Online Change-point and Anomaly Detection for Complex and High-dimensional Data

by

Xiaojun Zheng

Department of Statistical Science
Duke University

Defense Date: May 20, 2024

Approved:

Simon Mak, Supervisor

Sayan Mukherjee

Ezra Miller

Yao Xie

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Department of Statistical Science
in the Graduate School of Duke University
2024

ABSTRACT

Online Change-point and Anomaly Detection for Complex and High-dimensional Data

by

Xiaojun Zheng

Department of Statistical Science
Duke University

Defense Date: May 20, 2024

Approved:

Simon Mak, Supervisor

Sayan Mukherjee

Ezra Miller

Yao Xie

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistical Science in the Graduate School of Duke University
2024

Copyright © 2024 by
Xiaojun Zheng
All rights reserved except the rights granted by the Creative Commons
Attribution-Noncommercial Licence

Abstract

With remarkable advances in sensing and experimental technologies, scientists and engineers now have access to massive datasets with complex forms for decision-making, and the ability to efficiently process and analyze high-dimensional datasets in real time is more critical than ever. Anomaly detection and change point detection are pivotal in various applications, from video surveillance to solar activity monitoring, enabling timely responses to dynamic changes and aberrant behaviors within the data.

This dissertation offers three main contributions to the field. In Chapter 2, we introduce a novel persistence diagram-based method for change point detection, and the key component is to leverage the embedded topological structured from topological data analysis. In Chapter 3, we extend the Robust Principal Component Analysis to accommodate Exponential Family distributions. This extension enables the decomposition of data into low-rank and sparse matrices, where the sparse matrix effectively captures anomalies essential for process monitoring and diagnosis. In Chapter 4, we propose a Bayesian change-point detection framework, which incorporates image and spatial features learned from Gaussian Markov random fields. We investigate the effectiveness of our proposed methods through extensive numerical experiments, and demonstrate their applicability in diverse fields such as engineering, physics, and computer vision.

Dedication

To Mom and Dad.

Contents

Abstract iv

List of Tables ix

List of Figures x

Acknowledgements xiii

1 Introduction 1

2 PERCEPT: a new online change-point detection method using topological data analysis 4

2.1 Introduction 4

2.2 Preliminaries and Motivation 6

2.2.1 TDA Preliminaries 6

2.2.2 Existing Baseline Methods 9

2.3 Persistence Diagram-based Change Detection 11

2.3.1 Persistence Histogram Binning 11

2.3.2 Test Statistic 13

2.3.3 Connection between EDD and Topology 16

2.3.4 Persistence Cluster Binning 17

2.3.5 Weight Optimization 19

2.4 Simulation Experiments 21

2.4.1 Shape Change 22

2.4.2 Noise Change 22

2.4.3 Increasing Dimensionality 24

2.4.4 EDD vs. ARL 26

2.5 Applications 26

2.5.1 Solar Flare Change Detection 26

2.5.2 Human Gesture Change Detection 30

3 e^{RPCA}: Robust Principal Component Analysis for Exponential Family Distributions 33

3.1 Introduction 33
3.2 Background and Motivation 36
3.2.1 Robust PCA 36
3.2.2 Exponential PCA 38
3.2.3 Steel Defect Detection Application 39
3.3 e^{RPCA}: RPCA for Exponential Family Distributions 41
3.3.1 Optimization Formulation 41
3.3.2 Optimization Algorithm 45
3.3.3 Hyperparameter Tuning 50
3.3.4 Computational Complexity 50
3.4 Numerical Experiments 52
3.4.1 Single-Group Setting 53
3.4.2 Multi-Group Setting 56
3.4.3 Recommendations 57
3.5 Applications 59
3.5.1 Steel Defect Detection 59
3.5.2 Crime Mapping in Atlanta 61
4 BLAST: Bayesian Online Structure-aware Change-point Detection 65
4.1 Introduction 65
4.2 Background and Motivation 69
4.2.1 Gaussian Markov Random Field 69
4.2.2 Deep Gaussian Markov Random Fields 71
4.2.3 Existing Baseline Methods 72
4.2.4 Street Scene Monitoring 73
4.3 Bayesian Online Structure-aware Change-point Detection 75
4.3.1 Image Prior Elicitation 75

4.3.2 Image Posterior Update 78
4.3.3 Bayesian Change-point Detection 79
4.4 Simulation Experiment 83
4.4.1 Spatial Correlation Change 84
4.4.2 Visual Structure Change 86
4.4.3 Structure Intensity Change 88
4.5 Applications 89
4.5.1 Street Scene Monitoring 89
4.5.2 Drift Tube Chambers Monitoring 90
5 Conclusion 99
Appendix A PERCEPT 102
A.1 Experiments for approximating the ARL and EDD 102
A.2 Details on the lower star filtration 102
A.3 Connection on distributional & bottleneck distances 103
A.4 Increasing Dimensionality on geometrix structures 103
Appendix B ERPCA 105
B.1 Closed-form Optimization Updates for θ 105
B.2 Simulation Set-up for $\mathbf{L}$ and $\mathbf{S}$ 105
Bibliography 107
Biography 120

List of Tables

2.1 Total computation time (in seconds) for the compared methods on the solar flare application. 30

3.1 Common distributions from the (one-parameter) exponential family (3.4) . . 39

B.1 Closed-form updates of (3.19) for common (one-parameter) exponential family distributions. 105

List of Figures

2.1 An illustration of the persistent homology pipeline, from point cloud data to a filtration of simplicial complexes to a persistence diagram. 8

2.2 (left) Point cloud data generated from two disjoint circles, and (right) its corresponding persistence diagram for 0-D and 1-D holes. 9

2.3 (a) Snapshots of the solar flare around the true change-point $t^* = 49$. (b,c) Hotelling's T^2 and Wasserstein distance statistics at each time t. 10

2.4 Visualizing the general workflow for PERCEPT. 12

2.5 (a) Persistence histograms for a pre-change and post-change solar flare image. (b) Visualizing the intervals used for the weighted ℓ_2 detection statistic. . . 15

2.6 (a) A sample pre-change and post-change PD. (b) Pre- and post-change cluster centers and its corresponding persistence cluster. 19

2.7 Test statistics using PERCEPT, Hotelling's T^2, MMD and Wasserstein distance at each time t, for the case of geometric shape changes. 23

2.8 The test statistics using PERCEPT, Hotelling's T^2, MMD and Wasserstein distance at each time t, for the case of noise changes. 24

2.9 The test statistics using PERCEPT, Hotelling's T^2, MMD and Wasserstein distance, with higher-dimensional geometric structures. 25

2.10 EDD–ARL curves for 2-D circle experiment with noise change for PERCEPT, Hotelling's T^2, MMD, and the Wasserstein distance method. 27

2.11 (a) Freudenthal triangulation of the solar flare image. (b) "Tilted" persistence diagram of the solar flare image in (a). 28

2.12 Detection statistics for each method in the solar flare monitoring application. 29

2.13 An illustration of the pipeline for human gesture change detection application. 31

2.14 Detection statistics for each method in the human gesture change detection application. 32

3.1 (Left) The uncorrupted steel sheet image from Severstal, and (right) corresponding binary images generated with Bernoulli noise. 40

3.2 Visualizing the estimated $\mathbf{L}$ and $\mathbf{S}$ from RPCA, along with the estimated $\mathbf{L}$ from ePCA, for the steel defect detection application. 41

3.3 Boxplots of recovery errors for the $\mathbf{L}$ and $\mathbf{S}$ for single-group simulations across different distributions. 54

3.4 Visualizing the true and recovered low-rank matrix **L** and sparse anomalies **S** for the single-group Bernoulli simulations. 55

3.5 Visualizing the true and recovered low-rank matrix **L** and sparse anomalies **S** for the single-group Exponential simulations. 56

3.6 Visualizing the true and recovered low-rank matrix **L** and sparse anomalies **S** for the single-group Poisson simulations. 56

3.7 Boxplots of recovery errors for the **L** and **S** for multi-group simulations across different distributions. 58

3.8 Visualizing the estimated **L** and **S**, using the e^{RPCA}, RPCA and ePCA in the steel defect detection application. 60

3.9 Visualizing the estimated low-rank matrix **L** from e^{RPCA} for the crime monitoring application. 62

3.10 Spatial visualization of the burglary rate from the estimated **L** on Monday, Friday, and Sunday at 9 a.m. and 10 p.m. for the crime monitoring application. 63

3.11 Visualizing the absolute values of the estimated anomaly matrices $\mathbf{S}_1$ (summer) and $\mathbf{S}_2$ (winter) from e^{RPCA} for the crime monitoring application. . . . 64

3.12 Plotting the non-zero entries in $\mathbf{S}_1$ and $\mathbf{S}_2$ for Beat 403 (residential) and Beat 503 (commercial) from e^{RPCA} for the crime monitoring application. 64

4.1 Illustration of the Deep GMRF. 72

4.2 (a) Snapshots of the street scene with a car driving starting at $t^* = 16$. (b,c) Hotelling's T^2 and MONAD statistics at each time t. 74

4.3 Constructing the precision matrix using DGMRF. 78

4.4 Visualizing the pre-change and post-change true images along with their generated samples: spatial correlation change 85

4.5 Results for the setting of spatial correlation change. 86

4.6 Visualizing the pre-change and post-change true images along with their generated samples: structure change 87

4.7 Results for the setting of structure change 87

4.8 Visualizing the pre-change and post-change true images along with their generated samples: intensity change 88

4.9 Results for the setting of intensity change. 89

4.10 Snapshots of the street scene with a car driving starting at $t^* = 16$. 90

4.11 Results for each compared method in the street scene monitoring application. 91

4.12 The experimental apparatus at Legnaro Laboratory, with four drift-tube chambers. 91

4.13 Examples of data matrices under the normal configuration and the anomalous configurations. 93

4.14 Results for lowering the front-end threshold to 75% (75 mV) 94

4.15 Results for lowering the front-end threshold to 50% (50 mV) 94

4.16 Results for lowering the front-end threshold to 25% (25 mV) 95

4.17 Results for reducing voltage to 75% (-900 V) 96

4.18 Results for reducing voltage to 50% (-600 V) 96

4.19 Results for reducing voltage to 25% (-300 V) 97

A.1 Visualizing a regular point (left) and a saddle point (right) in the lower star filtration. 103

A.2 The test statistics using PERCEPT, Hotelling's T^2, MMD and Wasserstein distance at each time t for the 3-D spheres/ ellipsoids, under noise changes. 104

Acknowledgements

I extend my deepest gratitude to my advisor, Simon Mak, whose guidance has been indispensable throughout my doctoral journey. I am truly fortunate to have been introduced to the field of change point and anomaly detection under his mentorship. Words fall short of expressing my happiness and appreciation for the opportunity to work with him. His passion for research and his insightful vision for solving practical problems have deeply inspired me. I am profoundly grateful for the chance to learn from his exceptional thought processes, which have significantly honed my problem-solving skills and continually fuel my own passion for discovery.

I also owe a sincere thanks to the members of my dissertation committee—Sayan Mukherjee, Ezra Miller, and Yao Xie. Collaborating with them has greatly enriched my research experience and broadened my perspectives in invaluable ways. Their mentorship has been pivotal in shaping my academic path and approach to research. Thank you for your invaluable contributions and guidance.

My deep appreciation extends to all my collaborators across various projects, including those not highlighted in this dissertation. A special thanks to Liyan Xie, Marc Ryser, Kaleigh Fetcko-Fayad, Devon Godfrey, Henry Kirveslahti, and Katherine Peters—your expertise and insights have been instrumental in my growth as a researcher. I am also grateful for my fellow students in the department for their companionship and the enriching discussions we have ever had. Additionally, I cherish the summer I spent at Merck, which was an incredible opportunity that allowed me to develop further as a biostatistician and gain valuable industry experience.

I am immensely grateful to Matthew Wright, my undergraduate advisor at St. Olaf College, whose inspiration and encouragement were crucial in my decision to pursue a PhD. I feel truly fortunate to have been introduced to the field of Topological Data Analysis under his mentorship.

Heartfelt thanks go to my cherished friends Chenyu Liu, Bokang Wang, Zhiyue Wang, Xinyi Niu, Kainan Wang, Yue Li, Yunran Chen, and Zhangjie Hong. Some of you have

been by my side for over half my life. Despite the distance and the rare opportunities to meet in person, your emotional support has been a source of immense strength and joy throughout my PhD journey.

A special thank you to my boyfriend Chao Ouyang, who has been my constant source of comfort and happiness for as long as I have been in the US. Your unwavering support has been my anchor. To my baby cat Simba, thank you for the boundless joy and companionship you've brought into our lives since the pandemic began.

Lastly, I owe an immeasurable amount of gratitude to my parents. You have always encouraged me to forge my own path, supporting and respecting my decisions, even when they provided tough lessons rather than immediate successes. Your unconditional love, support, and belief in me have been the foundation of my dreams, allowing me to pursue a life filled with ambition and fulfillment. This thesis is dedicated to you. I also extend a heartfelt thank you to my dog brother, Mantou, for bringing joy and steadfast company to my mom in my absence.

1. Introduction

With remarkable advances in sensing and experimental technologies, scientists and engineers now have access to massive datasets with complex forms for decision-making, and the ability to efficiently process and analyze high-dimensional datasets in real time is more critical than ever. Anomaly detection and change point detection are pivotal in various applications, from video surveillance (Ramachandra & Jones, 2020) to solar activity monitoring (Y. Xie et al., 2012), enabling timely responses to dynamic changes and aberrant behaviors within the data.

Traditional change-point detection methods largely focus on likelihood ratio tests, which presume that observations are independently and identically distributed, both before and after the change (Basseville & Nikiforov, 1993; Poor & Hadjiliadis, 2008; Siegmund, 1985; Tartakovsky et al., 2015). When the pre- and post-change distributions are known, one can apply the cumulative sum (CUSUM) detection rule, which has been proved to be optimal (Lorden, 1971; Moustakides, 1986). In practice, the post-change distribution is typically unknown, in which case one can sequentially estimate it and construct a generalized likelihood ratio test (Lai, 1998). However, a key limitation with these traditional methods is that they may perform poorly for *high-dimensional* data with *complex* low-dimensional structure. This is because the distributions in such setting would typically be difficult to approximate well using parametric families, and the required density ratio would thus be difficult to estimate as well.

This dissertation focuses on the development of new methods for online change point and anomaly detection in high-dimensional data settings. Each chapter introduces a novel method tailored to specific challenges and data structures, reinforcing the overarching goal of enhancing the accuracy and efficiency of detecting changes and anomalies.

In Chapter 2, we propose a new method, called PERsistence diagram-based ChangE-PoinT detection (PERCEPT), which leverages the learned topological structure from Topological Data Analysis (TDA) to sequentially detect changes. TDA provides a set of data analysis tools for extracting embedded topological structures from complex high-dimensional

datasets. In recent years, TDA has been a rapidly growing field which has found success in a wide range of applications, including signal processing, neuroscience and network analysis. In these applications, the online detection of changes is of crucial importance, but this can be highly challenging since such changes often occur in low-dimensional embeddings within high-dimensional data streams. The proposed method PERCEPT follows two key steps: it first learns the embedded topology as a point cloud via persistence diagrams, then applies a non-parametric monitoring approach for detecting changes in the resulting point cloud distributions. This yields a non-parametric, topology-aware framework which can efficiently detect online geometric changes. This method exemplifies the application of advanced topological constructs in real-time monitoring, such as in solar flare activity and human gesture recognition.

In Chapter 3, the focus shifts to the anomaly detection of data matrices. Robust Principal Component Analysis (RPCA) is a widely used method for recovering low-rank structure from data matrices corrupted by significant and sparse outliers. These corruptions may arise from occlusions, malicious tampering, or other causes for anomalies, and the joint identification of such corruptions with low-rank background is critical for process monitoring and diagnosis. However, existing RPCA methods and their extensions largely do not account for the underlying probabilistic distribution for the data matrices, which in many applications are known and can be highly non-Gaussian. We thus propose a new method called Robust Principal Component Analysis for Exponential Family distributions (e^{RPCA}), which can perform the desired decomposition into low-rank and sparse matrices when such a distribution falls within the exponential family. We present a novel alternating direction method of multiplier optimization algorithm for efficient e^{RPCA} decomposition, under either its natural or canonical parametrization. The effectiveness of e^{RPCA} is then demonstrated in two applications: the first for steel sheet defect detection, and the second for crime activity monitoring in the Atlanta metropolitan area.

In Chapter 4, we propose a novel Bayesian onLine Structure-Aware change deTection (BLAST) procedure, which leverages an underlying deep Gaussian Markov random field

(GMRF) to capture prior knowledge on expected image structures. Using this elicited deep GMRF prior, BLAST employs a carefully-constructed online Bayesian cumulative sum (CUSUM) procedure for scalable image monitoring. In particular, at each time-step, its posterior run length distribution (the key ingredient for Bayesian CUSUM monitoring) can be computed in $\mathcal{O}(M^2)$, where M is the number of image pixels. We demonstrate the effectiveness of BLAST compared to the state-of-the-art in a suite of numerical experiments and in two applications, the first on street scene monitoring and the second on drift tube monitoring for particle physics experiments.

Together, these chapters contribute to the broader field of online change point and anomaly detection by presenting novel methods that handle the complexities of high-dimensional and evolving data streams. The proposed method enhances our ability to detect subtle yet significant changes, offering robustness against noise and adaptability to different data structures. This dissertation demonstrates practical implementations, showcasing the potential of these methodologies in real-world scenarios. The collective insights from these studies underscore the importance of developing specialized, feature-aware techniques for monitoring and analysis in the age of information.

2. PERCEPT: a new online change-point detection method using topological data analysis

2.1 Introduction

Topological Data Analysis (TDA) is a thriving field at the intersection of statistics, machine learning, and algebraic topology, which has gained traction in recent years. TDA methods provide a set of tools for studying the shapes of complex high-dimensional datasets, by extracting its underlying low-dimensional geometric structures. This field of study has found success in a wide range of applications, including signal processing (Perea & Harer, 2015), computational biology (Cang et al., 2015), time series analysis (Seversky et al., 2016), and neuroscience (Sizemore et al., 2019).

Despite promising developments in recent years, there has been little work on integrating topological structure for sequential change-point detection, which is a fundamental problem in many of the aforementioned applications. Change-point detection here refers to the detection of a possible change in signal distribution over time. Traditional change-point detection methods largely focus on likelihood ratio tests, which presume that observations are independently and identically distributed, both before and after the change (Basseville & Nikiforov, 1993; Poor & Hadjiliadis, 2008; Siegmund, 1985; Tartakovsky et al., 2015). When the pre- and post-change distributions are known, one can apply the cumulative sum (CUSUM) detection rule, which has been proved to be optimal (Lorden, 1971; Moustakides, 1986). In practice, the post-change distribution is typically unknown, in which case one can sequentially estimate it and construct a generalized likelihood ratio test (Lai, 1998).

However, a key limitation with these traditional methods is that they may perform poorly for *high-dimensional* data with *complex* low-dimensional structure. This is because the distributions in such setting would typically be difficult to approximate well using parametric families, and the required density ratio would thus be difficult to estimate as well. One solution in this high-dimensional setting is to first extract the underlying low-dimensional structure from data, and then construct detection statistics based on the

extracted information. CUSUM-type detection algorithms were proposed in (L. Xie et al., 2020) and (Jiao et al., 2018) to detect changes characterized by an unknown subspace structure in the covariance matrix. These methods work well when the true low-dimensional embedding is precisely the embedding used in the detection procedure, i.e., a linear subspace. In complex problems where the true embedding is nonlinear, however, this model misspecification may result in considerable deterioration in detection performance (Molloy & Ford, 2017). There is thus a need for a high-dimensional change detection method which can integrate a broader yet realistic framework for modeling low-dimensional structure.

In recent years, the rise of TDA methods suggests that, for many problems, the underlying data may have embedded *topological* structure. Indeed, in such problems, the extracted topology from TDA often captures intuitive features from high-dimensional data which are interpretable for the practitioner. For example, in computer vision, topological features can represent segmented regions of a 3D shape (Beksi & Papanikolopoulos, 2016). Similarly, for time series, periodic signals can be captured by topological features via Taken's Embedding (Seversky et al., 2016). However, the integration of such structure for change detection has largely remained unexplored. A recent work (Islambekov et al., 2019) proposed a TDA detection approach for time series data, by converting this to a sequence of so-called Betti numbers (Edelsbrunner & Harer, 2008), which capture the number of k-dimensional holes on a topological surface. However, Betti numbers are known to provide a weak summary of topology (Ghrist, 2008) and thus may be insensitive to certain topological changes. Another recent work (Ofori-Boateng et al., 2021) utilized a richer topological summary called a *persistence diagram* (Edelsbrunner & Harer, 2008), which uniquely captures the topological features of the data at different spatial resolutions (more on this in Section 2.2.1). With persistence diagrams computed at each point in time, the method then makes use of the Wasserstein distance between diagrams from adjacent times as the detection test statistic. However, there are two limitations with this approach: (i) it relies on the Wasserstein distance, which may not fully capture the topological differences between two persistence diagrams (more on this later), and (ii) its test statistic relies on information from only the

immediate past, which can greatly reduce detection efficiency.

To address this, we propose a new *non-parametric*, *topology-aware* framework called PERsistence diagram-based ChangE-PoinT detection (PERCEPT). As in (Ofori-Boateng et al., 2021), PERCEPT leverages the extracted persistence diagrams (which capture topological features of the data at each time) for change detection. However, instead of using the Wasserstein distance of diagrams from adjacent times, PERCEPT extends a recent non-parametric change detection method (L. Xie & Xie, 2021) to detect changes directly on the diagram point clouds. This yields two advantages: (i) it offers a distribution-based approach which amplifies changes in topological features; and (ii) its test statistic makes use of data within a past time window, which addresses information loss. We demonstrate the effectiveness of PERCEPT over existing methods in a suite of simulation experiments and in applications to solar flare monitoring and human gesture detection.

The rest of the paper is organized as follows. Section 2.2 provides preliminaries on persistent homology and motivations. Section 2.3 outlines the PERCEPT methodology. Section 2.4 demonstrates the effectiveness of this method in numerical experiments. Section 2.5 applies the method to solar flare monitoring and human gesture detection.

2.2 Preliminaries and Motivation

We first provide a brief overview of TDA, then discuss two baseline methods, the Hotelling's T^2 statistic and the Wasserstein distance approach in (Ofori-Boateng et al., 2021), for high-dimensional change-point detection. We then motivate the proposed PERCEPT method via our motivating solar flare detection application.

2.2.1 TDA Preliminaries

A primary tool in TDA is *persistent homology*, which extracts topological features (e.g., connected components, holes, and their higher-dimensional analogues) from point cloud data. In what follows, we provide a brief overview of persistent homology, following (Ghrist, 2008) and (Edelsbrunner & Harer, 2008). For a given point cloud dataset, persistent homology provides a representation of this as a *simplicial complex*, defined as a set of vertices,

edges, triangles, and their higher-dimensional counterparts. A common simplicial complex is the *Rips complex*, which depends on a single scale parameter ϵ. At a given radius $\epsilon > 0$, the Rips complex contains all edges between any two points whose distance is at most ϵ, and contains triangular faces for any three points whose pairwise distances are at most ϵ. Figure 2.1 (adapted from (Han et al., 2018)) illustrates this for a toy dataset. Clearly, a single ϵ cannot capture all geometric structures embedded in the data. Thus, a sequence of scale parameters is used to build a sequence (or *filtration*) of simplicial complexes. This filtration provides a means for extracting topological structure from the data: zero-dimensional holes (or connected components), one-dimensional holes, and their higher-dimensional analogues.

Under this framework, suppose a topological feature appears in the filtration at some radius ϵ and disappears at a larger radius $\epsilon' > \epsilon$. The pair (ϵ, ϵ') gives the *persistence* of the feature, with ϵ and ϵ' being its *birth* and *death*, respectively. A prominent topological feature in the point cloud data would have long persistence, whereas a small or noisy topological feature would have short persistence. The persistence information from all topological features can be captured by an (untilted) *persistence diagram* (PD), a collection of points in $\mathbb{R}^2$ where each feature is represented by a point (ϵ, ϵ'), with ϵ is its birth time and ϵ' is its death time. Figure 2.1 illustrates the persistent homology pipeline from point cloud data to a persistence diagram. We will distinguish this untilted PD from its tilted variant later.

We use a simple example to illustrate this pipeline from topological structures to a PD. Figure 2.2 (left) shows the point cloud data generated from two disjoint circles with radii 1, and Figure 2.2 (right) shows the corresponding PD of this data. For the 0-D homology, which captures connected components (black points in Figure 2.2 right), we observe many points with small persistence, but one point which persists for a long time. This last point with large persistence suggests the data has two connected components, which is indeed true. For the 1-D homology, which captures one-dimensional holes (red points in Figure 2.2 right), we observe two red points with large persistence, which reflects the two holes within the circles. The choice of 0-D or 1-D homology for change detection will depend on prior

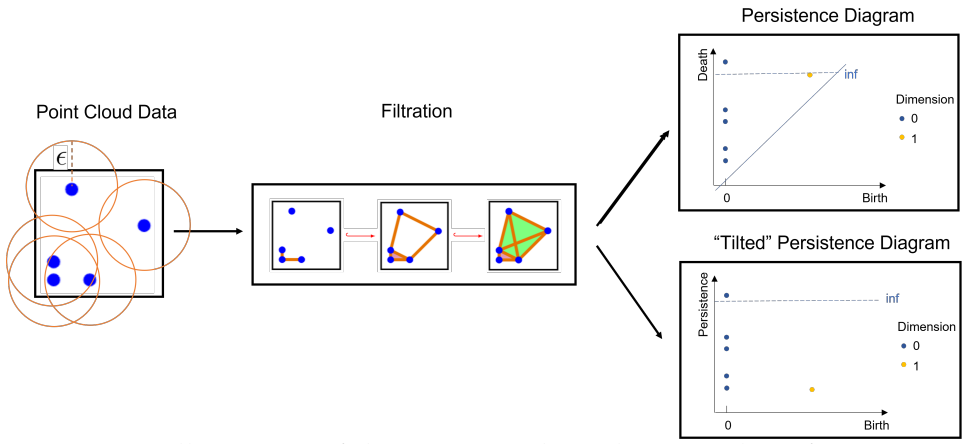


FIGURE 2.1: An illustration of the persistent homology pipeline, from point cloud data to a filtration of simplicial complexes to a persistence diagram. The Rips complex with radius ϵ in the left plot corresponds to the second simplicial complex in the filtration.

knowledge of the data, and will be discussed later in numerical experiments.

From a dimension reduction perspective, one might wonder how the low-dimensional (namely, 2-D) nature of persistence diagrams can serve as an effective feature extraction tool. One reason is that, when such topology is present in data, the embedded geometric structure is typically low-dimensional enough to be captured on the 2-D persistence diagrams, even when the data is high-dimensional. This *targeted* extraction of *topological* features thus allows for a reduced dimension of the feature space over other dimension reduction approaches. See Chazal, 2016 for further discussion, with high-dimensional applications in signal processing (Perea & Harer, 2015) and neuroscience (Sizemore et al., 2019).

One potential limitation of PDs for change detection, however, is its inability to distinguish translation shifts in data. One can show that for any point cloud data, its PD representation remains the same after any translation shifts. This is not necessarily a drawback for persistent homology, which aims to extract *geometric* structure rather than translation shifts. For problems both geometric changes and translation shifts are important to detect, we propose later a simple modification which addresses this in Section 2.3.2.

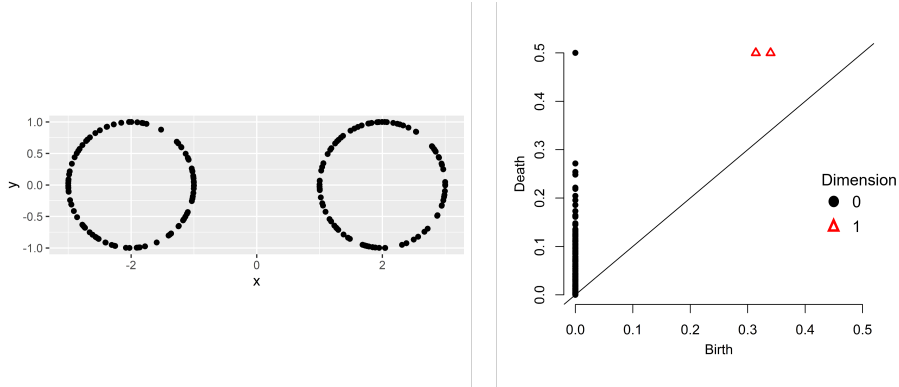


FIGURE 2.2: (left) Point cloud data generated from two disjoint circles, and (right) its corresponding persistence diagram for 0-D and 1-D holes.

2.2.2 Existing Baseline Methods

A classic baseline change detection approach for multivariate data is the Hotelling's T^2 statistic (Hotelling, 1947), which makes use of the mean and covariance structure of data. Given data $\boldsymbol{x}_t \in \mathbb{R}^p$ at time t, $t = 1, \cdots, T$, the Hotelling's T^2 statistic is defined as

$$(\bar{\boldsymbol{x}} - \boldsymbol{\mu}_0)^\intercal \Sigma_0^{-1} (\bar{\boldsymbol{x}} - \boldsymbol{\mu}_0), \tag{2.1}$$

where $\bar{\boldsymbol{x}}$ is the sample mean vector, $\boldsymbol{\mu}_0$ and Σ_0 is the nominal mean vector and covariance matrix, respectively (this is typically given or estimated from reference data). The vanilla Hotelling's T^2 statistic is calculated using *only* data at the current time t, with all historical data discarded. To compute the test statistic more efficiently, it can be coupled with the CUSUM scheme (Page, 1954), which makes use of a cumulative sum of the statistic over time. The resulting detection statistic S_t^H is then given by:

$$S_t^H = (S_{t-1}^H)^+ + (\bar{\boldsymbol{\mu}}_{t-w,t} - \hat{\boldsymbol{\mu}}_0)^\intercal \widehat{\Sigma}_0^{-1} (\bar{\boldsymbol{\mu}}_{t-w,t} - \hat{\boldsymbol{\mu}}_0) - d^H, \quad S_0^H = 0, \tag{2.2}$$

where $(x)^+ = \max(x, 0)$, $\bar{\boldsymbol{\mu}}_{t-w,t}$ denotes the sample average of the data vectors $\{\boldsymbol{x}_{t-w}, \ldots, \boldsymbol{x}_t\}$, and $\hat{\boldsymbol{\mu}}_0$ and $\widehat{\Sigma}_0$ are the *pre-change* mean vector and covariance matrix estimated from historical data. Here, d^H is a drift parameter that can also be estimated using historical data. When the data is known to be concentrated on a linear subspace, one can adapt the Hotelling's T^2 test by first performing Principal Component Analysis (PCA), then using the resulting principal components as data within equation (2.2). With this, a change-point

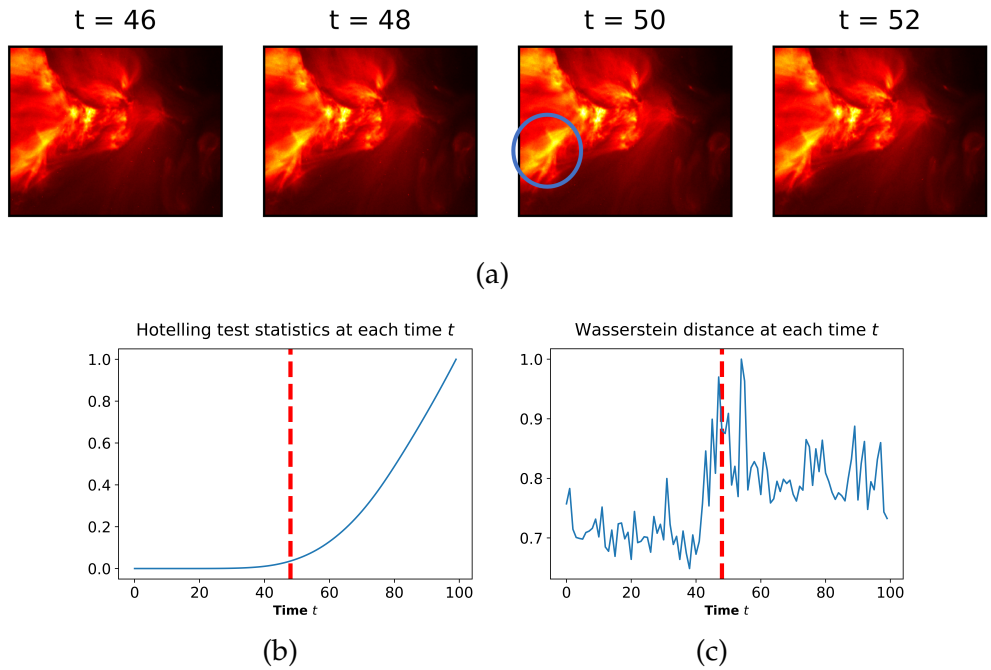


FIGURE 2.3: (a) Snapshots of the solar flare around the true change-point $t^* = 49$. (b,c) Hotelling's T^2 and Wasserstein distance statistics at each time t; the true change-point t^* is marked in red.

is then declared at time t when the statistic S_t^H first exceeds a pre-specified threshold b.

The second baseline method, the Wasserstein distance approach in (Ofori-Boateng et al., 2021), integrates topology in the following way. The Wasserstein distance (of order 1) between two distributions P and Q on sample space Ω is defined as:

$$\mathcal{W}_1(P,Q) := \min_{\gamma \in \Gamma(P,Q)} \left\{ \mathbb{E}_{(\omega,\omega')\sim\gamma} \left[c(\omega,\omega') \right] \right\}.$$

Here, $c(\cdot,\cdot)$ is a metric on Ω (we use the Euclidean norm), and $\Gamma(P,Q)$ denotes the collection of all (Borel) probability measures on $\Omega \times \Omega$ with marginal distributions P and Q. With this, the Wasserstein distance method is straight-forward. First, at each time t, the persistence diagram $\mathcal{D}_t \in \mathbb{R}^2$ is computed from data $\boldsymbol{x}_t$. Next, the Wasserstein distances $S_t^W = \mathcal{W}_1(\mathcal{D}_t, \mathcal{D}_{t+1})$ are computed between the PDs from adjacent time frames. One then declares a change-point at time t when the statistic S_t^W exceeds a pre-specified threshold b.

We investigate these two baseline methods for a motivating solar flare monitoring ap-

plication (more on this in Section 2.5). The data consists of image snapshots (232×292 pixels) of a solar flare, captured by the Solar Dynamics Observatory[1] at NASA. Figure 2.3a shows several snapshots before and after the true change-point at $t^* = 49$, where the flare bursts become more pronounced and bright. However, such a change can be quite subtle visually, thus making monitoring a challenging task (Y. Xie et al., 2012). We applied the above two baseline methods, with the Hotelling's T^2 applied by first vectorizing the image data, then applying PCA to extract the top 15 principal components. Figures 2.3 (b) and (c) show the detection statistics as a function of time t. We see that both methods experience a large detection delay from the true change-point at $t^* = 49$. For the Hotelling's T^2, the test statistic increases slowly after the change-point, which suggests it does not capture well the changed image features. Similarly, the Wasserstein distance statistic appears highly unstable and unable to detect the change-point. Given the limitations of the two baseline methods, we introduce next a new non-parametric, topology-aware method which provides a new framework for integrating low-dimensional geometric information for online monitoring.

2.3 Persistence Diagram-based Change Detection

We now introduce the proposed PERsistence diagram-based ChangE-PoinT (PERCEPT) monitoring method. We first describe the histogram binning for PDs, then show how the extracted histograms can be used for change detection. We then present useful insights for PERCEPT, and discuss methodological developments for bin and weight optimization. Figure 2.4 visualizes the workflow for PERCEPT; we will describe each step in detail below.

2.3.1 Persistence Histogram Binning

The first step in PERCEPT is to construct the so-called *persistence histograms*, a novel histogram representation which captures persistence information from the computed PDs. This histogram binning serves two purposes: it provides a robust way for filtering noise

[1] See https://sdo.gsfc.nasa.gov/assets/docs/SDO_Guide.pdf.

