7-2022

Dynamic Algorithms and Asymptotic Theory for Lp-norm Data Analysis

Mayur Dhanaraj
mxd6023@rit.edu

Dynamic Algorithms and Asymptotic Theory for Lp-norm Data Analysis

by

Mayur Dhanaraj

A dissertation submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical and Computer Engineering

Kate Gleason College of Engineering
Rochester Institute of Technology
Rochester, New York

July, 2022

Dynamic Algorithms and Asymptotic Theory for Lp-norm Data Analysis

by

Mayur Dhanaraj

Committee Approval:

We, the undersigned committee members, certify that we have advised and/or supervised the candidate on the work described in this dissertation. We further certify that we have reviewed the dissertation manuscript and approve it in partial fulfillment of the requirements of the degree of Doctor of Philosophy in Electrical and Computer Engineering.

Dr. Panos P. Markopoulos — Date
Dissertation Advisor

Dr. Sohail Dianat — Date
Dissertation Committee Member

Dr. Eli Saber — Date
Dissertation Committee Member

Dr. Andreas Savakis — Date
Dissertation Committee Member

Dr. Basca Jadamba — Date
Dissertation Defense Chairperson

Certified by:

Dr. Andres Kwasinski — Date
Ph.D. Program Director, Electrical and Computer Engineering

Dynamic Algorithms and Asymptotic Theory for Lp-norm Data Analysis

by

Mayur Dhanaraj

Submitted to the
Kate Gleason College of Engineering
Ph.D. Program in Electrical and Computer Engineering
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Electrical and Computer Engineering
at the Rochester Institute of Technology

Abstract

The focus of this dissertation is the development of outlier-resistant stochastic algorithms for Principal Component Analysis (PCA) and the derivation of novel asymptotic theory for Lp-norm Principal Component Analysis (Lp-PCA). Modern machine learning and signal processing applications employ sensors that collect large volumes of data measurements that are stored in the form of data matrices, that are often massive and need to be efficiently processed in order to enable machine learning algorithms to perform effective underlying pattern discovery. One such commonly used matrix analysis technique is PCA. Over the past century, PCA has been extensively used in areas such as machine learning, deep learning, pattern recognition, and computer vision, just to name a few. PCA's popularity can be attributed to its intuitive formulation on the L2-norm, availability of an elegant solution via the singular-value-decomposition (SVD), and asymptotic convergence guarantees. However, PCA has been shown to be highly sensitive to faulty measurements (outliers) because of its reliance on the outlier-sensitive L2-norm. Arguably, the most straightforward approach to impart robustness against outliers is to replace the outlier-sensitive L2-norm by the outlier-resistant L1-norm, thus formulating what is known as L1-PCA. Exact and approximate solvers are proposed for L1-PCA in the literature. On the other hand, in this big-data era, the data matrix may be very large and/or the data measurements may arrive in streaming fashion. Traditional L1-PCA algorithms are not suitable in this setting. In order to efficiently process streaming data, while being resistant against outliers, we propose a stochastic L1-PCA algorithm that computes the dominant principal component (PC) with formal convergence guarantees. We further generalize our stochastic L1-PCA algorithm to find multiple components by propose a new PCA framework that maximizes the recently proposed Barron loss. Leveraging Barron loss yields a stochastic algorithm with a tunable robustness parameter that allows the user to control the amount of outlier-resistance required in a given application. We demonstrate the efficacy and robustness of

our stochastic algorithms on synthetic and real-world datasets. Our experimental studies include online subspace estimation, classification, video surveillance, and image conditioning, among other things. Last, we focus on the development of asymptotic theory for Lp-PCA. In general, Lp-PCA for $p < 2$ has shown to outperform PCA in the presence of outliers owing to its outlier resistance. However, unlike PCA, Lp-PCA is perceived as a "robust heuristic" by the research community due to the lack of theoretical asymptotic convergence guarantees. In this work, we strive to shed light on the topic by developing asymptotic theory for Lp-PCA. Specifically, we show that, for a broad class of data distributions, the Lp-PCs span the same subspace as the standard PCs asymptotically and moreover, we prove that the Lp-PCs are specific rotated versions of the PCs. Finally, we demonstrate the asymptotic equivalence of PCA and Lp-PCA with a wide variety of experimental studies.

Acknowledgments

There are a number of people without whom this dissertation might not have been possible, and to whom I am forever indebted.

To my mother, Dr. Gnaneswari Gopal for being the backbone of my family, for being the main reason behind all my successes, and for her relentless support. Extra special thanks to you mother.

To my doctoral advisor, Dr. Panos P. Markopoulos for being an excellent mentor, a patient listener, and an extraordinary teacher. I have imbibed a great work ethic and professional personality from him. This work would not have be possible without his support and remarkable expertise. Many thanks to you, Professor Markopoulos.

To the members of my dissertation committee, Dr. Sohail Dianat, Dr. Eli Saber, Dr. Andreas Savakis, and Dr. Basca Jadamba, for their valuable feedback and advise, leading to significant improvement of this dissertation.

To Ms. Rebecca Ziebarth, Dr. Andres Kwasinski, and Dr. Edward Hensel for their constant support and invaluable advise in times of need.

To my late father Dhanaraj Guntoor Muniswamy, my late paternal aunt Amruthamma M, and my late maternal uncle Sumanth Kumar for paving a path towards a career in engineering and technology and their blessings.

Moreover, I thank each member of my family for being very supportive to me throughout my life. My sisters Mrunalini and Mrudula have supported me relentlessly over the years and enabled me to complete this work. In addition, I thank Mr. Mahesh Narayana for inspiring me to pursue a career in Machine Learning and offering constant encouragement through his words of wisdom.

Furthermore, I offer thanks to my ardent girlfriend, Prakruthi Manjunath for her persistent support and motivation throughout my PhD. Additionally, I thank our pet dog, Zazu Bean for being a constant companion and a great stress-buster.

I take this opportunity to further offer thanks to my best friends Akash M. Bushan and Prasidh P. Kumar for their continuous friendship and having my back at all times.

I would like to thank the Rochester Institute of Technology for offering me an opportunity and the facilities to pursue a PhD. I extend my sincere thanks to our funding agencies, including the U.S. National Science Foundation, the U.S. National Geospatial-Intelligence Agency, and the U.S. Air Force Research Lab for funding our research.

Finally, I would like to thank my colleagues at MILOS lab for their collaboration, valuable discussions, and exchange of ideas over the past 5 years. Specifically, I thank Dr. Dimitis Chachlakis, Manish Sharma, Masha Mozzafari, and Ian Tomeo for their collaboration and discussions throughout the years. Special thanks to Dr. Dimitris Chachlakis for his continued assistance and friendship.

Dedicated to my hardworking and loving mother, Dr. Gnaneswari Gopal.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This doctoral dissertation focuses on three directions of research, namely, dynamic algorithms for L1-norm data analysis, stochastic algorithms for outlier-resistant PCA, and novel asymptotic theory for Lp-norm principal component analysis (Lp-PCA). The latter two lines of research are our main research thrusts.

Since its dawn more than a century ago, **P**rincipal **C**omponent **A**nalysis (PCA) has been applied in a wide variety of areas including machine learning, signal processing, computer vision, and pattern recognition, to name a few [1]. Given a collection of data points, PCA strives to estimate the lower dimensional subspace wherein data presence is maximized. Some of PCA's advantages include its intuitive problem formulation, low-cost solution via the Singular Value Decomposition (SVD) [2], and its asymptotic optimality, among others. However, since PCA relies on the L2 (Frobenius) norm, which places squared emphasis on each datum including outliers (peripheral measurements that lie far from nominal data), it is sensitive to outliers and tends to provide inaccurate results in their presence. In this big-data era, real-world datasets are often outlier corrupted and warrant the use of outlier-resistant PCA solvers. L1-norm Principal Component Analysis (L1-PCA) is one such robust counterpart that has gained recent research traction owing to the availability of exact solvers [3] and multiple approximate algorithms [4–7]. L1-PCA straightforwardly replaces the squared emphasis placed by the L2-norm in PCA by linear emphasis using the L1-norm. L1-PCA has demonstrated similar performance compared to traditional PCA on nominal data, while being significantly better on outlier-corrupted data.

Big and/or streaming real-valued matrix data. Many modern datasets are extremely large and/or their constituent data measurements arrive in streaming fashion, possibly corrupted by

outliers. In some applications, the underlying subspace of the streaming data may change over time and needs to be tracked. Employing batch solvers in these settings and solving the problem from scratch each time a new measurement becomes available is ineffective, inefficient, and infeasible in some cases. In order to reliably process big and/or streaming outlier-corrupted data, while being able to track any subspace change, we focus on developing methods for incremental and adaptive L1-PCA.

Streaming complex-valued matrix data. Important applications in wireless communications and wireless sensor array processing rely on complex-valued measurements. PCA algorithms proposed to analyze real-valued data may yield degraded performance on complex-valued data. Algorithms specifically designed to handle complex-valued data are required to maintain and leverage the intrinsic complex-valued nature. As is the case with real-valued datasets, complex-valued data come with outliers and in order to process such data reliably, we need outlier-resistant algorithms. Additionally, some applications including direction-of-arrival (DoA) estimation encounter streaming complex-valued data. As discussed earlier, employing batch PCA solvers to handle streaming data is inefficient and may be infeasible in some cases. In order to process streaming complex-valued data reliably and efficiently, we propose an incremental L1-PCA algorithm for complex-valued data.

Streaming tensor-valued data. Advanced Internet of Things (IoTs) employ sensors that collect large volumes of measurements across diverse sensing modalities. Such data come with inherent multilinear (tensor) structures and correlations. In order to learn/discover meaningful underlying patterns from such data, we must maintain and leverage their tensor structure [8]. To this end, PCA has been generalized to handle tensor data [9, 10] and recently L1-norm based algorithms were proposed to reliably process tensor data corrupted with outliers [11–16]. Similar to data matrices, tensor data may arrive in streaming fashion in some applications. In order to process dynamic tensor data efficiently, we focus on the development of algorithms for dynamic L1-norm tensor analysis

Robust stochastic PCA. In order to process big data with low cost, multiple algorithms have been proposed, including the power method [17]. Although the power method is faster than SVD in practice, for massive N and/or D, the power method is restrictive. A different class of algorithms for incremental PCA offer gradient based methods that rely on single sample (stochastic) updates. The landmark stochastic PCA algorithm of Oja in [18] paved the way in this direction and owing to the recent surge of data availability, stochastic PCA algorithms have gained significant popularity. Despite the noteworthy efficiency of stochastic PCA algorithms, they are notoriously sensitive to outliers. In order to offer robustness against outliers while maintaining high efficiency, we propose

algorithms for outlier-resistant stochastic PCA.

Asymptotic theory for Lp-PCA. Owing to its outlier-resistance, Lp-PCA, for $p < 2$ has recently gained remarkable popularity in the research community. For instance, L1-PCA has attracted significant research interest over the past decade leading to the development of exact and multiple approximate solutions [3, 4, 6, 7]. Recent works have demonstrated the robustness of Lp-PCA against outliers on real-world datasets [19–22]. Although Lp-PCA outperforms PCA in the presence of outliers, its asymptotic characteristics are largely unknown and it is perceived as a ***robust heuristic***. While PCA tends to the dominant eigenvectors of the covariance matrix as the number of processed points increase, the asymptotic properties of Lp-PCA remain to date unknown. In this line of research, we aim at shedding light onto the asymptotic convergence guarantees of Lp-PCA. We focus on developing the theory for asymptotic Lp-PCA, assuming the data is drawn from a large family of well-known Elliptical distributions.

1.1 Contributions

Our contributions are organized in the following three chapters of this dissertation.

