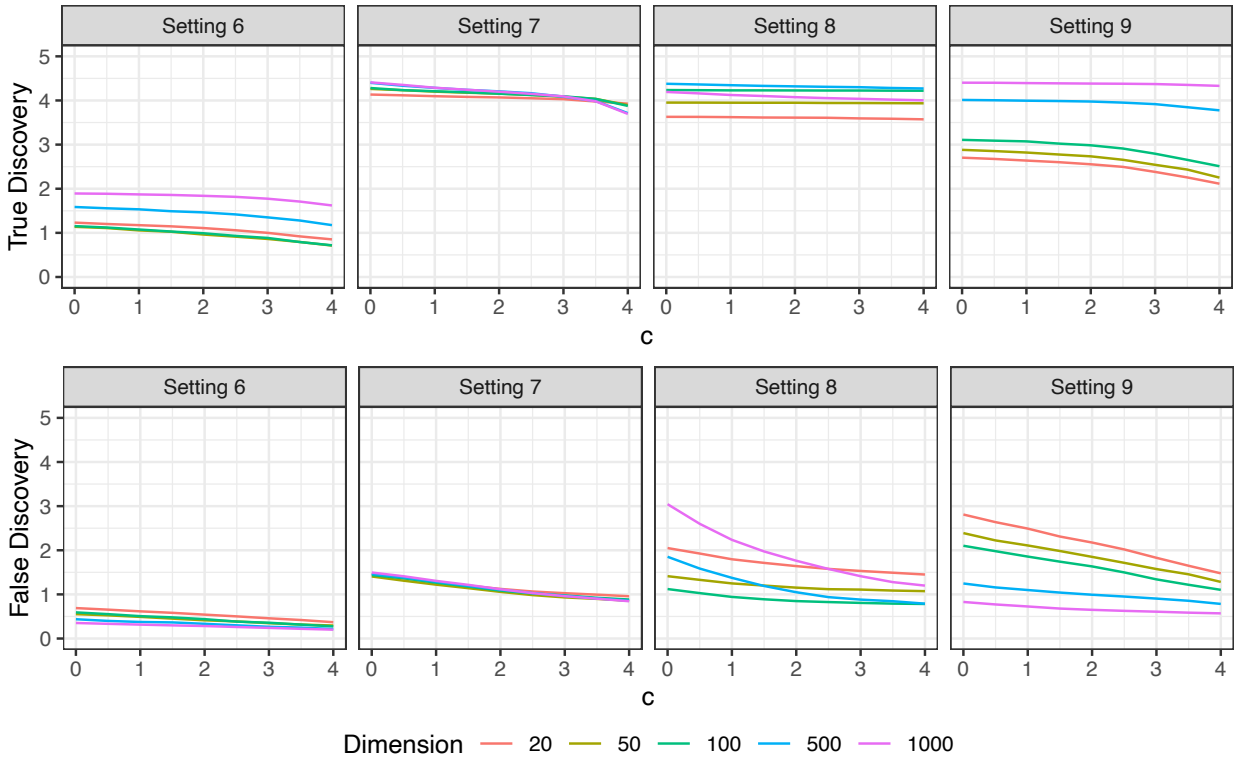

FIGURE A.1. The average number of true discoveries and false discoveries under different values of c's setting. Each has 1,000 replicates.

We can see from Figure A.1 that the average true discoveries decreases faster after roughly $c = 2$. For average false discoveries, it decreases slower after roughly $c = 2$. Though the exact

position of change varies a bit across different settings, we recommend $c = 2$ since it reaches a good balance between power and false discovery rate.

Bibliography

Arlot, S., Celisse, A. & Harchaoui, Z. (2019), 'A kernel multiple change-point algorithm via model selection.', *Journal of Machine Learning Research* **20**(162), 1–56.

Bennett, C., Gale, S. D., Garrett, M. E., Newton, M. L., Callaway, E. M., Murphy, G. J. & Olsen, S. R. (2019), 'Higher-order thalamic circuits channel parallel streams of visual information in mice', *Neuron* **102**(2), 477–492.

Berkes, P., Orbán, G., Lengyel, M. & Fiser, J. (2011), 'Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment', *Science* **331**(6013), 83–87.

Braun, U., Harneit, A., Pergola, G., Menara, T., Schäfer, A., Betzel, R. F., Zang, Z., Schweiger, J. I., Zhang, X., Schwarz, K. et al. (2021), 'Brain network dynamics during working memory are modulated by dopamine and diminished in schizophrenia', *Nature Communications* **12**(1), 1–11.

Chen, H., Chen, S. & Deng, X. (2019), 'A universal nonparametric event detection framework for neuropixels data', *bioRxiv* .