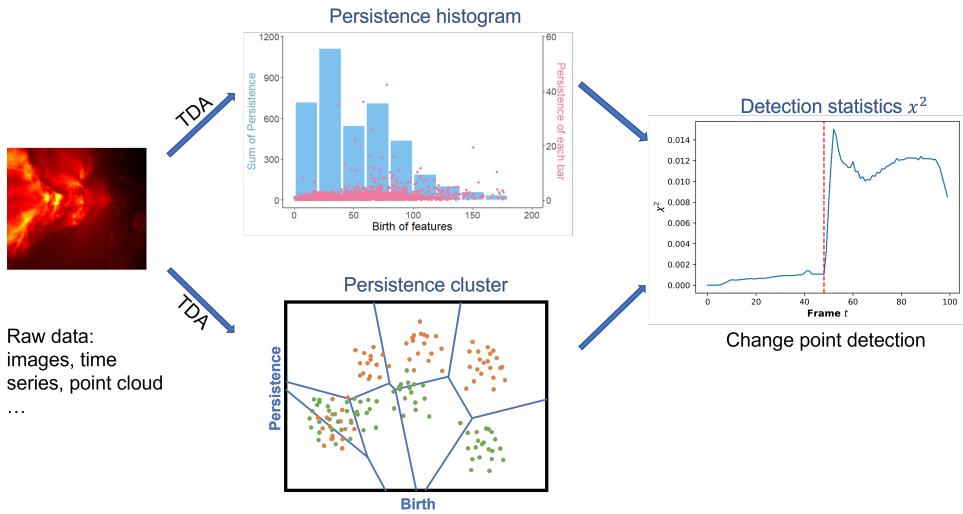


FIGURE 2.4: Visualizing the general workflow for PERCEPT. The data can take a variety of forms, including images, time series or point cloud data. Persistence homology is used to extract useful geometric features from data to persistence diagrams (Section 2.2.1). These diagrams are then binned using either persistence histograms or persistence clusters (Sections 2.3.1 and 2.3.4, respectively), then used within an ℓ_2 test statistic for monitoring (Section 2.3.2).

in the PD data, and allows us to leverage recent developments in histogram-based change detection methods within PERCEPT (shown next). To facilitate this, we assume all PDs are given in their *tilted* representation, where a feature is represented by a point $(\epsilon, \epsilon' - \epsilon)$, with ϵ is its birth time and $\epsilon' - \epsilon$ is its persistence time. Figure 2.1 provides an illustration.

Assume, as before, that the PDs $\mathcal{D}_t \in \mathbb{R}^2$ are computed for the data $\boldsymbol{x}_t$ at each time $t = 1, \cdots, T$. Further assume that the birth range for the PDs (i.e., the x-axis on $\mathcal{D}_t$) is partitioned into the L bins: $[0, b_1)$, $[b_1, b_2)$, …, $[b_{L-1}, b_L)$, where b_l is the right break-point for the l-th bin. With this, we can now bin the point cloud data $\mathcal{D}_t$. Let $f_{t,l}$ be the sum of persistences for points in $\mathcal{D}_t$ within the l-th bin, i.e.:

$$f_{t,l} = \sum_{(u,v) \in \mathcal{D}_t, u \in [b_{l-1}, b_l)} v, \tag{2.3}$$

where $v = \epsilon' - \epsilon$ is the persistence time of point (u, v), and let $\omega_{t,l} = f_{t,l} / \sum_{l'=1}^{L} f_{t,l'}$ be its corresponding proportion. The histogram frequencies and distribution of the PD $\mathcal{D}_t$ can thus be represented by the vectors $\boldsymbol{f}_t = (f_{t,1}, \cdots, f_{t,L})$ and $\boldsymbol{\omega}_t = (\omega_{t,1}, \cdots, \omega_{t,L})$,

respectively. We will call $\boldsymbol{\omega}_t$ the *persistence histogram* of the PD $\mathcal{D}_t$. Figure 2.5 (a) visualizes this binning procedure. We will discuss the choice of L in Section 2.3.5.

In the absence of prior information on pre- and post-change data, the persistence histogram breakpoints $b_1, \cdots, b_L$ can be placed in such a way that the resulting histogram bins have equal widths. The same bins are then used throughout the monitoring procedure. When there is prior information on only the pre-change data, the breakpoints $b_1, \cdots, b_L$ can be chosen such that there is (roughly) an equal sum of persistences within each histogram bin. When prior information are available on both pre- or post-change data, one can employ more elaborate binning approaches; see Section 2.3.4 for further discussion.

As an illustration, consider again the solar flare images from Figure 2.3. Using the lower star filtration ((Bendich et al., 2011), discussed later in Section 2.5.1), we can obtain persistence diagrams for each flare image. Figure 2.5 (a) shows the binned persistence histogram (with $L = 10$ bins) of a PD computed from a pre-change flare image, and Figure 2.5 (b) shows the persistence histogram for a "post-change" image using the same bins. Given a significant change in topological structure, the corresponding pre- and post-change persistence histograms (which capture topology information) should be sufficiently different to capture this change. We will leverage this next to formulate our test statistic.

2.3.2 Test Statistic

The second step in PERCEPT is to use the extracted persistence histograms to construct a monitoring test statistic. The idea is as follows. Suppose the bins $[0, b_1), [b_1, b_2), \cdots, [b_{L-1}, b_L)$ are fixed. Then, at each time t, one can treat the observed persistence histograms $\boldsymbol{f}_t$ as *data* sampled from an underlying discrete distribution with L levels. Let p_{pre} denote this so-called *persistence distribution* (with L levels) prior to the change, and p_{post} be the persistence distribution after the change. The goal of detecting topological changes can then be thought of as testing for differences between the persistence distributions p_{pre} and p_{post}. We thus desire a test which investigates the following hypotheses:

$$H_0 : p_{\text{pre}} = p_{\text{post}}, \quad H_A : p_{\text{pre}} \neq p_{\text{post}}. \tag{2.4}$$

To test (2.4), we extend a recent non-parametric test in (L. Xie & Xie, 2021) for detecting changes on discrete distributions. At the current time t, we search for all possible change points at time $k < t$ within a fixed-sized window. To investigate whether time k is a change point, we will consider four consecutive time intervals (see Figure 2.5b): the first two intervals are immediately before time k, and the last two intervals are immediately after k. All four intervals have the same time length of $\lfloor (t-k)/2 \rfloor$. We call the first group of intervals (representing potential *pre*-change times) as "Group 1", and the latter group (representing potential *post*-change times) as "Group 2".

Suppose the histogram breakpoints $b_1, \cdots, b_L$ are given. Consider now the persistence histogram proportions within the two time intervals in Group 1 and the two intervals in Group 2; we denote this as $\boldsymbol{\omega}_{t,k}^{[1,1]}$, $\boldsymbol{\omega}_{t,k}^{[1,2]}$, $\boldsymbol{\omega}_{t,k}^{[2,1]}$ and $\boldsymbol{\omega}_{t,k}^{[2,2]}$, respectively. This non-parametric weighted-ℓ_2 statistic can then be defined as

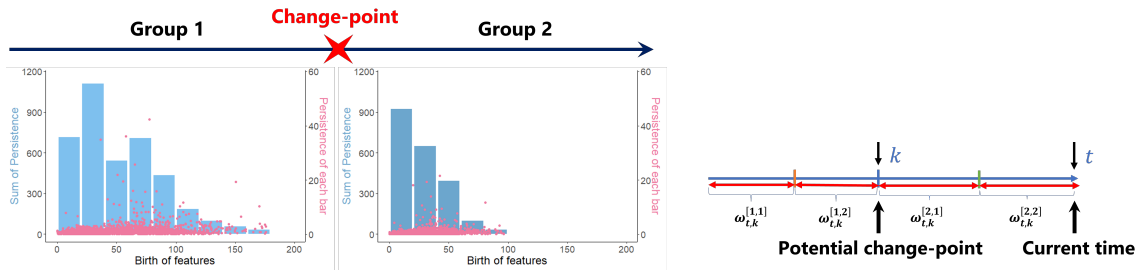
$$\chi_{t,k} = (\boldsymbol{\omega}_{t,k}^{[1,1]} - \boldsymbol{\omega}_{t,k}^{[2,1]})^{\intercal} \Sigma (\boldsymbol{\omega}_{t,k}^{[1,2]} - \boldsymbol{\omega}_{t,k}^{[2,2]}). \tag{2.5}$$

Here, $\Sigma = \mathrm{diag}\{\sigma_1 . \cdots, \sigma_L\}$ is the diagonal weight matrix, where $\sigma_l \geq 0$, $l = 1, \cdots, L$. Note that, if time k were indeed a change point, the resulting test statistic $\chi_{t,k}$ would likely be large, thus providing evidence for a change. In the absence of prior information on pre- and post-change data, an intuitive choice of Σ is the identity matrix, which assumes equal weights over persistence histogram bins. However, when some prior data are available on either the pre- or post-change PDs, the weights in Σ can be optimized to better highlight distributional differences. Section 2.3.5 provides further discussion on this.

Using the statistic (2.5), an online detection procedure is then given by the stopping time:

$$\mathcal{T} = \inf\{t : \chi_t^{\max} \geq b\}, \quad \chi_t^{\max} = \max_{0 \leq k \leq t} \chi_{k,t}, \tag{2.6}$$

where b is a pre-specified threshold parameter. Here, $\mathcal{T}$ is the time at which the procedure raises an alarm indicating a change-point has occurred before time t. In other words, when the test statistic $\chi_t^{\max}$ exceeds a certain threshold, a change is declared indicating the persistence histograms before and after time k are sufficiently different.



(a) (b)

FIGURE 2.5: (a) Persistence histograms for a pre-change and post-change solar flare image. (b) Visualizing the intervals used for the weighted ℓ_2 detection statistic.

The threshold b in (2.6) can be set by controlling the false alarm rate to a desired pre-specified level, or, equivalently, by controlling the average run length (ARL, more on this later) to be above a pre-specified level. For the above weighted-l_2 statistic, theoretical approximations are available for the ARL as a function of threshold b (L. Xie & Xie, 2021):

$$\mathbb{E}_\infty[\mathcal{T}] = \frac{1}{2} \frac{b^{-1} e^{b^2/(2\sigma_p^2)} [2\pi\sigma_p^2]^{1/2}}{\int_{[4b^2/(m_1\sigma_p^2)]^{1/2}}^{[4b^2/(m_0\sigma_p^2)]^{1/2}} y\nu^2(y)dy} (1 + o(1)).$$

Here, $\sigma_p^2 = 4[\sum_{l=1}^{L} \sigma_i^2 p_i^2 (1-p_i)^2 + \sum_{i \neq j} \sigma_i \sigma_j p_i^2 p_j^2]$, $p = p_{\text{pre}}$ is the pre-change distribution, m_0 and m_1 are two known constants, and $\nu(\cdot)$ is a special function that is easy to calculate. Further details can be found in (L. Xie & Xie, 2021). With this, one can set the threshold b to bound the ARL $\mathbb{E}_\infty[\mathcal{T}]$, which then controls the false alarm rate. Furthermore, for computational efficiency, one may adopt a window-limited procedure which considers only $k \in [t - m_1, t - m_0]$ in (2.6), instead of searching over all possible change-points $k < t$. Typically, m_1 is set to be larger than the desired detection delay, and m_0 is set to ensure the reliable estimation of the persistence histogram distribution. In later experiments, we set $m_0 = 20$ and $m_1 = 80$, which seems to provide good performance.

As discussed earlier, one potential limitation of PDs for change detection is that it might not capture translation shifts of the data. For problems where both geometric changes and translation shifts are of interest, an easy fix is as follows. One can perform the proposed PERCEPT procedure (2.6) (which can detect geometric changes) in parallel with standard

procedures which are adept at mean-shift detection, such as Shewhart control charts (Shewhart, 1925), exponential weighting mean average charts (Roberts, 1959), CUSUM test (Lau et al., 2018; Page, 1954), etc. The joint monitoring of these two statistics will enable the joint tracking of both geometric changes and mean shifts.

2.3.3 Connection between EDD and Topology

We now provide a useful connection between the expected detection delay (EDD) and topology, which sheds light on how PERCEPT may be useful for topological change detection. Consider two fundamental metrics in online change-point detection: EDD and Average Run Length (ARL). Let $\mathbb{E}_\infty$ denote the expectation under the probability measure when there is no change-point, and let $\mathbb{E}_0$ denote the expectation under the probability measure when the change happens at time 0. For a given stopping time $\mathcal{T}$ of a monitoring procedure, its ARL is defined as $\mathbb{E}_\infty[\mathcal{T}]$, the expected run length to false alarm when there is no change, and the EDD is defined as $\mathbb{E}_0[\mathcal{T}]$, the number of samples needed to detect the change. Theoretically, the EDD is known to be linearly related to the $\log(\text{ARL})$ (Tartakovsky et al., 2015).

Next, we introduce the bottleneck distance, a standard metric for topological distance (Ghrist, 2008). Suppose, for two point cloud datasets with different topologies, one computes its (untilted) PDs $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively. The extracted topological features in these PDs can then be compared via a *matching* η. This matching is performed in two steps: (i) it pairs each point in the first PD $\mathcal{D}_1$ with a point in the second PD $\mathcal{D}_2$ or a point on the diagonal line, and (ii) it pairs each point in $\mathcal{D}_2$ with a point in $\mathcal{D}_1$ or a point on the diagonal. The *bottleneck distance* (Ghrist, 2008) between the PDs $\mathcal{D}_1$ and $\mathcal{D}_2$ is then defined as:

$$d_B(\mathcal{D}_1, \mathcal{D}_2) = \inf_{\eta} \sup_{\boldsymbol{y} \in \mathcal{D}_1} ||\boldsymbol{y} - \eta(\boldsymbol{y})||_\infty.$$

Here, the supremum is taken over all matched points in $\mathcal{D}_1$, and the infimum is taken over all possible matchings η. The bottleneck distance can also be viewed as the Wasserstein distance of order ∞. Clearly, a larger bottleneck distance indicates the extracted features

from the first PD are quite different from that for the second PD (and vice versa). This then suggests the topology for the first dataset is markedly different from that for the second dataset. This link between the bottleneck distance and topological differences is formalized by the Stability Theorem (Cohen-Steiner et al., 2007), a key theorem in TDA.

With this, the EDD of the proposed PERCEPT method can then be linked to the bottleneck distance of the topologies of the pre- and post-change data. Recall that PERCEPT makes use of the non-parametric l_2-statistics on the underlying persistence diagrams. It is known (L. Xie & Xie, 2021) that the EDD for such a procedure can be upper bounded by:

$$\text{EDD} \leq \frac{2b}{(\min_i \Sigma_{ii})||p_{\text{pre}} - p_{\text{post}}||_2^2}, \tag{2.7}$$

where b is the pre-specified detection threshold and Σ_{ii} is the i-th diagonal entry of Σ. In other words, the larger the ℓ_2-difference between the pre- and post-change persistence distributions p_{pre} and p_{post}, the smaller the EDD for PERCEPT. We can then show (technical details in Appendix A.3) that, under certain asymptotic approximations, $||p_{\text{pre}} - p_{\text{post}}||_2^2$ can be lower bounded by the bottleneck distance between a pre-change PD $\mathcal{D}_{\text{pre}}$ and post-change PD $\mathcal{D}_{\text{post}}$, which quantifies the change in topology. This, together with (2.7), suggests that, for PERCEPT, *the greater (or smaller) the topological difference is between pre- and post-change data, the smaller (or greater) its expected detection delay tends to be*, which is as desired.

2.3.4 Persistence Cluster Binning

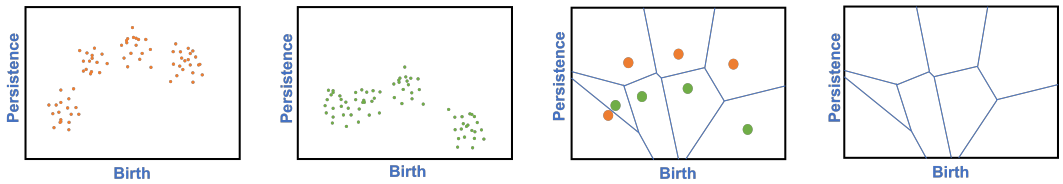
The persistence histogram binning approach in Section 2.3.1 can be viewed as partitioning the persistence space into vertical *rectangular* regions, which are then used to bin the PDs for monitoring. This may have two limitations. First, since persistences are summed within each bin, the procedure can distinguish topological features with different birth times, but not features with similar birth times but different persistences. Second, the restriction of partitions to be vertical and rectangular may hamper the identification of regions of greatest change between the pre- and post-change distributions. When training

data are available on either the pre- or post-change PDs, we present an alternate novel *persistence clustering* approach which can yield a more informed partition of the persistence space.

Consider first the case where training data are available for both pre- and post-change regimes. The idea is to find a clustering of these point clouds, so that the corresponding partition of the persistence space can discriminate well the pre- and post-change distributions. We construct these *persistence clusters* as follows. First, we perform k-means clustering (Lloyd, 1982) on the pre-change PD point clouds, which returns C_{pre} cluster centers $\mathcal{C}_{\text{pre}} = \{\boldsymbol{c}_1, \cdots, \boldsymbol{c}_{C_{\text{pre}}}\}$. Then we do the same on the post-change PDs to obtain C_{post} cluster centers $\mathcal{C}_{\text{post}} = \{\boldsymbol{c}'_1, \cdots, \boldsymbol{c}'_{C_{\text{post}}}\}$. Using the combined centers $\mathcal{C} = \mathcal{C}_{\text{pre}} \cup \mathcal{C}_{\text{post}}$, we then form a *Voronoi diagram* using centers $\mathcal{C}$ (Aurenhammer, 1991), i.e., a partition of $\mathbb{R}^2$ to its closest point in $\mathcal{C}$. Figure 2.6 visualizes this persistence clustering procedure. The number of clusters can be determined by the elbow method in k-means clustering (Ketchen & Shook, 1996).

Guided by prior knowledge, the modeler may choose to use alternate clustering approaches in lieu of k-means clustering, such as BIRCH (T. Zhang et al., 1996), DBSCAN (Ester et al., 1996), and spectral clustering (J. Shi & Malik, 2000). In particular, distribution-based clustering methods, e.g., DBSCAN or other recent methods (Krishna et al., 2019), may be able to capture distributional features on the PD space, thus leading to clusters which can better discriminate pre- and post-change distributions. We make use of the standard k-means clustering for persistence binning in later experiments, which turns out to perform quite well, but aim to explore more sophisticated clustering approaches as future research.

There may, of course, be cases where only pre-change (and not post-change) data are available, i.e., when the modeler has prior information on data topology before a change but not after. In such cases, one may perform k-means clustering (or other clustering approaches) on the pre-change PDs for persistence binning.



(a) (b)

FIGURE 2.6: (a) A sample pre-change (orange) and post-change (green) PD. (b) Pre- and post-change cluster centers and its corresponding persistence cluster via Voronoi diagrams.

With these clusters, we then employ the same weighted-ℓ_2 test statistic in (2.5). The only difference is that, instead of taking $\boldsymbol{\omega}_{t,k}^{[1,1]}$, $\boldsymbol{\omega}_{t,k}^{[1,2]}$, $\boldsymbol{\omega}_{t,k}^{[2,1]}$ and $\boldsymbol{\omega}_{t,k}^{[2,2]}$ as persistence histogram proportions, we take these as the proportions over persistence clusters. As Figure 2.6 (b) shows, these persistence clusters may yield improved discrimination between the pre- and post-change distributions compared to the earlier persistence histograms, particularly when there is a large amount of points (or features) captured in the persistence diagrams. When there is only a small number of features, however, we would recommend the use of the persistence histogram approach instead, since there may be insufficient data to fit the more complex persistence clusters. We will demonstrate this later in numerical experiments.

2.3.5 Weight Optimization

Similarly, given available training data on both the pre-change and post-change PDs, the weights $\Sigma = \text{diag}\{\sigma_1, \cdots, \sigma_L\}$ in the test statistic (2.5) can also be carefully specified to amplify differences between the pre- and post-change distributions. For this, we adopt the weight optimization approach in (L. Xie & Xie, 2021), which aims to find weights Σ to maximize the worst-case weighted ℓ_2 distance. This can be formulated as:

$$\max_{\sigma \geq 0, g(\sigma) \leq 1} f(\sigma), \quad g(\sigma) := \max_{p_{\text{pre}} \in \Delta} \sum_i \sigma_i^2 p_{\text{pre},i}^2,$$
$$f(\sigma) := \min_{p_{\text{pre}}, p_{\text{post}}} \left\{ \sum_i \sigma_i (p_{\text{pre},i} - p_{\text{post},i})^2 : p_{\text{pre}}, p_{\text{post}} \in \Delta, ||p_{\text{pre}} - p_{\text{post}}||_2 \geq \rho \right\}. \tag{2.8}$$

Here, $p_{\text{pre,i}}$ and $p_{\text{post,i}}$ denote the i-th entry in the vector p_{pre} and p_{post}, respectively. The minimization in $f(\sigma)$ is taken over all possible pre- and post-change distributions p_{pre} and p_{post} (within the probability simplex Δ) that are ρ-separable for a given $\rho > 0$. The choice of ρ depends on the scale of change that one wants to detect; in our experiments later, ρ is set to be 0.1. For persistence histograms, the optimal number of bins L can be optimized simultaneously by selecting L which yields the highest $f(\sigma)$ in (2.8). In practice, the training pre- and post-change data are required for estimating the persistence distributions p_{pre} and p_{post}. Further details on this optimization can be found in (L. Xie & Xie, 2021). These optimized weights can then be used within the test statistic (2.5) for change detection.

In implementation, we have found that a small modification of (2.8) can yield noticeably improved performance. Note that when $p_{\text{pre,i}}/p_{\text{post,i}}$ or $p_{\text{post,i}}/p_{\text{pre,i}}$ is exceedingly larger from 1, and $|p_{\text{pre,i}} - p_{\text{post,i}}|$ is small for some i, it may be difficult to pick out bin i as an important bin using the above formulation. However, such a bin i distinguishes the pre- and post-change distributions because their relative difference is large. Thus, we see that this optimization is distribution-dependent; when p_{pre} and p_{post} are extremely different from the uniform distribution, the above weights may not be ideal for distribution discrimination. In practice, we suggest using the *relative* difference of p_{pre} and p_{post} instead of the *absolute* difference in the above formulation, i.e., minimizing $\sum_i \sigma_i \{(p_{\text{pre,i}} - p_{\text{post,i}})/p_{\text{post,i}}\}^2$ instead of $\sum_i \sigma_i (p_{\text{pre,i}} - p_{\text{post,i}})^2$ for $f(\sigma)$ in (2.8).

As before, there may be scenarios where only pre-change data are available. In such a setting, one could use the weights which minimize the best-case weighted ℓ_2 distance over just the pre-change distributions. Specifically, we can divide the pre-change data into

two groups, with binned distributions denoted as p_{pre_1} and p_{pre_2}. We then replace $p_{\text{pre,i}}$ and $p_{\text{post,i}}$ with $p_{\text{pre}_1,\text{i}}$ and $p_{\text{pre}_2,\text{i}}$, take the maximization in $f(\sigma)$, and choose weights which minimize $f(\sigma)$ in (2.8). With such a choice, the detection statistics are expected to be small prior to the change-point, and increase after the change-point.

2.4 Simulation Experiments

We now explore the performance of PERCEPT in a suite of simulation studies where the data is generated with topological structure. We investigate several change scenarios, including topology changes, noise changes, and its scalability in higher dimensions.

Three baseline methods are used here for comparison. The first is the aforementioned parametric Hotelling's T^2 test, using the 15 extracted principal components from PCA on the original data. The second is the Wasserstein distance method proposed in (Ofori-Boateng et al., 2021), which makes use of the Wasserstein distance between PDs in adjacent times. Details on both methods can be found in Section 2.2.2. The third method is the maximum mean discrepancy (MMD) test (Gretton et al., 2012; S. Li et al., 2015), a widely used non-parametric change detection method. Given a class of functions $\mathcal{F}$ and two distributions p and q, the MMD distance between p and q is defined as $\text{MMD}_{\mathcal{F}}(p,q) = \sup_{f\in\mathcal{F}}(\mathbb{E}_{x\sim p}[f(x)] - \mathbb{E}_{y\sim q}[f(y)])$. When $\mathcal{F}$ is a reproducing kernel Hilbert space (RKHS) associated with kernel function $K(\cdot,\cdot)$, this MMD statistic can be written as:

$$S_t^M = \frac{1}{n_{\text{pre}}^2}\sum_{i,i'=1}^{n_{\text{pre}}} K(\boldsymbol{x}_i,\boldsymbol{x}_{i'}) + \frac{1}{n_{\text{post}}^2}\sum_{j,j'=1}^{n_{\text{post}}} K(\boldsymbol{x}_j,\boldsymbol{x}_{j'}) - \frac{2}{n_{\text{pre}}n_{\text{post}}}\sum_{i=1}^{n_{\text{pre}}}\sum_{j=1}^{n_{\text{post}}} K(\boldsymbol{x}_i,\boldsymbol{x}_j). \tag{2.9}$$

In our implementation, we used the standard Gaussian radius basis function (RBF) kernel $K(\cdot,\cdot)$, where the kernel bandwidth is chosen using the so-called "median trick" (Bernhard et al., 2018), i.e., set to be the median of the pairwise distances between data points.

The simulation set-up is as follows. We simulate data $\boldsymbol{x}_1,\cdots,\boldsymbol{x}_T$ with $T = 400$, and the change-point is set at time $t^* = 200$. In other words, $\boldsymbol{x}_1,\cdots,\boldsymbol{x}_{200}$ are generated from the pre-change distribution, and $\boldsymbol{x}_{201},\cdots,\boldsymbol{x}_{400}$ are generated from the post-change distribution.

In our simulations, this point cloud data is generated with topological structure from two simple geometric shapes, the unit sphere and the ellipsoid (in varying dimensions). A quick inspection of the PDs shows the presence of many persistence features, thus we decided to use the persistence clustering approach across all simulation studies. The goal is to have PERCEPT learn this topological structure from data, then leverage such structure to quickly identify the change-point. Here, we make use of the 1-D homology for PERCEPT (see Section 2.2.1), since both the circle and ellipsoid have a hole structure.

2.4.1 Shape Change

We first consider the case of geometric shape changes, where the pre-change data is sampled with noise from the unit two-dimensional (2-D) circle, and the post-change data is sampled with noise from a 2-D ellipse. Two noise settings are considered for this experiment: $N(0, 0.05)$ and $N(0, 0.10)$. Figure 2.7 shows the detection statistics from PERCEPT, Hotelling's T^2 and MMD under the two noise settings. We see that both PERCEPT and MMD are able to quickly detect the change: both monitoring statistics peak up immediately after the change-point at $t = 200$. The Hotelling's T^2 statistic, however, shows a relatively larger delay, which is unsurprising since such a method is adept at detecting mean and covariance shifts, but not shape changes. This suggests that, by integrating topological structure, PERCEPT can yield improved performance over the Hotelling's T^2. For the Wasserstein distance method, the statistic experiences a large spike at exactly $t = 200$ for the noise setting $N(0, 0.05)$, but is unable to detect the change at the larger noise level $N(0, 0.10)$. This affirms the earlier observation that the Wasserstein distance is a weaker metric which may not be able to distinguish local topological differences.

2.4.2 Noise Change

Next, we consider the case of noise changes, where the pre- and post-change data are generated from the same unit 2-D circle (or ellipse), but with noise levels $N(0, 0.05)$ and $N(0, 0.10)$, respectively. Figure 2.8 shows the test statistics from PERCEPT and the Hotelling's T^2 for the circle and ellipse. We see that PERCEPT can quickly detect the

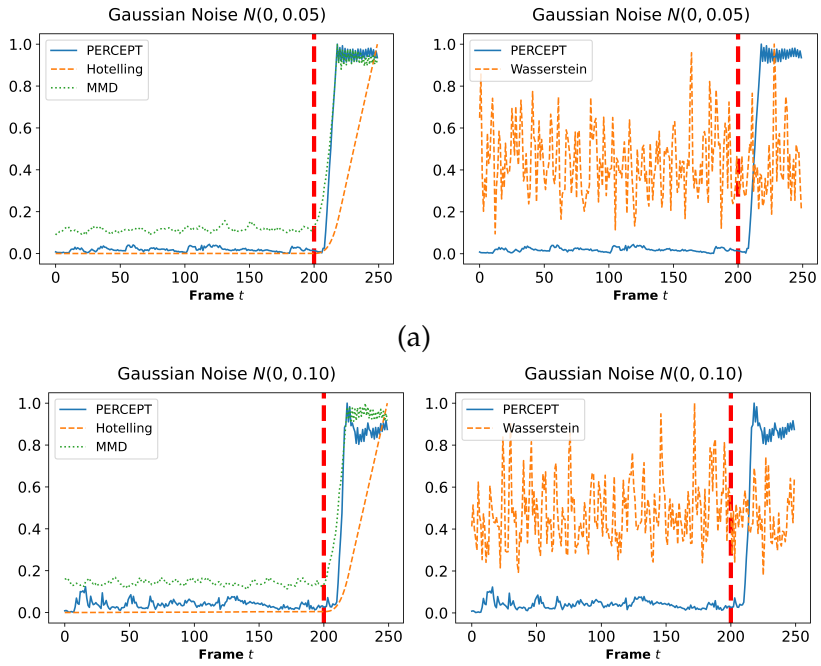
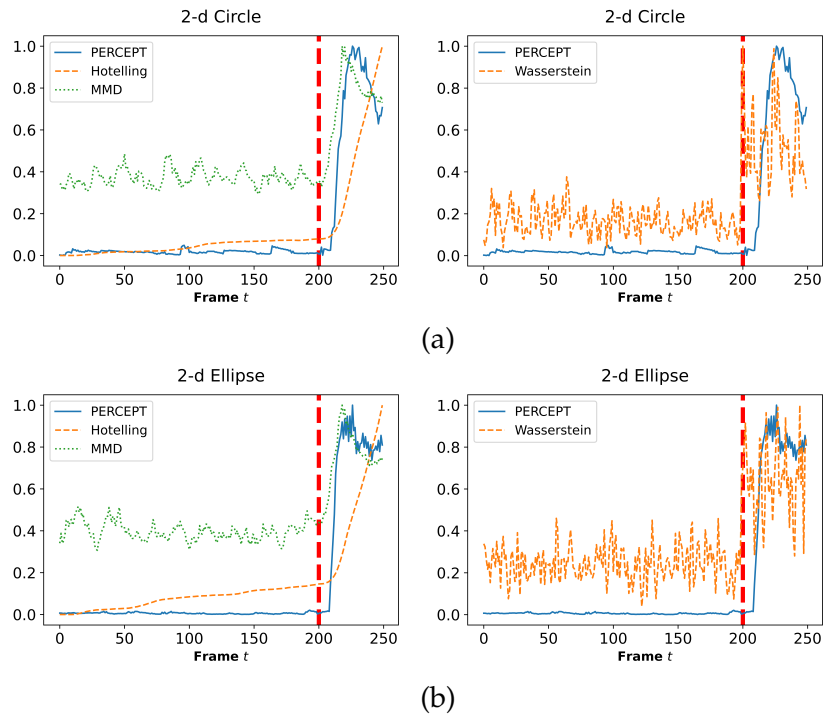


FIGURE 2.7: (a) Test statistics using PERCEPT, Hotelling's T^2, MMD and Wasserstein distance at each time t, for the setting of Gaussian noise $N(0, 0.05)$. The vertical red dashed lines indicate the true change-point $t^* = 200$. (b) The same comparison for the setting of Gaussian noise $N(0, 0.10)$.

underlying change: its pre-change test statistic is quite stable, and it peaks up quickly after the change-point at $t = 200$. The Wasserstein distance approach also seems to perform quite well here, although its pre-change statistic is noticeably more unstable, which may lead to frequent false alarms (i.e., lower ARL). On the other hand, the Hotelling's T^2 statistic increases noticeably more slowly after the change, which suggests a larger detection delay; this indicates that Hotelling's T^2 can indeed be slow at detecting small variance changes. For the MMD statistic, we see that while it peaks up after the change-point, its pre-change statistic is quite unstable and volatile, which again leads to frequent false alarms. This is not unexpected, since the MMD approach does not leverage the underlying topological structure for change detection, which may result in inefficient monitoring performance.



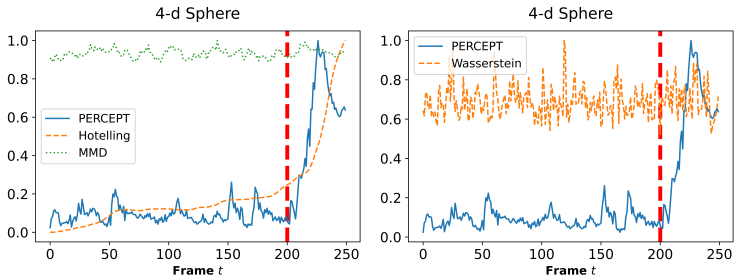
(a)

(b)

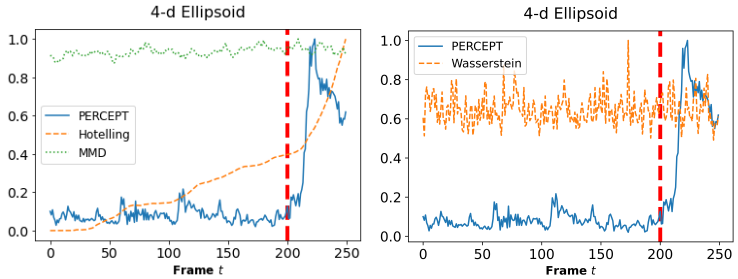
FIGURE 2.8: (a) The test statistics using PERCEPT, Hotelling's T^2, MMD and Wasserstein distance at each time t for the 2-D circle, where the noise changes from $N(0, 0.05)$ to $N(0, 0.10)$. Again, the vertical red dashed lines indicate the true change-point at $t^* = 200$.

2.4.3 Increasing Dimensionality

We now investigate how well these methods perform on data generated on higher-dimensional geometric structures, namely, the 3-D and 4-D unit spheres and ellipsoids. We consider here the same noise change as in Section 2.4.2, and we compared the performance of PERCEPT with the classic parametric Hotelling's T^2 test, MMD statistics and Wasserstein distance. Figure 2.9a shows the detection statistics for the 4-D sphere, and the detection statistics for the 3-D sphere can be found in the Appendix. We see that the proposed PERCEPT method consistently outperforms existing methods: its pre-change statistics are stable, and its post-change statistics peak up quickly after the change. Comparatively, the increase in the Hotelling-T^2 statistic is more subdued after the change (which results in greater detection delay), and the MMD statistic pre-change is noticeably more unstable (which results in increased false alarms). A likely reason for the poorer



(a)



(b)

FIGURE 2.9: (a) The test statistics using PERCEPT, Hotelling's T^2, MMD and Wasserstein distance at each time t for the 4-D sphere, where the noise changes from $N(0, 0.05)$ to $N(0, 0.10)$. Again, the vertical red dashed lines indicate the true change-point at $t^* = 200$. (b). Same for 4-D ellipsoids.

performance of the MMD statistic is that it does not integrate topological structure, which when present, can improve monitoring performance. The Wasserstein distance approach is again unable to detect this change, for similar reasons as before. Furthermore, comparing with the lower-dimensional setting in Figure 2.8, PERCEPT appears to yield greater improvements over existing methods, which is unsurprising since it leverages the underlying low-dimensional topological structure in the high-dimensional data. Figure 2.9b shows the detection statistics for the 4-D ellipsoid, and the detection statistics for the 3-D ellipsoid can be found in the Appendix. Again, we see that PERCEPT yields improved performance over the existing benchmarks, despite having slightly more unstable pre-change statistics. These experiments suggest that, by learning and integrating low-dimensional topological structure, PERCEPT can efficiently detect changes when such structures are present in high-dimensional data.

2.4.4 EDD vs. ARL

Finally, we compare the performance of these methods via the two metrics introduced in Section 2.3.3: the Average Run Length (ARL) and the Expected Detection Delay (EDD). Recall that ARL measures the expected run length to a false alarm when there is no change, and EDD measures how quickly the change is detected. A method with large ARL and small EDD is desired, since this means less false alarms and smaller detection delay. We approximate the ARL and EDD of the compared methods using different thresholds b, under the 2-D circle experiment with Gaussian noise $N(0, 0.05)$ for pre-change regime and Gaussian noise $N(0, \sigma^2)$ for post-change regime, where $\sigma^2 = 0.09$, 0.10 and 0.11. We adopted the experiments from (L. Xie & Xie, 2021), and details can be found in Appendix A.1.

Figure 2.10 plots the EDD vs. log-ARL curves for the 2-D circle experiment. There are two interesting observations. First, for all noise levels and ARL levels, we see that PERCEPT yields much lower EDD compared to the Hotelling's T^2 procedure, which suggests that it indeed yields improved detection performance by integrating topological structure. Second, the proposed method is more robust to noise perturbations. As the noise level decreases, the EDD for the Hotelling's T^2 becomes noticeably larger for fixed ARL levels. Again, this is not surprising, since this change is difficult to detect without first identifying the underlying topological structure. In contrast, the EDD for PERCEPT is more stable and nearly remains the same as the noise level decreases, which shows the robustness of PERCEPT. The relationship between ARL and EDD for the 4-D case is not shown here since it is expensive to compute, but similar conclusions are expected from earlier results.