1. **Chapter 1: Dynamic L1-norm Data Analysis.** In this chapter, we present our work on dynamic algorithms for matrix data (real-valued and complex-valued) and tensor data based on the outlier-resistant L1-norm. This chapter is further divided into the following three sections:

   (a) *Incremental and adaptive L1-PCA.* In this line of research, we present an algorithmic framework for the incremental and adaptive implementation of outlier-resistant L1-norm PCA (L1-PCA) [4]. *We would like to note that the majority of this work was completed as part of the author's Master thesis [23–25] and finalized during the early stage of his doctoral studies.*

   (b) *Incremental algorithm for complex-valued L1-PCA.* We propose the first incremental algorithm for complex L1-PCA and employ it for online direction finding in this direction of research. Our algorithm is applied for online direction-of-arrival-estimation under the presence of jammers, demonstrating its capability to perform efficient direction estimation while identifying and rejecting misleading jamming signals.

   (c) *Dynamic Algorithms for L1-norm Tensor Analysis.* In this work, we present Dynamic L1-Tucker: a scalable method for incremental L1-norm based tensor analysis, with the

ability to (i) provide quality estimates of the sought-after multilinear bases, (ii) detect and reject outliers in an online fashion, and (iii) adapt to any nominal subspace changes.

2. **Chapter 2: Robust Stochastic Principal Component Analysis.** In this chapter, we propose stochastic algorithms for outlier-resistant PCA that can process streaming data with low cost. For the case of a single principal component (PC) estimation, we propose a modified stochastic L1-PCA algorithm based on mean absolute projection maximization and for multiple PCs, we propose a generalized framework for stochastic PCA based on the Barron loss [26]. Our generalized framework comes with a tunable robustness parameter α that can be handcrafted to achieve a trade-off between higher robustness and faster convergence. We show in theory that by tuning the robustness parameter α we can vary the step-size of the proposed algorithm, with particular values of α resulting in the coincidence of the proposed algorithm with that of Oja, while other lower values of α yielding robust counterparts including the robust stochastic PCA algorithm in [27]. We offer a variety of experimental studies on both synthetic and real-world data to demonstrate the efficacy of the proposed methods.

3. **Chapter 3: Asymptotic Theory for Lp-norm Principal Component Analysis.** This chapter focuses on the development of asymptotic theory for Lp-PCA, when the data follows a broad family of Elliptical distributions. Firstly, for the case of single PC, we prove that the asymptotic L1-PC coincides with the asymptotic standard PC, therefore to the dominant eigenvector or the data covariance matrix. Next, for the case of multiple PCs, we extend this theory and prove the asymptotic convergence of the Lp-PCA subspace to the L2-PCA subspace, and therefore to the dominant eigensubspace. Moreover, we show that although the dominant asymptotic Lp-PCs span the same subspace as that of the dominant eigenvectors of the covariance matrix of the distribution, the asymptotic Lp-PCs are specific rotated versions of the dominant eigenvectors. We offer an algorithm to derive the specific rotation matrix to obtain the asymptotic Lp-PCs, starting from the dominant eigenvectors. Finally, we leverage the proposed theory to initialize the iterative algorithms of Lp-PCA to obtain faster and better convergence on synthetic and multiple real-world datasets.

Chapter 2

Dynamic L1-norm Data Analysis

2.1 Introduction

Principal Component Analysis (PCA) is a cornerstone of signal processing, data analysis, and machine learning [28, 29]. Broadly, given a collection of data points, PCA seeks a small number of orthogonal directions (principal components) that define a subspace wherein data presence is maximized. Over the past decades, PCA has found numerous applications in wireless communications [30, 31], machine learning [32, 33], pattern recognition [1], video/image processing [34], and biomedical signal processing [35–37], among other fields.

Mathematically, PCA approximates data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$ by the low-rank matrix $\mathbf{Q}\mathbf{S}^\top \in \mathbb{R}^{D \times N}$, where $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}_K$ and $K < d = \text{rank}(\mathbf{X})$, so that the L2-norm of the approximation error is minimized. That is, PCA is formulated as [2]

$$(\mathbf{Q}_{L2}, \mathbf{S}_{L2}) = \underset{\substack{\mathbf{Q}\in\mathbb{R}^{D\times K},\ \mathbf{Q}^\top\mathbf{Q}=\mathbf{I}_K \\ \mathbf{S}\in\mathbb{R}^{N\times K}}}{\text{argmin}} \|\mathbf{X} - \mathbf{Q}\mathbf{S}^\top\|_F^2, \tag{2.1}$$

where the squared Frobenius norm (or L2-norm) $\|\cdot\|_F^2$ returns the sum of the squared entries of its matrix argument. Observing that, for any given $\mathbf{Q}$, $\mathbf{S} = \mathbf{X}^\top\mathbf{Q}$ minimizes the error in (2.1), $\mathbf{Q}_{L2}$ can be found as solution to the equivalent projection maximization problem

$$\mathbf{Q}_{L2} = \underset{\mathbf{Q}\in\mathbb{R}^{D\times K},\ \mathbf{Q}^\top\mathbf{Q}=\mathbf{I}_K}{\text{argmax}} \left\|\mathbf{Q}^\top\mathbf{X}\right\|_F^2. \tag{2.2}$$

Accordingly, $\mathbf{S}_{L2} = \mathbf{X}^\top \mathbf{Q}_{L2}$. Principal component matrix $\mathbf{Q}_{L2}$ consists of the K-dominant left singular-vectors of $\mathbf{X}$, obtained through standard Singular-Value Decomposition (SVD) [38], with cost $\mathcal{O}(NDmin(N,D))$.

In the *big data* era, real-world datasets often contain irregular or corrupted measurements that lie far from the nominal data subspace. Such measurements are commonly referred to as "outliers" [39] and may appear due to various causes, such as intermittent sensor malfunctions, errors in data transcription, transmission or labeling, deliberate jamming, or other sporadic changes in the sensing environment. Standard PCA is known to be very fragile in the presence of outliers, even if they appear as a small fraction of the processed data. The reason is that the L2-norm objective in (2.2), $\|\mathbf{Q}^\top \mathbf{X}\|_F^2 = \sum_{n=1}^{N} \|\mathbf{Q}^\top \mathbf{x}_n\|_2^2$, places squared emphasis on each and every datum, therefore benefiting unfavorably peripheral points. Therefore, for practical applications, there is a need for robust/outlier-resistant PCA alternatives [40–43].

2.1.1 Outlier-resistant PCA

To counteract the impact of outliers, researchers have focused on robust PCA and subspace estimation alternatives. The term "Robust PCA" (RPCA) commonly refers to methods that seek to rewrite the available data matrix as the summation of a low-rank component that describes the sought-after subspace and a sparse component that captures outliers [40, 42, 44–46]. A rather more straightforward approach substitutes the outlier-responsive L2-norm in (2.2) with the robust L1-norm, effectively removing the squared emphasis that PCA places on the magnitude of each datum [3–5]. This *L1-projection-maximization* PCA formulation, commonly referred to as *L1-PCA*, is mathematically defined as

$$\mathbf{Q}_{L1} = \underset{\mathbf{Q} \in \mathbb{R}^{D \times K},\ \mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_K}{\operatorname{argmax}} ||\mathbf{Q}^\top \mathbf{X}||_1, \tag{2.3}$$

where L1-norm $\|\cdot\|_1$ returns the sum of the absolute entries of its matrix argument. Alternatively, *L1-error-minimization* PCA replaces the L2-norm by the L1-norm in the error-minimization formulation of (2.1) [47–49]. Interestingly, contrary to what is true for standard PCA in (2.1)-(2.2), L1-projection-maximization PCA and L1-error-minimization PCA are not equivalent formulations. Rotation-invariant and optimally zero-centered variants of RPCA and L1-PCA have also been studied in [50–55]. In this work, we focus on L1-PCA in the form of (2.3).

In [5], Kwak proposed an early approximate solver for (2.3) with complexity $\mathcal{O}(N^2DK)$. The solver of [5] first approximates the dominant L1-PC ($K = 1$) of $\mathbf{X}$ and then computes the remaining $K-1$

L1-PCs through a sequence of deflating nullspace projections. In [6], Nie et al. approximated jointly all $K \geq 1$ L1-PCs of $\mathbf{X}$ with a "non-greedy" alternating-optimization algorithm of complexity $\mathcal{O}(N^2DK + NK^3)$. A semi-definite programming (SDP) approach for (2.3) was proposed in [7] with cost $\mathcal{O}(KN^{3.5}log(1/\epsilon) + KL(N^2 + DN))$, for desired accuracy ϵ. Authors in [56] presented a low-cost/high-performance L1-PCA/SVD hybrid model. More recently, [4] introduced *L1-BF*, a bit-flipping-based approximate solver for (2.3), with cost $\mathcal{O}(NDmin\{N, D\}+N^2(K^4+DK^2)+NDK^3)$. [4] showed that L1-BF attains very low (if any) performance degradation in the L1-PCA metric, often outperforming its counterparts. The exact solution to L1-PCA was delivered in [3], where authors reformulated (2.3) as an equivalent combinatorial optimization problem over NK $\{\pm 1\}$ variables. Moreover, [3] showed formally that, contrary to standard PCA, the K L1-PCs in (2.3) are to be computed jointly. The algorithms of [3] solve (2.3) exactly with complexity $\mathcal{O}(2^{NK})$, in general, or $\mathcal{O}(N^{dK-K+1})$ when $d = \text{rank}(\mathbf{X})$ is a constant with respect to N. A state-of-the-art algorithm for L1-PCA of complex-valued data was recently presented in [57].

L1-PCA has found a plethora of applications over the past few years. In [58], L1-PCA was used for outlier identification and elimination. In [59–66], It was used for robust image-fusion, face recognition, and dynamic video foreground/background extraction. In [67, 68], L1-PCA was used for DoA estimation. Authors in [69] proposed an L1-PCA-based nearest-subspace classifier for radar-based indoor motion recognition. L1-PCA-informed reduced-rank filtering for robust interference suppression was presented in [66]. A method for iterative re-weighted L1-PCA was recently presented in [70]. Our research on incremental and adaptive L1-PCA relies on the batch algorithms of L1-PCA and for the sake of simplicity, we offer a brief review of theory and algorithms for L1-PCA in Section 2.2.

2.1.2 Adaptive algorithms for L1-PCA of real-valued data

Modern day datasets are often extremely large and in some cases of interest, data measurements arrive in an online fashion. In addition such big and/or streaning data may be corrupted by outliers. Moreover in some applications, the underlying subspace of the streaming data may change over time and needs to be tracked. In such scenarios, utilizing traditional batch solvers to solve the problem from scratch each time a new measurement becomes available leads to severe computationally inefficiency. Therefore, in order to process big and/or streaming outlier-corrupted data efficiently and reliably, while being able to track any subspace change, we develop algorithms for incremental and adaptive L1-PCA in Section 2.3.

2.1.3 Incremental algorithm for L1-PCA of complex-valued data

Some applications including Direction of Arrival (DoA) estimation involve streaming complex-valued data, wherein the received snapshots may be corrupted by jamming signals, resulting in undesirable performance degradation if traditional PCA based techniques are employed [57]. In order to diminish the effect of jammers, L1-PCA has been used in place of PCA for DoA estimation [57, 68]. However, in an online direction finding scenario, the snapshots are received in streaming fashion and re-running L1-PCA algorithms from scratch each time a snapshot is received it computationally inefficient. In this line of research, we propose the first incremental algorithm for complex L1-PCA in section Section 2.4 and employ it for online direction finding.