Chen, H., Chen, X. & Su, Y. (2018), 'A weighted edge-count two-sample test for multivariate and object data', *Journal of the American Statistical Association* **113**(523), 1146–1155.

Chen, H. & Friedman, J. H. (2017), 'A new graph-based two-sample test for multivariate and object data', *Journal of the American Statistical Association* **112**(517), 397–409.

Chen, H. & Zhang, N. (2015), 'Graph-based change-point detection', *Annals of Statistics* **43**(1), 139–176.

Chen, H. & Zhang, N. R. (2013), 'Graph-based tests for two-sample comparisons of categorical data', *Statistica Sinica* **23**(4), 1479–1503.

Chen, R., Canales, A. & Anikeeva, P. (2017), 'Neural recording and modulation technologies', *Nature Reviews Materials* **2**(2), 1–16.

Cho, H. & Fryzlewicz, P. (2015), 'Multiple-change-point detection for high dimensional time series via sparsified binary segmentation', *Journal of the Royal Statistical Society. Series B (Statistical*

Methodology) **77**(2), 475–507.

Chu, L. & Chen, H. (2019), 'Asymptotic distribution-free change-point detection for multivariate and non-euclidean data', *Annals of Statistics* **47**(1), 382–414.

Chung, J. E., Magland, J. F., Barnett, A. H., Tolosa, V. M., Tooker, A. C., Lee, K. Y., Shah, K. G., Felix, S. H., Frank, L. M. & Greengard, L. F. (2017), 'A fully automated approach to spike sorting', *Neuron* **95**(6), 1381–1394.

Dong, F., He, Y., Wang, T., Han, D., Lu, H. & Zhao, H. (2020), 'Predicting viral exposure response from modeling the changes of co-expression networks using time series gene expression data', *BMC bioinformatics* **21**(1), 1–18.

Dutta, B., Andrei, A., Harris, T. D., Lopez, C. M., O'Callahan, J., Putzeys, J., Raducanu, B. C., Severi, S., Stavisky, S. D., Trautmann, E. M., Welkenhuysen, M. & Shenoy, K. V. (2019), The neuropixels probe: A cmos based integrated microsystems platform for neuroscience and brain-computer interfaces, *in* '2019 IEEE International Electron Devices Meeting (IEDM)', pp. 10.1.1–10.1.4.

Enikeeva, F. & Harchaoui, Z. (2019), 'High-dimensional change-point detection under sparse alternatives', *Annals of statistics* **47**(4), 2051–2079.

Evans, D. A., Stempel, A. V., Vale, R., Ruehle, S., Lefler, Y. & Branco, T. (2018), 'A synaptic threshold mechanism for computing escape decisions', *Nature* **558**(7711), 590–594.

Friedman, J. H. & Rafsky, L. C. (1979), 'Multivariate generalizations of the wald-wolfowitz and smirnov two-sample tests', *The Annals of Statistics* **7**(4), 697–717.

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

Fryzlewicz, P. (2020), 'Detecting possibly frequent change-points: Wild binary segmentation 2 and steepest-drop model selection', *Journal of the Korean Statistical Society* **49**(4), 1027–1070.

Gardner, R. J., Lu, L., Wernle, T., Moser, M.-B. & Moser, E. I. (2019), 'Correlation structure of grid cells is preserved during sleep', *Nature neuroscience* **22**(4), 598–608.

Gaucher, Q., Panniello, M., Ivanov, A. Z., Dahmen, J. C., King, A. J. & Walker, K. M. (2020), 'Complexity of frequency receptive fields predicts tonotopic variability across species', *eLife* **9**, e53462.

Harchaoui, Z., Moulines, E. & Bach, F. (2008), Kernel change-point analysis, *in* 'Advances in Neural Information Processing Systems', Vol. 21.

Henze, N. & Penrose, M. D. (1999), 'On the multivariate runs test', *The Annals of Statistics* **27**(1), 290–298.

James, N. A. & Matteson, D. S. (2015), 'ecp: An r package for nonparametric multiple change point analysis of multivariate data', *Journal of Statistical Software, Articles* **62**(7), 1–25.

Jiang, Y., Oldridge, D. A., Diskin, S. J. & Zhang, N. R. (2015), 'Codex: a normalization and copy number variation detection method for whole exome sequencing', *Nucleic acids research* **43**(6), e39–e39.

Johnson, G. A., Badea, A., Brandenburg, J., Cofer, G., Fubara, B., Liu, S. & Nissanov, J. (2010), 'Waxholm space: an image-based reference for coordinating mouse brain research', *Neuroimage* **53**(2), 365–372.

Jun, J. J., Steinmetz, N. A., Siegle, J. H., Denman, D. J., Bauza, M., Barbarits, B., Lee, A. K., Anastassiou, C. A., Andrei, A., Aydın, Ç. et al. (2017), 'Fully integrated silicon probes for high-density recording of neural activity', *Nature* **551**(7679), 232–236.