2.5 Applications

2.5.1 Solar Flare Change Detection

We now demonstrate the effectiveness of PERCEPT in the earlier motivating solar flare detection problem. A solar flare is an intense emission of radiation on the Sun's atmosphere, and the monitoring of such changes via satellite imaging is critical for predicting

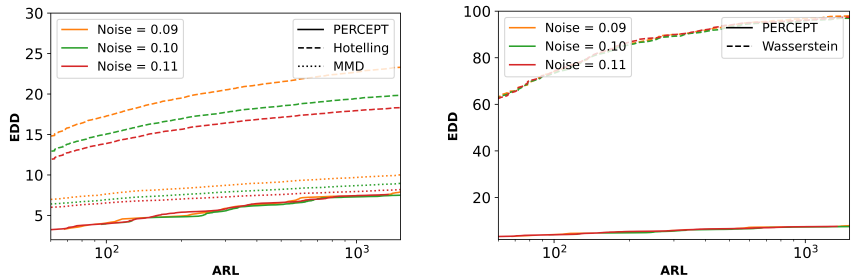
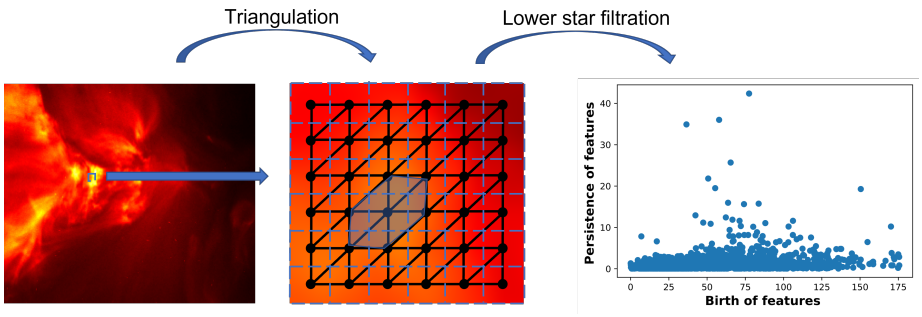


FIGURE 2.10: EDD–ARL curves for 2-D circle experiment with noise change, using varying post-change noise levels for PERCEPT, Hotelling's T^2, MMD, and the Wasserstein distance method.

geomagnetic storms (Y. Xie et al., 2012). This detection can be highly challenging, however, as noted in Section 2.2.2, due to the high dimensionality of images and the subtlety of such a change. Recent work in image processing shows that a wide range of image features can be captured via topology (Bendich et al., 2011), which suggests our topology-aware approach may provide a solution. The data used here are $T = 100$ image snapshots taken by the SDO at NASA, where the true change-point is at $t^* = 49$. Further details can be found in Section 2.2.2.

To apply PERCEPT, we need to first map the image data to a filtration, or a sequence of simplicial complexes (see Section 2.2.1 for details). We will make use of the so-called *lower star filtration*, which has shown success in capturing useful topological features in image data (Bendich et al., 2011). Let $f(x)$ be a mapping from each pixel location x to its intensity value, and for a given ϵ, define the *sublevel set* of f as $X(\epsilon) = \{x | f(x) \leq \epsilon\}$. One can then form a simplicial complex from $X(\epsilon)$ by first considering image pixels as vertices on a grid, then triangulating this grid by placing an edge between two points that are horizontally, vertically, or diagonally adjacent, and a triangular face for any three adjacent points forming a triangle. This is known as the Freudenthal triangulation (Freudenthal, 1942). Figure 2.11a visualizes this mapping from image to simplicial complex. For a sequence $0 < \epsilon_1 < \epsilon_2 < \cdots < \epsilon_m$, the *sublevel set filtration* of the image is defined as $X_i = X(\epsilon_i)$, $i = 1, \cdots, m$, which forms a sequence of nested simplicial complexes. This filtration can be used to construct a PD (Figure 2.11b) as described in Section 2.2.1. One



(a) (b)

FIGURE 2.11: (a) Freudenthal triangulation of the solar flare image. (b) "Tilted" persistence diagram of the solar flare image in (a).

can view this filtration pipeline as a way to extract important image topological features as a PD point cloud. Here, the 0-D homology is used for PERCEPT, since different solar flares can be characterized by connected components in the lower star filtration.

With the mapped PD in hand for each image, we can then proceed with the detection framework outlined in Section 2.3. Since there are a wide range of solar flares, we assume here that only pre-change training data are available, and we use the persistence binning approach in Section 2.3.1 and equal weights over persistence histogram bins. A quick inspection of the computed PDs show only a small amount of points (or features), thus we decided to use persistence histograms instead of persistence clusters (see Section 2.3.4). Here, the histogram breakpoints $b_1, \cdots, b_L$ are chosen such that there is (roughly) an equal sum of persistences within each histogram bin in the first solar flare image. The Hotelling's T^2 is performed using the 30 extracted principal components from PCA on the image data, and the MMD test is performed on the image data directly, with the RBF kernel, and the bandwidth is chosen using the "median trick" as described in Section 2.4. The Wasserstein distance approach is performed on the mapped PDs from the lower star filtration.

Figure 2.12 shows the proposed detection statistic $\chi_t^{\max}$ using $L = 10$ histogram bins, the Hotelling's T^2, the MMD test statistics and the Wasserstein distance statistic. For PERCEPT, we note a sudden increase in the test statistic after the true change-point $t^* = 49$ (vertical red dashed line), which suggests the proposed method is able to pick out

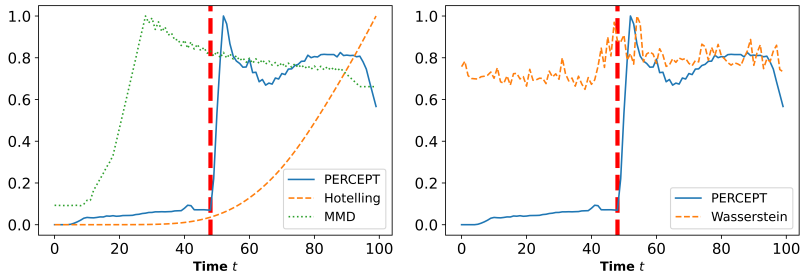


FIGURE 2.12: Detection statistics for each method in the solar flare monitoring application. Vertical red dashed lines indicate the true change-point at $t^* = 49$.

the underlying topological change in the images with little detection delay. Comparatively, for the Hotelling's T^2, the increase in its test statistic is much more subdued and gradual, which indicates a much larger detection delay. The MMD approach yields poor performance here: its pre-change statistics are highly volatile and unstable, and its post-change statistics experience a decrease after the change-point. The Wasserstein distance approach is not able to detect the change, since its test statistic does not increase significantly at the change-point. This is in line with results from earlier simulations, and an in-depth discussion of plausible reasons is provided there. By learning and integrating low-dimensional topological structure within a non-parametric change detection framework, PERCEPT can yield significant improvements over existing methods when such structure is indeed present for image data.

Despite the required mapping from images to PDs via the lower star filtration, PERCEPT appears to be quite efficient. This is due in part to the availability of well-maintained packages for TDA algorithms. Table 2.1 summarizes the computation time for each method in this application, where persistence homology (for PERCEPT) is performed using the Python package `Ripser` (Tralie et al., 2018). Here, the lower star filtration on 100 images takes around 30 seconds on a standard desktop computer, which is fast given these images are quite large (232×292 pixels). The Wasserstein distance approach takes quite long due to optimal matching. With this filtration computed, the detection statistic $\chi_t^{\max}$ can then be evaluated with minimal additional computation, thus enabling efficient detection.

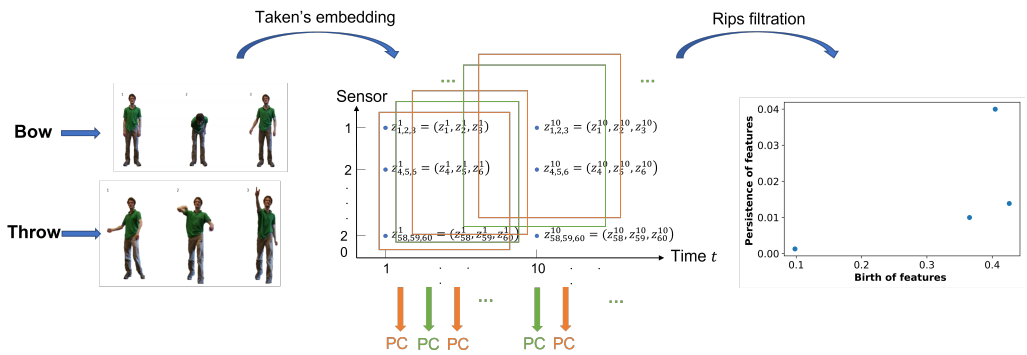
Table 2.1: Total computation time (in seconds) for the compared methods on the solar flare application. Computation is performed on a standard desktop computer using an AMD Ryzen 5 2600 processor with 6 cores and 12 threads.

PERCEPT	Hotelling	MMD	Wasserstein
32.3 s	2.1 s	3.4 s	341.8 s

2.5.2 Human Gesture Change Detection

Next, we investigate the performance of PERCEPT in a human gesture detection application. The detection of human body gestures is an important task in computer vision (Turaga et al., 2008), and has immediate applications in visual surveillance and sign language interpretation (Oh et al., 2011). One challenge for this detection lies in the high-dimensional time series data, and the low-dimensional embedding of human body gestures within such data. Recent developments in time series analysis suggest that many time series features can be captured via TDA (further details below), and thus PERCEPT may be promising for this task. To explore this, we will use the human gestures dataset from the Microsoft Research Cambridge-12 Kinect (Fothergill et al., 2012), which consists of time series observations of human skeletal body part movements, collected from 20 sensors on 30 people performing 12 distinct gestures. We study in particular the transition of gestures from a “bow" to a “throw" sequence, as shown in Figure 2.13a.

To apply PERCEPT, we need to first transform the multi-dimensional time series to point cloud data, on which the usual TDA pipeline (see Section 2.2.1) can then be performed. A popular transformation is Taken’s embedding (Gidea & Katz, 2017), which is widely used in TDA. Suppose we observe the d-dimensional time series $\{z_k(t), t \geq 0\}$, $k = 1, \cdots, d$ (here, $d = 20 \times 3 = 60$, since each of the 20 body sensors return a 3-D coordinate). For each time t, define the point $\boldsymbol{z}_t = (z_1(t), \cdots, z_d(t))$ as the cross-section of the multivariate time series. With a given window size w, Taken’s embedding returns the point cloud representation $\mathcal{Z}_t = \{\boldsymbol{z}_i, \cdots, \boldsymbol{z}_{i+w-1}\}$ at each time t. After this transformation, one can then perform the standard TDA filtration in Section 2.2.1 to convert the point cloud



(a) (b) (c)

FIGURE 2.13: (a) Visualizing the two gestures: taking a bow (top) and throwing an object (bottom). (b) Embedding of a multivariate time series to point cloud (PC) data via Taken's embedding, to a persistence diagram via persistence homology. (c) "Tilted" persistence diagram of the embedded point cloud at $t = 510$.

$\mathcal{Z}_t$ to a persistence diagram $\mathcal{D}_t$. It can be shown (Gidea & Katz, 2017) that this embedding captures key periodicity and dynamic system information on the multivariate time series. Figure 2.13b provides a visualization of Taken's embedding. Here, we set the window size w to be approximately the periodicity of the gestures, which is based on prior knowledge. We further make use of 1-D homology for PERCEPT, since it is known that periodicity of time series can be represented as 1-D holes via Taken's embedding (Gidea & Katz, 2017).

With this, we can then apply PERCEPT for detecting gesture changes. A quick inspection shows that there are limited points in the computed PDs $\mathcal{D}_t$. Thus we use the persistence histograms in Section 2.3.1 rather than the persistence clusters in Section 2.3.4. We then used 60 frames from "bow" and "throw" sequences to choose the number of histogram bins and optimize weights. The Hotelling's T^2 is performed using the 30 extracted principal components from PCA on the embedded point cloud data $\mathcal{Z}_t$, and the MMD test is performed on the embedded point cloud data $\mathcal{Z}_t$ directly with the RBF kernel, and the bandwidth is chosen using the "median trick" as described in Section 2.4. The Wasserstein distance approach is performed on the mapped PDs from the Rips filtration.

Figure 2.14 shows the proposed detection statistic $\chi_t^{\max}$ using $L = 2$ histogram bins, the Hotelling's T^2, the MMD test statistics, and the Wasserstein distance over time, with the

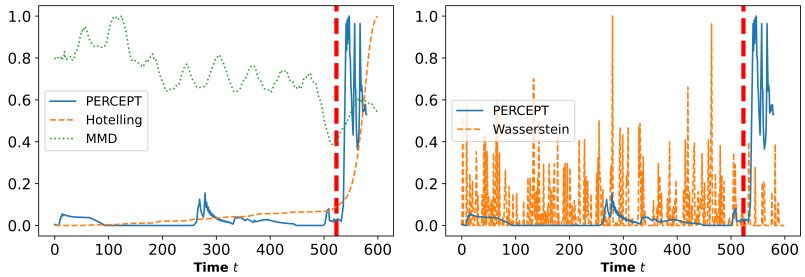


FIGURE 2.14: Detection statistics for each method in the human gesture change detection application. Vertical dashed lines indicate the true change-point from "bow" to "throw".

vertical line indicating the true change-point (i.e., the transition from "bow" to "throw"). For PERCEPT, we see the test statistic is relatively stable pre-change and experiences a large jump immediately after the change. This suggests our method has a large ARL and small detection delay, which is as desired. The Hotelling's T^2 again experiences a much more subtle and gradual increase after the change, which indicates a large detection delay. As before, the MMD method performs poorly: its pre-change statistic is highly unstable pre-change and experiences a notable decrease post-change, which suggests it is unable to detect the change at all. The Wasserstein distance method is again not able to detect the change-point. This demonstrates the advantages of a topology-aware non-parametric change detection framework: when such a low-dimensional structure exists and can be leveraged, one can achieve efficient monitoring performance even with high-dimensional data.

3. e^{RPCA}: Robust Principal Component Analysis for Exponential Family Distributions

3.1 Introduction

With remarkable advances in sensing and experimental technologies, scientists and engineers now have access to massive datasets with complex forms for decision-making. The efficient harnessing of such data, in particular, the extraction of background structure and deviating anomalies (arising from occlusions, malicious tampering, process defects, or other causes for outliers), becomes ever more important for timely process monitoring, diagnosis and improvement. The increasingly complex form of such data further necessitates a careful consideration and integration of its underlying probabilistic distribution, which is often known and can be highly non-Gaussian. To tackle these challenges, we propose a novel Robust Principal Component Analysis method for Exponential Family Distributions (e^{RPCA}, for short), that leverages structure on this probabilistic distribution from the exponential family, to jointly perform anomaly detection and background extraction from massive and complex data matrices. The e^{RPCA} is expected to outperform the state-of-the-art when the distribution of such data matrices is non-Gaussian and can reliably be inferred from domain knowledge; we shall show this later in experiments.

Our approach is motivated by two ongoing applications, on defect detection for steel sheet manufacturing and burglary monitoring in the Atlanta metropolitan area. For the first application, the timely detection of defects (e.g., gashes and dents) in steel sheet manufacturing is crucial for quality control. Recent developments in quanta image sensing (QIS; Charbon, 2014) have shown promise for the desired high-frequency imaging, but these systems typically capture image intensities via binary bits. The efficient identification of potential defects for timely diagnosis thus poses a challenge with such large binary images. For the second application, the detection of regular and irregular burglary activities is paramount for crime monitoring and prevention. Reported burglary data naturally take the form of counts, and can be observed at high spatiotemporal resolution. Using such

high-dimensional count data for timely extraction of regular and irregular crime patterns is thus a critical challenge. In both applications, both the identification of background structure, e.g., regular crime activity, and its corresponding anomalies, e.g., irregular crime activity, are important for timely decision-making from high-dimensional and complex data matrices. We return to these two applications later in Section 3.5.

A widely-used method for joint extraction of structure and sparse anomalies from a data matrix $\mathbf{M} \in \mathbb{R}^{p \times q}$ is the Robust Principal Component Analysis (RPCA) method (Candès et al., 2011). RPCA decomposes $\mathbf{M}$ into the sum of two matrices $\mathbf{L}$ and $\mathbf{S}$, such that $\mathbf{L}$ is a low-rank matrix (modeling structure) and $\mathbf{S}$ is a sparse matrix (capturing anomalies). Such a decomposition can be optimized via convex optimization methods via its tightest convex relaxation using the nuclear and l_1-norms; more on this later. There has been much subsequent work on efficient optimization algorithms for the RPCA decomposition, including the use of augmented Lagrangian multipliers (Z. Lin et al., 2010; Yuan & Yang, 2013), accelerated proximal gradient (Bouwmans & Zahzah, 2014), alternating minimizing approaches (Gu et al., 2016) and low-rank matrix fitting (Shen et al., 2014). There is also notable (albeit less) work on the RPCA when $\mathbf{M}$ is observed with random noise. This includes the stable principal component pursuit approach (Zhou et al., 2010), which relaxes the equality constraint $\mathbf{M} = \mathbf{L} + \mathbf{S}$ to account for the presence of small measurement errors; more on this later. Such an approach, however, does not factor for the specific probabilistic distribution for $\mathbf{M}$, which in many applications may be known or can be reliably inferred. (Ding et al., 2011) proposed a hierarchical Bayesian approach for decomposing a noisy matrix into its low-rank and sparse components, but such an approach again does not factor in the non-Gaussian distribution of $\mathbf{M}$.

There is also a complementary line of work on extending the standard principal component analysis (PCA) for non-Gaussian noise distributions. This includes (Collins et al., 2001), which proposed a modification of PCA that generalizes to a broad class of so-called exponential family distributions (Casella & Berger, 2002) via Bregman distances. The exponential family covers a broad range of parametric distributions encountered in applications,

including the Bernoulli, Poisson, Exponential, and Gaussian distributions. (Mohamed et al., 2008) investigated a fully probabilistic extension of PCA for the exponential family. (L. Liu et al., 2018) presented the "Exponential PCA" (or ePCA) approach, which uses recent developments in random matrix theory and shrinkage for efficient estimation of low-rank structure under exponential family noise. Such methods, however, do not account for nor facilitate the identification of sparse anomalies in $\mathbf{M}$, which is critical in our aforementioned motivating applications.

To tackle these limitations, we thus propose a new e^{RPCA} method that facilitates the *joint* extraction of low-rank structure and sparse anomalies, in the setting where data matrices are generated from the exponential family distribution. The e^{RPCA} leverages a novel optimization formulation for this decomposition, which integrates information on the underlying probabilistic distribution of $\mathbf{M}$ via its likelihood function. We then present an alternating direction method of multiplier (Boyd et al., 2011) optimization algorithm, which incorporates this distributional structure for efficient decomposition under either its natural or canonical parametrization. Finally, we demonstrate the effectiveness of the e^{RPCA} over existing methods in a suite of numerical experiments and for our two motivating applications on steel defect detection and crime monitoring. In particular, we show that when the distribution of the data matrices is markedly non-Gaussian and can be reliably inferred from domain knowledge, the e^{RPCA} can leverage this information for improved extraction of low-rank structure and sparse anomalies over the state-of-the-art.

This paper is organized as follows. Section 3.2 provides background on the RPCA, the ePCA, and their recent extensions, then discusses their limitations for our motivating application. Section 3.3 outlines the proposed e^{RPCA}, including its formulation and optimization algorithm, including a discussion on hyperparameter tuning and scalability for large data matrices. Section 3.4 presents a suite of numerical experiments investigating the performance of e^{RPCA} and existing methods, under different distributions of $\mathbf{M}$ from the exponential family. Section 3.5 explores the e^{RPCA} in the aforementioned two motivating applications.

3.2 Background and Motivation

We first provide an overview of the robust PCA (Candès et al., 2011), the exponential PCA (L. Liu et al., 2018) and its extensions, then motivate the proposed e^{RPCA} via our steel defect detection application.

3.2.1 Robust PCA

RPCA (Candès et al., 2011) is a widely-used method for jointly recovering low-rank structure and anomalies from a data matrix $\mathbf{M} \in \mathbb{R}^{p \times q}$ with significant corruptions on a sparse number of entries. For recovering low-rank structure, it is well-known that the standard PCA approach (Hastie et al., 2009; Pearson, 1901) can be highly sensitive to sparse and large outliers in $\mathbf{M}$; a single large outlier can greatly skew its estimated structure. PCA also cannot perform the task of detecting and isolating these sparse anomalies, which as mentioned before is critical for process diagnosis and quality control. To address such limitations, RPCA makes use of the following decomposition of the data matrix $\mathbf{M}$:

$$\min_{\mathbf{L},\mathbf{S}} \|\mathbf{L}\|_* + \lambda \|\mathbf{S}\|_1, \quad \text{s.t.} \quad \mathbf{L} + \mathbf{S} = \mathbf{M}. \tag{3.1}$$

Here, $\|\mathbf{A}\|_*$ is the nuclear norm (the sum of the singular values of $\mathbf{A}$), and $\|\mathbf{A}\|_1$ is the matrix ℓ_1-norm (the sum of absolute values of entries in $\mathbf{A}$). Note that the nuclear norm $\mathbf{A}$ can be viewed as the tightest convex relaxation for the rank of $\mathbf{A}$, and its ℓ_1-norm similarly serves as a convex relaxation of the number of non-zero entries in $\mathbf{A}$. Thus, Equation (3.1) decomposes the data matrix $\mathbf{M}$ as the sum of a low-rank matrix $\mathbf{L}$ and a sparse matrix $\mathbf{S}$, which facilitates the desired recovery of the underlying low-rank structures and anomalies from $\mathbf{M}$. The parameter $\lambda > 0$ controls the trade-off between low-rankedness and sparsity in this decomposition.

The formulation (3.1), which can be shown to be convex, can be efficiently optimized via a variety of scalable algorithms. A popular approach (Z. Lin et al., 2010; Yuan & Yang, 2013) is to iteratively minimize the following augmented Lagrange multiplier (ALM)

formulation:

$$\ell(\mathbf{L}, \mathbf{S}, \mathbf{Y}) := \|\mathbf{L}\|_* + \lambda\|\mathbf{S}\|_1 + \langle \mathbf{Y}, \mathbf{M} - \mathbf{L} - \mathbf{S}\rangle_F + \frac{\mu}{2}\|\mathbf{M} - \mathbf{L} - \mathbf{S}\|_F^2, \tag{3.2}$$

where $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product. Here, the equality constraint on $\mathbf{L} + \mathbf{S} = \mathbf{M}$ is replaced by its Lagrangian form, $\mathbf{Y}$ is the so-called Lagrange multiplier matrix, and $\mu > 0$ a positive constant. A generic Lagrange multiplier algorithm (Bertsekas, 1982) can then be applied to iteratively solve (3.2). During the k-th iteration, one would optimize $(\mathbf{L}_k, \mathbf{S}_k) = \arg\min_{\mathbf{L},\mathbf{S}} \ell(\mathbf{L}, \mathbf{S}, \mathbf{Y}_k)$, then update the Lagrange multiplier matrix via $\mathbf{Y}_{k+1} = \mathbf{Y}_k + \mu(\mathbf{M} - \mathbf{L}_k - \mathbf{S}_k)$. These two steps are repeated until convergence. The parameter $\mu > 0$ can be viewed as the step size for updating the Lagrangian multiplier matrix. (Z. Lin et al., 2010) provides further technical details on the validity and optimality of ALM. Further extensions include (Xue et al., 2019; Yang & Zou, 2015), which explored various exact and inexact ALM approaches; (Guyon et al., 2012), which proposed a linearized alternating direction optimization approach with adaptive penalties; and (Bouwmans & Zahzah, 2014), which investigated the use of accelerated proximal gradient algorithms for performing this decomposition.

The formulation (3.1), however, does not account for the presence of noise in the observed data matrix $\mathbf{M}$, as $\mathbf{M}$ is assumed to decompose into the low-rank signal $\mathbf{L}$ and sparse anomalies $\mathbf{S}$ without noise. In many problems, including our motivating applications on steel defect detection and crime monitoring, such noise is ubiquitous and unavoidable; it arises either from the measurement process or as a realization of the data-generating process. To account for this, (Zhou et al., 2010) investigated the following extension of RPCA, which they call the stable principal component pursuit (Stable-PCP):

$$\min_{\mathbf{L},\mathbf{S}} \|\mathbf{L}\|_* + \lambda\|\mathbf{S}\|_1 \quad \text{s.t.} \quad \|\mathbf{M} - \mathbf{L} - \mathbf{S}\|_F \leq \delta. \tag{3.3}$$

The inequality in (3.3) relaxes the equality constraint $\mathbf{L} + \mathbf{S} = \mathbf{M}$ in (3.1) to account for a small amount of deviation δ resulting from noise. One limitation of the Stable-PCP, however, is that it does not factor in the *parametric form* of the underlying noise, which

is often known in applications. For example, in imaging applications, the imaging system often dictates the parametric distribution used when observing the data matrix $\mathbf{M}$ (Y. Cao & Xie, 2015). As we shall see later, the use of this information on noise structure can greatly improve the recovery of both the low-rank signal $\mathbf{L}$ and its anomalies $\mathbf{S}$, particularly when such noise is large and non-Gaussian. We will investigate this further in later numerical experiments.

3.2.2 Exponential PCA

The extension of the standard PCA for non-Gaussian noise has been explored in a series of papers; such work showed that when the parametric form of this noise is known, the integration of this structure can greatly improve the recovery of the low-rank signal. A common family of distributions is the exponential family distribution (Casella & Berger, 2002). Given a single parameter θ, the (one-parameter) exponential family is a family of distributions with probability density (or mass) function:

$$p_\theta(m) = \exp\left\{\eta(\theta)t(m) + a(\theta) + b(m)\right\}. \tag{3.4}$$

Here, $\eta(\theta)$ is the *canonical* parameterization of parameter θ, $t(m)$ is the sufficient statistic of the distribution, and $a(\theta)$ and $b(m)$ are fixed and known functions of θ and data m, respectively. Such a specification defines a broad range of common distributions, including the Poisson, Bernoulli, Exponential, and Gaussian distributions (see Table 3.1). We will denote a random variable following this distribution as $M \sim \text{ExpFam}\{\theta; \eta(\cdot), t(\cdot), a(\cdot), b(\cdot)\}$. An early work on integrating this non-Gaussian structure within PCA is (Collins et al., 2001), which proposed a PCA extension that generalizes to the exponential family distribution via Bregman distances. This is further extended in (Mohamed et al., 2008) via a fully probabilistic extension of PCA leveraging hybrid Monte Carlo sampling.

A recent development on this front is the ePCA method in (L. Liu et al., 2018). The key idea is to leverage the eigendecomposition of a new covariance matrix estimator, constructed via moment calculations, shrinkage, and random matrix theory. ePCA begins with the sample covariance matrix of the data, then applies a series of operations, including di-

Table 3.1: Common distributions from the (one-parameter) exponential family (3.4): the Poisson, Bernoulli, Exponential, and Gaussian (with known variance σ^2) distributions.

Distribution	$\eta(\theta)$	$t(m)$	$a(\theta)$	$b(m)$
Poisson	$\log\theta$	m	$-\theta$	$-\log(x!)$
Bernoulli	$\log\left(\frac{\theta}{1-\theta}\right)$	m	$\log(1-\theta)$	0
Exponential	$-\theta$	m	$\log\theta$	0
Gaussian	$\frac{\theta}{\sigma}$	$\frac{m}{\sigma}$	$-\frac{\theta^2}{2\sigma^2}$	$-\frac{\log(2\pi\sigma^2)}{2} - \frac{m^2}{2\sigma^2}$

agonal debiasing, homogenization, shrinkage, heterogenization, and scaling (guided by the underlying exponential family model), to improve this covariance estimator. The corresponding low-rank representation can finally be obtained via an eigendecomposition of this modified covariance estimator. Further details of ePCA and its theoretical justification can be found in (L. Liu et al., 2018).

While such work on extending the standard PCA for non-Gaussian distributions is promising, there has been little work on leveraging such non-Gaussian noise for *jointly* recovering structure and anomalies in the presence of significant sparse outliers. This *combined* setting of non-Gaussian noise with sparse anomalies arises in a broad range of modern problems, including our two later applications on steel defect detection and crime monitoring. The aforementioned approaches, which tackle *only* the setting of non-Gaussian noise or sparse corruptions, can thus yield poor low-rank recovery and anomaly detection performance, as we will see next.

3.2.3 Steel Defect Detection Application

We first investigate these existing methods for our motivating steel defect detection application (further details can be found in Section 3.5.1). This application features the two defining challenges motivating our method: (i) non-Gaussian noise with (ii) significant sparse anomalies. For (i), the high-frequency imaging of steel sheets can be performed via quanta image sensing (Charbon, 2014), which has shown improved performance over more

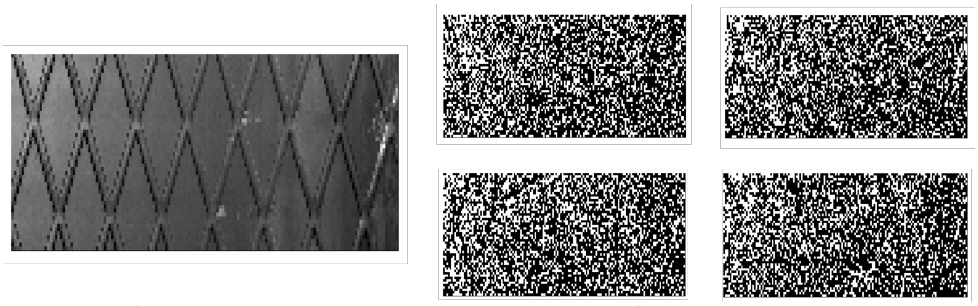


FIGURE 3.1: (Left) The uncorrupted steel sheet image from Severstal, and (right) corresponding binary images generated with Bernoulli noise.

conventional multi-bit systems (e.g., complementary metal-oxide semiconductor imaging (Charbon, 2014)) due to higher frequency imaging with lower read noise (Chan, 2022). QIS is a photon-counting device that captures image intensities using binary bits (Charbon, 2014), which can be modeled via i.i.d. Bernoulli noise (Chan, 2022). For (ii), anomalies arise in the form of defects in the steel manufacturing process, e.g., gashes, dents, or inhomogeneities on the steel sheet. These defects result in significant anomalies that are sparse on the imaged surface, and the primary objective is to quickly detect such anomalies for process diagnosis.

Fig. 3.1 (left) shows the uncorrupted image of a steel sheet from a steel industry company Severstal (PAO Severstal, 2019). We see that the steel sheet has a "criss-cross" background structure, which can be well-represented via a low-rank decomposition. It also has visible defects from the manufacturing process, e.g., bumps and dents in white, particularly on the right side. To mimic QIS, we generate synthetic binary images by first normalizing the uncorrupted image intensities, then sampling $n = 500$ binary images from i.i.d. Bernoulli distributions with parameters taken as such intensities. Fig. 3.1 (right) shows several binary images generated in this fashion. We then explore the performance of RPCA and ePCA for jointly estimating the background structure of the steel sheet and its associated defects (further details on this set-up in Section 3.5.1).

Fig. 3.2 shows the estimated low-rank structure $\mathbf{L}$ and its estimated anomalies (i.e., defects) $\mathbf{S}$ using RPCA, along with the estimated low-rank structure using ePCA. For

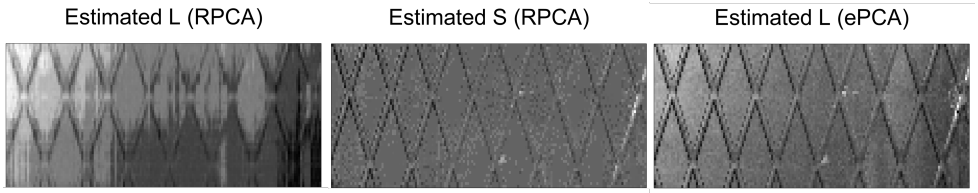


FIGURE 3.2: Visualizing the estimated low-rank structure **L** from RPCA and its anomalies **S**, along with the estimated low-rank structure from ePCA, for the steel defect detection application.

RPCA, we see it yields a mediocre recovery of the criss-cross background, which is expected since it does not factor in the underlying *non-Gaussian* noise from QIS. Because of this, the estimated anomalies from RPCA erroneously capture the cross-cross background and fail to pinpoint the desired defects. The ePCA yields a slightly improved recovery of the background, but the recovered **L** also captures the underlying defects (in white), which is undesirable. This is again unsurprising since ePCA does not account for the presence of *sparse* anomalies. The combined setting of non-Gaussian noise with sparse anomalies thus poses a challenge for structure recovery and anomaly detection using these existing methods. We present next the proposed e^{RPCA} approach for tackling these challenges.

3.3 e^{RPCA}: RPCA for Exponential Family Distributions

We now outline the proposed Robust Principal Component Analysis for exponential family distributions (e^{RPCA}) for two settings: the "single-group" setting, where all observations share the *same* anomalies, and the "multi-group" setting, where anomalies may *change* between different groups of observations. We first formulate the optimization problem for the e^{RPCA}, then present an efficient optimization algorithm and recommendations for tuning parameters.

3.3.1 Optimization Formulation

3.3.1.1 Single-Group Setting

Consider first the single-group setting. Suppose that we observe a collection of matrices $\mathbf{M}_1, \ldots, \mathbf{M}_n \in \mathbb{R}_{p\times q}$. Further suppose the entries of each $\mathbf{M}_i = (M_{i,j,k})_{j=1,\ldots,p;k=1,\ldots,q}$, follow

the earlier exponential family model (3.4):

$$M_{i,j,k} \overset{indep.}{\sim} \text{ExpFam}(\theta_{j,k}; \eta(\cdot), t(\cdot), a(\cdot), b(\cdot)), \tag{3.5}$$

for $i = 1, \ldots, n$, $j = 1, \ldots, p$ and $k = 1, \ldots, q$. The random variable $M_{i,j,k}$ thus models the randomly corrupted observation given the true (unobserved) signal $\theta_{j,k}$. As before, we assume that prior knowledge is available on the class of noise distribution (e.g., Bernoulli), hence the functions $\eta(\cdot)$, $t(\cdot)$, $a(\cdot)$ and $b(\cdot)$ are known, and only the parameter matrix $\Theta = (\theta_{j,k})_{j=1,\ldots,p;k=1,\ldots,q}$ needs to be estimated.

Following RPCA, we assume the parameter matrix Θ can be decomposed as $\Theta = \mathbf{L} + \mathbf{S}$, where $\mathbf{L}$ is a low-rank matrix capturing background structure, and $\mathbf{S}$ is a sparse matrix that models for sparse anomalies. Let $l(\theta; m)$ be the negative log-likelihood function for the exponential family distribution (3.4) given a single data point m, defined as:

$$l(\theta; m) = -\eta(\theta)t(m) - a(\theta) - b(m). \tag{3.6}$$

One appeal of the exponential family is that, with the canonical parametrization $\eta(\theta)$ (or alternate careful parametrizations of θ), the above negative log-likelihood can be made convex in the transformed parameter. This is important for our decomposition algorithm later; it allows for efficient parameter updates via computationally efficient convex optimization algorithms.

We can then formulate the optimization problem of the penalized maximum likelihood estimator (Casella & Berger, 2002) for Θ as:

$$\min_{\mathbf{L},\mathbf{S},\Theta} \sum_{i=1}^{n} \sum_{j=1}^{p} \sum_{k=1}^{q} \frac{l(\theta_{j,k}; M_{i,j,k})}{n} + \alpha \|\mathbf{L}\|_* + \beta \|\mathbf{S}\|_1 \quad \text{s.t.} \qquad \Theta = \mathbf{L} + \mathbf{S}. \tag{3.7}$$

The first term in (3.7) is the standard maximum likelihood estimator for the parameter matrix Θ. As in RPCA, the second term penalizes the rank of the background structure $\mathbf{L}$ via its tightest convex relaxation, and the third term penalizes the number of non-zero entries in $\mathbf{S}$ via its tightest convex relaxation. The parameters $\alpha > 0$ and $\beta > 0$ control

the severity of each penalty term; Section 3.3.3 provides recommendations on how such parameters should be set.

Given that the noise corruption follows an exponential family distribution, one can then plug in the corresponding negative log-likelihood function l, and solve for $\mathbf{L}$ and $\mathbf{S}$ to extract the underlying low-rank structure and sparse anomalies. For example, in our steel defect application, where the image is subject to Bernoulli noise, the e^{RPCA} formulation becomes:

$$\min_{\mathbf{L},\mathbf{S},\Theta} \sum_{i=1}^{n}\sum_{j=1}^{p}\sum_{k=1}^{q}\left\{\frac{-M_{i,j,k}\log(\theta_{j,k})}{n} - \frac{(1-M_{i,j,k})\log(1-\theta_{j,k})}{n}\right\} + \alpha\|\mathbf{L}\|_* + \beta\|\mathbf{S}\|_1 \quad \text{s.t.} \quad \Theta = \mathbf{L}+\mathbf{S}. \tag{3.8}$$

Similar formulations can be adopted for other distributions from the exponential family (see Table 3.1).