2.1.4 Dynamic L1-norm algorithms for tensor-valued data

State-of-the-art machine learning and signal processing applications employ sensors that collect large volumes of measurements across diverse sensing modalities, with innate multilinear (tensor) structure. In order to learn/discover meaningful underlying patterns from such data, we must maintain and leverage their tensor structure [8]. To this end, standard PCA algorithms have been generalized to handle tensor data [10, 71]. However, these methods have been shown to be produce inaccurate results in the presence of outliers. In order to reliably process tensor data corrupted by outliers, L1-norm based tensor processing algorithms were proposed [11–16]. On the other hand, tensor data may arrive in streaming fashion in some applications and vanilla tensor analysis algorithms result in exorbitant costs if used in this setting. In order to process streaming tensor data efficiently, while offering sturdy outlier resistance, we focus on the development of algorithms for dynamic L1-norm tensor analysis in Section 2.5.

2.1.5 Contributions

Our contributions in this chapter are summarized as follows:

1. First, we propose efficient algorithmic solutions for both incremental and adaptive L1-PCA. Our first algorithm computes L1-PCA incrementally, processing one measurement at a time, with low computational and memory requirements; thus, this algorithm applies to big data and streaming data applications. Our second algorithm combines the merits of the first one with the additional ability to track changes in the nominal signal subspace.

2. Next, we present the first incremental method for complex L1-PCA and employ it for online direction finding, as snapshots naturally arrive in a streaming fashion.

3. Finally, we present Dynamic L1-Tucker: an algorithm for dynamic and outlier-resistant Tucker analysis of tensor data.

4. The superior robustness and the efficacy of our algorithms are corroborated by our experimental studies on synthetic data and multiple real-world datasets. We present experiments on online video background/foreground separation, glare/shadow artifact removal in facial images, and online direction-of-arrival estimation.

The rest of this chapter is organized as follows. In Section 2.2, we provide a brief overview of the theory and algorithms for L1-PCA. In Section 2.3, we present our research on incremental and adaptive L1-PCA. In Section 2.4, we propose the first incremental algorithm for complex-valued L1-PCA. Finally, in Section 2.5, we present dynamic algorithms for L1-Tucker.

2.2 L1-norm Principal-Component Analysis

Exact Solution. Authors in [3] showed that, if

$$\mathbf{B}_{\text{opt}} = \underset{\mathbf{B}\in\{\pm 1\}^{N\times K}}{\text{argmax}} \; ||\mathbf{X}\mathbf{B}||_*, \tag{2.4}$$

where nuclear norm $\|\cdot\|_*$ returns the sum of the singular values of its matrix argument, then L1-PCA in (2.3) is solved by

$$\mathbf{Q}_{L1} = \Phi(\mathbf{X}\mathbf{B}_{\text{opt}}) \tag{2.5}$$

where, for any tall matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ with SVD $\mathbf{A} \overset{\text{SVD}}{=} \mathbf{U}\mathbf{\Sigma}_{n\times n}\mathbf{V}^\top$, $\Phi(\mathbf{A}) = \mathbf{U}\mathbf{V}^\top$. In addition, [3] showed that $||\mathbf{X}^\top\mathbf{Q}_{L1}||_1 = ||\mathbf{X}\mathbf{B}_{\text{opt}}||_*$ and

$$\mathbf{B}_{\text{opt}} = \text{sgn}\left(\mathbf{X}^\top\mathbf{Q}_{L1}\right). \tag{2.6}$$

Therefore L1-PCA in (2.3) can be cast as an equivalent combinatorial optimization problem over antipodal binary variables ("bits") in $\{\pm 1\}$. The first optimal algorithm in [3] performs exhaustive search over the entire feasibility set of (2.4), $\{\pm 1\}^{N\times K}$, to obtain a solution $\mathbf{B}_{\text{opt}}$ with exponential complexity $\mathcal{O}(2^{NK})$. The second optimal algorithm in [3] first constructs a polynomial-size subset

of $\{\pm 1\}^{N \times K}$, wherein a solution to (2.4) is proven to exist; then it searches exhaustively within this set to obtain $\mathbf{B}_{\text{opt}}$ and, after the SVD step in (2.5), returns the solution to L1-PCA, with overall cost $\mathcal{O}(N^{dK-K+1})$, polynomial in N.

Efficient L1-PCA Through Bit Flipping (L1-BF). L1-BF is a state-of-the-art efficient algorithm for L1-PCA, based on optimal single-bit-flipping iterations [4]. L1-BF has similar cost to standard PCA (i.e., SVD), exhibits sturdy resistance against outliers, and has been shown to outperform most of its counterparts in the L1-PCA metric. In the sequel, we offer a brief review of L1-BF, since it is a core component of the incremental and adaptive L1-PCA calculators proposed in this work.

L1-BF initiates at bit matrix $\mathbf{B}(0) \in \{\pm 1\}^{N \times K}$ (arbitrary or better –see [4]) and executes a sequence of optimal single-bit-flipping iterations, across which the metric in (2.4) monotonically increases. Specifically, at each iteration, L1-BF examines all NK bits and recognizes the single one that, when flipped, offers the highest increase to the metric of (2.4). That is, at the t-th iteration, L1-BF finds

$$(n,k) = \underset{\substack{(m,l) \in \{1,2,\ldots,N\} \\ \times \{1,2,\ldots,K\}}}{\operatorname{argmax}} \left\| \mathbf{X}\mathbf{B}(t) - 2\left[\mathbf{B}(t)\right]_{m,l} \mathbf{x}_m \mathbf{e}_{l,K}^\top \right\|_*, \tag{2.7}$$

where $\mathbf{e}_{l,K}$ denotes the l-th column of the size-K identity matrix $\mathbf{I}_K$ and $\mathbf{x}_m$ is the m-th column of data matrix $\mathbf{X}$. Thereafter, L1-BF flips the (n,k)-th bit of $\mathbf{B}(t)$ setting

$$\mathbf{B}(t+1) = \mathbf{B}(t) - 2[\mathbf{B}(t)]_{n,k} \mathbf{e}_{n,N} \mathbf{e}_{k,K}^\top. \tag{2.8}$$

Bit flipping terminates at iteration t if the nuclear norm in (2.4) cannot further increase by any single-bit flip. Upon termination, L1-BF returns $\hat{\mathbf{B}} = \mathbf{B}(t)$ as an approximation to $\mathbf{B}_{\text{opt}}$ in (2.4) and $\hat{\mathbf{Q}} = \Phi(\mathbf{X}\mathbf{B}(t))$ as an approximation to $\mathbf{Q}_{L1}$ in (2.3), in accordance with (2.5). It was shown in [4] that the bit flipping iterations converge, since the metric of (2.4) (i) is upper bounded by its exact solution and (ii) increases monotonically across the iterations. Henceforth, for brevity in notation, we summarize the L1-BF procedure as $(\hat{\mathbf{Q}}, \hat{\mathbf{B}}) = \text{L1BF}(\mathbf{X}; \mathbf{B}(0); K)$. A pseudo-code for L1-BF is provided in Figure 2.1 and the code is available in [72]. The computational complexity of L1-BF is $\mathcal{O}(NDmin\{N,D\} + N^2(K^4 + DK^2) + NDK^3)$.

Algorithm L1-BF

Input: $\mathbf{X} \in \mathbb{R}^{D \times N}$, init. $\mathbf{B} \in \{\pm 1\}^{N \times K}$, $K \leq \text{rank}(\mathbf{X})$,

1: $\mathbf{B} \leftarrow \texttt{BF}(\mathbf{X}, \mathbf{B}, K)$
2: $(\mathbf{U}, \mathbf{\Sigma}_{K \times K}, \mathbf{V}) \leftarrow \texttt{SVD}(\mathbf{XB})$
3: $\mathbf{Q} \leftarrow \mathbf{U}\mathbf{V}^{\top}$

Output: $(\hat{\mathbf{B}}, \hat{\mathbf{Q}}) \leftarrow (\mathbf{B}, \mathbf{Q})$

Function: $\mathbf{B} \leftarrow \texttt{BF}(\mathbf{X}_{D \times N}, \mathbf{B}, K)$

1: $\omega \leftarrow K||\mathbf{X}[\mathbf{B}]_{:,1}||_2$
2: while true (or terminate at NK iterations)
3: for $m \in \{1, 2, \ldots, N\}$, $l \in \{1, 2, \ldots, K\}$
4: $a_{m,l} \leftarrow ||\mathbf{XB} - 2[\mathbf{B}]_{m,l}\mathbf{x}_m\mathbf{e}_{l,K}^{\top}||_*$
5: $(n, k) \leftarrow \text{argmax}_{m,l}\ a_{m,l}$
6: if $\omega < a_{n,k}$
7: $[\mathbf{B}]_{n,k} \leftarrow -[\mathbf{B}]_{n,k}$, $\omega \leftarrow a_{n,k}$
8: else, break
9: Return $\mathbf{B}$

Figure 2.1. Pseudocode of L1-BF.

2.3 Contribution 1: Incremental and Adaptive L1-PCA

Modern datasets often comprise a very large number of data points, N, of high dimensionality, D. In such cases, jointly analyzing all measurements in $\mathbf{X} \in \mathbb{R}^{D \times N}$ ("batch" PCA) may be of prohibitively high computational cost. Moreover, in streaming data applications, data points are initially unavailable and arrive sequentially (e.g., in video processing [73–78] and dynamic face-ID [79]). Clearly, appending new points to the previously collected data matrix and recalculating batch PCA anew may lead to unsustainably high computational and storage complexity. In view of the above, big and/or streaming data processing can be significantly benefited by online/incremental alternatives of PCA.

Finally, in many applications, the data subspace of interest shifts during the data collection process. Oftentimes, recalculating anew the shifted subspace is an inefficient approach. For such applications, experts prefer instead to start from the already calculated subspace and *adapt* to the new one –an approach also known as subspace tracking [41, 80].

In this work, we present an algorithmic framework for the incremental and adaptive implementation of outlier-resistant L1-norm PCA (L1-PCA) [3, 56]. The first proposed algorithm (L1-IPCA) computes L1-PCA incrementally, processing one measurement at a time, with low computational and memory requirements. In contrast to previous approaches in the literature, this algorithm evaluates each new incoming data point by means of an L1-PCA-informed reliability criterion and discards points that appear to be outliers. Our second algorithm for adaptive L1-PCA (L1-APCA)

has the additional capability of tracking changes to the nominal signal subspace. The proposed algorithms are evaluated in an array of experimental studies on subspace estimation/tracking, video surveillance (foreground/background separation), image conditioning, and direction-of-arrival (DoA) estimation/tracking.

2.3.1 Related Works

For big and streaming data applications, researchers have long focused on incremental PCA solutions. Similar to batch PCA, incremental/adaptive PCA calculators perform well on clean data, but experience significant performance degradation when they encounter outliers. This observation has prompted extensive research on corruption-resistant incremental and adaptive PCA [80–96]. Thorough reviews of robust incremental and adaptive algorithms for PCA are offered in [41, 78, 82]; in the sequel, we list few of the many notable works in the area. [92] presents an incremental RPCA implementation based on projected gradient descent. Online RPCA (OR-PCA) in [84] uses a stochastic model for accepting or rejecting new data points. [91] presents an online version of Robust Subspace Learning (RSL) in [46], which detects outliers and replaces them by neighboring nominal points. A kernel-based method for robust and fast incremental PCA was proposed in [96]. Grassmannian Robust Adaptive Subspace Tracking Algorithm (GRASTA) [93] operates on possibly sparsely sampled data matrices and tries to track underlying subspace changes, while staying robust against corruptions. The works in [63–65] offer the first incremental L1-PCA algorithms in the literature, proposed for compressed-sensed domain video surveillance and visual object tracking.

2.3.2 Proposed Algorithms for Incremental and Adaptive L1-PCA

Incremental L1-PCA

The first proposed algorithm (L1-IPCA) calculates incrementally the K L1-PCs of data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$ as its columns arrive in a streaming fashion. In the case that all columns of $\mathbf{X}$ are initially available, the proposed algorithm processes them one-by-one for complexity savings.