Kovács, S., Li, H., Bühlmann, P. & Munk, A. (2020), 'Seeded binary segmentation: A general methodology for fast and optimal change point detection', *arXiv preprint arXiv:2002.06633* .

Krupic, J., Bauza, M., Burton, S. & O'Keefe, J. (2018), 'Local transformations of the hippocampal cognitive map', *Science* **359**(6380), 1143–1146.

Lavielle, M. & Teyssiere, G. (2006), 'Detection of multiple change-points in multivariate time series', *Lithuanian Mathematical Journal* **46**(3), 287–306.

Lewis, C. M., Bosman, C. A. & Fries, P. (2015), 'Recording of brain activity across spatial scales', *Current opinion in neurobiology* **32**, 68–77.

Matteson, D. S. & James, N. A. (2014), 'A nonparametric approach for multiple change point analysis of multivariate data', *Journal of the American Statistical Association* **109**(505), 334–345.

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

Pachitariu, M., Steinmetz, N. A., Kadir, S. N., Carandini, M. & Harris, K. D. (2016), 'Fast and accurate spike sorting of high-channel count probes with kilosort', *Advances in neural information processing systems* **29**.

Park, J., Phillips, J. W., Guo, J.-Z., Martin, K. A., Hantman, A. W. & Dudman, J. T. (2022), 'Motor cortical output for skilled forelimb movement is selectively distributed across projection neuron classes', *Science Advances* **8**(10), eabj5167.

Peel, L. & Clauset, A. (2015), Detecting change points in the large-scale structure of evolving networks, Vol. 29.

Sauerbrei, B. A., Guo, J.-Z., Cohen, J. D., Mischiati, M., Guo, W., Kabra, M., Verma, N., Mensh, B., Branson, K. & Hantman, A. W. (2020), 'Cortical pattern generation during dexterous movement is input-driven', *Nature* **577**(7790), 386–391.

Schneider, D. M., Nelson, A. & Mooney, R. (2014), 'A synaptic and circuit basis for corollary discharge in the auditory cortex', *Nature* **513**(7517), 189–194.

Steinmetz, N. A., Aydin, C., Lebedeva, A., Okun, M., Pachitariu, M., Bauza, M., Beau, M., Bhagat, J., Böhm, C., Broux, M. et al. (2021), 'Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings', *Science* **372**(6539), eabf4588.

Steinmetz, N. A., Koch, C., Harris, K. D. & Carandini, M. (2018), 'Challenges and opportunities for large-scale electrophysiology with neuropixels probes', *Current opinion in neurobiology* **50**, 92–100.

Steinmetz, N., Pachitariu, M., Stringer, C., Carandini, M. & Harris, K. (2019), 'Eight-probe neuropixels recordings during spontaneous behaviors'.

Stringer, C., Pachitariu, M., Steinmetz, N., Reddy, C. B., Carandini, M. & Harris, K. D. (2019), 'Spontaneous behaviors drive multidimensional, brainwide activity', *Science* **364**(6437), eaav7893.

Vostrikova, L. Y. (1981), 'Detecting "disorder" in multidimensional random processes', *Doklady Akademii Nauk* **259**(2), 270–274.

Wang, G., Zou, C. & Yin, G. (2018), 'Change-point detection in multinomial data with a large number of categories', *The Annals of Statistics* **46**(5), 2020–2044.

Wang, T. & Samworth, R. J. (2016), 'High-dimensional changepoint estimation via sparse projection', *arXiv preprint arXiv:1606.06246* .

Woodroofe, M. (1976), 'Frequentist properties of bayesian sequential tests', *Biometrika* **63**(1), 101–110.

Woodroofe, M. (1978), 'Large deviations of likelihood ratio statistics with applications to sequential testing', *The Annals of Statistics* **6**(1), 72–84.

Yao, Y.-C. (1988), 'Estimating the number of change-points via schwarz'criterion', *Statistics & Probability Letters* **6**(3), 181–189.

Zhang, N. R. (2005), *Change-point detection and sequence alignment: statistical problems of genomics*, stanford university.

Zhang, N. R., Siegmund, D. O., Ji, H. & Li, J. Z. (2010), 'Detecting simultaneous changepoints in multiple sequences', *Biometrika* **97**(3), 631–645.

Zhang, Y. & Chen, H. (2021), 'Graph-based multiple change-point detection', *arXiv preprint arXiv:2110.01170* .

Zhu, Y. & Chen, H. (2021), 'Limiting distributions of graph-based test statistics on sparse and dense graphs', *arXiv preprint arXiv:2108.07446* .