A natural question is whether the parameter matrix Θ itself is suitable for the desired low-rank plus sparse decomposition $\mathbf{L}+\mathbf{S}$, or whether such a decomposition is better suited on some transformation of Θ. One computational advantage of the decomposition of Θ in (3.7) is that for the common distributions in Table 3.1, one can show that the negative log-likelihood $l(\theta; m)$ is convex in θ (see Boyd and Vandenberghe, 2004). As we shall see later, this convexity is useful for developing efficient optimization algorithms for solving the e^{RPCA} formulation (3.7).

An alternate decomposition may be via its *canonical* parameterization $\eta(\theta)$ (see Table 3.1). For example, in our steel defect application, suppose one expects the low-rank structure $\mathbf{L}$ to arise in the canonical parameter matrix, rather than Θ. Defining the matrix $\mathbf{H} = (\eta_{j,k})_{j=1,\cdots,p;k=1,\cdots,q}$ as:

$$\eta_{j,k} = \log\frac{\theta_{j,k}}{1-\theta_{j,k}}, \quad j=1,\ldots,p,\ k=1,\ldots,q,$$

the following formulation may be more appropriate:

$$\min_{\mathbf{L},\mathbf{S},\mathbf{E}} \sum_{i=1}^{n}\sum_{j=1}^{p}\sum_{k=1}^{q}\left\{-\frac{M_{i,j,k}\eta_{j,k}}{n} + \frac{\log(1+e^{\eta_{j,k}})}{n}\right\} + \alpha\|\mathbf{L}\|_* + \beta\|\mathbf{S}\|_1 \quad \text{s.t.} \quad \mathbf{H} = \mathbf{L}+\mathbf{S}. \tag{3.9}$$

Similar formulations can be adopted for other exponential family distributions. This canonical decomposition again retains a convex formulation in the natural parameter η, and does not require additional constraints on $\mathbf{H}$ when solving (3.9), since the range of the canonical parameter η is over the reals (Casella & Berger, 2002). In problems where there is domain knowledge on where low-rank structure is expected to arise, such information should be used first and foremost for guiding the formulation of this low-rank plus sparse decomposition.

3.3.1.2 Multi-Group Setting

Consider next the multi-group setting, where between groups of observed matrices, the underlying low-rank structure remains the same but the sparse anomalies may change. This arises, e.g., for our crime monitoring application, where one may have common activities within each week, but different (and sparse) anomalies from week to week. Suppose the observed matrices form $G > 1$ groups, namely, $\{\mathbf{M}_i^{[1]}\}_{i=1}^{n_1}$, $\{\mathbf{M}_i^{[2]}\}_{i=1}^{n_2}$, ... , $\{\mathbf{M}_i^{[G]}\}_{i=1}^{n_G}$, where the anomalies may vary between different groups. As before, suppose the entries of each $\mathbf{M}_i^{[g]} = (M_{i,j,k}^{[g]})_{j=1,\ldots,p;k=1,\ldots,q}$ independently follow the exponential family distribution (3.4) with parameters in matrix $\Theta^{[g]}$.

With this, the multi-group e^{RPCA} can be formulated as:

$$\min_{\mathbf{L},\mathbf{S}_1,\cdots,\mathbf{S}_G,\Theta} \sum_{j=1}^{p}\sum_{k=1}^{q}\left(\sum_{g=1}^{G}\sum_{i=1}^{n_g}\frac{l(\theta_{j,k}^{[g]};M_{i,j,k}^{[g]})}{n_g}\right) + \alpha\|\mathbf{L}\|_* + \sum_{g=1}^{G}\beta_g\|\mathbf{S}_g\|_1 \tag{3.10}$$
$$\text{s.t. } \Theta_g = \mathbf{L} + \mathbf{S}_g,\ g = 1,\ldots,G.$$

Here, each of the G parameter matrices is decomposed as $\Theta_g = \mathbf{L} + \mathbf{S}_g$, where $\mathbf{L}$ models the common low-rank structure, and $\mathbf{S}_g$ models the different sparse anomalies within each group. The parameter $\alpha > 0$ controls the low-rank penalty, and the parameters $\beta_1,\ldots,\beta_G > 0$ control the sparsity within each group. One can similar adopt an alternate decomposition via the canonical parametrization $\eta(\theta)$ (see Table 3.1).

3.3.2 Optimization Algorithm

Next, we present efficient optimization algorithms for solving the single-group and multi-group e^{RPCA} formulations (3.7) and (3.10). These algorithms make use of the alternating direction method of multiplier (ADMM) method (Boyd et al., 2011), which has been widely used in large-scale optimization problems in image processing (Chan et al., 2016) and statistical learning (Zhu, 2017). ADMM optimizes problems of the form:

$$\min_{\mathbf{x},\mathbf{z}} \quad f(\mathbf{x}) + g(\mathbf{z}) \quad \text{s.t.} \quad \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{c}, \tag{3.11}$$

where f and g are convex functions of $\mathbf{x}$ and $\mathbf{z}$. Note that the original RPCA formulation (3.1) fits in the above form, where with $\mathbf{x} = \mathbf{L}$ and $\mathbf{z} = \mathbf{S}$, we have $f(\mathbf{x}) = \|\mathbf{L}\|_*$ and $g(\mathbf{z}) = \|\mathbf{S}\|_1$. With this, the key steps for ALM are to (i) minimize the augmented Lagrangian form (3.2) iteratively for first $\mathbf{L}$ then $\mathbf{S}$ (given other parameters fixed), (ii) update the Lagrange multiplier matrix via $\mathbf{Y}_{k+1} = \mathbf{Y}_k + \mu(\mathbf{M} - \mathbf{L}_k - \mathbf{S}_k)$. These steps are then iterated until the solution converges. One can show that this ADMM algorithm enjoys appealing convergence properties for RPCA optimization; see Z. Lin et al., 2010 for details. We will adapt ADMM for solving the single-group and multi-group problems.

3.3.2.1 Single-Group Setting

Consider first the single-group e^{RPCA} problem (3.7). Its corresponding augmented Lagrangian form can be written as:

$$\min_{\mathbf{L},\mathbf{S},\Theta} \sum_{i=1}^{n}\sum_{j=1}^{p}\sum_{k=1}^{q} \frac{l(\theta_{j,k}; M_{i,j,k})}{n} + \alpha\|\mathbf{L}\|_* + \beta\|\mathbf{S}\|_1 + \langle \mathbf{Y}, \Theta - \mathbf{L} - \mathbf{S}\rangle_F + \frac{\mu}{2}\|\Theta - \mathbf{L} - \mathbf{S}\|_F^2, \tag{3.12}$$

where $\mathbf{Y}$ is the Lagrange multiplier matrix, and $\mu > 0$ is a constant. The key differences of the above e^{RPCA} formulation from (3.2) are the additional parameter matrix Θ to optimize and the additional negative log-likelihood term in the objective.

Our optimization of (3.7) proceeds as follows. First, for fixed Θ and $\mathbf{L}$, the optimal $\mathbf{S}$ that minimizes (3.12) can be solved in closed form via the following lemma.

Lemma 1. (Cai et al., 2010) *For $\tau > 0$, the optimal solution $\mathbf{S}^*$ to the following problem,*

$$\min_{\mathbf{S}} \tau\|\mathbf{S}\|_1 + \frac{1}{2}\|\mathbf{X} - \mathbf{S}\|_F^2, \tag{3.13}$$

is given by:

$$S^*_{jk} = \mathcal{S}_\tau(\mathbf{X}) := sgn(X_{jk}) \max(|X_{jk}| - \tau, 0), \tag{3.14}$$

for $j = 1, \ldots, p$ *and* $k = 1, \ldots, q$.

This is known as the *pointwise soft thresholding* solution. As a direct corollary, the optimal $\mathbf{S}$ that minimizes (3.12) given fixed Θ and $\mathbf{L}$, i.e.:

$$\mathbf{S}^* = \arg\min_{\mathbf{S}} \left\{ \beta \|\mathbf{S}\|_1 + \frac{\mu}{2} ||\Theta - \mathbf{L} - \mathbf{S} + \frac{1}{\mu}\mathbf{Y}||_F^2 \right\}, \tag{3.15}$$

can be solved via pointwise soft thresholding (see Algorithm 1 for specific expression).

Similarly, for fixed Θ and $\mathbf{S}$, the optimal $\mathbf{L}$ that minimizes (3.12) can be solved in closed form via the lemma below.

Lemma 2. (Cai et al., 2010) *Let* $\mathbf{X} = \mathbf{U\Sigma V}^T$ *be the SVD of* $\mathbf{X}$. *Then for* $\tau > 0$, *the optimal solution* $\mathbf{L}^*$ *to the following problem*

$$\arg\min_{\mathbf{L}} \tau \|\mathbf{L}\|_* + \frac{1}{2}\|\mathbf{X} - \mathbf{L}\|_F^2 \tag{3.16}$$

is given by:

$$\mathbf{L} = \mathcal{D}_\tau(\mathbf{X}) := \mathbf{U}\mathcal{S}_\tau(\Sigma)\mathbf{V}^T. \tag{3.17}$$

This is the *singular value thresholding* (SVT) solution. With this, the $\mathbf{L}$ that optimizes (3.12) given fixed Θ and $\mathbf{S}$:

$$\mathbf{L}^* = \arg\min_{\mathbf{L}} \left\{ \alpha \|\mathbf{L}\|_* + \frac{\mu}{2} ||\Theta - \mathbf{L} - \mathbf{S} + \frac{1}{\mu}\mathbf{Y}||_F^2 \right\}, \tag{3.18}$$

can be solved via SVT (see Algorithm 1 for expression).

Finally, we need to optimize (3.12) for the parameter matrix Θ given $\mathbf{L}$ and $\mathbf{S}$. Note that this can be *decoupled* into pq separate optimization problems for each entry of Θ, i.e., for $j = 1, \ldots, p$ and $k = 1, \ldots, q$:

$$\begin{aligned} \theta^*_{j,k} &= \arg\min_{\theta_{j,k}} \left\{ \sum_{i=1}^{n} \frac{l(\theta_{j,k}; M_{i,j,k})}{n} + \frac{\mu}{2} \left(\theta_{j,k} - L_{j,k} - S_{j,k} + \frac{1}{\mu} Y_{j,k} \right)^2 \right\} \\ &=: \arg\min_{\theta_{j,k}} \zeta_{j,k}(\theta_{j,k}; L_{j,k}, S_{j,k}, Y_{j,k}). \end{aligned} \tag{3.19}$$

For the exponential family, it is known that the negative log-likelihood $l(\theta; m)$ is *convex* in θ for the common distributions in Table 3.1 (see Boyd and Vandenberghe, 2004). Thus, given $\mathbf{L}$ and $\mathbf{S}$, we can optimize for Θ using gradient descent methods and enjoy standard convergence guarantees (Nesterov, 1983; Nocedal & Wright, 1999). For certain exponential family distributions (e.g., the Bernoulli), one can further obtain closed-form solutions for (3.19) that can be exploited for efficient optimization (see Appendix for such closed-form solutions for specific distributions). The decoupled problem (3.19) can further be sped up via parallel optimization on each entry of Θ.

With this in hand, the proposed optimization algorithm is presented in Algorithm 1. We begin with an initial estimate on $\mathbf{L}$ via a low-rank SVD approximation of the observation mean $\bar{\mathbf{M}} = (1/n)\sum_{i=1}^{n} \mathbf{M}_i$, with $\mathbf{Y}$ initialized at $\mathbf{0}$ and Θ initialized at $\bar{\mathbf{M}}$. Next, we update $\mathbf{L}$ as the singular value thresholding solution to (3.18) given current iterates for $\mathbf{S}$, $\mathbf{Y}$ and Θ, then update $\mathbf{S}$ as the soft thresholding solution to (3.15) given current $\mathbf{L}$, $\mathbf{Y}$ and Θ. The parameter matrix Θ is then optimized via (3.19) either gradient descent methods (see, e.g., Nocedal and Wright, 1999; the L-BFGS algorithm in (D. C. Liu & Nocedal, 1989) was used in later experiments) or the closed-form updates in Table B.1 (see Appendix). Finally, the Lagrange multiplier matrix $\mathbf{Y}$ is updated via:

$$\mathbf{Y} \leftarrow \mathbf{Y} + \mu(\Theta - \mathbf{L} - \mathbf{S}), \tag{3.20}$$

where $\mu > 0$ is a step size parameter. These steps are then iterated until convergence. Algorithm 1 summarizes the detailed steps of this optimization procedure.

It is worth noting that, in Algorithm 1, the step size μ is fixed as a constant, i.e., $\mu_t = \mu$ over different iterations t. This can be justified as follows. For the ALM formulation of RPCA (3.2), one can show that (see Z. Lin et al., 2010) if $(\mu_t)_{t=1}^{\infty}$ is a non-decreasing sequence and $\sum_{t=1}^{\infty} {\mu_t}^{-1} = \infty$, then the iteratively updated matrices $\mathbf{L}^{[t]}$ and $\mathbf{S}^{[t]}$ converge to an optimal solution $(\mathbf{L}^*, \mathbf{S}^*)$ for the RPCA problem (3.1). Furthermore, if μ_t is bounded above, one can show that the iterative solutions $(\mathbf{L}^{[t]}, \mathbf{S}^{[t]})$ can reach ϵ-optimality from the optimal solution (i.e., within ϵ of the desired RPCA objective in (3.1)) after $t = \mathcal{O}(1/\epsilon)$ algorithm iterations (Z. Lin et al., 2010). Given such convergence guarantees for the RPCA, we thus adopt a similar strategy of constant step size μ for the e^{RPCA} via Algorithm 1.

Algorithm 1 Single-group e^{RPCA} optimization via ADMM

Inputs: Data matrices $\mathbf{M}_1, \ldots, \mathbf{M}_n$, initial parameters $(\mathbf{S}^{[0]}, \mathbf{Y}^{[0]}, \Theta^{[0]})$, penalty parameters α, β, μ.
Initialize $\mathbf{S} = \mathbf{S}^{[0]}$, $\mathbf{Y} = \mathbf{Y}^{[0]}$ and $\Theta = \Theta^{[0]}$. Set $t = 0$.
while not converge **do**
  $\mathbf{L}^{[t+1]} \leftarrow \mathcal{D}_{\alpha/\mu}(\Theta^{[t]} - \mathbf{S}^{[t]} + \frac{1}{\mu}\mathbf{Y}^{[t]})$
  $\mathbf{S}^{[t+1]} \leftarrow \mathcal{S}_{\beta/\mu}(\Theta^{[t]} - \mathbf{L}^{[t+1]} + \frac{1}{\mu}\mathbf{Y}^{[t]})$
  for $j = 1, \ldots, p$ and $k = 1, \ldots, q$
    $\theta_{j,k}^{[t+1]} \leftarrow \arg\min_{\theta_{j,k}} \zeta_{j,k}(\theta_{j,k}; L_{j,k}^{[t+1]}, S_{j,k}^{[t+1]}, Y_{j,k}^{[t]})$
  end for
  $\mathbf{Y}^{[t+1]} \leftarrow \mathbf{Y}^{[t]} + \mu(\Theta^{[t+1]} - \mathbf{L}^{[t+1]} - \mathbf{S}^{[t+1]})$
  Update $t \leftarrow t + 1$.
end while
Outputs: Optimized parameters $(\mathbf{S}^{[t]}, \mathbf{L}^{[t]}, \Theta^{[t]})$.

We further note that, while theoretical convergence guarantees are difficult to establish for Algorithm 1 (due in large part to the iterative estimation of the unknown natural parameter matrix), empirical experiments later suggest that the employed algorithm yields satisfactory convergence and optimization performance.

3.3.2.2 Multi-Group Setting

Consider next the multi-group e^{RPCA} problem (3.10), where there are multiple groups of observed matrices $\{\mathbf{M}_i^{[1]}\}_{i=1}^{n_1}, \{\mathbf{M}_i^{[2]}\}_{i=1}^{n_2}, \ldots, \{\mathbf{M}_i^{[G]}\}_{i=1}^{n_G}$ that share a common low-rank structure $\mathbf{L}$ but different sparse anomalies $\mathbf{S}_1, \ldots, \mathbf{S}_G$. We adopt a similar augmented Lagrangian form for optimization, given by:

$$\begin{aligned}\min_{\mathbf{L},\mathbf{S}_1,\ldots,\mathbf{S}_G,\Theta} & \sum_{j=1}^{p}\sum_{k=1}^{q}\left(\sum_{g=1}^{G}\sum_{i=1}^{n_g}\frac{l(\theta_{j,k}^{[g]}; M_{i,j,k}^{[g]})}{n_g}\right) + \alpha\|\mathbf{L}\|_* \\ & + \sum_{g=1}^{G}\left\{\beta_g\|\mathbf{S}_g\|_1 + \langle \mathbf{Y}_g, \Theta_g - \mathbf{L} - \mathbf{S}_g\rangle_F + \frac{\mu}{2}\|\Theta_g - \mathbf{L} - \mathbf{S}_g\|_F^2\right\}.\end{aligned} \tag{3.21}$$

Similar to before, $\mathbf{Y}_1, \ldots, \mathbf{Y}_G$ are Lagrange multiplier matrices, and $\mu > 0$ is a constant.

The optimization problem (3.21), unfortunately, is harder to solve than the single-group problem (3.12). This is due to the fact that, as the low-rank structure $\mathbf{L}$ is shared over all groups, its optimization, given fixed Θ, $\mathbf{S}_1, \ldots, \mathbf{S}_G$, is no longer in closed form. We thus adopt the following two-stage algorithm to find an approximate solution. For Stage 1, we

Algorithm 2 Multi-group e^{RPCA} optimization via ADMM

Inputs: Data matrices $\{\mathbf{M}_i^{[1]}\}_{i=1}^{n_1}, \ldots, \{\mathbf{M}_i^{[G]}\}_{i=1}^{n_G}$, initial parameters for Stage 1 $\{(\mathbf{S}^{[0]}, \mathbf{Y}^{[0]}, \Theta^{[0]})\}$ and Stage 2 $\{(\mathbf{S}_g^{[0]}, \mathbf{Y}_g^{[0]}, \Theta_g^{[0]})\}_{g=1}^{G}$, penalty parameters α, β, μ.

Stage 1: Let $\widetilde{\mathbf{L}}$ be the low-rank structure with common anomalies $\mathbf{S}$ and parameters Θ.
Initialize $\mathbf{S} = \mathbf{S}^{[0]}$, $\mathbf{Y} = \mathbf{Y}^{[0]}$ and $\Theta = \Theta^{[0]}$.
Optimize $\widetilde{\mathbf{L}}$ using Algorithm 1.
Stage 2: Fix $\mathbf{L} = \widetilde{\mathbf{L}}$.
for $g = 1, \ldots, G$
  Initialize $\mathbf{S}_g = \mathbf{S}_g^{[0]}, \mathbf{Y}_g = \mathbf{Y}_g^{[0]}, \Theta_g = \Theta_g^{[0]}$. Set $t = 0$.
  while not converge **do**
    $\mathbf{S}_g^{[t+1]} \leftarrow \mathcal{S}_{\beta_g/\mu}(\Theta_g^{[t]} - \mathbf{L} + \frac{1}{\mu}\mathbf{Y}_g^{[t]})$
    for $j = 1, \ldots, p$ and $k = 1, \ldots, q$
      $\theta_{g,j,k}^{[t+1]} \leftarrow \arg\min_{\theta_{j,k}} \zeta_{g,j,k}(\theta_{g,j,k}; L_{j,k}, S_{g,j,k}^{[t+1]}, Y_{g,j,k}^{[t]})$
    end for
    $\mathbf{Y}_g^{[t+1]} = \mathbf{Y}_g^{[t]} + \mu(\Theta_g^{[t+1]} - \mathbf{L} - \mathbf{S}_g^{[t+1]})$
    Update $t \leftarrow t + 1$.
  end while
end for
Outputs: Optimized parameters $\{(\mathbf{S}_g^{[t]}, \Theta_g^{[t]})\}_{g=1}^{G}, \mathbf{L}$.

obtain an *estimate* $\widetilde{\mathbf{L}}$ of the low-rank matrix $\mathbf{L}$ under the approximation $\mathbf{S} = \mathbf{S}_1 = \cdots = \mathbf{S}_G$, i.e., all groups have the same sparse anomalies, and thus $\Theta = \Theta_1 = \cdots = \Theta_G$. This can be achieved via a direct application of the earlier single-stage ADMM algorithm. The optimization of $\widetilde{\mathbf{L}}$ (given common parameters Θ and common anomalies $\mathbf{S}$) yields the closed-form singular value thresholding update:

$$\widetilde{\mathbf{L}}^{[t]} \leftarrow \mathcal{D}_{\alpha/\mu}\left(\Theta^{[t]} - \mathbf{S}^{[t]} + \frac{1}{\mu}\mathbf{Y}^{[t]}\right). \tag{3.22}$$

Similarly, the optimization of $\mathbf{S}$ (given $\widetilde{\mathbf{L}}$ and Θ) and Θ (given $\widetilde{\mathbf{L}}$ and $\mathbf{S}$) yields closed-form updates from (3.15) and (3.19), respectively. For Stage 2, with $\mathbf{L}$ *fixed* at this estimated $\widetilde{\mathbf{L}}$, we then cyclically optimize the group-dependent sparse anomaly matrices $\mathbf{S}_1, \ldots, \mathbf{S}_G$ and parameter matrices $\Theta_1, \ldots, \Theta_G$ via a similar ADMM algorithm on (3.21); such closed-form updates are provided in Algorithm 2. The detailed steps for this two-stage optimization procedure are outlined in Algorithm 2.

3.3.3 Hyperparameter Tuning

Finally, careful tuning of the hyperparameters α, β, and μ is needed for accurate recovery of the low-rank structure and sparse anomalies. For the standard RPCA, (Candès et al., 2011) showed that with $\alpha = 1$, $\beta = 1/\sqrt{\max(p,q)}$ and $\mu = pq/(4||\Theta||_1)$, one achieves the theoretical recovery of $\mathbf{L}$ and $\mathbf{S}$ in an asymptotic sense. We found that such a specification works reasonably well for the e^{RPCA} as well for both single-group and multi-group settings, in the absence of any prior knowledge on the rank of $\mathbf{L}$ or the degree of sparsity in $\mathbf{S}$. We note, however, that this specification presumes no noise in the observation of $\mathbf{L} + \mathbf{S}$; when large non-Gaussian noise is present, we found that a larger choice of μ may yield improved recovery performance.

In many applications, there may be guiding prior information on the rank of $\mathbf{L}$ or the sparsity level of $\mathbf{S}$, which can integrated for hyperparameter tuning. We illustrate how this can be done for the single-group setting; a similar strategy can be used for multiple groups. Suppose we have a desired upper bound on the rank $\text{rank}(\mathbf{L}) \leq r$ and the proportion of non-zero entries $\%\text{nz}(\mathbf{S}) := \#\{S_{j,k} \neq 0\}/(pq) < s$. Starting with an initial hyperparameter setting $\alpha^{[0]} = 1$, $\beta^{[0]} = 1/\sqrt{\max(p,q)}$ and $\mu = pq/(4||\Theta||_1)$, we first perform the e^{RPCA} optimization (Algorithm 1) with $\alpha = \alpha^{[0]}$ and $\beta = \beta^{[0]}$ to estimate the low-rank structure $\mathbf{L}$ and corresponding anomalies $\mathbf{S}$. If the rank of the estimated $\mathbf{L}$ exceeds r, we then iteratively increase α. Otherwise, if the proportion of sparse entries in the estimated $\mathbf{S}$ exceeds s, we then iteratively increase β. Algorithm 3 summarizes this hyperparameter tuning procedure.

3.3.4 Computational Complexity

Another appealing property of the e^{RPCA} is that, in addition to leveraging the underlying exponential family structure, it also permits closed-form efficient updates in the optimization algorithm. We investigate next the computational complexity of e^{RPCA} optimization algorithm, in terms of the data matrix dimensions p and q as well as sample size n.

For the single-group setting (Algorithm 1), each iterative update of $\mathbf{L}$ involves an SVD operation that requires $\mathcal{O}(pq\min(p,q))$ work. Each iterative update of $\mathbf{S}$ and $\mathbf{Y}$ requires $\mathcal{O}(pq)$ work. Each iterative update for the parameter $\theta_{j,k}$ (of which there are pq in total)

Algorithm 3 Hyperparameter tuning for single-group e^{RPCA}

Inputs: Hyperparameter step sizes $\eta_\alpha > 0$, $\eta_\beta > 0$.
Condition $(*)$: $\text{rank}(\mathbf{L}) \leq r$, $\%\text{nz}(\mathbf{S}) < s$

Initialize $\alpha^{[0]} = 1$, $\beta^{[0]} = 1/\sqrt{\max(p,q)}$. Set $t = 0$.
Optimize $(\mathbf{L}, \mathbf{S})$ with $\alpha = \alpha^{[0]}, \beta = \beta^{[0]}$ via Algorithm 2.
While $(*)$ is not satisfied **do**
  if $\text{rank}(\mathbf{L}) > r$ **then**
    $\alpha^{[t]} \leftarrow \alpha^{[t-1]} + \eta_\alpha \sqrt{t}$
  else $\alpha^{[t]} \leftarrow \alpha^{[t-1]}$
  end if
  if $\%\text{nz}(\mathbf{S}) > s$ **then**
    $\beta^{[t]} \leftarrow \beta^{[t-1]} + \eta_\beta \sqrt{t}$
  else $\beta^{[t]} \leftarrow \beta^{[t-1]}$
  end if
  Optimize $(\mathbf{L}, \mathbf{S})$ with $\alpha = \alpha^{[t]}, \beta = \beta^{[t]}$ via Algorithm 2.
  Update $t \leftarrow t + 1$.
  Stop if $\alpha^{[t]} = \alpha^{[t-1]}$ and $\beta^{[t]} = \beta^{[t-1]}$.
end while
Outputs: Optimized hyperparameters $(\alpha^{[t]}, \beta^{[t]})$.

requires $\mathcal{O}(1)$ work, but this involves a one-shot computation of the sample mean $\bar{\mathbf{M}} = n^{-1} \sum_{i=1}^{n} \mathbf{M}_i$ (see the closed-form updates in the Appendix) that requires $\mathcal{O}(npq)$ work. Summarizing the above, the single-group Algorithm 1 thus requires an initial $\mathcal{O}(npq)$ work for pre-processing, and $\mathcal{O}(pq \min(p,q))$ work per optimization iteration. For the multi-group setting (Algorithm 2), we need to consider both Stage 1 and Stage 2. For Stage 1, one can follow the above rationale to show that this requires an initial $\mathcal{O}(npq)$ work for pre-processing, and $\mathcal{O}(pq \min(p,q))$ work per optimization iteration. For Stage 2, the parameter updates within each group require $\mathcal{O}(pq)$ work, thus incurring a total work of $\mathcal{O}(Gpq)$ per optimization iteration.

With a relatively small sample size n and number of groups G, the key computational bottleneck for e^{RPCA} thus lies in the SVT updates that each require $\mathcal{O}(pq \min(p,q))$ work. This will not be too burdensome for data matrices with large p and small q (or vice versa), but may be time-consuming when both p and q are large. Luckily, with modern computing architecture, such operations can be greatly sped up via multi-thread processing and GPU acceleration; see, e.g., (Fatahalian et al., 2004; Kontoghiorghes, 2005). As this is the primary bottleneck for e^{RPCA}, we have found such tools to be greatly useful in scaling up

this decomposition approach for massive data matrices.

We also provide next a brief comparison of computational complexity with the standard RPCA and the ePCA. For the standard RPCA (3.1) (with appropriate modifications for the considered noisy setting with multiple data matrices; see Section 3.4 for details), the ALM approach requires the same running time per optimization iteration of $\mathcal{O}(pq\min(p,q))$. The ePCA incurs much higher computation relative to the other two methods, particularly for large data matrices. The key computational bottleneck in ePCA is the eigendecomposition of a $pq \times pq$ covariance matrix, which requires $\mathcal{O}((pq)^3)$ work. Clearly, for massive data matrices with both p and q large, the e^{RPCA} and standard RPCA are much more computationally efficient compared to the ePCA.

3.4 Numerical Experiments

We now explore the performance of the proposed e^{RPCA} in a suite of simulation experiments. Here, the underlying parameter matrix $\Theta \in \mathbb{R}^{p\times q}$ follows the presumed low-rank plus sparse decomposition $\Theta = \mathbf{L} + \mathbf{S}$. The low-rank matrix $\mathbf{L}$ is simulated by first generating a matrix with independent entries from the Gaussian distribution $\mathcal{N}(\zeta, \gamma^2)$, then truncating all but the largest $p/5$ singular values via SVD. The sparse matrix $\mathbf{S}$ is simulated with $p^2/20$ uniformly selected non-zero entries, where non-zero entry values are uniformly sampled from the interval $[L, U]$. The corresponding data matrices are then generated from Θ following several common non-Gaussian exponential family distributions, including the Bernoulli, Exponential, and Poisson distributions. Since the entries of Θ should be constrained to specific intervals depending on the choice of exponential family distribution, the simulation parameters ζ, γ^2, L, U need to be carefully chosen to adhere to such constraints; for brevity, we provided specific settings of these parameters in the Appendix.

For comparison, we adopt two baseline approaches. The first is the standard RPCA approach (Candès et al., 2011). Since *multiple* data matrices $\mathbf{M}_1, \ldots, \mathbf{M}_n$ are observed with *noise*, the RPCA formulation (3.1) can be modified as follows for fair comparison:

$$\min_{\mathbf{L},\mathbf{S},\blacksquare} \sum_{i=1}^{n}\sum_{j=1}^{p}\sum_{k=1}^{q} \frac{(M_{i,j,k} - \theta_{j,k})^2}{2n\hat{\sigma}^2} + \|\mathbf{L}\|_* + \lambda\|\mathbf{S}\|_1 \quad \text{s.t.} \qquad \Theta = \mathbf{L} + \mathbf{S}, \tag{3.23}$$

where $\hat{\sigma}$ is the sample standard deviation of the data matrices, and λ is set via recommended

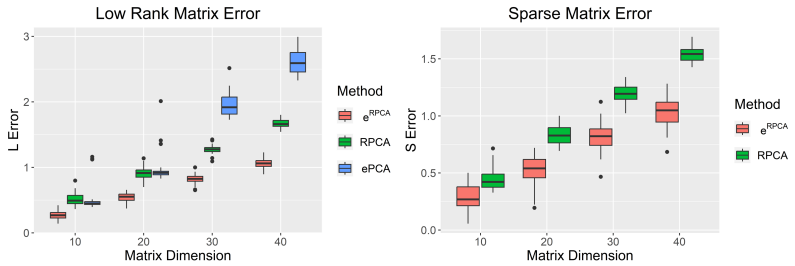
settings in (Candès et al., 2011). The formulation (3.23) makes two modifications in its first term, which account for standard Gaussian noise as well as multiple data matrices $\mathbf{M}_1, \ldots, \mathbf{M}_n$. This is analogous to the Stable-PCP formulation (3.3) from (Zhou et al., 2010) (for which we could not find an implementation online), where the constraint in (3.3) is relaxed via the first term in (3.23). This is then optimized using the aforementioned ALM approach (Z. Lin et al., 2010; Yuan & Yang, 2013) (see Section 3.2.1), with the recommended step size $\mu = pq/(4||\hat{\Theta}||_1)$ in (Candès et al., 2011), where $\hat{\Theta}$ is the maximum likelihood estimator (Casella & Berger, 2002) of Θ. The second baseline is the ePCA method (L. Liu et al., 2018), which can leverage the underlying non-Gaussian noise for extracting low-rank structure, but not for detection of sparse anomalies. These methods are compared with the e^{RPCA} for the single-group and multi-group settings.

3.4.1 Single-Group Setting

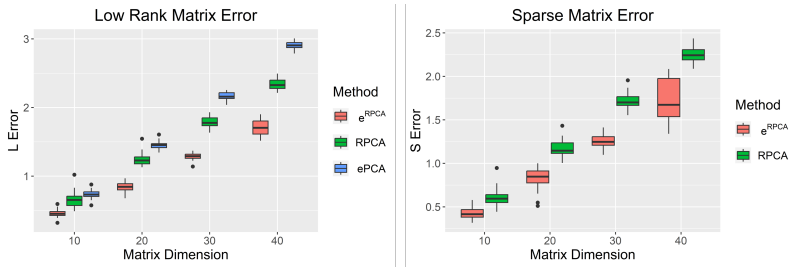
In the single-group experiments, we generated the low-rank matrix $\mathbf{L}$ and sparse anomalies $\mathbf{S}$ as described previously, then simulated $n = 500$ data matrices $\mathbf{M}_1, \ldots, \mathbf{M}_n$ from the parameter matrix $\Theta = \mathbf{L} + \mathbf{S}$. We then compared each method on its recovery of the low-rank structure $\mathbf{L}$ and the sparse anomalies $\mathbf{S}$ in terms of Frobenius error. These experiments are performed for matrices of dimensions $p \times p$, $p = 10, 20, 30$ and 40, with each setting replicated for 30 trials.

3.4.1.1 Bernoulli Distribution

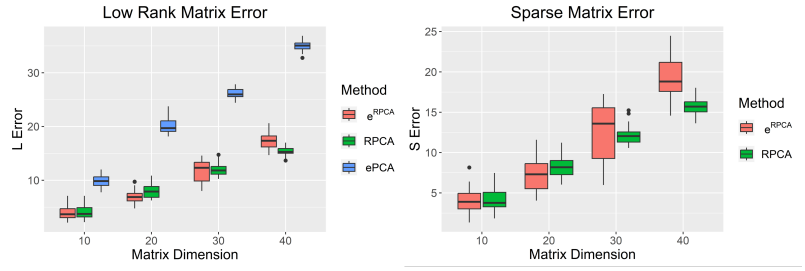
Consider first noise drawn from the Bernoulli distribution, where each matrix entry follows $M_{i,j,k} \overset{indep.}{\sim} \text{Bern}(\theta_{j,k})$. Fig. 3.3a shows the recovery errors for $\mathbf{L}$ and $\mathbf{S}$ (in Frobenius norm) using the proposed e^{RPCA}, the standard RPCA (with appropriate modifications detailed earlier) and the ePCA; note that the latter does not provide an estimate of $\mathbf{S}$. For $\mathbf{L}$, we see that the e^{RPCA} yields a noticeably improved recovery of the low-rank structure compared to existing methods, particularly as dimension p increases. Similarly, for $\mathbf{S}$, we observe an improved recovery of the underlying sparse anomalies for the e^{RPCA} compared to RPCA, with this improvement again growing as dimension p increases. This suggests, that with prior information on the underlying non-Gaussian noise, the integration of such information can greatly improve the joint recovery of both the underlying low-rank structure



(a) Bernoulli distribution.



(b) Exponential distribution.

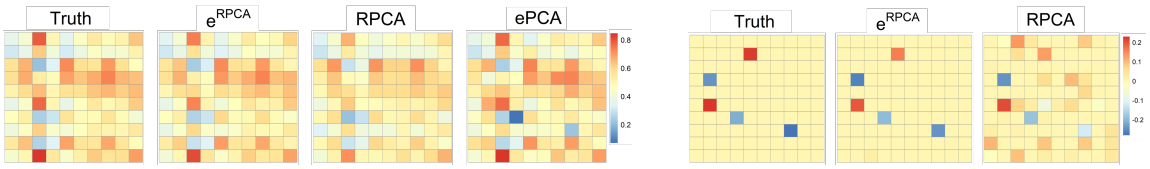


(c) Poisson distribution.

FIGURE 3.3: Boxplots of recovery errors (in Frobenius norm) for the low-rank matrix **L** (left), and sparse anomalies **S** (right) as a function of matrix dimension p for single-group simulations across different distributions.

and sparse anomalies.

Figs. 3.4a and 3.4b visualize the recovery of **L** and **S**, respectively, for one simulation experiment in $p = 10$ dimensions. We see that the standard RPCA yields a noticeably poorer recovery of the true **L** compared to e^{RPCA}, which is unsurprising as it does not account for the underlying non-Gaussian noise form. This, in turn, resulted in the erroneous detection of many "anomalies" that were not truly anomalies. The ePCA, while providing slightly better recovery of **L** to the RPCA, can be seen to be highly sensitive to the underlying sparse anomalies, resulting in significant deterioration of the recovery. The proposed



(a) The true and recovered **L**.

(b) The true and recovered **S**.

FIGURE 3.4: Visualizing the true and recovered low-rank matrix **L** and sparse anomalies **S** in one simulation in $p = 10$ dimensions, for the single-group Bernoulli simulations.

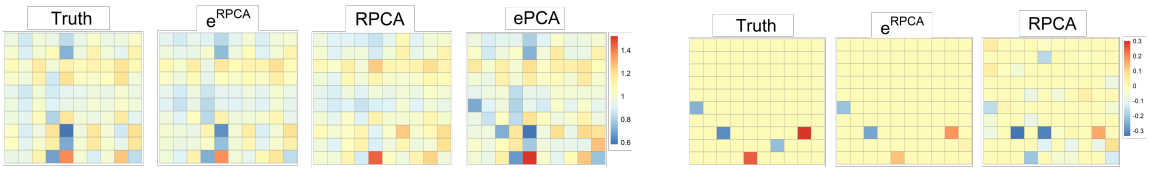
e^{RPCA}, by incorporating the underlying non-Gaussian noise within the desired low-rank plus sparse decomposition, yields improved recovery of both the low-rank structure **L** and the sparse anomalies **S**.