To initialize, we first collect a small batch of n data points from $\mathbf{X}$ in $\mathbf{Y}_0 = [\mathbf{X}]_{:,1:n} \in \mathbb{R}^{D \times n}$, with $\text{rank}(\mathbf{Y}_0) \geq K$. Then, we run L1-BF iterations on $\mathbf{Y}_0$, with some initialization $\mathbf{B} \in \{\pm 1\}^{n \times K}$, to obtain the first approximate L1-PCA solution $(\hat{\mathbf{Q}}_0, \hat{\mathbf{B}}_0) = \text{L1BF}(\mathbf{Y}_0; \mathbf{B}; K)$.

When a new data point $\mathbf{x}_i^{(\text{in})} = [\mathbf{X}]_{:,n+i}$ arrives, for $i = 1, 2, \ldots, N-n$, we first pass it through an L1-PCA-informed reliability check. Specifically, the *L1-reliability* of $\mathbf{x}_i^{(\text{in})}$ is defined as its angular proximity to the previously calculated L1-PCs $\hat{\mathbf{Q}}_{i-1}$,

$$r\left(\mathbf{x}_i^{(\text{in})};\ \hat{\mathbf{Q}}_{i-1}\right) = \frac{\left\|\hat{\mathbf{Q}}_{i-1}^\top \mathbf{x}_i^{(\text{in})}\right\|_2^2}{\left\|\mathbf{x}_i^{(\text{in})}\right\|_2^2}. \tag{2.9}$$

Based on the outlier resistance of L1-PCA, (2.9) constitutes a local measure for determining weather $\mathbf{x}_i^{(\text{in})}$ is nominal or outlying/corrupted.

If $r(\mathbf{x}_i^{(\text{in})};\ \hat{\mathbf{Q}}_{i-1}) \leq \tau$, for some predetermined reliability threshold $\tau \in [0, 1)$ (practically τ is set close to 1), then $\mathbf{x}_i^{(\text{in})}$ is disregarded as a probable outlier and we maintain the previous L1-PCA solution $(\hat{\mathbf{Q}}_i, \hat{\mathbf{B}}_i) = (\hat{\mathbf{Q}}_{i-1}, \hat{\mathbf{B}}_{i-1})$ and the previous memory matrix $\mathbf{Y}_i = \mathbf{Y}_{i-1}$. If, on the other hand, $r(\mathbf{x}_i^{(\text{in})};\ \hat{\mathbf{Q}}_{i-1}) > \tau$, then $\mathbf{x}_i^{(\text{in})}$ passes successfully the L1-reliability check and it is admitted for processing; the i-th L1-PCA update $(\hat{\mathbf{Q}}_i, \hat{\mathbf{B}}_i)$ is computed as follows. First, $\mathbf{x}_i^{(\text{in})}$ is appended to $\mathbf{Y}_{i-1}$, forming the augmented memory matrix

$$\tilde{\mathbf{Y}}_{i-1} = \left[\mathbf{Y}_{i-1},\ \mathbf{x}_i^{(\text{in})}\right] \in \mathbb{R}^{D \times (n+1)}. \tag{2.10}$$

Then, motivated by the optimality condition in (2.6), we use

$$\tilde{\mathbf{B}}_{i-1} = \text{sgn}\left(\tilde{\mathbf{Y}}_{i-1}^\top \hat{\mathbf{Q}}_{i-1}\right) \in \{\pm 1\}^{(n+1) \times K} \tag{2.11}$$

to initialize L1-BF iterations on $\tilde{\mathbf{Y}}_{i-1}$; at the end of these iterations, we obtain

$$\left(\hat{\mathbf{Q}}_i,\ \hat{\mathbf{B}}_i\right) = \text{L1BF}\left(\tilde{\mathbf{Y}}_{i-1}; \hat{\mathbf{B}}_{i-1}; K\right). \tag{2.12}$$

We notice that the number of data points stored in memory increased from n in $\mathbf{Y}_{i-1}$ to $n+1$ in $\tilde{\mathbf{Y}}_{i-1}$. In order to maintain limited storage and computational cost, we proceed with discarding one of the points in $\tilde{\mathbf{Y}}_{i-1}$. Specifically, similar to [63], we discard the point with the lowest L1-reliability, as defined in (2.9). Formally, we identify

$$j_{\min} = \underset{j=1,2,\ldots,n+1}{\text{argmin}}\ r\left(\left[\tilde{\mathbf{Y}}_{i-1}\right]_{:,j}; \hat{\mathbf{Q}}_i\right) \tag{2.13}$$

Proposed Algorithm L1-IPCA

Input: $\mathbf{Y}$, K, τ

$\mathbf{B} \leftarrow$ arbitrary
$(\mathbf{B}, \mathbf{Q}) \leftarrow \texttt{L1BF}(\mathbf{Y}, \mathbf{B}, K)$
When $\mathbf{x}$ arrives,
  $(\mathbf{B}, \mathbf{Q}, \mathbf{Y}) \leftarrow \texttt{uL1BF}(\mathbf{Y}, \mathbf{x}, \mathbf{Q}, K, \tau)$

Output: $(\hat{\mathbf{B}}, \hat{\mathbf{Q}}) \leftarrow (\mathbf{B}, \mathbf{Q})$

Function: $(\mathbf{B}, \mathbf{Q}, \mathbf{Y}, \tau) \leftarrow \texttt{uL1BF}(\mathbf{Y}, \mathbf{x}, \mathbf{Q}, K, \tau)$

$r \leftarrow \texttt{L1rel}(\mathbf{x}, \mathbf{Q})$
if $r > \tau$
  $\tilde{\mathbf{Y}} \leftarrow [\mathbf{Y}, \mathbf{x}]$
  $\mathbf{B} \leftarrow \text{sgn}(\tilde{\mathbf{Y}}^\top \mathbf{Q})$
  $(\mathbf{B}, \mathbf{Q}) \leftarrow \texttt{L1BF}\ (\tilde{\mathbf{Y}}, \mathbf{B}, K)$
  $r_j \leftarrow \texttt{L1rel}([\tilde{\mathbf{Y}}]_{:,j}, \mathbf{Q})$, $j = 1, \ldots, n+1$
  $j_{\min} \leftarrow \text{argmin}_{j=1,\ldots,n+1}\, r_j$
  $\mathbf{Y} \leftarrow [\tilde{\mathbf{Y}}]_{:,\{1,\ldots,n+1\} \setminus j_{\min}}$
Return $\mathbf{B}$, $\mathbf{Q}$, $\mathbf{Y}$

Function: $r \leftarrow \texttt{L1rel}\ (\mathbf{x}, \mathbf{Q})$

$r \leftarrow \frac{||\mathbf{Q}^\top \mathbf{x}||_2^2}{||\mathbf{x}||_2^2}$
Return r

Figure 2.2. Pseudocode of proposed L1-IPCA algorithm.

and discard the $(j_{\min})$-th column of $\tilde{\mathbf{Y}}_{i-1}$, setting

$$\mathbf{Y}_i = \left[\tilde{\mathbf{Y}}_{i-1}\right]_{:,\{1,\ldots,n+1\} \setminus j_{\min}} \in \mathbb{R}^{D \times n}. \tag{2.14}$$

In view of the above, the proposed algorithm has two lines of defense against outliers. First, the reliability of an incoming point is evaluated by means of the previously computed L1-PCs, thus protecting the incremental L1-PCA procedure against processing outliers. Second, any point that passes the reliability check, is processed by the outlier-resistant L1-BF procedure. A detailed description of L1-IPCA is provided in Figure 2.2. A complexity analysis follows.

According to [4], L1-BF returns the K (approximate) L1-PCs of $\tilde{\mathbf{Y}}_i \in \mathbb{R}^{D \times n}$ with cost $O(nD \min\{n, D\} + n^2 K^2 (K^2 + \min\{n, D\}))$, for any i. Since i takes values $1, 2, \ldots, N-n$, the total cost of L1-IPCA is $O(NnD \min\{n, D\} + Nn^2 K^2 (K^2 + \min\{n, D\}))$ –i.e., linear in N. If $n > D$, the cost is simplified to $ONn^2 K^2 (K^2 + D))$; if $D \geq n$, the cost is simplified to $O(Nn^2 (K^4 + K^2 n + D))$.

At this point, it is worth noting that the pioneering work in [64] also proposed L1-BF updates for incremental L1-PCA in compressed-sensed-domain video background tracking. L1-IPCA differs from L1-PC-updating method of [64] in the following ways. First, L1-IPCA updates the L1-PCs

using only new points that pass successfully the L1-reliability check; on the other hand, [64] does not apply a reliability check and processes every incoming point. Thus, L1-IPCA has an additional line of defense against outliers. Second, instead of (2.11), the algorithm of [64] sets the L1-BF-initialization matrix to $\tilde{\mathbf{B}}_i = \left[\hat{\mathbf{B}}_{i-1}^\top, \mathbf{b}_{\text{exact}}\right]^\top$, where

$$\mathbf{b}_{\text{exact}} = \underset{\mathbf{b}\in\{\pm 1\}^K}{\text{argmax}} \left\| \tilde{\mathbf{Y}}_{i-1} \left[\hat{\mathbf{B}}_{i-1}^\top, \mathbf{b}\right]^\top \right\|_*. \tag{2.15}$$

To identify $\mathbf{b}_{\text{exact}}$, [64] evaluates each of the 2^{K-1} candidate solutions of (2.15), with cost exponential in K. The proposed simpler initialization in (2.11) costs only $\mathcal{O}(KnD)$. Finally, there is a difference in the way L1-IPCA and the method of [64] drop points from the memory matrix $\tilde{\mathbf{Y}}_{i-1}$ to define $\mathbf{Y}_i$.

Adaptive L1-PCA

L1-IPCA presented above considers, implicitly, that the sought-after subspace is constant across all processed data. In many applications however it is desired to track a dynamic signal subspace that changes across the collected data points. In this section, we propose an algorithm for adaptive L1-PCA (L1-APCA) that derives by two main modifications of L1-IPCA.

Modification 1: Adjustable reliability threshold. When the nominal signal subspace changes significantly, new incoming points may fail the L1-reliability check and be discarded. Assuming that outliers appear rather sporadically, when multiple incoming points fail the reliability check one after the other, we have a strong indication that the signal subspace has changed. Thus, in L1-APCA we consider adjustable reliability threshold that decreases every time that an incoming point fails the reliability check and resets to its original high value whenever an incoming point is admitted for processing. Specifically, let threshold τ_i denote the reliability threshold by which the i-th incoming point $\mathbf{x}_i^{(\text{in})}$ is evaluated, with initialization $\tau_1 = \tau_{\max}$. If $r\left(\mathbf{x}_i^{(\text{in})};\ \hat{\mathbf{Q}}_{i-1}\right) < \tau_i$ and $\mathbf{x}_i^{(\text{in})}$ fails the reliability check, then we reduce $\tau_{i+1} = \tau_i \rho$, for some predetermined decrease ratio ρ in $(0, 1]$. Clearly, $\rho = 1$ corresponds to fixed threshold, as used in L1-IPCA. If $\mathbf{x}_i^{(\text{in})}$ passes the reliability check, then τ_{i+1} is reset to $\tau_{\max}$.

Modification 2: Preservation of recent measurements. Consider a change of the nominal signal subspace and assume that $\mathbf{x}_i^{(\text{in})}$ is the first point from the new subspace that passes the reliability check (possibly after threshold reduction.) $\mathbf{x}_i^{(\text{in})}$ will be inserted to $\tilde{\mathbf{Y}}_{i-1}$ and participate to the computation of $\hat{\mathbf{Q}}_i$. However, since n out of the $n+1$ points in $\tilde{\mathbf{Y}}_{i-1}$ are drawn from the

Proposed Algorithm L1-APCA

Input: $\mathbf{Y}$, K, q, $\tau_{\max}$, ρ

$\mathbf{B} \leftarrow$ arbitrary, $\tau \leftarrow \tau_{\max}$
$(\mathbf{B}, \mathbf{Q}) \leftarrow \texttt{L1BF}(\mathbf{Y}, \mathbf{B}, K)$
When $\mathbf{x}$ arrives,
  $(\mathbf{B}, \mathbf{Q}, \mathbf{Y}, \tau) \leftarrow \texttt{aL1BF}(\mathbf{Y}, \mathbf{x}, \mathbf{Q}, K, q, \tau_{\max}, \tau, \rho)$

Output: $(\hat{\mathbf{B}}, \hat{\mathbf{Q}}) \leftarrow (\mathbf{B}, \mathbf{Q})$

Function:
$(\mathbf{B}, \mathbf{Q}, \mathbf{Y}, \tau) \leftarrow \texttt{aL1BF}(\mathbf{Y}, \mathbf{x}, \mathbf{Q}, K, q, \tau_{\max}, \tau, \rho)$

$r \leftarrow \texttt{L1rel}(\mathbf{x}, \mathbf{Q})$
if $r > \tau$
  $\tilde{\mathbf{Y}} \leftarrow [\mathbf{Y}, \mathbf{x}]$, $\tau \leftarrow \tau_{\max}$
  $\mathbf{B} \leftarrow \text{sgn}(\tilde{\mathbf{Y}}^\top \mathbf{Q})$
  $(\mathbf{B}, \mathbf{Q}) \leftarrow \texttt{L1BF}\ (\tilde{\mathbf{Y}}, \mathbf{B}, K)$
  $r_j \leftarrow \texttt{L1rel}([\tilde{\mathbf{Y}}]_{:,j}, \mathbf{Q})$, $j = 1, \ldots, n+1$
  $j_{\min} \leftarrow \text{argmin}_{j=1,\ldots,n+1-q}\, r_j$
  $\mathbf{Y} \leftarrow [\tilde{\mathbf{Y}}]_{:,\{1,\ldots,n+1\}\setminus j_{\min}}$
else
  $\tau \leftarrow \tau\rho$
Return $\mathbf{B},\ \mathbf{Q},\ \mathbf{Y}, \tau$

Function: $r \leftarrow \texttt{L1rel}\ (\mathbf{x}, \mathbf{Q})$

$r \leftarrow \frac{||\mathbf{Q}^\top \mathbf{x}||_2^2}{||\mathbf{x}||_2^2}$
Return r

Figure 2.3. Pseudocode of proposed L1-APCA algorithm.

previous signal subspace, $\hat{\mathbf{Q}}_i$ will probably remain similar to $\hat{\mathbf{Q}}_{i-1}$. Thus, when the L1-reliability of the points in $\tilde{\mathbf{Y}}_{i-1}$ is evaluated by means of $\hat{\mathbf{Q}}_i$, $\mathbf{x}_i^{(\text{in})}$ may be found to be the least reliable and as such be discarded from $\tilde{\mathbf{Y}}_{i-1}$. Certainly, such an event would inhibit the subspace tracking process. Therefore, in L1-APCA, we modify L1-IPCA so that the $q < n$ most recently collected points $\tilde{\mathbf{Y}}_{i-1}$ cannot be discarded. That is, the i-th memory matrix is defined as $\mathbf{Y}_i = [\tilde{\mathbf{Y}}_{i-1}]_{:,\{1,\ldots,n+1\}\setminus j_{\min}} \in \mathbb{R}^{D \times n}$ where

$$j_{\min} = \underset{j=1,2,\ldots,n+1-q}{\text{argmin}}\ r\left(\left[\tilde{\mathbf{Y}}_{i-1}\right]_{:,j}; \hat{\mathbf{Q}}_i\right). \tag{2.16}$$

A detailed description of L1-APCA is offered in the pseudocode of Figure 2.3.

2.3.3 Experimental Studies

Subspace tracking with L1-APCA on synthetic data. We consider data matrix $\mathbf{X} \in \mathbb{R}^{5 \times 200}$, the first 90 columns of which are drawn from $\mathcal{N}(\mathbf{0}_5, \alpha \mathbf{z}\mathbf{z}^\top)$, where $\|\mathbf{z}\|_2 = 1$ and $\alpha = 100$. The last

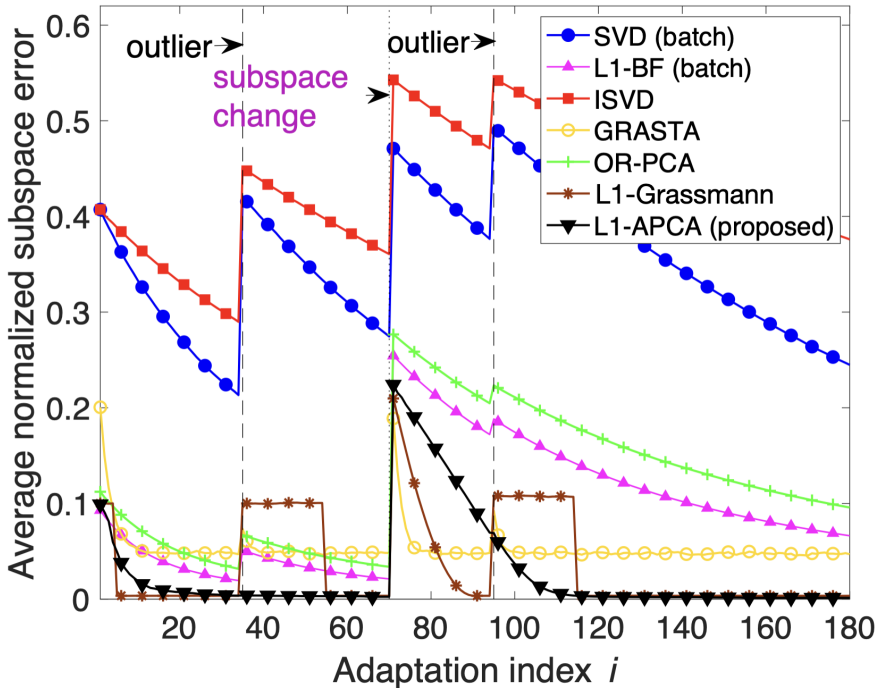


Figure 2.4. Subspace tracking experiment. Average subspace error versus adaptation index i. $D = 5$, $N = 200$, $K = 1$. Subspace change after 90 points. $\alpha = 100$, $\beta = 4000$, $\mathbf{z}^\top \mathbf{z}' = 0.88$, $\mathbf{p}^\top \mathbf{z} = 0.31$, $\mathbf{p}^\top \mathbf{z}' = 0.2$. $n = 20$, $q = 0.8n$, $\tau = 0.9$, $\rho = 0.5$.Outliers in $[\mathbf{X}]_{:,5}$, $[\mathbf{X}]_{:,55}$, and $[\mathbf{X}]_{:,115}$.

110 columns of $\mathbf{X}$ are drawn from $\mathcal{N}(\mathbf{0}_5, \alpha \mathbf{z}' \mathbf{z}'^\top)$, where $\|\mathbf{z}'\|_2 = 1$ and $\mathbf{z}'^\top \mathbf{z} = 0.88$. All entries of $\mathbf{X}$ are corrupted by AWGN from $\mathcal{N}(0, 1)$. Columns 5, 55, and 115 are once again corrupted additively by outliers drawn from $\mathcal{N}(\mathbf{0}_5, \beta \mathbf{p}\mathbf{p}^\top)$, where $\|\mathbf{p}\|_2 = 1$, $\mathbf{p}^\top \mathbf{z} = 0.31$, $\mathbf{p}^\top \mathbf{z}' = 0.2$ and $\beta = 4000$. We set L1-APCA parameters $n = 20$, $\tau = 0.9$, $\rho = 0.5$, and $q = 0.8n = 16$.

In Figure 2.4, we plot the normalized subspace error, calculated as $e_i = \frac{1}{2}\|\hat{\mathbf{q}}_i \hat{\mathbf{q}}_i^\top - \mathbf{z}\mathbf{z}^\top\|^2$ for $i \leq 90$ and $e_i = \frac{1}{2}\|\hat{\mathbf{q}}_i \hat{\mathbf{q}}_i^\top - \mathbf{z}'\mathbf{z}'^\top\|^2$ for $i > 90$. Together with L1-APCA, we plot again the performance of batch SVD, batch L1-BF [4], ISVD [97], GRASTA [93], OR-PCA [85], and the method of [65], which we refer to as L1-Grassmann. We notice that the L2-norm based methods, SVD and ISVD, deviate from the nominal subspace when they process outliers and display slower response to subspace changes. GRASTA, OR-PCA, L1-Grassmann [65], and batch L1-BF exhibit resistance against outliers and respond quickly to nominal-subspace changes. L1-APCA outperforms all counterparts in terms of outlier resistance and attains the lowest error upon convergence. We observe that GRASTA adapts slightly faster than L1-APCA to the subspace change, converging though to higher subspace error. L1-Grassmann also adapts slightly faster than L1-APCA to the subspace change, however it is considerably more susceptible to outliers.

Background/Foreground Separation in Video Sequences.

For this experiment, we use a video recorded at a shopping center in Portugal, available in the

(a)

(b) (c) (d) (e) (f) (g) (h)

(i) (j) (k) (l) (m) (n) (o)

Figure 2.5. Video processing experiment. (a) Original frame. Background extracted by (b) ISVD, (c) GRASTA , (d) PCP, (e) OR-PCA, (f) RPCA, (g) ReProCS, and (h) L1-IPCA (proposed). Foreground extracted by (i) ISVD, (j) GRASTA, (k) PCP, (l) OR-PCA, (m) RPCA, (n) ReProCS, and (o) L1-IPCA (proposed).

standard CAVIAR database [98]. The video consists of $N = 438$ frames of size 202×269 pixels. Video processing is carried out as follows. Each video frame is vectorized and arranged as a column of data matrix $\mathbf{X} \in \mathbb{R}^{54338 \times 438}$. We set $n = 20$ and $\tau = 0.98$ and apply L1-IPCA to compute the $K = 5$ L1-PCs of the video sequence $\hat{\mathbf{Q}} \in \mathbb{R}^{54338 \times 5}$. The background of the i-th frame $\mathbf{x}_i = [\mathbf{X}]_{:,i}$ is obtained by projecting it onto the computed K L1-PCs as $\mathbf{x}_i^{(\text{back})} = \hat{\mathbf{Q}}\hat{\mathbf{Q}}^\top \mathbf{x}_i$; then, the foreground component is obtained as $\mathbf{x}_i^{(\text{fore})} = \mathbf{x}_i - \mathbf{x}_i^{(\text{back})}$.

In Figure 2.5, we present the 200-th frame of the processed video sequence, together with its background as extracted by (b) ISVD, (c) GRASTA, (d) PCP, (e) OR-PCA, (f) RPCA [45], (g) ReProCS [83], and (h) L1-IPCA (proposed). In addition, we show the foreground of the frame, as extracted by (i) ISVD, (j) GRASTA, (k) PCP, (l) OR-PCA, (m) RPCA, (n) ReProCS, and (o) L1-IPCA (proposed). We observe that the background computed by ISVD exhibits a non-negligible "ghostly" appearance of the walking man, whose blurred/inaccurate figure also appears in the foreground. GRASTA, PCP, and RPCA exhibit similar performance, with a hazy appearance of the man in the background; OR-PCA performs slightly better than the previous methods. The proposed L1-IPCA, together with ReProCS, demonstrate similarly high performance, returning clean background and a well defined outline of the man and his shadow in the foreground.