3.4.1.2 Exponential Distribution

Consider the case where noise is drawn from the Exponential distribution; thus each entry follows $M_{i,j,k} \overset{indep.}{\sim} \text{Exp}(\theta_{j,k})$ and $\theta_{j,k}$ is its rate parameter. Here, since the sparse decomposition arises in its rate parameter (which is inversely proportional to its mean), we performed the RPCA after taking the entry-wise inverse of the data matrices. Fig. 3.3b shows the reconstruction errors for **L** and **S** (in Frobenius norm) using the e^{RPCA}, the standard RPCA and the ePCA, where again the latter does not provide an estimate of **S**. As before, the e^{RPCA} yields improved recovery of both the low-rank structure **L** and the sparse anomalies **S**, with this improvement gap growing as dimension p increases. Figs. 3.5a and 3.5b visualize the recovery of **L** and **S**, respectively, for a simulation experiment in $p = 10$ dimensions. The standard RPCA, which does not factor for non-Gaussian noise, can be seen to yield poorer recovery of **L** and erroneous detection of numerous false "anomalies". The ePCA offers a slightly better recovery of **L**, but this is highly corrupted by the underlying sparse outliers. By integrating both non-Gaussian noise and sparse anomalies within an efficient decomposition framework, the e^{RPCA} facilitates an accurate recovery of both **L** and **S**.

3.4.1.3 Poisson Distribution

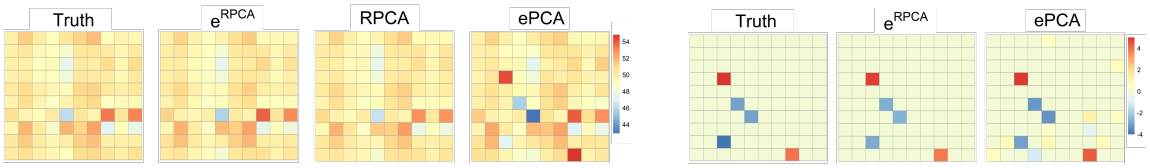
Finally, consider noise drawn from the Poisson distribution, where each matrix entry follows $M_{i,j,k} \overset{indep.}{\sim} \text{Pois}(\theta_{j,k})$. Fig. 3.3c shows the recovery errors for **L** and **S** using the



(a) The true and recovered **L**.

(b) The true and recovered **S**.

FIGURE 3.5: Visualizing the true and recovered low-rank matrix **L** and sparse anomalies **S** in one simulation in $p = 10$ dimensions, for the single-group Exponential simulations.



(a) The true and recovered **L**.

(b) The true and recovered **S**.

FIGURE 3.6: Visualizing the true and recovered low-rank matrix **L** and sparse anomalies **S** in one simulation in $p = 10$ dimensions, for the single-group Poisson simulations.

compared methods. Here, we see that the e^{RPCA} and standard RPCA have comparable performance, with both having considerably lower errors than the ePCA. In particular, e^{RPCA} yields slightly lower errors for both **L** and **S** in lower dimensions, while the standard RPCA performs slightly better in higher dimensions. One likely reason for this is that, with sufficiently large rate parameters θ, the resulting Poisson noise can be well-approximated by a Gaussian distribution (Feller, 1968). Thus, for Poisson noise, both the e^{RPCA} and the standard RPCA yield good performance. Figs. 3.6a and 3.6b visualize the recovered **L** and **S** for an experiment in $p = 10$ dimensions. We see that the e^{RPCA} and the standard RPCA both provide good recovery of both the low-rank structure and sparse anomalies, with the standard RPCA again erroneously identifying more false "anomalies".

3.4.2 Multi-Group Setting

Next, we performed multi-group experiments with $G = 2$ groups. We generated the low-rank matrix **L** and sparse anomalies $\mathbf{S}_1$ and $\mathbf{S}_2$ as before, then simulated $n_1 = n_2 = 250$ data matrices for each group, with $\Theta_1 = \mathbf{L} + \mathbf{S}_1$ and $\Theta_2 = \mathbf{L} + \mathbf{S}_2$. We then compared methods on its recovery of the low-rank structure **L** and the sparse anomalies **S** in terms of Frobenius error. These experiments are again performed for $p \times p$ matrices, where $p = 10, 20, 30$ and

40, with each setting replicated for 30 trials.

3.4.2.1 Bernoulli Distribution

Consider first the multi-group setting with Bernoulli noise. Fig. 3.7a shows the recovery errors for $\mathbf{L}$ and $\mathbf{S}$ using the e^{RPCA}, the RPCA, and the ePCA, where again the latter does not provide an estimate for $\mathbf{S}$. We see that, for $\mathbf{L}$, the e^{RPCA} offers improved recovery to competing methods, with the improvement growing larger as dimension p increases. Furthermore, the e^{RPCA} appears to yield greater improvement in this multi-group experiment compared to the earlier single-group experiment. This is not too surprising, as these competing methods do not factor in the different sparse anomalies between different groups. Similar observations can be made for the recovery of sparse anomalies $\mathbf{S}$.

3.4.2.2 Exponential Distribution

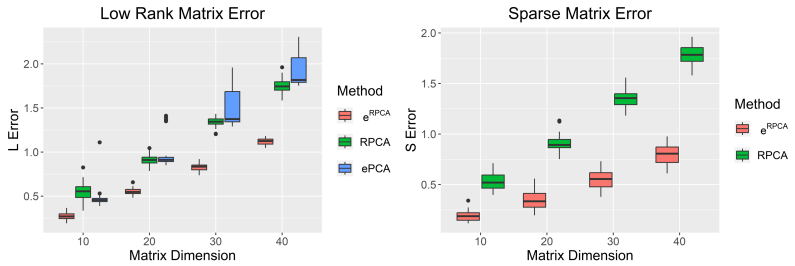
Consider next the multi-group setting with Exponential noise. Fig. 3.7b shows the recovery errors for $\mathbf{L}$ and $\mathbf{S}$ using e^{RPCA} and competing methods. For $\mathbf{L}$, we again see a marked improvement for the proposed method, with this improvement growing as dimension p increases. As before, these improvements appear more pronounced for the multi-group set-up compared to the single-group set-up. Similar observations hold for the recovery of anomalies in $\mathbf{S}$.

3.4.2.3 Poisson Distribution

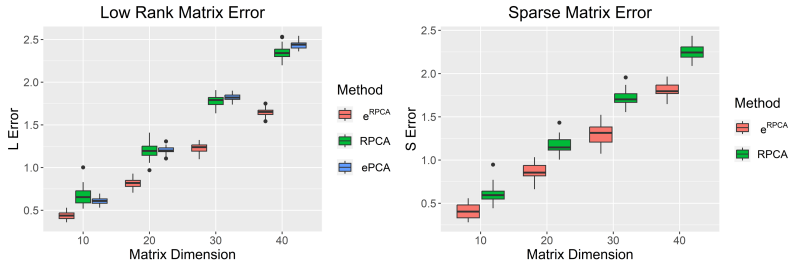
Finally, consider the multi-group setting with Poisson noise. Fig. 3.7c shows the recovery errors for $\mathbf{L}$ and $\mathbf{S}$. Interestingly, while the earlier single-group experiments showed comparable results for the e^{RPCA} and RPCA, the multi-group experiments show a noticeable improvement for the e^{RPCA}. This can again be explained by the presence of different sparse anomalies within different groups, which is not accounted for in the standard RPCA. Similar conclusions hold for recovering the sparse anomalies $\mathbf{S}$.

3.4.3 Recommendations

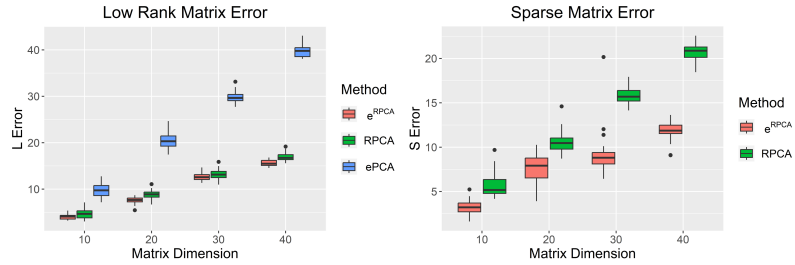
We provide here a brief recommendation summarizing when the e^{RPCA} is expected to outperform existing methods, following the simulation results above. First, when the underlying noise distribution is known or can be reliably inferred (e.g., from domain knowl-



(a) Bernoulli distribution.



(b) Exponential distribution.



(c) Poisson distribution.

FIGURE 3.7: Boxplots of recovery errors (in Frobenius norm) for the low-rank matrix **L** (left), and sparse anomalies **S** (right) as a function of matrix dimension p for multi-group simulations across different distributions.

edge), the use of such information can greatly improve performance for structure recovery and anomaly detection, particularly when such a distribution is markedly non-Gaussian. Second, when this distribution is elicited well, the improvement of the e^{RPCA} over the state-of-the-art can grow larger as matrix dimensions increase. Finally, when present, the use of multi-group information (where sparse anomalies may change from group to group) can greatly improve recovery performance via the proposed e^{RPCA}.

3.5 Applications

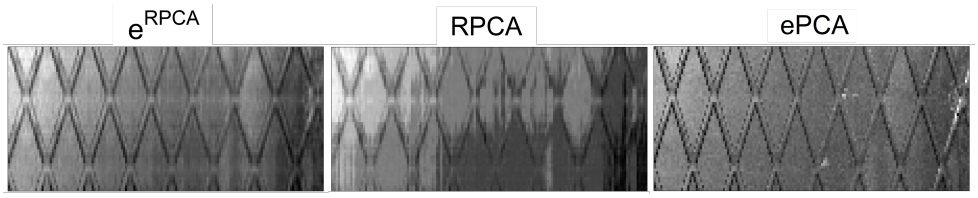
We now explore the use of the proposed e^{RPCA} in two practical applications. The first is our motivating problem on steel defect detection, and the second is a crime monitoring application in the city of Atlanta.

3.5.1 Steel Defect Detection

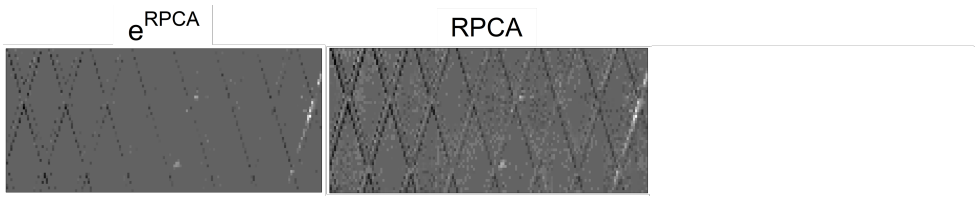
Consider first the motivating steel defect detection problem from Section 3.2.3. Steel manufacturing is essential in many facets of modern manufacturing, including the production of automobiles, electronics, furniture, infrastructure, and shipbuilding. The automated monitoring of steel defects, e.g., gashes, dents, or other inhomogeneities, thus plays a critical role in maintaining high product quality at low operation costs. Defects are typically detected via careful monitoring of images of the steel sheets taken from high-frequency cameras. Recent studies, e.g., (Chan, 2022), have shown that quanta image sensing (QIS, Charbon, 2014) may offer improved higher frequency imaging with lower read noise over more conventional multi-bit imaging systems (e.g., complementary metal-oxide semiconductor imaging (Charbon, 2014)). The key challenge in defect detection using QIS is that its image intensities take the form of binary bits, which can be well-modeled via i.i.d. Bernoulli noise (Chan, 2022); this thus presents an appropriate application for the e^{RPCA}.

We adopt the same set-up as Section 3.2.3 for numerical experiments. First, an uncorrupted steel sheet image with visible defects is taken from Severstal (PAO Severstal, 2019) (see Fig. 3.1 left). Next, to mimic QIS, $n = 500$ binary images are generated by normalizing the uncorrupted image and sampling via i.i.d. Bernoulli noise (see Fig. 3.1 right). The compared methods include the proposed (single-group) e^{RPCA}, the standard RPCA (with modifications as discussed in Section 3.4), and the ePCA. For the e^{RPCA}, we made use of the default hyperparameter specification in Section 3.3.3, i.e., without any prior knowledge of rank or degree of sparsity.

Fig. 3.8a shows the recovered low-rank structure $\mathbf{L}$ using the standard RPCA, ePCA, and the proposed e^{RPCA}. Here, the desired structure to recover is the underlying "criss-cross" background pattern. As observed previously, the first two existing methods yield a visually mediocre recovery of this criss-cross pattern; one reason may be that neither



(a) Estimated low-rank structure (background pattern) **L**.



(b) Estimated sparse anomalies (defects) **S**.

FIGURE 3.8: Visualizing (a) the estimated low-rank structure (background pattern) **L** and (b) the recovered sparse anomalies (defects) **S**, using the e^{RPCA}, RPCA and ePCA in the steel defect detection application.

method accounts for the *joint* presence of sparse anomalies with non-Gaussian noise. The e^{RPCA}, by factoring in both properties, in turn, provides a noticeably smoother recovery of the cross-cross pattern without defects.

Fig. 3.8b shows the recovered sparse anomalies **S** using RPCA and e^{RPCA}; note that the ePCA does not provide an estimate of **S**. From Fig. 3.1, the desired anomalies to recover include a large gash on the right and two smaller inhomogeneous spots in the middle. We see that the RPCA returns a rather muddled recovery of such defects: while the large right gash is noticeable visually, the recovered **S** also picks up on the background criss-cross structure, which obfuscates other defects. Comparatively, the e^{RPCA} yields improved recovery of the underlying defects: it picks up not only the clear right gash and the more subtle inhomogeneities in the middle but also darker discolorations along the criss-cross structure on the left. The latter defects were not immediately evident at first glance from Fig. 3.1, but are indeed present upon further inspection from this analysis. Thus, by integrating information on non-Gaussian noise, the proposed e^{RPCA} appears capable of jointly recovering low-rank structures and identifying sparse defects for steel monitoring.

3.5.2 Crime Mapping in Atlanta

Next, we investigate the use of the e^{RPCA} for a crime monitoring application in the city of Atlanta. The identification of *regular* (background) and *irregular* (anomalous) criminal activities are clearly critical tasks for improving public safety: it helps inform law enforcement of hotspots and/or peak times for different crimes (Eck et al., 2005), and facilitates the diagnosis of abnormal crime spikes. As such, a key objective here is to identify regular and irregular crime patterns over different geographical regions and times.

Our data consists of reported burglaries from the Atlanta Police Department over two years: 2015 and 2016. Each burglary is recorded along with its hour of occurrence, as well as its spatial location in the form of police "beats" (or zones) for patrol, of which there are 79 beats in total. Given the fine spatiotemporal scales for the recorded burglaries, there is typically at most one recorded crime within each spatiotemporal "window", i.e., a combination of occurrence hour and police beat. Prior studies (Frith et al., 2022; Nobles et al., 2016) suggest that *weekly* recurrent trends may be common for burglaries, which we leverage next for specifying the underlying low-rank structure.

Since there is no ground truth available for $\mathbf{L}$ and $\mathbf{S}$ here, we forgo a full comparison with existing methods and instead investigate the extraction of useful burglary patterns (both regular and irregular) from the proposed e^{RPCA}. Since there is at most one recorded crime within nearly all spatiotemporal windows, we thus adopt a binarization of this data ("0" for no burglaries within the window, "1" for at least one burglary). With this, e^{RPCA} then proceeds using the Bernoulli distribution, with the underlying probability matrix Θ presumed to follow the low-rank plus sparse decomposition. We employ here the multi-group e^{RPCA} set-up, with $G = 2$ groups for the summer and winter seasons (more on this next). To incorporate information on weekly trends, the data matrices $\{\mathbf{M}_i^{[g]}\}_{i=1}^{n_g}$ are constructed week-by-week; each matrix is thus of dimensions 79×168, where $p = 79$ is the number of police beats, and $q = 24 \times 7 = 168$ is the number of hours in a week. We then take $n_g = 24$ weeks (12 weeks per season $\times$ 2 years) of reported crime data for both the summer and winter seasons. With this set-up, the low-rank matrix $\mathbf{L}$ models for weekly (regular) crime activity, and the sparse matrices $\mathbf{S}_1$ and $\mathbf{S}_2$ account for irregular burglary activity over the summer and winter seasons, respectively.

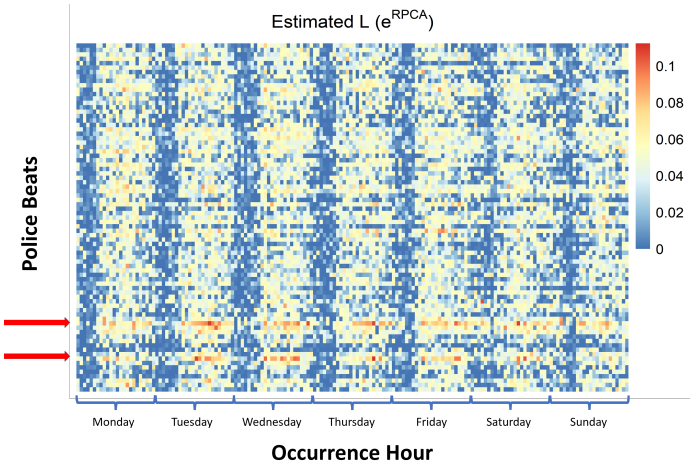
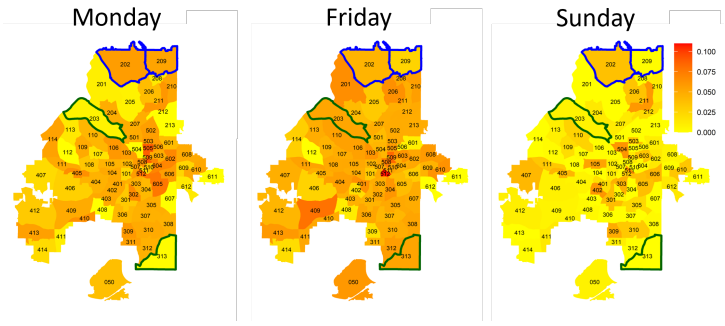


FIGURE 3.9: Visualizing the estimated low-rank matrix $\mathbf{L}$ from e^{RPCA} for the crime monitoring application. The two red arrows highlight two police beats: Beat 512 (downtown Atlanta) and Beat 509 (Midtown Atlanta).

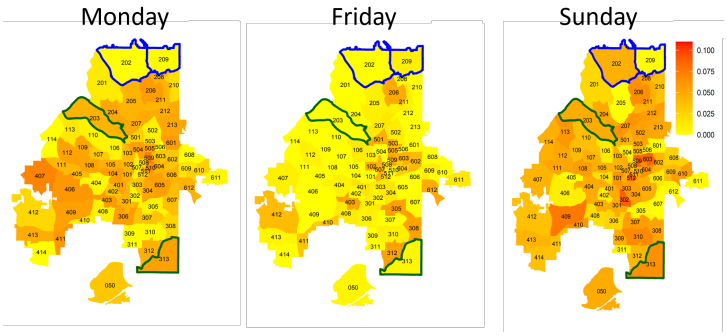
Fig. 3.9 shows the recovered low-rank matrix $\mathbf{L}$ using the two-group e^{RPCA}. We immediately see seven bright vertical bands, which suggest the presence of a daily trend in crime activity. In particular, for most beats, we observe an increased probability of burglary after dawn, which peaks during the day and decreases during the evening. Several beats, such as Beat 512 (downtown Atlanta) or 509 (Midtown Atlanta), have noticeably higher burglary rates compared to other beats, which is expected as such areas are highly urban and dense in terms of population. There also seems to be some interactions between occurrence hour and police beat, in that some beats have notably different peak hours than other beats.

Fig. 3.10 visualizes the spatial distribution of the estimated low-rank matrix $\mathbf{L}$ at various hours in the week. Certain beats, such as Beat 202 and 209 (outlined in blue), have noticeably higher burglary rates on Monday mornings, whereas other beats, such as Beat 203 or 313 (outlined in green), experience higher rates in the evenings. Upon further inspection, this is quite intuitive: the former beats (in blue) are primarily residential areas, where burglaries are expected to occur more often during weekday mornings (e.g., when individuals are at work), whereas the latter beats (in green) are primarily business districts, where burglaries typically occur more frequently at night (e.g. when businesses are closed). The estimated $\mathbf{L}$ reflects such intuitive patterns, thus suggesting the proposed e^{RPCA} can indeed recover useful insights for diagnosis.

Next, Fig. 3.11 visualizes the estimated sparse matrices $\mathbf{S}_1$ and $\mathbf{S}_2$, which models for



(a) Spatial visualization of burglary rates at 9am.



(b) Spatial visualization of burglary rates at 10pm.

FIGURE 3.10: Spatial visualization of the burglary rate from the estimated **L** on Monday, Friday, and Sunday at (a) 9 a.m. and (b) 10 p.m. for the crime monitoring application. Beats 202 and 209 (outlined in blue) have noticeably higher rates on Monday mornings, whereas Beats 203 or 313 (outlined in green) experience higher rates in the evenings.

anomalous (irregular) burglary activities over the summer and winter seasons, respectively. From a quick inspection, there appear to be some differences in anomalies between the two seasons, which supports our multi-group approach. To explore this difference, Fig. 3.12 plots the non-zero entries in $\mathbf{S}_1$ and $\mathbf{S}_2$ over Beats 403 (a residential area) and 503 (a commercial area). For Beat 403, it appears that after factoring in the weekly trends in **L**, it experiences more anomalous burglaries in the summer than in the winter during the evenings. A plausible explanation is that, in residential areas, burglaries during the daytime become more difficult due to more people being at home from summer break or hot weather. Similarly, for Beat 503, one observes a greater quantity of irregular burglaries in the summer than in the winter. A likely reason is that commercial areas typically experience a large influx of tourists and visitors, which may create greater opportunities for burglaries. Such

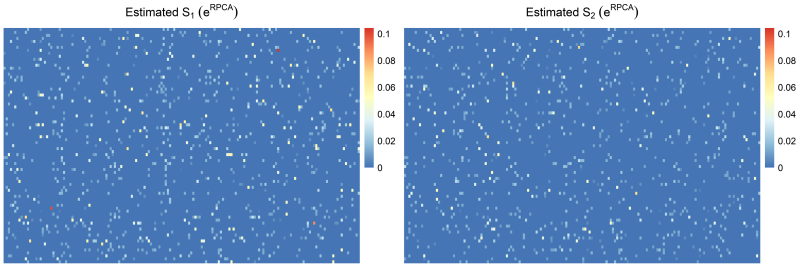


FIGURE 3.11: Visualizing the absolute values of the estimated anomaly matrices $\mathbf{S}_1$ (summer) and $\mathbf{S}_2$ (winter) from e^{RPCA} for the crime monitoring application.

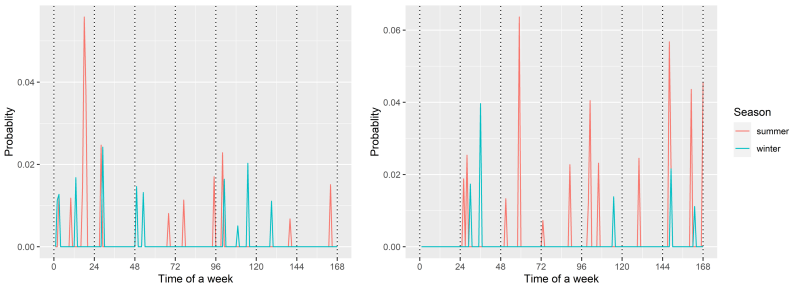


FIGURE 3.12: Plotting the non-zero entries in $\mathbf{S}_1$ (summer) and $\mathbf{S}_2$ (winter) for Beat 403 (left, residential) and Beat 503 (right, commercial) from e^{RPCA} for the crime monitoring application.

insights suggest that the proposed e^{RPCA} can indeed extract interpretable and useful insights for crime monitoring.

4. BLAST: Bayesian Online Structure-aware Change-point Detection

4.1 Introduction

The timely online detection of sudden local changes within massive image data is critical in a myriad of modern applications, from high-energy physics to surveillance. For such applications, however, there are three critical challenges that hinder effective change-point detection in practice. Consider, for example, the real-time monitoring of a high-security region, where one wishes to quickly detect sudden disturbances within the region. The first challenge (i) arises from the the *high-dimensional* nature of images from camera systems, which can consist upwards of thousands of pixels. To make timely decisions, a monitoring method thus needs to manipulate such high-dimensional images in a *scalable* fashion. The second challenge (ii) is stems from the observation that such changes typically involve *local* image structure, e.g., changes in edges and/or shapes. The timely detection of small local changes within large high-dimensional images is thus a difficult statistical problem. Finally (iii), a reliable quantification of *uncertainty* is essential for robust detection of potential changes, particularly in guiding downstream decisions (e.g., diagnosis or maintenance). Such uncertainty is critical not only for surveillance in quantifying threat risk, but also for scientific applications in establishing confidence in potential findings. Similar issues of scalability, locality and uncertainty arise in broad scientific and engineering applications; more on this later.

Despite its urgent practical need, there is scant literature on online change-point detection methods that fully address the above three challenges. Following existing literature, a change-point here is defined as a sudden change in the signal distribution over time. Much work in this literature focus on the cumulative sum (CUSUM) approach in (Page, 1954), which serves as a foundational technique for sequential analysis and detection of shifts in the distribution of a process (Siegmund, 1985). In (Lorden, 1971), a problem of optimal stopping is formulated, and simple rules are proposed which are asymptotically

optimal in an appropriate sense. Using the framework of optimal stopping theory, (Poor & Hadjiliadis, 2008) provides a comprehensive overview of quickest detection algorithms. (Basseville & Nikiforov, 1993) focuses on the design and investigation of online change detection algorithms. (Tartakovsky et al., 2015) systematically develops the theory of sequential hypothesis testing and quickest change-point detection, and highlights important applications where these theoretical results can be effectively utilized. (Hotelling, 1947) introduces the Hotelling's T^2 test, which is a multivariate statistical method used in change point detection to identify shifts in the mean vector of a process over time. More recent approaches investigated the use of kernel methods Gretton et al., 2012; S. Li et al., 2015 and topological structure Zheng et al., 2023. The above literature, however, are frequentist in nature; given our need for reliable uncertainty quantification (UQ), a Bayesian approach is desirable here. An early work on Bayesian online change-point detection is (West, 1986), which proposed a Bayesian CUSUM approach that leverages cumulative Bayes' factors for detecting persistent parameter shifts. This is further extended in (Fearnhead & Liu, 2007) and (Adams & MacKay, 2007), which introduced the notion of posterior "run length", i.e., the length of time since the most recent change-point. More recent developments include (Knoblauch & Damoulas, 2018), which explored the notion of model selection for spatiotemporal models; (Turner et al., 2009), which investigated changes for hyperparameter optimization Gaussian; and (Garnett et al., 2009; Saatci et al., 2010), which explored this for Gaussian process models.

The above literature largely focuses on lower-dimensional signals, which may scale poorly for high-dimensional image data (Challenge (i)). Recent approaches have been explored for this challenging setting. This includes (Okhrin et al., 2021), which uses generalized likelihood ratios on local characteristics of image regions-of-interest for computational scalability. (Otto, 2021) explored parallelized multivariate exponentially weighted moving average control charts, which leverages careful segmentation of image data. (Yan et al., 2018) utilized the decomposition of image data streams into functional means and sparse anomalies, then applied sequential likelihood-ratio tests for anomaly monitoring. Such

methods, however, do not provide a measure of monitoring uncertainty (Challenge (iii)), and also do not incorporate local image structure (e.g., edges and/or shapes) for change detection (Challenge (ii)), which can greatly diminish detection power for high-dimensional images.

Much of the work on integrating local image structure (Challenge (ii)) arises in the image processing literature, particularly on the task of video anomaly detection. To model embedded image structure, these methods largely employ variations of convolutional neural networks (CNNs); see (He et al., 2015; Kiela & Bottou, 2014). Given the large body of work on CNNs in machine learning (Krizhevsky et al., 2012; J. Wang et al., 2020), such CNN-based approaches are highly scalable for large image data, addressing Challenge (i). (Sultani et al., 2018) proposed to learn anomaly through the deep multiple instance ranking framework by leveraging weakly labeled training videos, i.e. the training labels (anomalous or normal) are at video-level instead of clip-level. (Luo et al., 2017) investigated anomaly detection in videos, by leveraging a Convolutional Neural Network for appearance encoding for each frame, and leveraging a Convolutional Long Short Term Memory for memorizing all past frames which corresponds to the motion information. A more recent work (Doshi & Yilmaz, 2021) explored a multi-objective deep-learning-based modeling framework for online anomaly detection. Such approaches, however, do not offer the desired quantification of monitoring uncertainty (Challenge (iii)) that naturally arises from a Bayesian procedure. Furthermore, for such deep models, the lack of uncertainty may result in overconfidence in the fitted model, resulting in erratic detection performance with many false alarms; we will see this later.

To jointly address Challenges (i)-(iii), we propose a new Bayesian onLine STructure-Aware change deTection (BLAST) procedure. BLAST leverages the deep Gaussian Markov Random Field (GMRF) model in (Sidén & Lindsten, 2020) as the underlying probabilistic model for image data. This deep GMRF offers two advantages: it enjoys much of the same scalability as GMRFs (widely used for large-scale spatial statistics modeling; see, e.g., (Hartman & Hössjer, 2008; Nguyen et al., 2016; Simpson et al., 2012), and can be interpreted as a

multi-layer CNN architecture (thus providing a flexible framework for modeling local image structures). With an elicited deep GMRF image prior, BLAST then employs a carefully-constructed Bayesian CUSUM procedure for online monitoring, adapting the framework in (Adams & MacKay, 2007). With matrix algebra tricks, the BLAST posterior run length distribution can be computed in $\mathcal{O}(M^2)$ work at each time-step, where M is the number of image pixels. This permits scalability for high-dimensional images, while also providing the desired modeling of image structure and quantification of Bayesian uncertainty. We then compare BLAST with the state-of-the-art in a suite of numerical experiments, then explore its effectiveness in two applications, the first on street scene monitoring and the second on drift tube monitoring for particle physics experiments.

Markov random field (MRF) is an undirected graphical model which offers a robust framework for modeling spatial dependencies in various data types (Geman & Geman, 1984). In MRFs, each node interacts only with its direct neighbors, enabling effective local data modeling. MRFs have found success in a wide range of applications, including computer vision (S. Z. Li, 1995), image processing (Kashyap & Chellappa, 1983), and spatial statistics (Song et al., 2008).

A Gaussian Markov Random Field (GMRF) is a type of MRF where the nodes represent random variables with Gaussian distributions, and has been popular for modeling large datasets where only local interactions matter (Rue & Held, 2005). Images are lattice-based data with local dependencies and thus have clear connections with spatial statistics. The dominate approach is to model such spatial dependencies in images using GMRFs. However, the integration of GMRF for change detection has largely remained unexplored. One work (Roy et al., 2016) investigates a change point estimation problem in the context of high dimensional Markov random-field models. It focuses on a single change point impacting the global network structure of the underlying Markov random field. For example, in many social network applications, global structural changes in the network are of primary interest. However, in some applications, nodewise change point analysis would typically be preferred given that the change happens in a local region, for example, biomolecules (Kolar et al.,

2010) (e.g. genes, proteins or metabolites) or image defects (Zheng et al., 2024) (e.g., gashes and dents in steel sheet manufacturing).

To address this, we propose a new nodewise framework called Bayesian Online Structure-aware Change-point Detection (BLAST). BLAST extends the formulation of Adams and MacKay (Adams & MacKay, 2007) with a novel prior specification. Specifically, BLAST learns important image features (e.g., edges) using offline pre-change data via the deep GMRF, an extension of GMRF (more on this in Section 4.2.2), and then integrates the trained model within Bayesian change-point detection for scalable monitoring. We demonstrate the effectiveness of BLAST compared to existing methods in a suite of simulation experiments and in applications to street scene and drift tube chambers monitoring.

The rest of the paper is organized as follows. Section 4.2 provides an overview of (deep) GMRF and motivations. Section 4.3 outlines the BLAST methodology. Section 4.4 demonstrates the effectiveness of this method in numerical experiments. Section 4.5 applies the method to street scene and drift tube chamber monitoring.

4.2 Background and Motivation

We first provide a brief overview of GMRF, and the extended deep GMRF, then discuss two baseline methods, the Hotelling's T^2 statistic and the Multi-Objective Neural Anomaly Detector (MONAD) in (Doshi & Yilmaz, 2021), for high-dimensional and image change-point detection. We then motivate the proposed BLAST method via our motivating street scene monitoring application.

4.2.1 Gaussian Markov Random Field

A Gaussian Markov random field (GMRF) $\mathbf{x}$ is an M-dimensional Gaussian vector with mean $\boldsymbol{\mu}$ and precision matrix $\mathbf{Q}$, so that

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}^{-1}).$$

For each GMRF, there exists a graph with vertices $\mathcal{V}$ corresponding to the elements in $\mathbf{x}$, and edges $\mathcal{E}$ define the conditional independencies of elements. Given the neighbors

of an element in $\mathbf{x}$, the element is conditionally independent of all other elements in $\mathbf{x}$. Mathematically, every GMRF has the property

$$i, j \in \mathcal{E} \longleftrightarrow Q_{i,j} \neq 0, \quad \text{for all } i \neq j,$$

and resulting in a sparse precision matrix $\mathbf{Q}$. This is one key property of GMRFs, and the "local" dependence structure can be used to simplify computations, especially in high-dimensional problems.

In the realm of image processing and computer vision, GMRFs can be formulated using convolution operations, which capture the neighborhood relationships of a pixel, and thus implicitly represent the Markov property of the GMRF. The covariance structure of a GMRF is often spatially invariant, meaning the dependence between two pixels in the image only depends on their relative locations, not their absolute locations. The sparsity of the precision matrix $\mathbf{Q}$ is a direct result of the Markov property. In terms of convolution, this means that only a few terms (defined by the filter/kernel) are used to compute the new value of a pixel in the output image.

Here, we introduce the equivalent definitions of GMRFs with an example of the second-order intrinsic GMRF or thin-plate spline model (Rue & Held, 2005), which can be defined by $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, (\mathbf{G}^\top \mathbf{G})^{-1})$, with

$$G_{ij} = \begin{cases} 4 & \text{for } i = j \\ -1 & \text{for } i \sim j \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

where $i \sim j$ denotes adjacency, and the precision matrix $\mathbf{Q} = \mathbf{G}^\top \mathbf{G}$. The non-zero elements in $\mathbf{G}$ can be represented through the stencil

$$\mathbf{w}_{\mathbf{G}} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}. \tag{4.2}$$

An equivalent definition of this model is to first define $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and then define $\mathbf{x}$

through the inverse transform $\mathbf{z} = \mathbf{G}\mathbf{x}$. It is trivial to show that $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, (\mathbf{G}^\top \mathbf{G})^{-1})$. The last equivalent definition of this model can be written as, $\mathbf{Z} = \text{conv}(\mathbf{X}, \mathbf{w}_\mathbf{G})$, where $\mathbf{Z}$ and $\mathbf{X}$ are image representations of the vectors $\mathbf{z}$ and $\mathbf{x}$. This GMRF model is defined using the convolution operator with a single filter, and this observation motivates the definition of Deep GMRF (Sidén & Lindsten, 2020).

4.2.2 Deep Gaussian Markov Random Fields

Now, we define a linear Deep GMRF using a standard Gaussian vector $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and then define $\mathbf{x}$ through the inverse transform

$$\mathbf{z} = \mathbf{G}_{\boldsymbol{\theta}}\mathbf{x} + \mathbf{b}_{\boldsymbol{\theta}}$$

, where $\mathbf{G}_{\boldsymbol{\theta}}$ is an invertible matrix (Sidén & Lindsten, 2020). This model can be equivalently represented through a Convolutional Neural Network (CNN) with L layers, and the output of layer l is defined as

$$\mathbf{Z}_l = \text{conv}(\mathbf{Z}_{l-1}, \mathbf{w}_l) + \mathbf{b}_l, \tag{4.3}$$

where $\mathbf{w}_l$ denotes the convolutional filter to layer l, and $\mathbf{b}_l$ is the bias adjustment for layer l (Sidén & Lindsten, 2020). Figure 4.1 is the illustration of the Deep GMRF. The observed data $\mathbf{Y}$ is the noisy measurement of the latent DGMRF $\mathbf{X}$, also denoted as $\mathbf{Z}_0$. The model parameters $\boldsymbol{\theta} = \{(\mathbf{w}_l, \mathbf{b}_l) : l = 1, \cdots, L\}$ are represented by the DGMRF.