2.4 Contribution 2: Incremental Complex L1-PCA for Direction-of-Arrival Estimation

Direction-of-Arrival (DoA) estimation is a fundamental component of source localization in many modern radar, sonar, and communications applications. For example, localizing interference/jamming permits its successful mitigation [99, 100], ensuring safety in critical navigation applications such as Global Navigation Satellite Systems (GNSS). In communications, direction finding enables location-specific delivery of information, even in areas where Global Positioning System (GPS) is ineffective [101]. In addition, with direction finding, automotive radars provide environment awareness, enabling autonomous cruise control and collision avoidance [102].

DoA estimation is commonly based on likelihood maximization [103], spectral estimation [104], compressive sensing [105, 106], and subspace-estimation [107]. In standard subspace-based direction finding, e.g., by means of the multiple-signal classification (MUSIC) algorithm [108], the receiver identifies the principal components (PCs) of a number of recorded snapshots, through their singular-value decomposition (SVD); the span of these components constitutes an estimate of the array-response subspace that corresponds to the source DoAs. In nominal system operation, when snapshots are only corrupted by additive white Gaussian noise (AWGN), such methods are known to offer asymptotically consistent DoA estimates, with high resolution [109].

However, even when a small part of the snapshots is corrupted by intense directional jamming, principal-component analysis (PCA) and thereon-based MUSIC DoA estimation can be highly misled. This corruption sensitivity of PCA can be attributed to its L2-norm-based formulation which benefits points that lie far from the signal subspace. To counteract the impact of jamming, PCA in the MUSIC procedure was recently replaced by the corruption-resistant L1-PCA [67, 68]. While PCA searches for the linear subspace that maximizes the L2-norm of the projected snapshots, L1-PCA seeks to maximize their L1-norm; thus, L1-PCA places reduced emphasis to jamming-corrupted, outlying snapshots. Incremental algorithms for L1-PCA of real-valued data were recently proposed in [24, 25, 64, 65]. Exact and approximate algorithms for L1-PCA of real-valued data were proposed in [3, 4]. Extensions of L1-PCA for tensor processing were also recently proposed in [11, 13, 14]. Authors in [57] presented the first algorithms for L1-PCA of complex-valued data. Other recent applications of L1-PCA include image reconstruction [59], video foreground/background analysis [63], radar-based motion recognition [69], visual object tracking [65], and reduced-rank filtering [66].

In DoA estimation, the snapshots arrive in a streaming fashion and some of them may be jammer

corrupted. Processing the entire data batch from scratch each time that a new snapshot is inserted to it is highly inefficient, in terms of both storage and computational cost. This motivates the development of incremental DoA estimators. In this work, we propose the first algorithm for incremental L1-PCA of complex data and, then, we apply this algorithm for online DoA estimation. Our experimental studies illustrate the computational efficiency and jamming resistance of the proposed method.

2.4.1 System Model and Problem Statement

We consider uniform linear antenna array (ULA) with D elements at positions $\{0, 1, \ldots, D-1\}d$, where d is the spacing reference unit (e.g., equal to half the wavelength). $K < D$ sources of interest, at the far-field, transmit signals with carrier frequency f_c and propagation speed c. The signals impinge on the array from distinct directions $\theta_1, \theta_2, \ldots, \theta_K \in \Theta = (-\frac{\pi}{2}, \frac{\pi}{2}]$, with respect to the broadside. Accordingly, the qth collected vector snapshot is of the form

$$\mathbf{x}_q = \sum_{k=1}^{K} \mathbf{s}(\theta_k)\xi_{k,q} + \mathbf{n}_q \in \mathbb{C}^{D\times 1}, \tag{2.17}$$

where, for any DoA $\theta \in \Theta$, the array response vector $\mathbf{s}(\theta)$ is

$$\mathbf{s}(\theta) = [1, \mathrm{e}^{-\mathrm{j}\frac{2d\pi f_c}{c}\sin(\theta)}, \ldots, \mathrm{e}^{-\mathrm{j}(D-1)\frac{2d\pi f_c}{c}\sin(\theta)}]^\top. \tag{2.18}$$

$\xi_{k,q} \sim \mathcal{CN}(0, \alpha_k)$ is the q-th transmitted symbol (power-scaled and channel processed) and $\mathbf{n}_q \sim \mathcal{CN}(\mathbf{0}_D, \sigma^2\mathbf{I}_D)$ models complex additive white Gaussian noise (AWGN). Based on (2.17), the nominal received-signal autocorrelation matrix is given by

$$\mathbf{R} = \mathbb{E}\{\mathbf{x}_q\mathbf{x}_q^H\} = \sum_{k=1}^{K} \alpha_k \mathbf{s}(\theta_k)\mathbf{s}(\theta_k)^H + \sigma^2\mathbf{I}_D. \tag{2.19}$$

The eigenvalue decomposition (EVD) $\mathbf{R} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^H$ returns $\mathbf{Q} = [\mathbf{Q}_{S_{D\times K}}\ \mathbf{Q}_{N_{D\times D-K}}]$, $\mathbf{Q}^H\mathbf{Q} = \mathbf{I}_D$, and $\mathbf{\Lambda} = \mathrm{diag}([\omega_1, \ldots, \omega_K, \mathbf{0}_{D-K,1}^\top]^\top) + \sigma^2\mathbf{I}_D$, for $\omega_1 \geq \ldots \geq \omega_K \geq 0$. Thus, the dominant K eigenvectors of $\mathbf{R}$ describe the subspace of the signals of interest interest –i.e., $\mathcal{S} = \mathrm{span}(\mathbf{Q}_S) = \mathrm{span}([\mathbf{s}(\theta_1), \ldots, \mathbf{s}(\theta_K)])$. Accordingly, for any $\theta \in \Theta$, it holds that

$$(\mathbf{I}_D - \mathbf{Q}_S\mathbf{Q}_S^H)\mathbf{s}(\theta) = \mathbf{0}_D \Leftrightarrow \theta \in \Theta_S = \{\theta_1, \ldots, \theta_K\}. \tag{2.20}$$

In lieu of exact knowledge of $\mathbf{R}$ and $\mathbf{Q}_S$, the receiver collects N coherent snapshots in

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{C}^{D \times N}, \tag{2.21}$$

estimates $\mathbf{Q}_S$ by the K L2-norm principal components (PCs) of the snapshots,

$$\mathbf{Q}_{\text{L2}} = \underset{\mathbf{Q} \in \mathbb{C}^{D \times K};\ \mathbf{Q}^H \mathbf{Q} = \mathbf{I}_K}{\operatorname{argmax}} \left\| \mathbf{Q}^H \mathbf{X} \right\|_F^2, \tag{2.22}$$

where $\| \cdot \|_F^2$ returns the sum of squared magnitudes of the entries of its martix argument. The solution to (2.22) is given simply by the K dominant left-singular vectors of $\mathbf{X}$. Accordingly, standard MUSIC computes the spectrum

$$P_{\text{MUSIC}}(\theta) = \left\| (\mathbf{I}_D - \mathbf{Q}_{\text{L2}} \mathbf{Q}_{\text{L2}}^H) \mathbf{s}(\theta) \right\|_F^{-2}, \theta \in \Theta \tag{2.23}$$

and detects $\hat{\theta}_1, \ldots, \hat{\theta}_K$ as the K angle arguments that yield local maxima (peaks) to spectrum $P_{\text{MUSIC}}(\theta)$.

It has been observed that even if few of the snapshots in $\mathbf{X}$ are corrupted by strong directional jamming, the metric of (2.22), $\|\mathbf{Q}^H \mathbf{X}\|_F^2 = \sum_{n=1}^{N} \|\mathbf{Q}^H \mathbf{x}_n\|_2^2$, can be significantly misled [3, 4]. Similar to [57, 67], in this work, we replace the jamming-responsive $\mathbf{Q}_{\text{L2}}$ in (2.23) by the complex L1-PCs of $\mathbf{X}$,

$$\mathbf{Q}_{\text{L1}} = \underset{\mathbf{Q} \in \mathbb{C}^{D \times K};\ \mathbf{Q}^H \mathbf{Q} = \mathbf{I}_K}{\operatorname{argmax}} \left\| \mathbf{Q}^H \mathbf{X} \right\|_1, \tag{2.24}$$

where $\|\cdot\|_1$ returns the sum of the magnitudes of the entries of its matrix argument. Specifically, for the practical case when the snapshots arrive in a streaming fashion, in this work we present and employ, for the first time, an incremental calculator of complex L1-PCA in (2.24).

2.4.2 Theoretical Preliminaries on Complex L1-PCA

Authors in [57] showed that L1-PCA in (2.3) is equivalent to the unimodular nuclear-norm maximization (UNM)

$$\underset{\mathbf{B} \in U^{N \times K}}{\text{maximize}} \ \|\mathbf{X}\mathbf{B}\|_*, \tag{2.25}$$

where $U = \{a \in \mathbb{C};\ |a| = 1\}$ is the set of unimodular complex numbers and the nuclear norm $\|\cdot\|_*$ returns the sum of the singular values of its matrix argument. Specifically, if $\mathbf{Q}_{\text{L1}}$ a solution to L1-PCA,

$$\mathbf{B}_{\text{unm}} = \text{phase}(\mathbf{X}^H \mathbf{Q}_{\text{L1}}) \in U^{N \times K} \tag{2.26}$$

is a solution to UNM in (2.25), where phase$(\cdot)$ returns the phases of the entries of its complex matrix argument.[1] Conversely, if $\mathbf{B}_{\text{unm}}$ is a solution to UNM, then

$$\mathbf{Q}_{\text{L1}} = \Phi(\mathbf{X}\mathbf{B}_{\text{unm}}) \tag{2.27}$$

is a solution to L1-PCA, where for any tall matrix $\mathbf{A}^{m \times n}$ with "thin" SVD $\mathbf{A} = \mathbf{U}_{m \times n}\mathbf{\Sigma}_{n \times n}\mathbf{V}^H_{n \times n}$, we define $\Phi(\mathbf{A}) = \mathbf{U}\mathbf{V}^H$. Moreover,

$$\|\mathbf{Q}_{\text{L1}}^H \mathbf{X}\|_1 = \text{Tr}\left(\mathbf{Q}_{\text{L1}}^H \mathbf{X}\mathbf{B}_{\text{unm}}\right) = \|\mathbf{X}\mathbf{B}_{\text{unm}}\|_*. \tag{2.28}$$

Motivated by (2.26) and (2.27), [57] proposed an iterative algorithm that approximates the exact L1-PCs $\mathbf{Q}_{\text{L1}}$ by the locally optimal $\hat{\mathbf{Q}}$. The algorithm initializes at an orthonormal matrix $\mathbf{Q}_0$ and iterates $\mathbf{B}_i = \text{phase}(\mathbf{X}^H \mathbf{Q}_{i-1})$ and $\mathbf{Q}_i = \Phi(\mathbf{X}\mathbf{B}_i)$, for $i = 1, 2, \ldots, t_{\text{conv}}$, where t_{conv} denotes the converging, or terminating iteration index. The objective metric $\|\mathbf{Q}_{i-1}^H \mathbf{X}\|_1$ is upper-bounded by $\|\mathbf{Q}_{\text{L1}}^H \mathbf{X}\|_1 = \|\mathbf{X}\mathbf{B}_{\text{unm}}\|_*$ and increases monotonically across iterations towards convergence. For simplicity in presentation, the two updates can be merged into the single one

$$\mathbf{Q}_i = \Phi(\mathbf{X}\ \text{phase}(\mathbf{X}^H \mathbf{Q}_{i-1})). \tag{2.29}$$

Upon convergence, or termination, the algorithm returns $\hat{\mathbf{Q}} = \mathbf{Q}_{t_{\text{conv}}}$. In the sequel, we refer to the above iterative procedure as $\hat{\mathbf{Q}} = \text{L1-PCA}(\mathbf{X},\ \mathbf{Q}_0)$.

2.4.3 Proposed Algorithm for Incremental Complex L1-PCA

We start with an initial basis $\mathbf{Q}(0) = \text{L1-PCA}(\mathbf{X}(0);\ \mathbf{Q}_0)$, computed on a small batch of snapshots $\mathbf{Y} = \mathbf{X}(0) = [\mathbf{X}]_{:,1:n}$, with $K \leq n < N$.

Then, for every new snapshot $\mathbf{x}_i = [\mathbf{X}]_{:,n+i}$, $i = 1, 2, \ldots, N - n$, arriving at the receiver, we first

[1] The phase of a complex number $a \in \mathbb{C}$ is given by $\frac{a}{|a|}$.

Algorithm 1: Incremental Complex L1-PCA (Proposed)

Input: $\mathbf{X} \in \mathbb{C}^{D \times N}$, $\mathbf{Q}$, n, τ

$\mathbf{Y} \leftarrow [\mathbf{X}]_{:,1:n}$
$\mathbf{Q} \leftarrow \texttt{L1-PCA}(\mathbf{Y}, \mathbf{Q})$
For $i = 1, 2, \ldots, N - n$
  $\mathbf{x} \leftarrow [\mathbf{X}]_{:,n+i}$
  $r \leftarrow p(\mathbf{Q}, \mathbf{x})$; from (2.30)
  if $r > \tau$
    $\tilde{\mathbf{Y}} \leftarrow [\mathbf{Y},\ \mathbf{x}]$
    $\mathbf{Q} \leftarrow \texttt{L1-PCA}(\tilde{\mathbf{Y}}, \mathbf{Q})$
    $j^* \leftarrow \text{argmin}_{j=1,\ldots,n+1}\, p(\mathbf{Q}, [\tilde{\mathbf{Y}}]_{:,j})$
    $\mathbf{Y} \leftarrow [\tilde{\mathbf{Y}}]_{:,\{1,\ldots,n+1\}\setminus j^*}$

Output: $\mathbf{Q}$

Function: $\mathbf{Q} \leftarrow \texttt{L1-PCA}(\mathbf{X}, \mathbf{Q})$ [57]

Input: $\mathbf{X}$, $\mathbf{Q}$

Until convergence/termination
  $\mathbf{Q} \leftarrow \Phi(\mathbf{X}\ \text{phase}(\mathbf{X}^H \mathbf{Q}))$

Output: $\mathbf{Q}$

Figure 2.6. The proposed incremental complex L1-PCA algorithm.

compute its reliability, in view of the current subspace estimate $\mathbf{Q}_{i-1}$, as

$$p(\mathbf{Q}(i-1), \mathbf{x}_i) = \frac{\left\|\mathbf{Q}(i-1)^H \mathbf{x}_i\right\|_2^2}{\|\mathbf{x}_i\|_2^2} \in [0, 1]. \tag{2.30}$$

If $p(\mathbf{Q}(i-1), \mathbf{x}_i) = 1$, then $\mathbf{x}_i$ is perfectly described by $\text{span}(\mathbf{Q}(i-1))$, while if $p(\mathbf{Q}(i-1), \mathbf{x}_i) = 0$, then $\mathbf{x}_i$ is orthogonal to the current subspace estimate $\text{span}(\mathbf{Q}(i-1))$. In view of this observation, if reliability $p(\mathbf{Q}(i-1), \mathbf{x}_i)$ is below a preset threshold τ, then $\mathbf{x}_i$ is deemed jammer-corrupted and hence rejected. Accordingly, our algorithm sets $\mathbf{Q}(i) = \mathbf{Q}(i-1)$ and $\mathbf{X}(i) = \mathbf{X}(i-1)$. Otherwise, $\mathbf{x}_i$ is deemed appropriate for processing and it is appended to the memory batch $\mathbf{X}(i-1)$, forming the augmented memory matrix

$$\tilde{\mathbf{Y}} = [\mathbf{X}(i-1),\ \ \mathbf{x}_i] \in \mathbb{C}^{D \times n+1}. \tag{2.31}$$

Subsequently, the L1-PCA basis estimate is updated as $\mathbf{Q}(i) = \text{L1-PCA}(\tilde{\mathbf{Y}}; \mathbf{Q}(i-1))$.

Next, in order to maintain a fixed computational and storage cost, we remove from $\mathbf{Y}$ the snapshot with the minimum reliability, setting $\mathbf{Y} = [\tilde{\mathbf{Y}}]_{:,\{1,2,\ldots,n+1\}\setminus j^*}$, where $j^* = \text{argmin}_{j=1,2\ldots,n+1}\, p(\mathbf{Q}, [\tilde{\mathbf{Y}}]_{:,j})$. The iterations of (2.29) are proven to converge for arbitrary initialization in [57]. In Figure 2.6 we present a pseudo-code of the proposed algorithm.

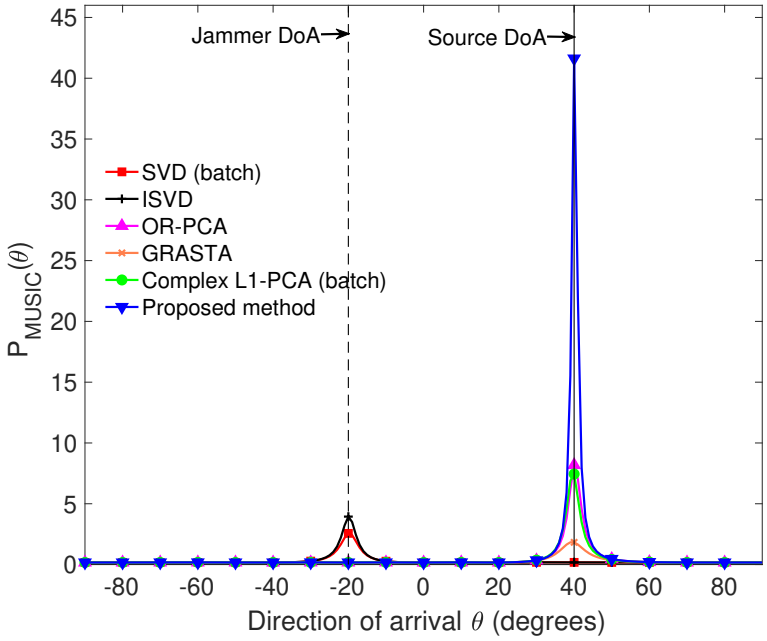


Figure 2.7. MUSIC spectrum versus DoA $\theta \in \Theta$ for $K = 1$ ($D = 6$, $N = 200$, and $n = 20$).

2.4.4 Experimental Studies

We consider DoA of interest $\theta_1 = 40°$ and receiver ULA with $D = 6$ elements collecting $N = 200$ snapshots. The signal-to-noise-ratio (SNR) of the incoming signal is $\text{SNR}_1 = 5$ dB. Snapshots 19, 85 and 155 are corrupted by directional jamming signal with DoA $\theta_j = -20°$ and $\text{SNR}_j = 15$ dB. We set $n = 20$ and $\tau = 0.7$ and employ the proposed method for incrementally approximating the $K = 1$ L1-PC of the received snapshots. At the i-th incremental update, DoA θ_1 is estimated by means of a spectrum in the form of (2.23), computed upon the approximate L1-PC $\mathbf{q}(i)$. In Figure 2.7, we plot the MUSIC spectrum after the last update. Together with the proposed method, we plot the spectrum attained by batch SVD, batch complex L1-PCA [57], incremental SVD (ISVD) [97], and two robust-PCA (RPCA) variants namely, online robust PCA (OR-PCA) [85], and Grassmanian robust adaptive subspace tracking algorithm (GRASTA) [93]. We observe that OR-PCA, GRASTA, and both L1-PCA methods point to the sought-after source DoA, while the SVD-based methods are misled and point to the jammer DoA.

Next, we repeat the experiment, setting $\theta_1 = 20°$, $\theta_j = 5°$, $\text{SNR}_1 = 6$dB, and $\text{SNR}_j = 16$dB, and compute over 10,000 realizations the root-mean-square-error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{10,000} \sum_{t=1}^{10,000} \left(\theta_1 - \hat{\theta}_{1,t}(i)\right)^2}, \tag{2.32}$$

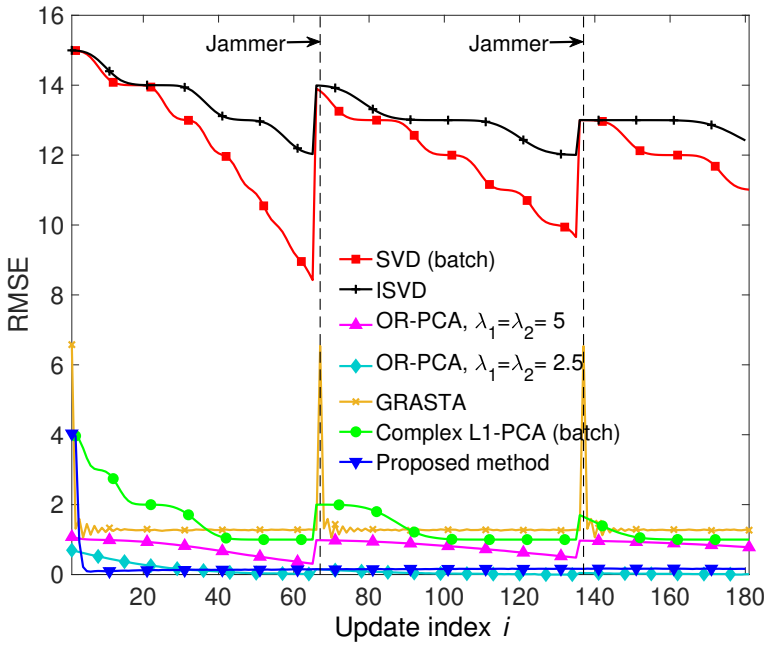


Figure 2.8. RMSE versus update index $i = 1, 2, \ldots, N - n + 1$ for $K = 1$ ($D = 6$, $N = 200$, and $n = 20$).

where $\hat{\theta}_{1,t}(i)$ is the estimated angle of arrival at the i-th update of the t-th realization. In Figure 2.8, we plot the RMSE attained by the proposed algorithm and its counterparts, versus subspace update index i. We notice that SVD and ISVD start at high RMSE, falsely pointing at the jammer; this is arguably because one of the first $n = 20$ points is corrupted by the jammer. The performance of both methods naturally improves as they process nominal points; however, when further corruptions occur (see dotted lines in Figure 2.8), SVD and ISVD are again misled towards the jammer. GRASTA begins at a relatively low error and quickly drops to RMSE lower than 1.5°; when jamming occurs, its RMSE increases momentarily but recovers again very quickly. OR-PCA with regularization parameters $\lambda_1 = \lambda_2 = 5$ also starts at low RMSE but jammer-corrupted snapshots have a longer impact on it; preferred (heuristically set) regularization parameters $\lambda_1 = \lambda_2 = 2.5$ yield lower RMSE, similar to that of the proposed algorithm. The batch complex L1-PCA algorithm of [57] and the proposed incremental algorithm start at equally low RMSE. The performance of batch L1-PCA decreases as it processes more nominal points; however, when the jammer is active, the batch algorithm processes the corrupted snapshots and its RMSE increases –albeit momentarily. On the other hand, the RMSE of the proposed algorithm drops quickly to a very low value (e.g., RMSE $< 0.25°$ after processing just 4 nominal points) and, since the method avoids processing jammer-corrupted snapshots, it stays close to zero across all updates $i = 1, 2, \ldots, N - n + 1$.