Given the definition of DGMRF, we can then write $\mathbf{x}$ as a GMRF with density

$$p(\mathbf{x}) = \frac{|\det(\mathbf{G}_{\boldsymbol{\theta}})|}{(2\pi)^{M/2}} \exp\Big(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \mathbf{G}_{\boldsymbol{\theta}}^\top \mathbf{G}_{\boldsymbol{\theta}}(\mathbf{x} - \boldsymbol{\mu})\Big), \tag{4.4}$$

with $\mathbf{G}_{\boldsymbol{\theta}} = \mathbf{G}_L \mathbf{G}_{L-1} \cdots \mathbf{G}_1$, and $\boldsymbol{\mu} = -\mathbf{G}_{\boldsymbol{\theta}}^{-1}\mathbf{b}_{\boldsymbol{\theta}}$. The determinant $\det(\mathbf{G}_{\boldsymbol{\theta}})$ can be computed as

$$\det(\mathbf{G}_{\boldsymbol{\theta}}) = \prod_{l=1}^{L} \det(\mathbf{G}_l), \tag{4.5}$$

The distribution of $\mathbf{x}$ is defined by a CNN that transforms the input image $\mathbf{X}$ to the output image $\mathbf{Z}$ which has iid. standard normal pixel values. The CNN captures the GMRF

structure, with the convolutional layers being instrumental in modeling local interactions between image pixels and thus capturing the Markov property of the GMRF.

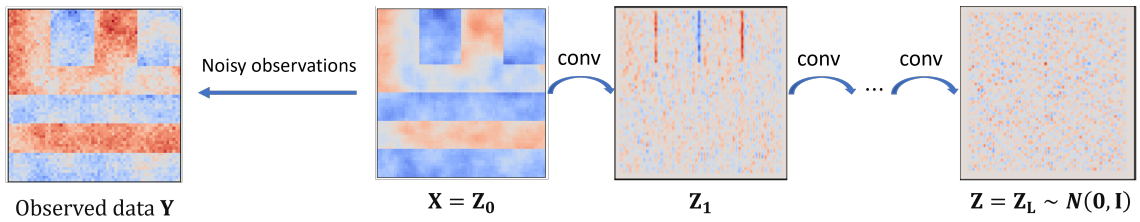


FIGURE 4.1: Illustration of the Deep GMRF. The observed data $\mathbf{Y}$ is the noisy measurement of the latent DGMRF $\mathbf{X}$, also denoted as $\mathbf{Z}_0$.

4.2.3 Existing Baseline Methods

A classic baseline change detection approach for multivariate data is the Hotelling's T^2 statistic (Hotelling, 1947), which makes use of the mean and covariance structure of data. Given data $\boldsymbol{x}_t \in \mathbb{R}^p$ at time t, $t = 1, \cdots, T$, the Hotelling's T^2 statistic is defined as

$$(\bar{\boldsymbol{x}} - \boldsymbol{\mu}_0)^\intercal \Sigma_0^{-1} (\bar{\boldsymbol{x}} - \boldsymbol{\mu}_0), \tag{4.6}$$

where $\bar{\boldsymbol{x}}$ is the sample mean vector, $\boldsymbol{\mu}_0$ and Σ_0 is the nominal mean vector and covariance matrix, respectively (this is typically given or estimated from reference data). The vanilla Hotelling's T^2 statistic is calculated using *only* data at the current time t, with all historical data discarded. To compute the test statistic more efficiently, it can be coupled with the CUSUM scheme (Page, 1954), which makes use of a cumulative sum of the statistic over time. The resulting detection statistic S_t^H is then given by:

$$S_t^H = (S_{t-1}^H)^+ + (\bar{\boldsymbol{\mu}}_{t-w,t} - \hat{\boldsymbol{\mu}}_0)^\intercal \widehat{\Sigma}_0^{-1} (\bar{\boldsymbol{\mu}}_{t-w,t} - \hat{\boldsymbol{\mu}}_0) - d^H, \quad S_0^H = 0, \tag{4.7}$$

where $(x)^+ = \max(x, 0)$, $\bar{\boldsymbol{\mu}}_{t-w,t}$ denotes the sample average of the data vectors $\{\boldsymbol{x}_{t-w}, \ldots, \boldsymbol{x}_t\}$, and $\hat{\boldsymbol{\mu}}_0$ and $\widehat{\Sigma}_0$ are the *pre-change* mean vector and covariance matrix estimated from historical data. Here, d^H is a drift parameter that can also be estimated using historical data. When the data is known to be concentrated on a linear subspace, one can adapt the Hotelling's T^2 test by first performing Principal Component Analysis (PCA), then using the resulting principal components as data within equation (4.7). With this, a change-point

is then declared at time t when the statistic S_t^H first exceeds a pre-specified threshold b.

The second baseline method, the Multi-Objective Neural Anomaly Detector (MONAD) in (Doshi & Yilmaz, 2021), integrates the deep learning techniques in the following way. The first module is to extract deep learning-based features, consisting of the future frame predicting error using a Generative Adversarial Network (GAN) (W. Liu et al., 2018), and location and appearance features using an object detection system called You Only Look Once (YOLO) (Redmon et al., 2016). The second module is to use the extracted features to construct a monitoring test statistics using the kNN (Euclidean) distance to the nominal space learned from training data. The proposed anomaly detection system computes the frame-level anomaly evidence δ_t:

$$\delta_t = \big(\max_i \{d_t^i\}\big)^m - \big(d_\alpha\big)^m, \tag{4.8}$$

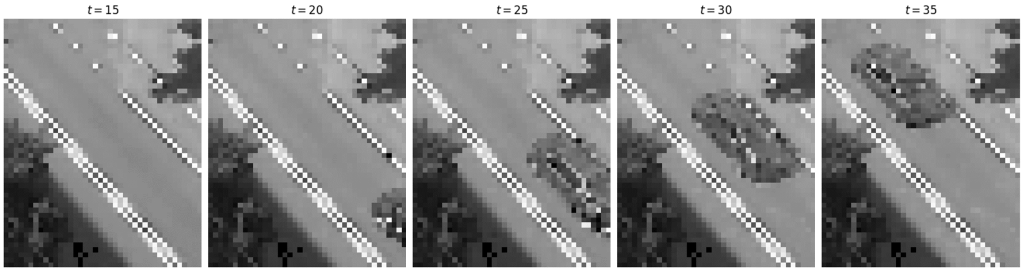
where m is the dimensionality of feature vector F_t^i for the detected object i, and d_α is the drift parameter that can also be trained using historical data under significance level α. Following a CUSUM-like procedure (Page, 1954), the test statistic S_t^M is defined as

$$S_t^M = \max\{S_{t-1}^M + \delta_t, 0\}, \quad S_0^M = 0. \tag{4.9}$$

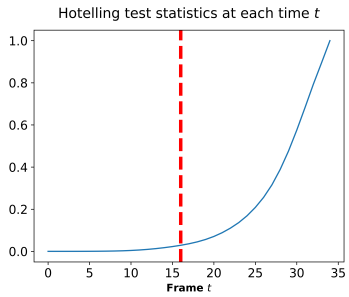
One then declares a change-point at time t when the test statistic S_t^M exceeds some pre-specified threshold h.

4.2.4 Street Scene Monitoring

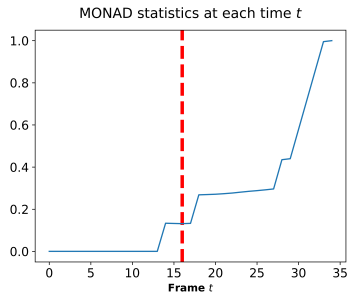
We investigate the above two baseline methods for a motivating street scene monitoring application (more on this in Section 4.5.1). Surveillance cameras are ubiquitous, and having humans monitor them constantly is not practical. In most cases, almost all of the video from a surveillance camera is unimportant and only unusual video segments are of interest. The data used here comprise $T = 35$ image snapshots taken from a static USB camera looking down on a scene of a two-lane street with bike lanes and pedestrian sidewalks (Ramachandra & Jones, 2020). Figure 4.2(a) shows snapshots of the street scene with a car entering starting at $t^* = 16$. We applied the above two baseline methods, with the



(a)



(b)



(c)

FIGURE 4.2: (a) Snapshots of the street scene with a car driving starting at $t^* = 16$. (b,c) Hotelling's T^2 and MONAD statistics at each time t; the true change-point t^* is marked in red.

Hotelling's T^2 applied by first vectorizing the image data, then applying PCA to extract the top 15 principal components. Figures 4.2(b) and 4.2(c) show the detection statistics as a function of time t. For the Hotelling's T^2, the test statistic increases slowly after the change-point, which is expected since it does not factor in the underlying image structure. Similarly, MONAD struggles with change-point detection, primarily due to its disregard for the uncertainty of image features, leading to numerous false alarms. Additionally, employing a deep-learning model for online data is expensive. It is important to note that the object detection system, YOLO, is pretrained on the MS-COCO dataset (T.-Y. Lin et al., 2014), which features high-resolution images of diverse, everyday scenes. However, in practical scenarios where images may be low-resolution or heavily noised, the fine details that YOLO is trained to recognize may not be as visible. Given the limitations of the two baseline methods, we introduce next a new Bayesian, structure-aware method which provides a broader framework for integrating spatial information for monitoring.

4.3 Bayesian Online Structure-aware Change-point Detection

We now introduce the proposed Bayesian Online Structure-aware Change-point Detection (BLAST) method by leveraging Bayesian inference and spatial information modeling. We first describe the filter learning through deep GMRF, then show how the trained filters can be used to construct a prior precision matrix $\mathbf{Q}_0$ for Bayesian change-point detection. The BLAST method consists of three key parts: (1) training image filters $\mathbf{w}_1, \cdots, \mathbf{w}_L$ via convolutional neural network using offline training images and constructing a precision matrix for the DGMRF using trained filters; (2) updating posterior image distribution from online noisy image observations; (3) leveraging posterior image distribution for scalable and structure-aware Bayesian change detection.

4.3.1 Image Prior Elicitation

The first part of BLAST is to construct the image prior, and it involves two steps. The first step is to learn important image features through the linear DGMRF using offline training data. The feature learning allows us to construct a precision matrix, which integrates the spatial relationships in images for faster and more accurate detection of change points.

Assume that the linear DGMRF is a L-layer convolution, with $\mathbf{w}_l$ being a 3×3 filter. We consider two special forms of convolution filters: plus (+) filters and sequential (seq) filters, with forms

$$+ : \begin{bmatrix} 0 & a_3 & 0 \\ a_2 & a_1 & a_4 \\ 0 & a_5 & 0 \end{bmatrix} \qquad \text{seq} : \begin{bmatrix} 0 & 0 & 0 \\ 0 & a_1 & a_2 \\ a_3 & a_4 & a_5 \end{bmatrix}$$

where $a_1, \cdots, a_5 \in \mathbf{R}$ are parameters to be learned. The benefits of these two types of filters are that they correspond to transforms for which $\det(\mathbf{G}_l)$ in Eq.4.5 can be cheaply computed. Here, we define a linear DGMRF through a sequence of convolution using the small, specially designed filters, which has cheap determinant calculations, and it is equivalent to a single convolution using a large filter. Additionally, such linear DGMRF

results in fewer parameters to learn compared to a convolution with a single large filter. The plus filters are useful when detecting structures such as edges, patterns or texture which are commonly seen in the images. The sequential filters are more useful to capture the dependencies between the neighboring pixels with the sequential patterns.

As illustrated in Figure 4.1, our objective is to train the filters and bias adjustments, parameterized by $\boldsymbol{\theta}$, given the observed image $\mathbf{Y}$. In our case, the observed data refers to the training data from the pre-change regime. We utilize a single observed data point, $\mathbf{y}_{\text{train}}$, to illustrate the filter learning process and then discuss how to incorporate the series of training images into the framework.

For the linear DGMRF, the marginal likelihood can be written as

$$p(\mathbf{y}_{\text{train}}|\boldsymbol{\theta}) = \left.\frac{p(\mathbf{y}_{\text{train}}|\mathbf{x},\boldsymbol{\theta})p(\mathbf{x}|\boldsymbol{\theta})}{p(\mathbf{x}|\mathbf{y}_{\text{train}},\boldsymbol{\theta})}\right|_{\mathbf{x}=\mathbf{x}^*}$$

for arbitrary value of $\mathbf{x}^*$. However, this expression requires the determinant of the posterior precision matrix $\det(\frac{1}{\sigma^2}\mathbf{I}_M + \mathbf{G}_{\boldsymbol{\theta}}^\top \mathbf{G}_{\boldsymbol{\theta}})$, which is computationally infeasible for large M. Instead, we use the variational evidence lower bound (ELBO; (David M. Blei & McAuliffe, 2017)) $\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi},\mathbf{y}_{\text{train}}) \leq \log p(\mathbf{y}_{\text{train}}|\boldsymbol{\theta})$ which can be written as

$$\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi},\mathbf{y}_{\text{train}}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{x})}[-\log q_{\boldsymbol{\phi}}(\mathbf{x}) + \log p(\mathbf{y}_{\text{train}},\mathbf{x}|\boldsymbol{\theta})],$$

where $q_{\boldsymbol{\phi}}(x)$ is the variational posterior approximation, which depends on variational parameter $\boldsymbol{\phi}$ (Kingma & Welling, 2013). For simplicity, we choose $q_{\boldsymbol{\phi}}(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\nu},\mathbf{S})$ with diagonal covariance matrix, and $\boldsymbol{\phi} = \{\boldsymbol{\nu},\mathbf{S}\}$. Then we can write

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi},\mathbf{y}_{\text{train}}) =& \frac{1}{2}\log|\det(\mathbf{S}_{\boldsymbol{\phi}})| - M\log\sigma_{\boldsymbol{\theta}} + \log|\det(\mathbf{G}_{\boldsymbol{\theta}})| \\ &- \frac{1}{2}\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{x})}\left[(\mathbf{G}_{\boldsymbol{\theta}}\mathbf{x}+\mathbf{b}_{\boldsymbol{\theta}})^\top(\mathbf{G}_{\boldsymbol{\theta}}\mathbf{x}+\mathbf{b}_{\boldsymbol{\theta}}) + \frac{1}{\sigma_{\boldsymbol{\theta}}^2}(\mathbf{y}_{\text{train}}-\mathbf{x})^\top\mathbf{I}_M(\mathbf{y}_{\text{train}}-\mathbf{x})\right],\end{aligned}$$

where constant terms have been omitted, and all parameters have been subscripted with $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$. Additionally, we use the reparametrization trick to replace the last expectation with the sum over N_q standard random samples $\boldsymbol{\varepsilon}_1,\cdots,\boldsymbol{\varepsilon}_{N_q}$, and set $\mathbf{x}_i = \boldsymbol{\nu}_{\boldsymbol{\phi}} + \mathbf{S}_{\boldsymbol{\phi}}^{1/2}\boldsymbol{\varepsilon}_i$ in

Algorithm 4 Filter Optimization Algorithm (Sidén & Lindsten, 2020)

Inputs: training data, model structure, learning rate, N_q, etc.

Initialize values for the model parameters $\boldsymbol{\theta} \subset \{(\mathbf{w}_l, \mathbf{b}_l)_{l=1,\cdots,L}, \sigma_2\}$ and the variational parameters $\boldsymbol{\phi} = \{\boldsymbol{\nu}, \mathbf{S}\}$.
Optimize the ELBO wrt. $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ using Adam to obtain $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\phi}}$.

Outputs: Optimized parameters $(\hat{\mathbf{w}}_l)_{l=1,\cdots,L}$

the sum (Kingma & Welling, 2013). This gives an unbiased estimator of the ELBO, which has low variance. Moreover, this estimator is differentiable with respect to $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ and can be used for stochastic gradient optimization with autodiff and backprop (Rumelhart et al., 1986). In practice, we use Adam, which is a popular optimization algorithm that adaptively changes the learning rate for each parameter (Kingma & Ba, 2015).

The aforementioned ELBO serves as the loss function for the single observed data point $\mathbf{y}_{\text{train}}$, which represents the training data from the pre-change regime. If multiple training images are available, the computation of loss can be performed across the entirety of the training data. Furthermore, this framework can be extended to matrix data with appropriate structure. For example, when applied to drift tube chamber applications, the filters do not map to image features. Instead, they are utilized to discern the relationships between the features of particles detected in the chambers.

The second step of image prior elicitation for BLAST is to use the trained filters $(\hat{\mathbf{w}}_l)_{l=1,\cdots,L}$ to construct the fitted precision matrix for the underlying true image in the pre-change regime. Figure 4.3 visualizes the workflow for constructing the prior precision matrix using DGMRF; we will discuss each step in detail below. Suppose the trained filters $(\hat{\mathbf{w}}_l)_{l=1,\cdots,L}$ are obtained using Algorithm 4. Then, we can compute the forward filter $\mathbf{w}_{\text{forward}}$ in $\mathbf{Z} = \text{conv}(\mathbf{X}, \mathbf{w}_{\text{forward}})$, where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, by convolving the trained filters, which are defined as follows:

$$\mathbf{w}_{\text{forward}} = \hat{\mathbf{w}}_1 * \cdots * \hat{\mathbf{w}}_L$$

We can then construct the inverse transform $\mathbf{G}$ in $\mathbf{z} = \mathbf{G}\mathbf{x}$ using $\mathbf{w}_{\text{forward}}$ given the equivalent definitions of GMRF. Specifically, we convert 2D image coordinates into a 1D index. This

mapping allows us to distribute the filter's effect across the image, taking into account the specific pixel locations. Then, it is trivial to show that the precision matrix of $\mathbf{x}$ is $\mathbf{G}^\top\mathbf{G}$.

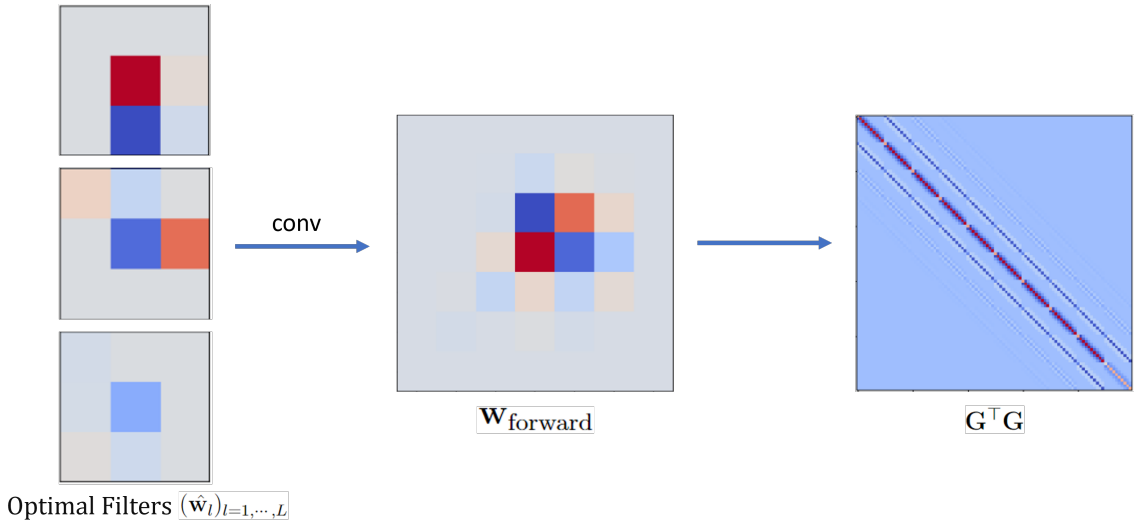


FIGURE 4.3: Constructing the precision matrix using DGMRF.

4.3.2 Image Posterior Update

The second part of BLAST is to update image posterior. To incorporate spatial information for the underlying true image $\mathbf{x}$, we employ a GMRF prior defined by:

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_0, \mathbf{Q}_0^{-1}).$$

Here, the mean vector $\boldsymbol{\mu}_0$ is set as the mean of the training images, and the precision matrix, $\mathbf{Q}_0$, is defined as $\mathbf{G}^\top\mathbf{G}$. This specification of $\mathbf{Q}_0$ effectively captures the expected image structure derived from offline training images.

Let $\{\mathbf{y}_t\}_{t=1}^T$ be the noisy image observations of $\mathbf{x}$; the likelihood function can be expressed as:

$$\mathbf{y}_t|\mathbf{x} \overset{iid}{\sim} \mathcal{N}(\mathbf{x}, \sigma^2\mathbf{I}_M), \quad t = 1, \cdots, T,$$

where M is the number of image pixels, and σ^2 represents the known variance.

In this context, the GMRF prior exhibits conjugacy, resulting in the posterior distribution also being a GMRF, with the image posterior mean and precision matrix updated in an online fashion via the following iterative scheme:

$$\mathbf{x}|\mathbf{y}_1, \cdots, \mathbf{y}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{Q}_t^{-1})$$

where:

- $\boldsymbol{\mu}_t = \mathbf{Q}_t^{-1}(\frac{1}{\sigma^2}\mathbf{I} \cdot \sum_{i=1}^{t} \mathbf{y}_i + \mathbf{Q}_0\boldsymbol{\mu}_0) = \mathbf{Q}_t^{-1}(\frac{1}{\sigma^2}\mathbf{I} \cdot \mathbf{y}_t + \mathbf{Q}_{t-1}^{-1}\boldsymbol{\mu}_{t-1})$
- $\mathbf{Q}_t = \frac{t}{\sigma^2}\mathbf{I} + \mathbf{Q}_0 = \frac{1}{\sigma^2}\mathbf{I} + \mathbf{Q}_{t-1}$.

A computational bottleneck is clear for updating the posterior mean $\boldsymbol{\mu}_t$ from online data: it requires the computation of the matrix inverse $\mathbf{Q}_t^{-1}$, which requires $\mathcal{O}(M^3)$ work. For high-dimensional images (i.e., with large M), this can be prohibitively expensive for online implementation. The following proposition shows that, with careful linear algebra manipulations, this posterior mean can be updated at each time t in $\mathcal{O}(M^2)$ work.

Proposition 1. *Let $\mathbf{Q}_0 = \mathbf{PDP}^{-1}$ be the eigendecomposition of the precision matrix $\mathbf{Q}_0$. Then the inverse of $\mathbf{Q}_t$ can be written as:*

$$\mathbf{Q}_t^{-1} = (\frac{t}{\sigma^2}\mathbf{I} + \mathbf{Q}_0)^{-1} = (\mathbf{P}(\frac{t}{\sigma^2}\mathbf{I} + \mathbf{D})\mathbf{P}^{-1})^{-1} = \mathbf{P}(\mathbf{D} + \frac{t}{\sigma^2}\mathbf{I})^{-1}\mathbf{P}^{-1}$$

In other words, the updated posterior mean $\boldsymbol{\mu}_t$ can be written as $\mathbf{P}(\mathbf{D} + t/\sigma^2\mathbf{I})^{-1}\mathbf{P}^{-1}$, where $\mathbf{P}$ and $\mathbf{D}$ are the eigenvectors and eigenvalues of $\mathbf{Q}_0$. The eigendecomposition of $\mathbf{Q}_0$, which requires $\mathcal{O}(M^3)$ work, can be performed offline after prior elicitation. Then, at each time t for online monitoring, the required inverse $\mathbf{Q}_t^{-1}$ can be computed as $\mathbf{P}(\mathbf{D} + t/\sigma^{-2}\mathbf{I})^{-1}\mathbf{P}^{-1}$ with $\mathcal{O}(M^2)$ work, as $\mathbf{D} + t/\sigma^{-2}\mathbf{I}$ is a diagonal matrix.

4.3.3 Bayesian Change-point Detection

The final step of BLAST is to integrate the GMRF prior within the Bayesian change-point detection framework. Suppose the true change-point occurs at time τ. Let

$$\mathbf{y}_t^{\text{pre}} \sim \mathcal{N}(\mathbf{x}^{\text{pre}}, \sigma^2\mathbf{I}_M) \quad \text{for} \quad t = 1, \cdots, \tau,$$

and

$$\mathbf{y}_t^{\text{post}} \sim \mathcal{N}(\mathbf{x}^{\text{post}}, \sigma^2\mathbf{I}_M) \quad \text{for} \quad t = \tau + 1, \cdots, T.$$

The goal of detecting the change in the observed images can then be thought of as detecting the differences between the underlying pre-change image $\mathbf{x}^{\text{pre}}$ and the underlying post-change image $\mathbf{x}^{\text{post}}$. We thus desire an approach that investigates the following hypothesis:

$$H_0 : \mathbf{x}^{\text{pre}} = \mathbf{x}^{\text{post}}, \quad H_A : \mathbf{x}^{\text{pre}} \neq \mathbf{x}^{\text{post}}. \tag{4.10}$$

To test (4.10), we adopt the Bayesian online change-point detection framework in (Adams & MacKay, 2007), and present an efficient message-passing algorithm. At each time t, within the Bayesian online change-point detection framework, we compute the probability distribution of the time since the last change-point, also known as the run length.

4.3.3.1 Algorithm

The algorithm proceeds as follows. At each time t, new data $\mathbf{y}_t$ arrives, and the predictive probability of observing $\mathbf{y}_t$ given the parameter estimates from $t-1$ is evaluated for every possible run length r_{t-1} as follows:

$$\pi_t^{(r_{t-1})} = p(\mathbf{y}_t|\mathbf{y}_1, \ldots, \mathbf{y}_{t-1}, r_{t-1}) \sim \mathcal{N}(\boldsymbol{\mu}_{t-1}, \mathbf{Q}_{t-1}^{-1} + \sigma^2\mathbf{I}).$$

The run length posterior probability $p(r_t|\mathbf{y}_{1:t})$ is proportional to the joint $p(r_t, \mathbf{y}_{1:t})$, which can be written recursively as

$$p(r_t, \mathbf{y}_{1:t}) = \sum_{r_{t-1}} p(\mathbf{y}_t|r_t, \mathbf{y}_{1:t-1}) p(r_t|r_{t-1}) p(r_{t-1}, \mathbf{y}_{1:t-1}).$$

This algorithm is recursive because we can forward message-pass the joint distribution $p(r_{t-1}, \mathbf{y}_{1:t-1})$ to the computation for $p(r_t, \mathbf{y}_{1:t})$. The transition probabilities are modelled by the change-point distribution $p(r_t|r_{t-1})$, which is the assumption that r_t is conditionally independent to everything else given r_{t-1}. Logically, there are two possibilities with the run length: (i) the run length grows by one, indicating that there is no change happens; and (ii) a run length of 0, indicating a change-point occurring at t. The change-point probability

is defined as

$$p(r_t|r_{t-1}) = \begin{cases} \mathrm{H}(\mathrm{r_{t-1}}+1) & \text{if } r_t = 0 \\ 1-\mathrm{H}(\mathrm{r_{t-1}}+1) & \text{if } r_t = r_{t-1}+1 \\ 0 & \text{otherwise.} \end{cases} \tag{4.11}$$

If T is a geometric random variable with a probability of success p, then $\mathrm{H}(\emptyset) = \mathrm{p}$ (Forbes et al., 2011). This kind of process is called *memoryless* because the hazard function does not depend on time.

For case (i), we compute the growth probability. For each possible run length value r_{t-1}, the joint probability $p(r_t, \mathbf{y}_{1:t})$ can be written as:

$$p(r_t = r_{t-1}+1, \mathbf{y}_{1:t}) = p(r_{t-1}, \mathbf{y}_{1:t-1})\pi_t^{(r_{t-1})}(1-\mathrm{H}(\mathrm{r_{t-1}}+1)).$$

For case (ii), we compute the change-point probability. The joint probability $p(r_t, \mathbf{y}_{1:t})$ is taking the summation over all possible r_{t-1}, written as:

$$P(r_t = 0, \mathbf{y}_{1:t}) = \sum_{r_{t-1}} p(r_{t-1}, \mathbf{y}_{1:t-1})\pi_t^{(r_{t-1})}\mathrm{H}(\mathrm{r_{t-1}}+1)$$

The run length posterior probability $p(r_t|\mathbf{y}_{1:t})$ can then be computed as

$$p(r_t|\mathbf{y}_{1:t}) = \frac{p(r_t, \mathbf{y}_{1:t})}{p(\mathbf{y}_{1:t})},$$

where $p(\mathbf{y}_{1:t}) = \sum_{r_t} p(r_t, \mathbf{y}_{1:t})$.

The complete algorithm is presented in Algorithm 5. We begin with an empty matrix $\mathrm{R} \in \mathbb{R}^{(\mathrm{T}+1)\times(\mathrm{T}+1)}$ to store the posterior probabilities of run lengths over time. We then specify the prior mean $\boldsymbol{\mu}_0$ and prior precision matrix $\mathbf{Q}_0$ on $\mathbf{x}^{\mathrm{pre}}$ as described in Section 4.3.1 and the hazard function on change-point. Next, we compute the growth probability and change-point probability utilizing three components: (i) the joint probability message from $t-1$; (ii) the predictive probability $\pi_t^{(r_{t-1})}$; and (iii) the hazard function $\mathrm{H}(\mathrm{r_{t-1}}+1)$. Finally, the run length probability is computed, and the model parameters are updated

Algorithm 5 Bayesian Change-point Detection Algorithm

Inputs: Vectorized images $\mathbf{y}_1, \cdots, \mathbf{y}_T$, prior mean $\boldsymbol{\mu_0}$, prior precision matrix $\mathbf{Q}_0$, hazard function on change-point $\mathrm{H}(\varnothing)$.

Initialize $p(r_0) = 1$. Set $t = 1$.
for $t = 1, \cdots, T$
Observe new data $\mathbf{y}_t$.
Compute predictive probability:

$$\pi_t^{(r_{t-1})} \sim \mathcal{N}(\boldsymbol{\mu}_{t-1}, \mathbf{Q}_{t-1}^{-1} + \sigma^2 \mathbf{I}).$$

Compute growth probabilities:

$$p(r_t = r_{t-1} + 1, \mathbf{y}_{1:t}) = p(r_{t-1}, \mathbf{y}_{1:t-1}) \pi_t^{(r_{t-1})} (1 - \mathrm{H}(\mathrm{r_{t-1}} + 1)).$$

Compute change-point probability:

$$P(r_t = 0, \mathbf{y}_{1:t}) = \sum_{r_{t-1}} p(r_{t-1}, \mathbf{y}_{1:t-1}) \pi_t^{(r_{t-1})} \mathrm{H}(\mathrm{r_{t-1}} + 1).$$

Compute evidence:

$$p(\mathbf{y}_{1:t}) = \sum_{r_t} p(r_t, \mathbf{y}_{1:t}).$$

Compute run length posterior:

$$p(r_t | \mathbf{y}_{1:t}) = \frac{p(r_t, \mathbf{y}_{1:t})}{p(\mathbf{y}_{1:t})}.$$

Update model parameters $\boldsymbol{\mu}_t$ and $\mathbf{Q}_t$:

$$\boldsymbol{\mu}_t = \mathbf{Q}_t^{-1} (\frac{1}{\sigma^2} \mathbf{I} \cdot \mathbf{y}_t + \mathbf{Q}_{t-1}^{-1} \boldsymbol{\mu}_{t-1})$$

$$\mathbf{Q}_t = \frac{t}{\sigma^2} \mathbf{I} + \mathbf{Q}_0.$$

Update $t \leftarrow t + 1$.

Outputs: Run length posterior $p(r_t | \mathbf{y}_{1:t})$ for all t.

as described in Section 4.3.2. These steps are iterated at each time t to perform online change-point detection. Algorithm 5 summarizes the detailed steps of the procedure.

4.3.3.2 Computational Complexity

We investigate next the computational complexity of BLAST, in terms of images with M pixels as well as time series length T. For the filter optimization (Algorithm 4), it requires $\mathcal{O}(M)$ work for a fixed number of iterations of optimization (Sidén & Lindsten, 2020). The covariance matrix construction requires again $\mathcal{O}(M)$ work to map through all pixels in the images. For the Bayesian change-point (Algorithm 5), the posterior computation at each time t requires $\mathcal{O}(M^3)$ work without applying Proposition 1. When applying Proposition 1 for online updating of image posterior, it requires $\mathcal{O}(M^2)$ work at each time t, and a one-shot computation of the eigendecomposition of $\mathbf{Q}_0^{-1}$ that requires $\mathcal{O}(M^3)$ work. Summarizing the above, the Bayesian change-point (Algorithm 5) requires an initial $\mathcal{O}(M^3)$ work for eigendecomposition, and $\mathcal{O}(T^2M^2)$ for posterior computation for the entire time series. Clearly, this is not optimal for a time series with a large T, and a simple solution is to place a cap on the possible values of r_t, which reduces the computational complexity to $\mathcal{O}(TM^2)$.

4.4 Simulation Experiment

We now explore the performance of BLAST in a suite of simulation studies where the data is generated with edge-structure. We investigate several change scenarios, including background correlation change, structure change and intensity change. Here, the generation process of the true images $\mathbf{X}$ adopts the idea of Gaussian Markov Random Field (GMRF) by incorporating the spatial correlation of pixels. More details will be provided in the later sections for each scenario.

The simulation set-up is as follows. We simulate data $\mathbf{Y}_1, \cdots, \mathbf{Y}_T$ with $T = 100$, and the change-point is set at time $t^* = 50$. In other words, $\mathbf{Y}_1, \cdots, \mathbf{Y}_{50}$ are generated from the pre-change distribution, and $\mathbf{Y}_{51}, \cdots, \mathbf{Y}_{100}$ are generated from the post-change distribution. In our simulation, this image data is generated with edge-structure from the true images shown in Figure 4.4, 4.6 and 4.8 (in varying ways). The goal is to have BLAST learn this edge-feature from data, and leverage such structure to construct the prior precision matrix and then quickly identify the change-point.

Four baseline methods are used here for comparison. The first is named Bayes_nonIP, which is a method using the Bayesian change-point framework with the non-informative prior precision matrix. In another word, we integrate no spatial structure information in the precision prior. Specially, we set the prior precision matrix as an diagonal matrix, with each entry representing the variance estimated from the training data. The second is the parametric Hotelling's T^2 test, using the 15 extracted principal components from PCA on the original data. The third is the Multi-Objective Neural Anomaly Detector (MONAD) proposed in (Doshi & Yilmaz, 2021), which makes use of the deep learning-based features to construct the test statistics. Details on methods 2 & 3 can be found in Section 4.2.3. The last method is the maximum mean discrepancy (MMD) test (Gretton et al., 2012; S. Li et al., 2015), a widely used non-parametric change detection method, using the 15 extracted principal components from PCA on the original data. Given a class of functions $\mathcal{F}$ and two distributions p and q, the MMD distance between p and q is defined as $\mathrm{MMD}_{\mathcal{F}}(p,q) = \sup_{f\in\mathcal{F}}(\mathbb{E}_{x\sim p}[f(x)] - \mathbb{E}_{y\sim q}[f(y)])$. When $\mathcal{F}$ is a reproducing kernel Hilbert space (RKHS) associated with kernel function $K(\cdot,\cdot)$, this MMD statistic can be written as:

$$S_t^M = \frac{1}{n_{\text{pre}}^2}\sum_{i,i'=1}^{n_{\text{pre}}} K(\mathbf{y}_i,\mathbf{y}_{i'}) + \frac{1}{n_{\text{post}}^2}\sum_{j,j'=1}^{n_{\text{post}}} K(\mathbf{y}_j,\mathbf{y}_{j'}) - \frac{2}{n_{\text{pre}}n_{\text{post}}}\sum_{i=1}^{n_{\text{pre}}}\sum_{j=1}^{n_{\text{post}}} K(\mathbf{y}_i,\mathbf{y}_j). \tag{4.12}$$

In our implementation, we used the standard Gaussian radius basis function (RBF) kernel $K(\cdot,\cdot)$, where the kernel bandwidth is chosen using the so-called "median trick" (Bernhard et al., 2018), i.e., set to be the median of the pairwise distances between data points.

4.4.1 Spatial Correlation Change

We first consider the case of spatial correlation changes, where the pre-change and post-change data are sampled with noise $\mathcal{N}(0,1)$ from the 25×25 true images shown in Figure 4.4. Specifically, the spatial correlation distance l of the underlying GMRF is set to 20 for the pre-change images and 30 for the post-change images, where a larger correlation distance

l results in greater smoothness in the images. To introduce specific visual structures, predefined edge patterns are overlaid on both the pre- post-change GMRF images (Figure 4.4).

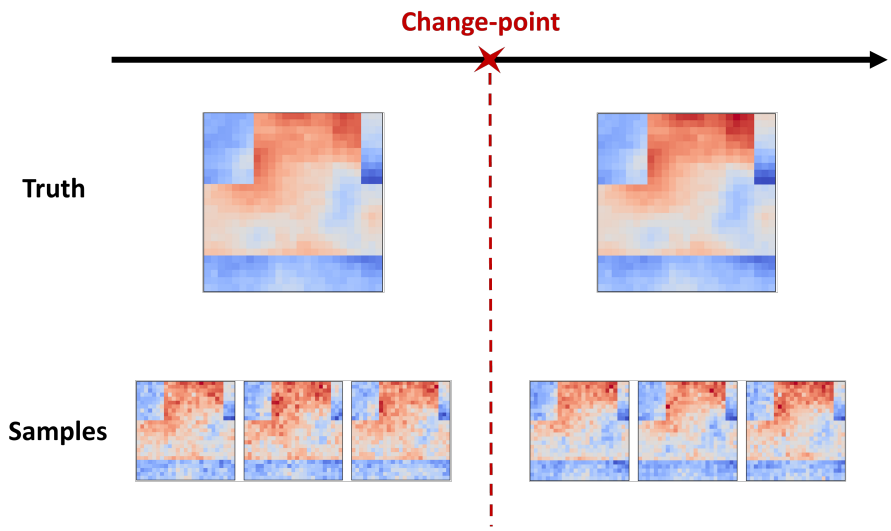
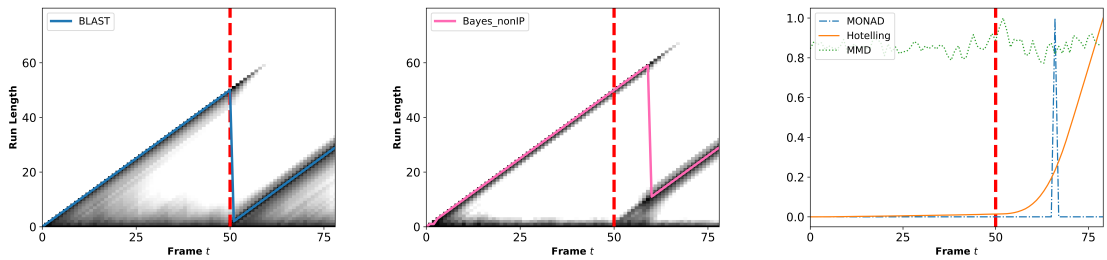


FIGURE 4.4: Visualizing the pre-change and post-change true images along with their generated samples: small spatial correlation distance (left) and large spatial correlation distance (right).

Figure 4.5(a) and 4.5(b) show the run length posterior at each time t for BLAST and Bayes_nonIP using a logarithmic color scale, and the saw-toothed lines are the maximum probability at each time point t. For BLAST, we observe a consistent increment by 1 in the run length with the maximum probability before $t = 50$, suggesting that no change occurs. The run length with the maximum probability then suddenly drops back to 0 at $t = 51$, indicating an immediate detection without delay. In contrast, while Bayes_nonIP does manage to detect this change, it suffers from a significant detection delay indicated by the maximum probability. The abrupt reduction of the run length to 0 at $t = 51$ signals a change-point. However, the probability of no change continuing to dominate until roughly $t = 60$, suggests the necessity of integrating the common image structures in the prior precision matrix for faster and more accurate detection of change points. Figure 4.5(c) shows the test statistics from MONAD, Hotelling's T^2 and MMD. The MONAD statistics is not able to detect the change at all because it does not consider image feature uncertainty.

In addition, there are no discernible objects in the data, and thus the pre-trained object detector fails. The Hotelling's T^2 statistic increases noticeably more slowly after the change, which suggests a larger detection delay. For the MMD statistic, we see that while it peaks up after the change-point, its pre-change statistic is quite unstable and volatile, which leads to increased false alarms.



(a) (b) (c)

FIGURE 4.5: Results for the setting of spatial correlation change, and the vertical red dashed line indicates the true change-point: (a) The run length posterior at each time t for BLAST using a logarithmic color scale. Darker indicates higher probability. The saw-toothed lines are the maximum probability at each time t for BLAST. (b). Same for Bayes_nonIP. (c) Test statistics using MONAD, Hotelling's T^2 and MMD at each time t.

4.4.2 Visual Structure Change

Next, we consider the case of the visual structure change, where the pre- and post-change data are sampled with noise $\mathcal{N}(0,1)$ from the 25×25 true images shown in Figure 4.6. The true images are generated using GMRF with the spatial correlation distance $l = 50$. The predefined edge patterns are overlaid on the GMRF pre- post-change images, with a segment missing from the upper-right block edge for post-change (Figure 4.6).

Figure 4.7(a) and 4.7(b) show the run length posterior at each time t for BLAST and Bayes_nonIP using a logarithmic color scale, and the saw-toothed lines are the maximum probability at each time point t. We see that the run length with the maximum probability for BLAST has a consistent increase by 1 before $t = 50$, suggesting that no change occurs. This run length sharply resets to 0 at $t = 51$ indicating an immediate detection of change without delay. A similar pattern is observed for Bayes_nonIP, though the run length which

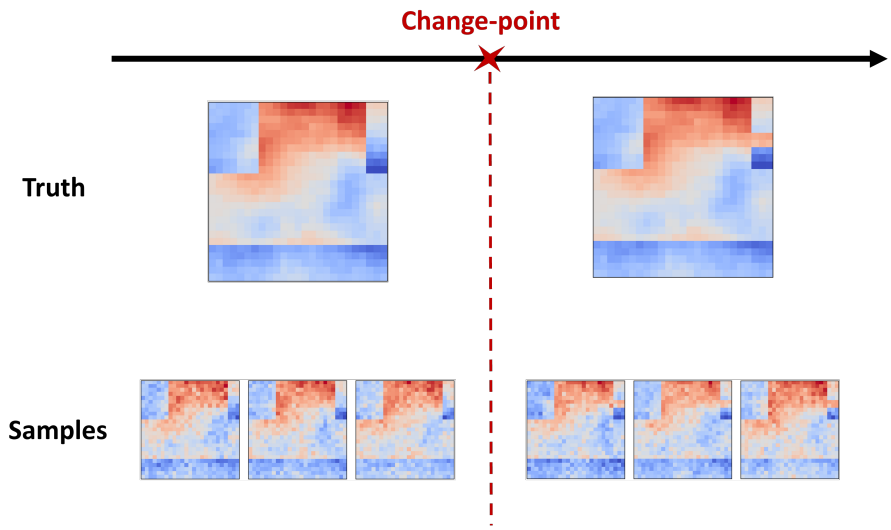
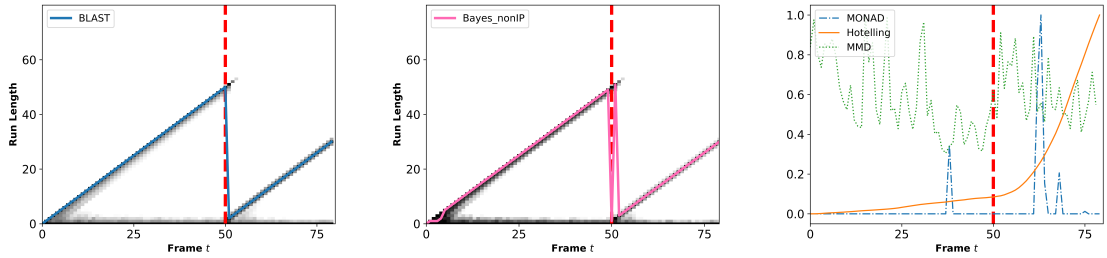


FIGURE 4.6: Visualizing the pre-change and post-change true images along with their generated samples: complete upper-right block edge (left) and a segment missing from the upper-right block edge (right).

shows the highest probability exhibits some instability in the initial time points. Around $t = 50$, there is noticeable fluctuation in the belief between the presence and absence of a change-point. Figure 4.7(b) shows the test statistics from MONAD, Hotelling's T^2 and MMD. The Hotelling's T^2 statistics shows a relatively larger detection delay. Neither MONAD nor MMD statistics succeeds in detecting the change.



(a) (b) (c)

FIGURE 4.7: Results for the setting of structure change, and the vertical red dashed line indicates the true change-point: (a) The Run length posterior at each time t for BLAST using a logarithmic color scale. Darker indicates higher probability. The saw-toothed lines are the maximum probability at each time t for BLAST. (b). Same for Bayes_nonIP. (c) Test statistics using MONAD, Hotelling's T^2 and MMD at each time t.

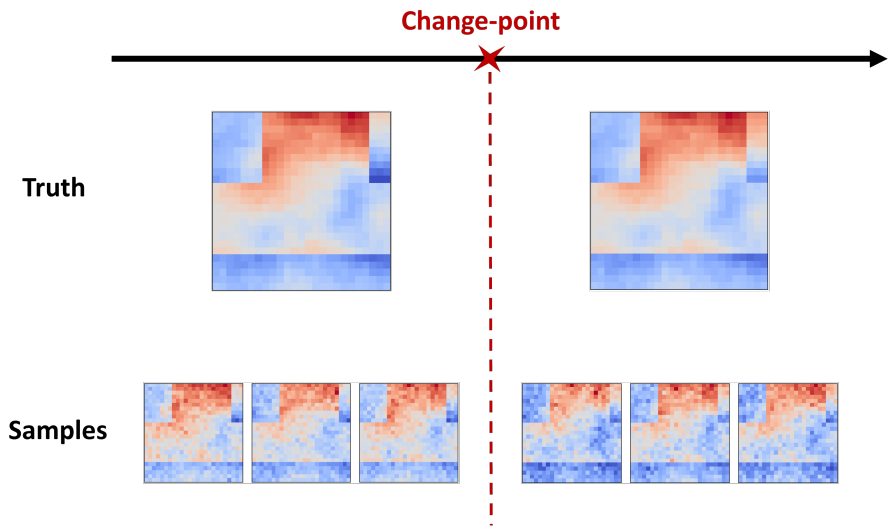


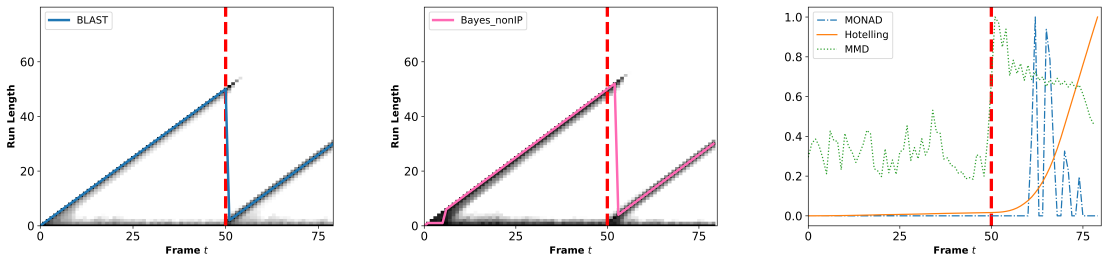
FIGURE 4.8: Visualizing the pre-change and post-change true images along with their generated samples: high intensity values of the block (left) and low intensity values of the block (right).

4.4.3 Structure Intensity Change

Finally, we investigate the case of the structure intensity change, where the pre- and post-change data are sampled with noise $\mathcal{N}(0,1)$ from the 25×25 true images shown in Figure 4.8. The true images are generated using GMRF with the spatial correlation distance $l = 50$. The predefined edge patterns are overlaid on the GMRF pre- post-change images, with the upper-right block intensity increased by 1 for post-change (Figure 4.8).

Figure 4.9 shows the results for all methods. We see that the run length with the maximum probability for BLAST increases by 1 perfectly, suggesting that no change occurs before $t = 50$. This run length suddenly drops back to 0 at $t = 51$, with no detection delay. A similar pattern is observed for Bayes_nonIP, though the run length showing the highest probability exhibits some instability in the initial time points and experiences a notable delay in detecting the change-point. Comparatively, the increase in the Hotelling's T^2 statistic is more subdued after the change, resulting in a greater detection delay, while the MMD statistic is noticeably more unstable in the pre-change, leading to increased false alarms). Although MONAD statistics can detect the change, it does so with a significant

detection delay. These experiments suggest that learning and integrating image structure while considering feature uncertainty can efficiently detect changes in image data.



(a) (b) (c)

FIGURE 4.9: Results for the setting of intensity change, and the vertical red dashed line indicates the true change-point: (a) The Run length posterior at each time t for BLAST using a logarithmic color scale. Darker indicates higher probability. The saw-toothed lines are the maximum probability at each time t for BLAST. (b) Same for Bayes_nonIP. (c) Test statistics using MONAD, Hotelling's T^2, and MMD at each time t.

4.5 Applications

4.5.1 Street Scene Monitoring

We now demonstrate the effectiveness of BLAST in the street scene monitoring problem. The data used here consists of $T = 35$ image snapshots taken from a static USB camera looking down on a scene of a two-lane street with bike lanes and pedestrian sidewalks (Ramachandra & Jones, 2020). See Figure 4.10 for snapshots with a car driving starting from $t^* = 16$.

We proceed with the detection framework outlined in Section 4.3. Here, the infrastructure features, such as bike lanes and pedestrian sidewalks are captured using 30 training images of the static street scenes with edge detector, and are then integrated in the prior precision matrix for quick detection. The Hotelling's T^2 test is applied using the 15 extracted principal components from PCA on the image data, and the MONAD statistics is performed on the extracted deep-learning features. The MMD test is performed on the image data directly, with the RBF kernel, and the bandwidth is selected using the "median trick" as described in Section 4.4.

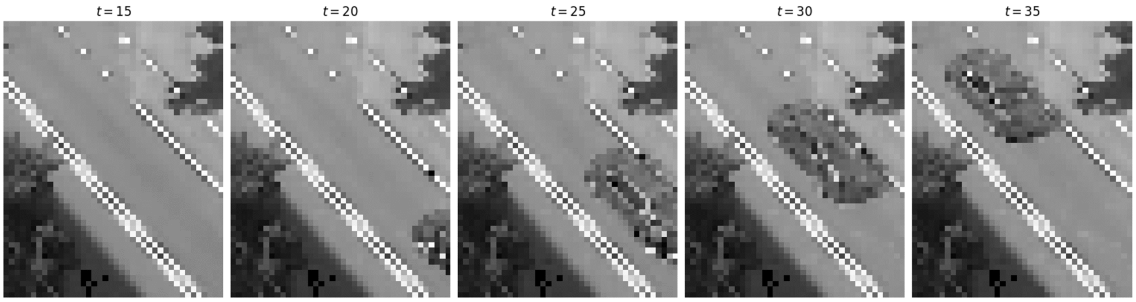
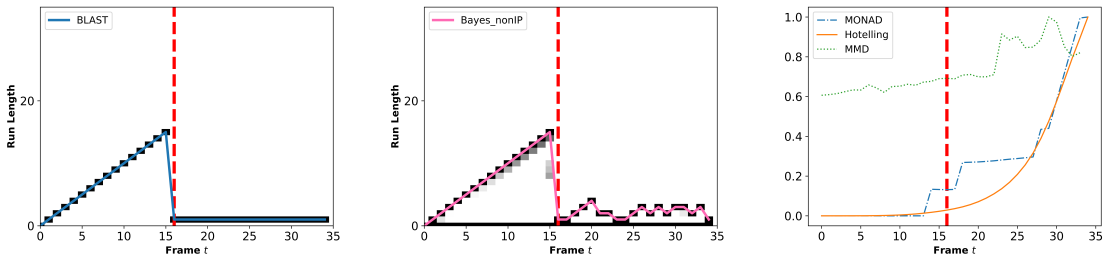


FIGURE 4.10: Snapshots of the street scene with a car driving starting at $t^* = 16$.

Figure 4.11 shows the results for BLAST, Bayes_nonIP, MONAD, the Hotelling T^2 and MMD. For BLAST, we note that the run length with the maximum probability incrementally increases by 1, consistently indicating the absence of any change (sign of cars) prior to $t = 16$. At $t = 17$, there is a precipitous drop back to 0 in the run length, signifying immediate detection of a car without delay. Since the car is moving overtime, the run length with the maximum probability remains at 0, indicating a change-point at all time $t > 17$. A similar pattern is observed for Bayes_nonIP, though the run length showing the highest probability exhibits some fluctuations after $t > 17$ suggesting that some snapshots are not differentiated by this method when the car is constantly moving. Comparatively, the Hotelling's T^2 shows a much more subdued and gradual increase in its statistic, indicating a significant detection delay. The MMD approach yields poor performance here: its pre-change statistics increases over time and its post-change statistics also increases but not significantly at the change-point. Thus, MMD is not able to detect the change. The MONAD statistics is also not able to detect the change: there is no large change in statistics at the change-point. This shows that, by learning and integrating image structure within the prior for the Bayesian change detection framework, the proposed method can yield significant improvements over existing methods.

4.5.2 Drift Tube Chambers Monitoring

Particle physics experiments are scientific investigations designed to explore the fundamental particles that make up matter and the forces that govern their interactions. Many



(a) (b) (c)

FIGURE 4.11: Results for each compared method in the street scene monitoring application. The vertical red dashed line indicates the true change-point.

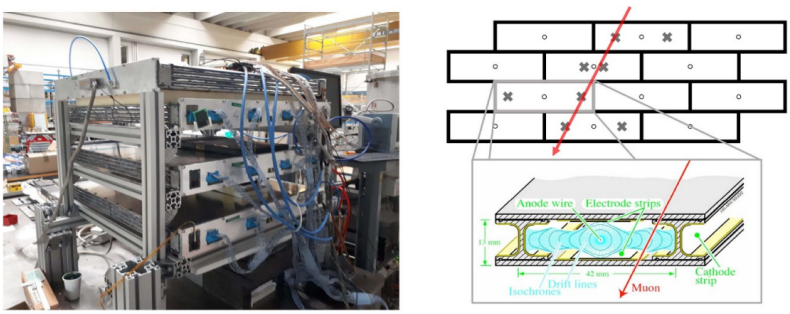


FIGURE 4.12: Left: the experimental apparatus at Legnaro Laboratory, with four drift-tube chambers, vertically stacked. Right: a schematic view of the cell (bottom) and an example of hit pattern left by a charged particle crossing a chamber (top).

particle physics experiments are conducted in large particle accelerators or colliders, such as the Large Hadron Collider (LHC) at the European Organization for Nuclear Research (CERN). These facilities accelerate particles to extremely high speeds (close to the speed of light) and then collide them. The collisions produce a variety of other particles, allowing scientists to study phenomena that occur at very high energy levels, similar to conditions just after the Big Bang. However, measurements of these experiments can be undermined even by minor malfunctions in the setup - such as issues with power supply, electronic configurations, or temperature variations in hardware components. Given the intricate nature, the precision of measurements required, and the vast scale of the experiments, data must be monitored continuously to ensure the quality and to promptly detect any faults.

In this work, we exploited a scaled-down version of these experiments, utilizing a set of drift tube (DT) chambers housed at the Legnaro Istituto Nazionale di Fisica Nucleare

(INFN) National Laboratory (Grosso et al., 2023). These chambers are essentially four layers of tubes filled with a special gas and subjected to an electric field (Figure 4.12). This field is created by placing an anodic wire at the center of each tube at a voltage of 3.6 kV and flanking it with two cathodic strips on the sides, each set to a voltage of -1.2 kV. When muons from cosmic rays pass through these tubes, they cause electrons in the gas to drift towards the wire. The front-end electronics then capture the time of arrival of these electrons, amplify the signal, and filter out any irrelevant background noise, focusing only on signals stronger than 100 mV. When conditions within the detector deviate from the normal, it can affect the precision of tracking muon paths. To mimic the most common sources of detector anomaly, the scientists deliberately created problems in the detector system to understand how these issues would affect the measurement. Specifically, we consider six basic measurements related to the passage of a muon through a DT chamber:

- Drift times t_i, where $i = 1, 2, 3, 4$: the times for electrons, ionized by the passing muon, to reach the detection wire for the four layers of tube.
- Slope θ: the angle at which the muon's path intersects the vertical axis.
- Number of hits n_{Hits}: the number of hits within a one-second time frame when a muon crosses the chamber.

We consider six anomalous configurations: (1-3) reducing the voltage of the cathodic strips to $75\%, 50\%, 25\%$ of their nominal value (-900 V, -600 V and -300 V, respectively) and (4-6) lowering the front-end threshold to $75\%, 50\%, 25\%$ of their nominal value (75 mV, 50 mV and 25 mV, respectively). The formal actions distort the electric field shape, while the latter ones mimic the sudden contribution of noise sources.

For our analysis, the data comprise 4050 events recorded under the normal chamber configuration, alongside 2700 events from each anomalous configuration. To learn the correlation information among the six fundamental measurements: drift time t_i (where $i = 1, 2, 3, 4$), slope θ and number of hits n_{Hits}, each data matrix was assembled with 27 events, structured into dimensions of 9×18. Here, $q = 6 \times 3$ represents the number of basic measurements across three events (forming a set), and $p = 9$ indicates the number of sets

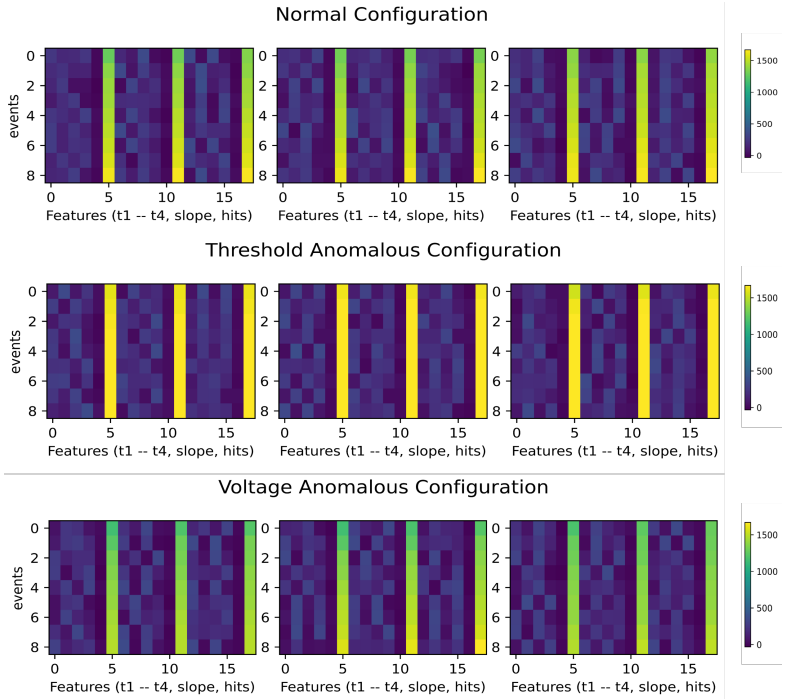


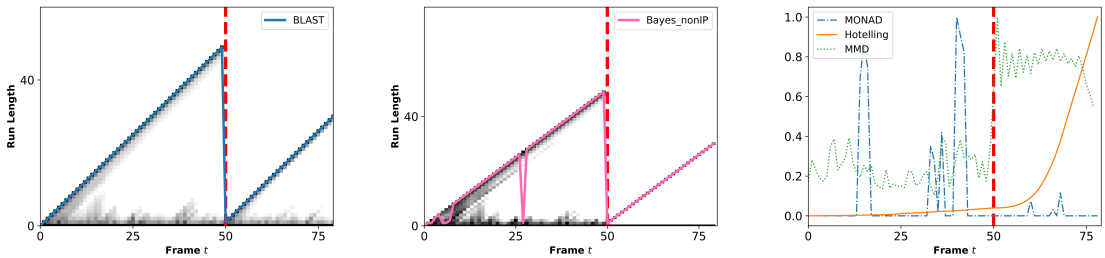
FIGURE 4.13: Examples of data matrices under the normal configuration (top), the anomalous configuration by lowering the front-end threshold to 75% (middle), the anomalous configuration by reducing the voltage to 75% (bottom).

of events. The data includes $T = 80$ data matrices, each assembling six basic measurements across events, with the true change-point identified at $t^* = 50$.

With this, we can then apply BLAST for detecting the anomalous chamber configuration. We use 30 data matrices to construct the prior precision matrix. The Hotelling's T^2 is performed using the 15 extracted principal components from PCA on the image data, and the MONAD statistics is performed on the extracted deep-learning features. The MMD test is performed on the image data directly, with the RBF kernel, and the bandwidth is chosen using the "median trick" as described in Section 4.4.

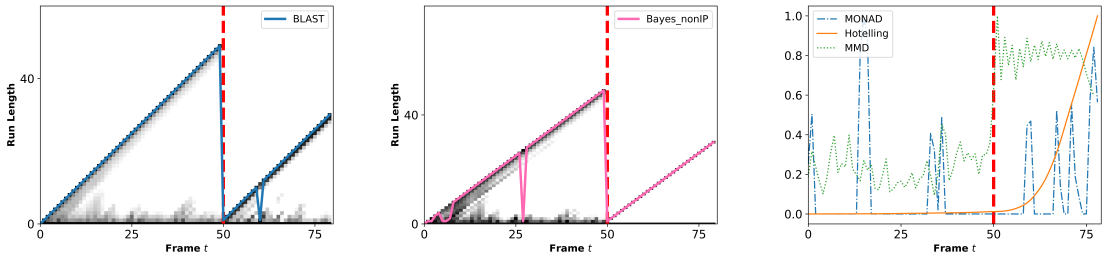
4.5.2.1 Threshold change

Figure 4.14(a) and (b) show the results for BLAST and Bayes_nonIP under the anomalous scenario of reducing the front-end threshold to 75%. Notably, the run length with the highest probability for BLAST exhibits a perfect increase by 1 until $t = 50$, indicating an



(a) (b) (c)

FIGURE 4.14: Results for lowering the front-end threshold to 75% (75 mV)

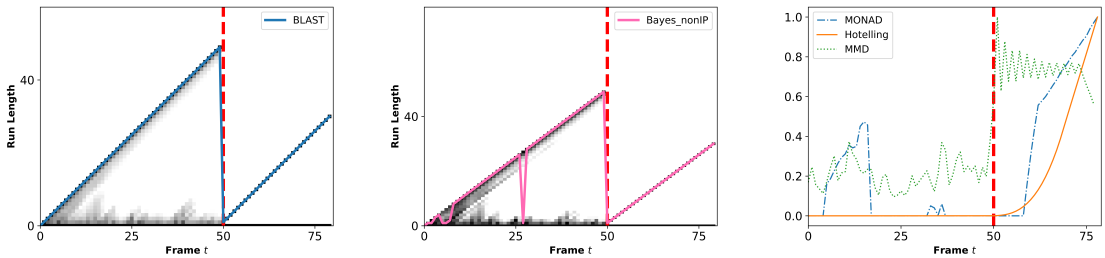


(a) (b) (c)

FIGURE 4.15: Results for lowering the front-end threshold to 50% (50 mV)

absence of change prior to this point. At $t = 51$, there is a precipitous drop back to 0 in the run length, signifying immediate detection without delay. In contrast, the pre-change run length for Bayes_nonIP demonstrates considerable instability, particularly before $t = 10$, potentially leading to numerous false alarms.

This disparity can be attributed to the prior precision matrix utilized by BLAST, which encapsulates the general structure of the six measurements in particle events. This integration allows the model to better understand the data before seeing the online observed matrices, resulting in quick understanding of the data structure and thus more stable pre-change evidence. Conversely, Bayes_nonIP processes basic measurements independently, disregarding potential correlations. Consequently, it depends heavily on online pre-change data to comprehend the data structure, resulting in unstable pre-change run length probabilities. Once the data structure is thoroughly understood — either from offline training or online pre-change data — both BLAST and Bayes_nonIP promptly detect anomalies.



(a) (b) (c)

FIGURE 4.16: Results for lowering the front-end threshold to 25% (25 mV)

However, in scenarios with limited pre-change data, Bayes_nonIP's detection capabilities may falter due to insufficient learning of the data structure from offline and sparse pre-change data.

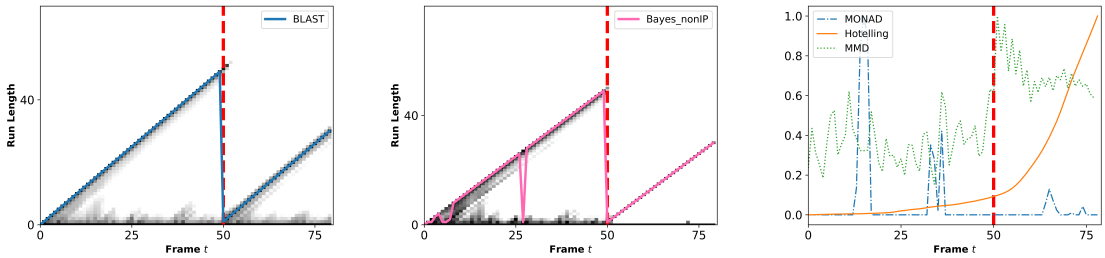
Figure 4.14(c) shows the detection statistics from MONAD, Hotelling's T^2 and MMD under the same anomalous conditions. For the MMD statistics, we see that while it peaks up after the change-point, its pre-change statistics are quite unstable and volatile, which leads to false alarms. On the other hand, the Hotelling's T^2 statistics increases noticeably more slowly after the change, which suggests a larger detection delay. However, the MONAD statistics is not able to detect the change at all. This suggests that, by integrating data structure, BLAST and Bayes_nonIP yield improved performance over MONAD, Hotelling's T^2 and MMD. Moreover, the adoption of a Bayesian framework in BLAST and Bayes_nonIP facilitates uncertainty quantification in change-point detection, offering a distinct advantage.

Figures 4.15 and 4.16 show the results under the anomalous conditions where the front-end threshold is reduced to 50% and 25%, respectively. The observations in these scenarios are consistent with those noted when the threshold was decreased to 75%. Specifically, for BLAST, the run length with the maximum probability incrementally increases by 1, consistently indicating the absence of any change prior to $t = 50$. The run length sharply reverts to 0 at $t = 51$, signifying an immediate detection without any delay. On the other hand, the pre-change run length associated with the maximum probability for Bayes_nonIP ex-

hibits significant instability, particularly before $t = 10$. This instability has the potential to trigger multiple false alarms, which could indicate less robust performance in early detection stages. Regarding the other methods, both MMD and the Hotelling's T^2 demonstrate the ability to detect changes; however, MMD suffers from unstable pre-change statistics, and the Hotelling's T^2 suffers from a significant delay in detection. Again, the MONAD approach is not able to detect the change.

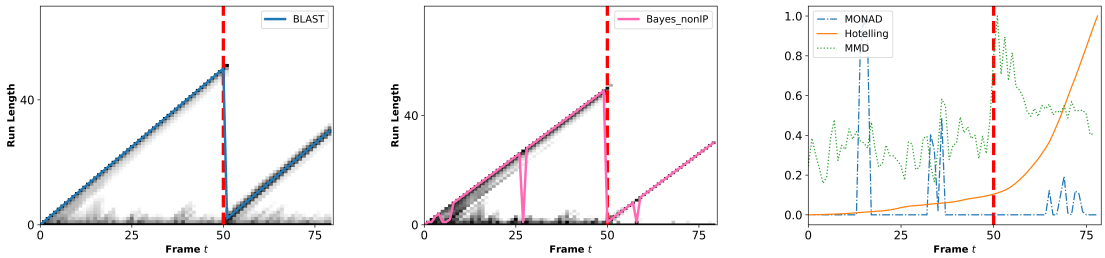
These results reinforce the observation that while BLAST exhibits consistent and prompt detection capabilities, Bayes nonIP struggles with early stability, leading to potential false alarms. Moreover, the comparative underperformance of MMD and the ineffectiveness of Hotelling's T^2 and MONAD in these scenarios highlight the superior detection proficiency of BLAST in handling threshold-based anomalies in particle physics experiments.

4.5.2.2 Voltage change



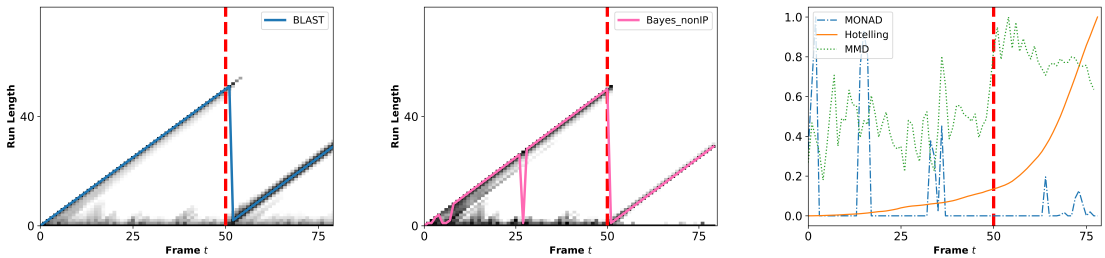
(a) (b) (c)

FIGURE 4.17: Results for reducing voltage to 75% (-900 V)



(a) (b) (c)

FIGURE 4.18: Results for reducing voltage to 50% (-600 V)



(a) (b) (c)

FIGURE 4.19: Results for reducing voltage to 25% (-300 V)

Figures 4.17, 4.18 and 4.19 detail the results under the anomalous scenarios of voltage reduction to 75%, 50%, and 25%. These results echo the results observed in the threshold change experiments, with BLAST consistently showing a gradual increase in run length with the highest probability up to $t = 50$, followed by an abrupt drop to 0 at $t = 51$, indicating swift anomaly detection. Conversely, Bayes_nonIP continues to display pre-change run length instability, especially before $t = 10$, thereby increasing the risk of false alarms.

A notable difference in the voltage reduction experiments is seen in the performance of MMD. While it manages to detect changes, the pre-change statistics are more unstable compared to the threshold-change experiments, which again leads to increased false alarms. MMD struggles to detect the change at a 50% voltage reduction and nearly fails at 25%. Hotelling's T^2, consistent with previous observations, remains more subdued after the change, which results in greater detection delay across all voltage reduction scenarios. MONAD statistics again is not able to detect the change.

These outcomes further emphasize the robust detection capability of BLAST, which, through its prior precision matrix, promptly understands and adapts to data structural changes. This advantage is contrasted with the limitations of Bayes_nonIP, particularly in scenarios with minimal pre-change data. The consistent underperformance of MMD and the non-detection by Hotelling's T^2 and MONAD accentuate the superior adaptability of BLAST and Bayes_nonIP, particularly in their ability to integrate data structure and

model uncertainty for more effective change-point detection in particle physics experiments.

5. Conclusion

To summarize, this dissertation presents innovative methods fr online change point and anomaly detection in high-dimensional data settings. Each chapter introduces a unique approach tailored to address specific challenges and data structures, ultimately aiming to improve the accuracy and efficiency of detecting changes and anomalies.

In Chapter 2, We have proposed a novel topology-aware, non-parametric monitoring framework called the Persistence Diagram-based Change-Point Detection (PERCEPT) method, which leverages tools from topological data analysis for change detection. The idea is to first extract the topological structure of the data via persistence diagrams, then leverage a non-parametric, histogram-based change detection approach on these diagrams to sequentially detect topological changes. A suite of simulation experiments and two applications show that, when the underlying change is topological in nature, the proposed PERCEPT approach yields noticeably improved detection performance over existing approaches.

There are several interesting future directions for refining PERCEPT for broader applications. First, exploring a more localized detection approach would be of interest, which can pinpoint and monitor local changes (e.g., local translation/rotation shifts) in topology. Second, there has been recent work on two-parameter persistence (Wright & Zheng, 2020), particularly on its robustness in extracting topological structure in the presence of noise; integrating such ideas within PERCEPT would allow for a more robust topology-aware monitoring approach. The third direction is to investigate broader uses of PERCEPT in applications for which TDA has found recent success, particularly neuroscience (H. Wang et al., 2020; H. Wang et al., 2022) and complex physical systems (Mak et al., 2018).

In Chapter 3, we proposed a new e^{RPCA} method for jointly recovering embedded low-rank structures and corresponding sparse anomalies from data matrices corrupted by non-Gaussian noise from the exponential family distribution. This method is directly motivated by two applications, the first for steel defect detection and the second for crime activity monitoring. The e^{RPCA} employs a novel optimization formulation that leverages the underlying

exponential noise structure for performing the desired low-rank plus sparse decomposition. We then presented an alternating direction method of multiplier algorithm for efficient optimization, both for the single-group e^{RPCA} (where anomalies are shared over all data matrices) and the multi-group e^{RPCA} (where anomalies may vary over different groups). We demonstrated the effectiveness of the proposed e^{RPCA} in a suite of numerical experiments with varying non-Gaussian noise and in our two motivating applications for steel defect detection and crime monitoring. We found that, when the underlying noise distribution is considerably non-Gaussian and can be reliably inferred from domain knowledge, our method can yield considerable improvements over the state-of-the-art.

Given promising results, there are numerous directions for impactful future work. First, a Bayesian extension of the e^{RPCA} would be of interest, as in many applications, it would be useful to have a reliable quantification of uncertainty for anomaly detection. Recent work on Bayesian matrix modeling (Yuchi et al., 2023) appears promising on this front. Another direction is the application of such methods for high-energy physics (Everett et al., 2021; Ji et al., 2022; K. Li et al., 2023), where there has been much recent work on using anomaly detection techniques to identify new particle activity from heavy-ion collisions; see, e.g., (Kasieczka et al., 2021). A key challenge is the non-Gaussian measurement noise in such systems (Flöss et al., 2023), and the e^{RPCA} can be highly useful in this setting. The decoupling of shared and unique features from diverse data sources (see, e.g., N. Shi and Kontar, 2022) is also of potential interest as future work.

In Chapter 4, we proposed a new Bayesian onLine Structure-Aware change deTection (BLAST), which tackles the urgent challenges of (i) scalable monitoring of high-dimensional images, (ii) capturing local image structure (e.g., edges, shapes) for quick change-detection, and (iii) reliable uncertainty quantification in monitoring. Such challenges are ubiquitous in broad scientific and engineering applications, from high-energy physics to surveillance. To address this, BLAST leverages the deep Gaussian Markov random field prior in Sidén and Lindsten, 2020, which captures prior knowledge on expected image structure. With this elicited prior, BLAST then employs a carefully-constructed online Bayesian CUSUM proce-

dure for scalable image monitoring. A key advantage of BLAST is that, at each time-step, the computation of its full posterior run length distribution can be performed in $\mathcal{O}(M^2)$ work (where M is the number of image pixels), thus facilitating timely monitoring of large images. We demonstrate the effectiveness of BLAST in a suite of numerical experiments and in two applications on street scene monitoring and particle physics experimentation.

Despite promising developments, there are several important directions for future investigation. These directions target the broader use of BLAST for modern but perhaps less explored applications of change-point detection. One such area is in the timely *novelty detection* of anomalous activity in particle physics experiments (Kasieczka et al., 2021), which contribute to discoveries of new particles (e.g., the Higgs Boson (Bass et al., 2021)). The key challenge there is the sheer volume of image measurement data collected, which can easily exceed terabytes of memory per second! We are exploring the incorporation of recent large-scale kernel learning methods (K. Li et al., 2024; K. Li & Mak, 2023) to further improve scalability for such a massive data setting; tackling this can catalyze recent promising high-energy physics developments (Everett et al., 2022; Ji et al., 2022; K. Li et al., 2023). We are also investigating the extension of BLAST for non-Gaussian image measurements. Such a situation arises in high-energy physics, where experimental observables are typically measured as counts (Ehlers et al., 2022; Fan et al., 2024; Ji et al., 2024). This non-Gaussian extension likely requires careful integration of convex programming for real-time scalability, following recent work in (Zheng et al., 2024).

Appendix A. PERCEPT

A.1 Experiments for approximating the ARL and EDD

Experiment 1: To compute the ARL, we generate n sequences of pre-change samples of length m. We create a random pool with M pre-change samples to reduce the computational cost. For each sequence, m samples are drawn randomly from the pool with replacement, and we repeat this process n times. Let $\mathcal{T}$ be the stopping time of the detection procedure, and if there is no change-point in the sequence of length m (all samples are drawn from the pre-change distribution), from the discussion in (L. Xie & Xie, 2021) we have:

$$P(\mathcal{T} > m) = P(\max_{0 \leq t \leq m} \chi_t < b) \approx \exp\{-m/\lambda\},$$

where λ is the estimation of the ARL. For any given threshold b, we could get the estimation of the ARL based on the percentage of sequences whose maximum online statistics is below the threshold, among all n sequences. More specifically, we approximate the ARL as $m/(-\ln \hat{p})$, where $\hat{p}$ is the percentage.

Experiment 2: To compute the EDD, we generate n sequences of pre-change samples of length m' followed by post-change samples of length m. The pre-change samples are only used as historical data, in order to construct the pre-change histograms $\boldsymbol{\omega}_{t,k}^{[1,1]}$, $\boldsymbol{\omega}_{t,k}^{[1,2]}$ as shown in Figure 2.5b. The online test statistics are calculated only from the onset of post-change samples to obtain the detection delay. We consider the same list of threshold b as we used in computing the ARL, and find the average detection delay over n sequences.

A.2 Details on the lower star filtration

When a new vertex is added in the sublevel set, the topological change depends on whether the vertex is a maximum, minimum, regular, or a saddle of the function. Figure A.1 visualizes a regular point and saddle point (in yellow), and the edges and faces in the sublevel sets (in blue). The topological features do not change after introducing a regular point, but the number of connected components decreases by one after introducing a saddle point.

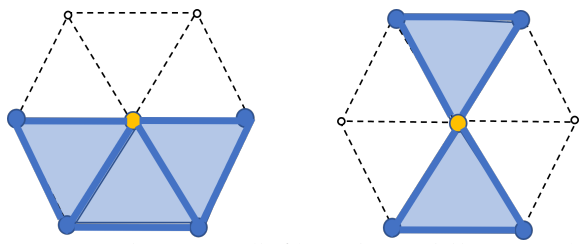


FIGURE A.1: Visualizing a regular point (left) and a saddle point (right) in the lower star filtration.

A.3 Connection on distributional & bottleneck distances

Here, we elaborate further on the connection between the difference between the pre- and post-change persistence distributions, $\|p_{\text{pre}} - p_{\text{post}}\|_2$, and its corresponding bottleneck distance $d_B(\mathcal{D}_{\text{pre}}, \mathcal{D}_{\text{post}})$. Since the bottleneck distance is defined on the sampled PDs $\mathcal{D}_{\text{pre}}$ and $\mathcal{D}_{\text{post}}$ at a given time t, we will make this connection on the *empirical* pre- and post-change persistence distributions $\hat{p}_{\text{pre}}$ and $\hat{p}_{\text{post}}$ at time t, respectively.

The following proposition provides a link between the empirical persistence distribution difference $\|\hat{p}_{\text{pre}} - \hat{p}_{\text{post}}\|_2$ and the bottleneck distance of its corresponding persistence diagrams $\mathcal{D}_{\text{pre}}$ and $\mathcal{D}_{\text{post}}$, under asymptotic conditions.

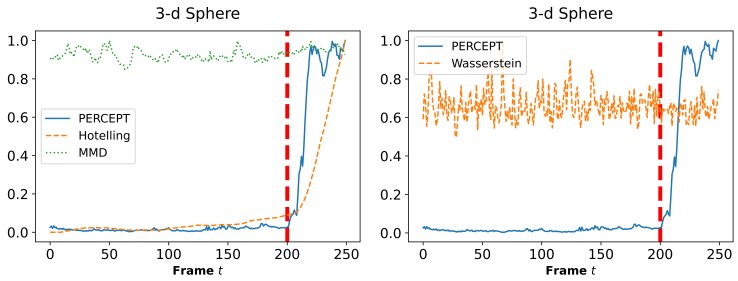
Proposition 2. *Let $\mathcal{D}_{\text{pre}}$ and $\mathcal{D}_{\text{post}}$ be the PDs of samples from pre- and post-change data at time t, and let $\hat{p}_{\text{pre}}$ and $\hat{p}_{\text{post}}$ be the corresponding persistence histograms given a fixed partition of L bins at time t. Suppose all birth times are unique, and suppose the number of histogram bins L are sufficiently large such that the persistence histograms at each time has at most one point in each bin. Then $||\hat{p}_{\text{pre}} - \hat{p}_{\text{post}}||_2^2 \geq d_B^2(\mathcal{D}_{\text{pre}}, \mathcal{D}_{\text{post}})$.*

In other words, as the number of histogram bins L goes to infinity, the distributional difference $||\hat{p}_{\text{pre}} - \hat{p}_{\text{post}}||_2$ can be lower bounded by the bottleneck distance $d_B(\mathcal{D}_{\text{pre}}, \mathcal{D}_{\text{post}})$. This, combined with Equation (7) of the main paper, suggests that the greater the topological difference is between pre- and post-change data, the smaller its detection delay, as desired.

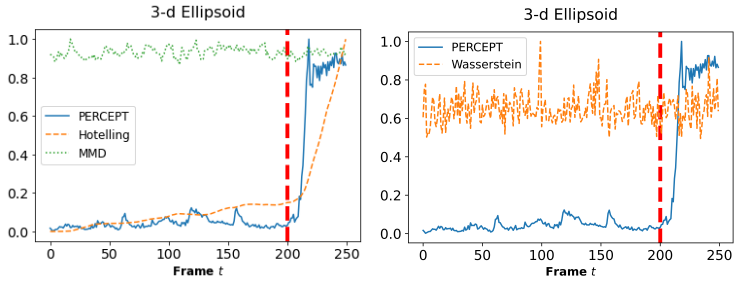
A.4 Increasing Dimensionality on geometrix structures

In Section 2.4.3, we investigate how well the methods perform on data generate on higher

dimensional geometric structures under noise change, and Figure A.2 shows the detection statistics for 3-D sphere and ellipsoids.



(a)



(c)

FIGURE A.2: (a) The test statistics using PERCEPT, Hotelling's T^2, MMD and Wasserstein distance at each time t for the 3-D sphere, where the noise changes from $N(0, 0.05)$ to $N(0, 0.10)$. Again, the vertical red dashed lines indicate the true change-point at $t^* = 200$. (b) Same for 3-D ellipsoids.

Appendix B. ERPCA

B.1 Closed-form Optimization Updates for θ

Table B.1: Closed-form updates of (3.19) for common (one-parameter) exponential family distributions.

Distribution	$\theta^*_{j,k}$
Poisson	root of $a\theta^2_{j,k} + b\theta_{j,k} + c = 0$ with the smallest loss, where $a = \mu,$ $b = -\mu(L_{j,k} + S_{j,k}) + Y_{j,k} + 1,$ $c = -\bar{M}_{j,k}$
Bernoulli	root of $a\theta^3_{j,k} + b\theta^2_{j,k} + c\theta_{j,k} + d = 0$ with the smallest loss, where $a = -\mu,$ $b = \mu(1 + L_{j,k} + S_{j,k}) - Y_{j,k},$ $c = 1 - \mu(L_{j,k} + S_{j,k}) + Y_{j,k},$ $d = -\bar{M}_{j,k}$
Exponential	root of $a\theta^2_{j,k} + b\theta_{j,k} + c = 0$ with the smallest loss, where $a = \mu,$ $b = -\mu(L_{j,k} + S_{j,k}) + Y_{j,k} + \bar{M}_{j,k},$ $c = -1$
Gaussian	$(\bar{M}_{j,k} + \mu(L_{j,k} + S_{j,k}) - Y_{j,k})/(1 + \mu)$

B.2 Simulation Set-up for $\mathbf{L}$ and $\mathbf{S}$

The settings for constructing the low-rank matrix $\mathbf{L}$ are as follows. For the Bernoulli distribution, we use $\mu = 0.5, \sigma = 0.15$. For the Exponential distribution, $\mu = 1, \sigma = 0.15$; and for the Poisson distribution, $\mu = 50, \sigma = 2$. When creating the sparse matrix $\mathbf{S}$, the spike ranges are $[0.2, 0.3]$ for Bernoulli distribution and Exponential distributions, and $[2, 5]$ for the Poisson distribution. Note that the Bernoulli parameters must lie between $[0, 1]$, so any out-of-range sampled values are set to zero or one (whichever is closest). For

the Exponential and Poisson distributions, only positive parameters are permitted, thus negative sampled values are set as zero.

Bibliography

Adams, R. P., & MacKay, D. J. (2007). Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*.

Aurenhammer, F. (1991). Voronoi diagrams — a survey of a fundamental geometric data structure. *ACM Computing Surveys*, *23*(3), 345–405.

Bass, S. D., De Roeck, A., & Kado, M. (2021). The Higgs boson implications and prospects for future discoveries. *Nature Reviews Physics*, *3*(9), 608–624.

Basseville, M., & Nikiforov, I. V. (1993). *Detection of abrupt changes: Theory and application*. Prentice Hall.

Beksi, W. J., & Papanikolopoulos, N. (2016). 3D region segmentation using topological persistence. *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1079–1084. https://doi.org/10.1109/IROS.2016.7759183

Bendich, P., Edelsbrunner, H., & Kerber, M. (2011). Computing robustness and persistence for images. *IEEE Transactions on Visualization and Computer Graphics*, *16*, 1251–1260. https://doi.org/10.1109/TVCG.2010.139

Bernhard, S., Smola, A. J., & Bach, F. (2018). *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT Press.

Bertsekas, D. P. (1982). *Constrained optimization and lagrange multiplier methods*. Academic Press.

Bouwmans, T., & Zahzah, E. H. (2014). Robust PCA via principal component pursuit: A review for a comparative evaluation in video surveillance. *Computer Vision and Image Understanding*, *122*, 22–34.

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, *3*(1), 1–122.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Cai, J.-F., Candès, E. J., & Shen, Z. (2010). A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, *20*(4), 1956–1982.

Candès, E. J., Li, X., Ma, Y., & Wright, J. (2011). Robust principal component analysis? *Association for Computing Machinery*, *58*(3).

Cang, Z., Mu, L., Wu, K., Opron, K., Xia, K., & Wei, G.-W. (2015). A topological approach for protein classification. *Molecular Based Mathematical Biology*, *3*(1), 140–162.

Cao, J., Wang, N., Zhang, J., Wen, Z., Li, B., & Liu, X. (2016). Detection of varied defects in diverse fabric images via modified RPCA with noise term and defect prior. *International Journal of Clothing Science and Technology*, *28*(4), 516–529.

Cao, Y., & Xie, Y. (2015). Poisson matrix recovery and completion. *IEEE Transactions on Signal Processing*, *64*(6), 1609–1620.

Casella, G., & Berger, R. L. (2002). *Statistical inference*. Thomson Learning.

Chan, S. H. (2022). What does a one-bit quanta image sensor offer? *arXiv preprint arXiv:2208.10350*.

Chan, S. H., Wang, X., & Elgendy, O. A. (2016). Plug-and-play ADMM for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, *3*(1), 84–98.

Charbon, E. (2014). Single-photon imaging in complementary metal oxide semiconductor processes. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *372*(2012), 20130100.

Chazal, F. (2016). High-dimensional topological data analysis. *Handbook of Discrete and Computational Geometry (3rd ed.)*

Cohen-Steiner, D., Edelsbrunner, H., & Harer, J. (2007). Stability of persistence diagrams. *Discrete & Computational Geometry*, *37*, 103–120.

Collins, M., Dasgupta, S., & Schapire, R. E. (2001). A generalization of principal components analysis to the exponential family. *Advances in Neural Information Processing Systems*, *14*.

Dai, J., Li, Y., He, K., & Sun, J. (2016). R-fcn: Object detection via region-based fully convolutional networks. *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 379–387.

David M. Blei, A. K., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, *112*(518), 859–877.

Department, A. P. (2019). Atlanta police make changes to four zone boundaries as part of beat redesign [Available: https://www.atlantapd.org/Home/Components/News/News/190/].

Ding, X., He, L., & Carin, L. (2011). Bayesian robust principal component analysis. *IEEE Transactions on Image Processing*, *20*(12), 3419–3430.

Doshi, K., & Yilmaz, Y. (2021). Online anomaly detection in surveillance videos with asymptotic bound on false alarm rate. *Pattern Recognition*, *114*, 107865.

Ebrahimi, S., Fleuret, J., Klein, M., Théroux, L.-D., Georges, M., Ibarra-Castanedo, C., & Maldague, X. (2021). Robust principal component thermography for defect detection in composites. *Sensors*, *21*(8), 2682.

Eck, J. E., Chainey, S., Cameron, J. G., Leitner, M., & Wilson, R. E. (2005). Mapping crime: Understanding hot spots [Available: https://www.ojp.gov/pdffiles1/nij/209393.pdf].

Edelsbrunner, H., & Harer, J. (2008). Persistent homology - a survey. *Contemporary Mathematics*, *453*, 257–282.

Ehlers, R., Angerami, A., Arora, R., Bass, S., Cao, S., Chen, Y., Du, L., Dai, T., Elfner, H., Fan, W., et al. (2022). Bayesian analysis of qgp jet transport using multi-scale modeling applied to inclusive hadron and reconstructed jet data. *arXiv preprint arXiv:2208.07950*.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 226–231.

Etten, A. V. (2018). You only look twice: Rapid multi-scale object detection in satellite imagery. *CoRR*, *abs/1805.09512*. http://arxiv.org/abs/1805.09512

Everett, D., Ke, W., Paquet, J.-F., Vujanovic, G., Bass, S., Du, L., Gale, C., Heffernan, M., Heinz, U., Liyanage, D., et al. (2021). Multisystem Bayesian constraints on the transport coefficients of qcd matter. *Physical Review C*, *103*(5), 054904.

Everett, D., Oliinychenko, D., Luzum, M., Paquet, J.-F., Vujanovic, G., Bass, S., Du, L., Gale, C., Heffernan, M., Heinz, U., et al. (2022). Role of bulk viscosity in deuteron production in ultrarelativistic nuclear collisions. *Physical Review C*, *106*(6), 064901.

Fan, W., Vujanovic, G., Bass, S., Angerami, A., Arora, R., Cao, S., Chen, Y., Dai, T., Du, L., Ehlers, R., et al. (2024). New metric improving bayesian calibration of a multistage approach studying hadron and inclusive jet suppression. *Physical Review C*, *109*(6), 064903.

Fatahalian, K., Sugerman, J., & Hanrahan, P. M. (2004). Understanding the efficiency of gpu algorithms for matrix-matrix multiplication. *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware*, 133–137.

Fearnhead, P. (2006). Exact and efficient bayesian inference for multiple changepoint problems. *Statistics and Computing*, *16*, 203–213.

Fearnhead, P., & Liu, Z. (2007). On-line inference for multiple changepoint problems. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, *69*(4), 589–605.

Feller, W. (1968). *An introduction to probability theory and its applications* (Vol. 1). Wiley. http://www.amazon.ca/exec/obidos/redirect?tag=citeulike04-20%7B%5C&%7Dpath=ASIN/0471257087

Flöss, T., Biagetti, M., & Meerburg, P. D. (2023). Primordial non-Gaussianity and non-Gaussian covariance. *Physical Review D*, *107*(2), 023528.

Forbes, C., Evans, M., Hastings, N., & Peacock, B. (2011). *Statistical distributions* (4th). John Wiley & Sons.

Fothergill, S., Mentis, H., Kohli, P., & Nowozin, S. (2012). Instructing people for training gestural interactive systems. *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12*, 1737–1746. https://doi.org/10.1145/2207676.2208303

Fox, K., & Tran, L. (2020). New sunspots potentially herald increased solar activity [Accessed: 2021-02-26].

Freudenthal, H. (1942). Simplizialzerlegungen von beschrankter flachheit. *Annals of Mathematics*, *43*(3), 580–582.

Frith, M. J., Bowers, K. J., & Johnson, S. D. (2022). Household occupancy and burglary: A case study using COVID-19 restrictions. *Journal of Criminal Justice*, *82*(100).

Garnett, R., Osborne, M. A., & Roberts, S. J. (2009). Sequential bayesian prediction in the presence of changepoints. *International Conference on Machine Learning*.

Geman, S., & Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PAMI-6*(6), 721–741.

Ghrist, R. (2008). Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society*, *45*(1), 61–75.

Gidea, M., & Katz, Y. A. (2017). Topological data analysis of financial time series: Landscapes of crashes. *Physica A: Statistical Mechanics and its Applications*, *491*, 820–834.

Green, P. J. (1995). Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, *82*(4), 711–732.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., & Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research*, *13*, 723–773.

Grosso, G., Lai, N., Letizia, M., Pazzini, J., Rando, M., Rosasco, L., Wulzer, A., & Zanetti, M. (2023). Fast kernel methods for data quality monitoring as a goodness-of-fit test. *Machine Learning: Science and Technology*, *4*(3), 035029.

Gu, Q., Wang, Z., & Liu, H. (2016). Low-rank and sparse structure pursuit via alternating minimization. In A. Gretton & C. C. Robert (Eds.), *Proceedings of the 19th international conference on artificial intelligence and statistics* (pp. 600–609). PMLR.

Guyon, C., Bouwmans, T., & Zahzah, E.-H. (2012). Foreground detection based on low-rank and block-sparse matrix decomposition. *IEEE International Conference on Image Processing*, 1225–1228.

Han, S., Okonek, T., Yadav, N., & Zheng, X. (2018). Distributions of matching distances in topological data analysis. *SIAM Undergraduate Research Online*, *13*.

Hartman, L., & Hössjer, O. (2008). Fast kriging of large data sets with Gaussian Markov random fields. *Computational Statistics & Data Analysis*, *52*(5), 2331–2349.

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (Vol. 2). Springer.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

Hotelling, H. (1947). Multivariate quality control illustrated by air testing of sample bombsights. McGraw-Hill.

Islambekov, U., Yuvaraj, M., & Gel, Y. (2019). Harnessing the power of topological data analysis to detect change points. *Environmetrics*, *31*, e2612.

Ji, Y., Mak, S., Soeder, D., Paquet, J. F., & Bass, S. A. (2024). A graphical multi-fidelity Gaussian process model, with application to emulation of expensive computer simulations. *Technometrics*, *66*(2), 267–281.

Ji, Y., Yuchi, H. S., Soeder, D., Paquet, J.-F., Bass, S. A., Joseph, V. R., Wu, C., & Mak, S. (2022). Multi-stage multi-fidelity Gaussian process modeling, with application to heavy-ion collisions. *arXiv preprint arXiv:2209.13748*.

Jiang, H., Kim, B., Guan, M., & Gupta, M. (2018). To trust or not to trust a classifier. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems* (pp. 5541–5552). Curran Associates, Inc.

Jiao, Y., Chen, Y., & Gu, Y. (2018). Subspace change-point detection: A new model and solution. *IEEE Journal of Selected Topics in Signal Processing*, *12*(6), 1224–1239.

Kashyap, R., & Chellappa, R. (1983). Estimation and choice of neighbors in spatial-interaction models of images. *IEEE Transactions on Information Theory*, *29*(1), 60–72.

Kasieczka, G., Nachman, B., Shih, D., Amram, O., Andreassen, A., Benkendorfer, K., Bortolato, B., Brooijmans, G., Canelli, F., Collins, J. H., Dai, B., Freitas, F. F. D., Dillon, B. M., Dinu, I.-M., Dong, Z., Donini, J., Duarte, J., Faroughy, D. A., Gonski, J., ... Yunus, M. (2021). The lhc olympics 2020 a community challenge for anomaly detection in high energy physics. *Reports on Progress in Physics*, *84*(12), 124201.

Ketchen, D. J., & Shook, C. L. (1996). The application of cluster analysis in strategic management research: An analysis and critique. *Strategic Management Journal*, *17*(6), 441–458. http://www.jstor.org/stable/2486927

Kiela, D., & Bottou, L. (2014). Learning image embeddings using convolutional neural networks for improved multi-modal semantics. *Conference on Empirical Methods in Natural Language Processing*.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations*.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Knoblauch, J., & Damoulas, T. (2018). Spatio-temporal Bayesian on-line changepoint detection with model selection. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th international conference on machine learning* (pp. 2718–2727). PMLR.

Kolar, M., Song, L., Ahmed, A., & Xing, E. P. (2010). Estimating time-varying networks. *The Annals of Applied Statistics*, *4*(1), 94–123.

Kontoghiorghes, E. J. (2005). *Handbook of parallel computing and statistics (statistics, textbooks and monographs)*. Chapman & Hall/CRC.

Kontogiorgis, S., & Meyer, R. R. (1989). A variable-penalty alternating directions method for convex optimization. *Mathematical Programming*, *83*, 29–53.

Krishna, A., Mak, S., & Joseph, R. (2019). Distributional clustering: A distribution-preserving clustering method. *arXiv preprint arXiv:1911.05940*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *25*, 1097–1105.

Lai, T. L. (1998). Information bounds and quick detection of parameter changes in stochastic systems. *IEEE Transactions on Information Theory*, *44*(7), 2917–2929.

Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business & Information Systems Engineering*, *6*, 239–242.

Lau, T. S., Tay, W. P., & Veeravalli, V. V. (2018). A binning approach to quickest change detection with unknown post-change distribution. *IEEE Transactions on Signal Processing*, *67*(3), 609–621.

Lee, B. S., & Haran, M. (2022). Picar: An efficient extendable approach for fitting hierarchical spatial models. *Technometrics*, *64*(2), 187–198.

Li, K., Balakirsky, M., & Mak, S. (2024). Trigonometric quadrature fourier features for scalable Gaussian process regression. *2024 International Conference on Artificial Intelligence and Statistics*.

Li, K., & Mak, S. (2023). ProSpar-GP: scalable Gaussian process modeling with massive non-stationary datasets. *arXiv preprint arXiv:2311.08752*.

Li, K., Mak, S., Paquet, J.-F., & Bass, S. A. (2023). Additive multi-index Gaussian process modeling, with application to multi-physics surrogate modeling of the quark-gluon plasma. *arXiv preprint arXiv:2306.07299*.

Li, S., Xie, Y., Dai, H., & Song, L. (2015). M-statistic for kernel change-point detection. *Proceedings of the Advances in Neural Information Processing Systems*, 3366–3374.

Li, S. Z. (1995). *Markov random field modeling in computer vision*. Springer-Verlag.

Liang, X., Ren, X., Zhang, Z., & Ma, Y. (2012). Repairing sparse low-rank texture. *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*, 482–495.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. *Computer Vision – ECCV 2014*, 740–755.

Lin, Z., Chen, M., & Ma, Y. (2010). The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*.

Liu, D. C., & Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, *45*(1-3), 503–528.

Liu, L., Dobriban, E., & Singer, A. (2018). Epca: High dimensional exponential family PCA. *Annals of Applied Statistics*, *12*(4), 2121–2150.

Liu, W., Luo, W., Lian, D., & Gao, S. (2018). Future frame prediction for anomaly detection - a new baseline. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6536–6545.

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, *28*(2), 129–137.

Lorden, G. (1971). Procedures for reacting to a change in distribution. *Annals of Mathematical Statistics*, *42*(6), 1897–1908.

Luenberger, D. G., & Ye, Y. (1984). *Linear and nonlinear programming* (Vol. 2). Springer.

Luo, W., Liu, W., & Gao, S. (2017). Remembering history with convolutional lstm for anomaly detection. *2017 IEEE International Conference on Multimedia and Expo (ICME)*, 439–444.

Mak, S., Sung, C.-L., Wang, X., Yeh, S.-T., Chang, Y.-H., Joseph, V. R., Yang, V., & Wu, C. F. J. (2018). An efficient surrogate model for emulation and physics extraction of large eddy simulations. *J Amer Statist Assoc*, *113*(524), 1443–1456.

Mohamed, S., Ghahramani, Z., & Heller, K. A. (2008). Bayesian exponential family PCA. *Advances in Neural Information Processing Systems*, *21*.

Molloy, T. L., & Ford, J. J. (2017). Misspecified and asymptotically minimax robust quickest change detection. *IEEE Transactions on Signal Processing*, *65*(21), 5730–5742.

Moustakides, G. V. (1986). Optimal stopping times for detecting changes in distributions. *Annals of Statistics*, *14*(4), 1379–1387.

NASA. (n.d.). Solar dynamics observatory [online] [Available: https://sdo.gsfc.nasa.gov/assets/docs/SDO_Guide.pdf].

Nesterov, Y. E. (1983). A method of solving a convex programming problem with convergence rate $\mathcal{O}\left(1/k^2\right)$. *Doklady Akademii Nauk*, *269*(3), 543–547.

Nguyen, L. V., Kodagoda, S., & Ranasinghe, R. (2016). Spatial sensor selection via gaussian markov random fields. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *46*(9), 1226–1239.

Nobles, M. R., Ward, J. T., & Tillyer, R. (2016). The impact of neighborhood context on spatiotemporal patterns of burglary. *Journal of Research in Crime and Delinquency*, *53*(5), 711–740.

Nocedal, J., & Wright, S. J. (1999). *Numerical optimization*. Springer.

Ofori-Boateng, D., Dominguez, I. S., Akcora, C., Kantarcioglu, M., & Gel, Y. R. (2021). Topological anomaly detection in dynamic multilayer blockchain networks. *Machine Learning and Knowledge Discovery in Databases. Research Track*, 788–804.

Oh, S., Hoogs, A., Perera, A., Cuntoor, N., Chen, C.-C., Lee, J. T., Mukherjee, S., Aggarwal, J. K., Lee, H., Davis, L., Swears, E., Wang, X., Ji, Q., Reddy, K., Shah, M., Vondrick, C., Pirsiavash, H., Ramanan, D., Yuen, J., … Desai, M. (2011). A large-scale benchmark dataset for event recognition in surveillance video, 3153–3160. https://doi.org/10.1109/CVPR.2011.5995586

Okhrin, Y., Schmid, W., & Semeniuk, I. (2021). New approaches for monitoring image data. *IEEE Transactions on Image Processing*, *30*, 921–933.

O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.

Otto, P. (2021). Parallelized monitoring of dependent spatiotemporal processes. In S. Knoth & W. Schmid (Eds.), *Frontiers in statistical quality control 13* (pp. 165–183). Springer International Publishing.

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

PAO Severstal. (2019). Severstal: Steel defect detection [Available: https://www.kaggle.com/competitions/severstal-steel-defect-detection].

Pearson, K. (1901). Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, *2*(11), 559–572.

Perea, J. A., & Harer, J. (2015). Sliding windows and persistence: An application of topological methods to signal analysis. *Foundations of Computational Mathematics*, *15*(3), 799–838.

Poor, H. V., & Hadjiliadis, O. (2008). *Quickest detection*. Cambridge University Press.

Ramachandra, B., & Jones, M. J. (2020). Street scene: A new dataset and evaluation protocol for video anomaly detection. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2569–2578.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788.

Roberts, S. W. (1959). Control chart tests based on geometric moving averages. *Technometrics*, *1*(3), 239–250.

Roy, S., Atchadé, Y., & Michailidis, G. (2016). Change Point Estimation in High Dimensional Markov Random-Field Models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *79*(4), 1187–1206.

Rue, H., & Held, L. (2005). *Gaussian markov random fields: Theory and applications*. Chapman; Hall/CRC.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*, 533–536.

Saatci, Y., Turner, R. D., & Rasmussen, C. E. (2010). Gaussian process change point models. *International Conference on Machine Learning*.

Seversky, L. M., Davis, S., & Berger, M. (2016). On time-series topological data analysis: New data and opportunities. *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1014–1022. https://doi.org/10.1109/CVPRW.2016.131

Shen, Y., Wen, Z., & Zhang, Y. (2014). Augmented Lagrangian alternating direction method for matrix separation based on low-rank factorization. *Optimization Methods and Software*, *29*(2), 239–263.

Shewhart, W. A. (1925). The application of statistics as an aid in maintaining quality of a manufactured product. *Journal of the American Statistical Association*, *20*(152), 546–548.

Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(8), 888–905.

Shi, N., & Kontar, R. A. (2022). Personalized pca: Decoupling shared and unique features. *arXiv preprint arXiv:2207.08041*.

Shih, F. (2010). *Image processing and pattern recognition: Fundamentals and techniques*. Wiley.

Sidén, P., & Lindsten, F. (2020). Deep gaussian markov random fields. *Proceedings of the 37th International Conference on Machine Learning*.

Siegmund, D. O. (1985). *Sequential analysis: Tests and confidence intervals*. Springer.

Simpson, D., Lindgren, F., & Rue, H. (2012). Think continuous: Markovian gaussian models in spatial statistics. *Spatial Statistics*, *1*, 16–29.

Sizemore, A. E., Phillips-Cremins, J. E., Ghrist, R., & Bassett, D. S. (2019). The importance of the whole: Topological data analysis for the network neuroscientist. *Network Neuroscience*, *3*(3), 656–673.

Smith, A. F. M. (1975). A bayesian approach to inference about a change-point in a sequence of random variables. *Biometrika*, *62*(2), 407–416.

Song, H.-R., Fuentes, M., & Ghosh, S. (2008). A comparative study of gaussian geostatistical models and gaussian markov random field models. *Journal of Multivariate Analysis*, *99*(8), 1681–1697.

Sultani, W., Chen, C., & Shah, M. (2018). Real-world anomaly detection in surveillance videos. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6479–6488.

Tartakovsky, A., Nikiforov, I., & Basseville, M. (2015). *Sequential analysis: Hypothesis testing and changepoint detection*. ser. Monographs on Statistics; Applied Probability 136. Boca Raton, London, New York: Chapman & Hall/CRC Press, Taylor & Francis Group.

Tralie, C., Saul, N., & Bar-On, R. (2018). Ripser.py: A lean persistent homology library for python. *The Journal of Open Source Software*, *3*(29), 925. https://doi.org/10.21105/joss.00925

Turaga, P., Chellappa, R., Subrahmanian, V. S., & Udrea, O. (2008). Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, *18*(11), 1473–1488. https://doi.org/10.1109/TCSVT.2008.2005594

Turner, R., Saatci, Y., & Rasmussen, C. (2009). Adaptive sequential bayesian change point detection. In Z. Harchaoui (Ed.), *Temporal segmentation workshop at nips*.

Venkatesan, R., & Li, B. (2017). *Convolutional neural networks in visual computing: A concise guide*. CRC Press.

Wang, H., Xie, L., Cuozzo, A., Mak, S, & Xie, Y. (2020). Uncertainty quantification for inferring Hawkes networks. *Advances in Neural Information Processing Systems*.

Wang, H., Xie, L., Xie, Y., Cuozzo, A., & Mak, S. (2022). Sequential change-point detection for mutually exciting point processes over networks [Accepted]. *Technometrics*.

Wang, J., ke, S., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., & Xiao, B. (2020). Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PP*, 1–1.

West, M. (1986). Bayesian model monitoring. *Journal of the Royal Statistical Society. Series B (Methodological)*, *48*(1), 70–78.

Wright, M., & Zheng, X. (2020). Topological data analysis on simple English wikipedia articles. *The PUMP Journal of Undergraduate Research*, *3*, 308–328. https://journals.calstate.edu/pump/article/view/2410

Xie, L., & Xie, Y. (2021). Sequential change detection by optimal weighted ℓ_2 divergence. *IEEE Journal on Selected Areas in Information Theory*, *2*(2), 747–761.

Xie, L., Xie, Y., & Moustakides, G. V. (2020). Sequential subspace change point detection. *Sequential Analysis*, *39*(3), 307–335.

Xie, Y., Huang, J., & Willett, R. (2012). Change-point detection for high-dimensional time series with missing data. *IEEE Journal of Selected Topics in Signal Processing*, *7*, 12–27. https://doi.org/10.1109/JSTSP.2012.2234082

Xue, Z., Dong, J., Zhao, Y., Liu, C., & Chellali, R. (2019). Low-rank and sparse matrix decomposition via the truncated nuclear norm and a sparse regularizer. *Vis. Comp.*, *35*, 1549–1566.

Yan, H., Paynabar, K., & Shi, J. (2018). Real-time monitoring of high-dimensional functional data streams via spatio-temporal smooth sparse decomposition. *Technometrics*, *60*(2), 181–197.

Yang, B., & Zou, L. (2015). Robust foreground detection using block-based RPCA. *Optik*, *126*(23), 4586–4590.

Yuan, X., & Yang, J. (2013). Sparse and low rank matrix decomposition via alternating direction method. *Pacific Journal of Optimization*, *9*, 167–180.

Yuchi, H. S., Mak, S., & Xie, Y. (2023). Bayesian uncertainty quantification for low-rank matrix completion. *Bayesian Analysis*, *18*(2), 491–518.

Zhang, T., Ramakrishnan, R., & Livny, M. (1996). Birch: An efficient data clustering method for very large databases. *SIGMOD Rec.*, *25*(2), 103–114.

Zhang, Z., Ganesh, A., Liang, X., & Ma, Y. (2012). TILT: Transform invariant low-rank textures. *International Journal of Computer Vision*, *99*, 1–24.

Zhang, Z., Matsushita, Y., & Ma, Y. (2011). Camera calibration with lens distortion from low-rank textures. *CVPR 2011*, 2321–2328.

Zheng, X., Mak, S., Xie, L., & Xie, Y. (2023). Percept: A new online change-point detection method using topological data analysis. *Technometrics*, *65*(2), 162–178.

Zheng, X., Mak, S., Xie, L., & Xie, Y. (2024). Erpca: Robust principal component analysis for exponential family distributions. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *17*(2), e11670.

Zhou, Z., Li, X., Wright, J., Candes, E., & Ma, Y. (2010). Stable principal component pursuit. *2010 IEEE International Symposium on Information Theory*, 1518–1522.

Zhu, Y. (2017). An augmented ADMM algorithm with application to the generalized lasso problem. *Journal of Computational and Graphical Statistics*, *26*(1), 195–204.

Biography

Xiaojun Zheng completed her BA in Mathematics, with distinction, and minors in Statistics and Management Studies, at St. Olaf College in May 2019. She began her PhD program in Statistical Science at Duke University in August 2019, where she was advised by Simon Mak. During her doctoral studies, she won the 2022 INFORMS Data Mining Best Student Paper Award, the 2023 Student Paper Competition organized by the Sections on Physical and Engineering Sciences (SPES) and Quality and Productivity (Q&P) of the American Statistical Association (ASA). Her award-winning paper, "PERCEPT: A New Online Change-Point Detection Method Using Topological Data Analysis," was published in Technometrics in 2023 and also received The American Statistical Association (ASA) Editor's Choice Award. After graduation, Xiaojun will begin her career as a Mathematical Statistician at the FDA.