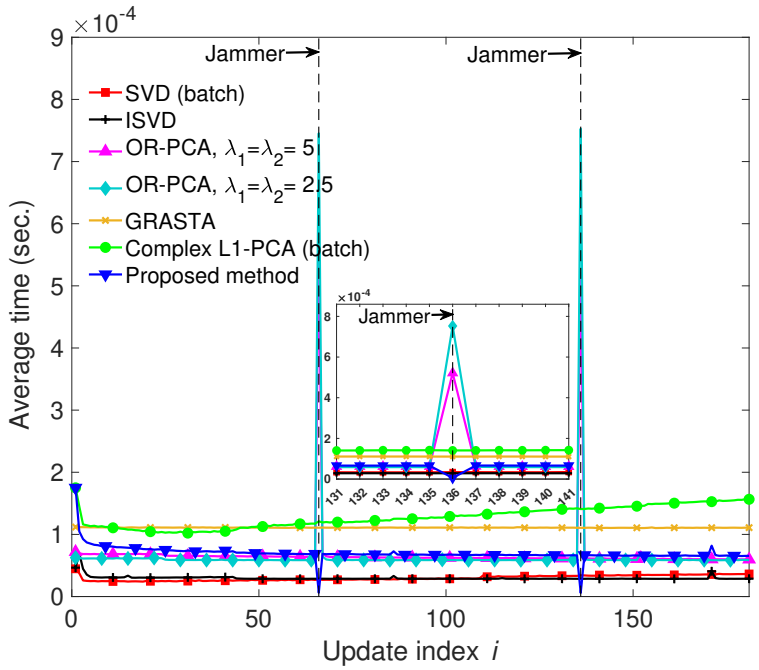


Figure 2.9. Average computation time versus update index $i = 1, 2, \ldots, N - n + 1$ for $K = 1$ ($D = 6$, $N = 200$, and $n = 20$).

In Figure 2.9, we plot the average computation time[2] needed by each algorithm for PC updates versus the update index i. We observe that SVD and ISVD are the fastest; though, we recognize that the cost of batch SVD slowly increases, while the cost of ISVD remains almost constant. GRASTA also needs almost constant computation time across i. In addition, we notice that the –otherwise nearly constant– computation time of OR-PCA exhibits a spike when a corrupted point is processed; this illustrates the intense effort made by the algorithm to recover from the corruption. Batch L1-PCA is the slowest among the compared methods, with computation time increasing across i. The proposed incremental algorithm is considerably faster than batch L1-PCA, exhibiting almost constant computation time across i. We also notice that, interestingly, the proposed method often identifies and discards the jammer-corrupted snapshots without processing them, resulting in steep instantaneous drops of the computation time.

[2]Reported computation times are measured in MATLAB R2017a, run on a computer equipped with Intel(R) core(TM) i7-6700 processor @ 3.40GHz and 32GB RAM.

2.5 Contribution 3: Dynamic Algorithms for L1-norm Tensor Analysis

Present-day datasets are collected across diverse sensing modalities and have an inherent N-way array structure, commonly referred to as *tensors* [9]. Much similar to PCA for matrix data analysis, Tucker decomposition is a standard method for tensor analysis, with many important applications in machine learning [110–112], pattern recognition [1, 32], communications [31, 113, 114], and computer vision [34, 115, 116], among other fields. Another successful tensor analysis tool is the Canonical Polyadic Decomposition (CPD) [71, 117], also known as Parallel Factor Analysis (PARAFAC), with many applications in machine learning, signal processing, and data mining.

In general, Tucker decomposition can be seen as a high-order extension of PCA [17], in the sense that it analyzes a collection of $(N \geq 1)$-way arrays to obtain an orthonormal basis for each mode. Importantly, as opposed to applying PCA on vectorized measurements, Tucker leverages and processes multi-way measurements in their natural tensor form, often resulting in superior generalization performance. Higher-Order Singular Value Decomposition (HOSVD) [10, 118] is the higher order equivalent of SVD used to solve Tucker. Higher-Order Orthogonal Iterations (HOOI) [10, 118] is another algorithm that solves Tucker.

The many advantages of Tucker decomposition in terms of its capability to handle tensor data in their natural form, leading to superior underlying pattern discovery has been studies thoroughly in the literature [9, 10, 71, 118]. However, as is the case with standard PCA, because of its formulation on the outlier-sensitive L2-norm, Tucker is very sensitive against faulty measurements (outliers). In order to rectify Tucker's outlier sensitivity, many researchers have focused on outlier-resistant formulation of Tucker. One common approach is the Higher-Order Robust PCA (HoRPCA) [119], wherein the tensor data is modeled as the sum of a low multi-linear rank tensor, representing nominal data and a sparse tensor capturing outliers. A more straightforward approach simply replaces the L2-norm by the outlier-resistant L1-norm, resulting in the formulation of L1-Tucker [11, 12]. Multiple approximate algorithms for L1-Tucker have been proposed in [11–14, 16, 120, 121].

On the other hand, tensor measurements arrive in a streaming fashion in some applications of interest. As a consequence, when new data becomes available, one needs to leverage the previous solution and update it to obtain a new solution incrementally. Additionally, incremental processing is preferred when the total number of measurements are too big to be stored and/or processed as a batch. To this end, an array of algorithms for incremental Tucker decomposition have been proposed in the literature, including Dynamic Tensor Analysis (DTA), Streaming Tensor Analysis

(STA), Window-based Tensor Analysis (WTA) [122, 123], and Accelerated Online Low-Rank Tensor Learning (ALTO) [124], to name just a few. Although these algorithms offer considerable speed up and computational efficiency, similar to batch Tucker analysis, they are sensitive against outlying measurements.

In this line of research, we present an outlier-resistant algorithm for Tucker based on the robust L1-norm. Our algorithm named Dynamic L1-Tucker is a scalable method that computes the L1-norm Tucker bases incrementally, with the ability to detect and reject outliers in an online fashion. Moreover, our algorithm is capable of tracking and adapting to nominal subspace changes.

2.5.1 Related Works

Streaming and robust matrix PCA has been thoroughly studied over the past decades [24, 80, 84, 93, 125, 126]. However, extending matrix PCA (batch or streaming) to tensor analysis is a non-trivial task that has been attracting increasing research interest. To date, there exist multiple alternative methods for batch tensor analysis (e.g., HOSVD, HOOI, L1-HOOI) but only few for streaming/dynamic tensor analysis. For example, DTA [122, 123] efficiently approximates the HOSVD solution by processing measurements incrementally, with a fixed computational cost per update. Moreover, DTA can track multi-linear subspace changes, weighing past measurements with a forgetting factor. STA [122, 123] is a fast alternative to DTA, particularly designed for time-critical applications. WTA is another DTA variant which, in contrast to DTA and STA, adapts to changes using considering only a sliding window of measurements. The ALTO method was presented in [124]. For each new measurement, ALTO updates the bases through a tensor regression model. In [127], authors presented another method for Low-Rank Updates to Tucker (LRUT). When a new measurement arrives, LRUT projects it on the current bases and few more randomly chosen orthogonal directions, forming an augmented core tensor. Then it updates the bases by standard Tucker (e.g., HOSVD) on this extended core. In [128], authors consider very large tensors and propose randomized algorithms for Tucker decomposition based on the TENSORSKETCH [129]. It is stated these algorithms can also extend for processing streaming data. Randomized methods for Tucker of streaming tensor data were also proposed in [130]. These methods rely on dimension-reduction maps for sketching the Tucker decomposition and they are accompanied by probabilistic performance guarantees. More methods for incremental tensor processing were presented in [131–134], focusing on specific applications, such as foreground segmentation, visual tracking, and video foreground/background separation.

Methods for incremental CPD/PARAFAC tensor analysis have also been presented. For instance,

authors in [135] consider the CPD/PARAFAC factorization model and assume that N-way measurements are streaming. They propose CP-Stream, an algorithm that efficiently updates the CPD every time a new measurement arrives. CP-stream can accommodate user-defined constraints in the factorization such as non-negativity. In addition, authors in [136] consider a Bayesian probabilistic reformulation of the CPD/PARAFAC factorization, assuming that the entries of the processed tensor are streaming across all modes, and develop a posterior inference algorithm (POST). Further, the problem of robust and incremental PARAFAC has also been studied and algorithms have been presented in [137, 138]. Typically, the application spaces of CPD and Tucker are complementary: CPD is preferred when uniqueness and interpretablity are needed; Tucker allows for the latent components to be related (dense core) and it is preferred for low-rank tensor compression and completion, among other tasks [71, 112].

The problem of both outlier-resistant and dynamic Tucker analysis remains to date largely unexplored and it is the main focus of this work.

For the sake of completeness, we offer a brief overview of batch Tucker and L1-Tucker decomposition in the sequel.

2.5.2 Tucker Decomposition

Given tensor measurements $\boldsymbol{\mathcal{X}}_t \in \mathbb{R}^{D_1 \times \ldots \times D_N}$, $t = 1, 2, \ldots, T$, Tucker decomposition of the measurement batch $\{\boldsymbol{\mathcal{X}}_t\}_{t=1}^T$ is formulated as

$$\underset{\{\mathbf{Q}_n \in \mathbb{S}_{D_n,d_n}\}_{n\in[N]}}{\text{max.}} \left\| \boldsymbol{\mathcal{X}} \times_{n\in[N]} \mathbf{Q}_n^\top \right\|_F^2, \tag{2.33}$$

where $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{D_1 \times \ldots \times D_N \times T}$ is the concatenation tensor, such that $\boldsymbol{\mathcal{X}}(:, :, \ldots, :, t) = \boldsymbol{\mathcal{X}}_t$, $\|\boldsymbol{\mathcal{X}} \times_{n\in[N]} \mathbf{Q}_n^\top\|_F^2 = \sum_{t\in[T]} \|\boldsymbol{\mathcal{X}}_t \times_{n\in[N]} \mathbf{Q}_n^\top\|_F^2$. For $d_n \leq D_n$ $\forall n \in [N]$, Tucker in (2.33) strives to compress the tensor measurements such that the overall variance preserved after compression is maximized by estimating N low-rank bases. Algorithms for Tucker decomposition include the HOSVD and HOOI. HOSVD is a higher order analog of SVD and it is a straightforward method that approximates the N bases in (2.33) by solving N PCA problems as follows

$$\underset{\mathbf{Q}\in\mathbb{S}_{D_n,d_n}}{\text{max.}} \|\texttt{mat}(\boldsymbol{\mathcal{X}}, n)^\top \mathbf{Q}\|_F^2. \tag{2.34}$$

Note that each PCA problem to estimate a basis for a particular mode is carried out disjointly of all others and may be done so in parallel to the computation of the bases for other modes. In contrast

L1-Tucker Decomposition (batch-processing) [139]

Input: $\boldsymbol{\mathcal{X}}$, $\{\mathbf{Q}_n\}_{n\in[N]}$

Until convergence/termination
  For $n \in [N]$
    $\mathbf{A} \leftarrow \text{mat}(\boldsymbol{\mathcal{X}} \times_{m\in[n-1]} \mathbf{Q}_m^\top \times_{k\in[N-n]+n} \mathbf{Q}_k^\top, n)$
    $\mathbf{B} \leftarrow \text{sgn}(\mathbf{A}^\top \mathbf{Q}_n)$
    $\mathbf{Q}_n \leftarrow \texttt{L1-BF}(\mathbf{A}, \mathbf{B})$

Return: $\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_N$

Function: $\texttt{L1-BF}(\mathbf{A}, \mathbf{B})$, $\quad \% \ \mathbf{B} \in \{\pm 1\}^{Q\times z}$

Until convergence/termination
  $(\bar{k}, \bar{l}) \leftarrow \underset{(k,l)\in[Q]\times[z]}{\text{argmax}} \left\| \mathbf{A} \left(\mathbf{B} - 2\mathbf{e}_{k,Q}\mathbf{e}_{l,z}^\top [\mathbf{B}]_{k,l} \right) \right\|_*$
  $\mathbf{B} \leftarrow \mathbf{B} - 2\mathbf{e}_{\bar{k},Q}\mathbf{e}_{\bar{l},z}^\top [\mathbf{B}]_{\bar{k},\bar{l}}$
$\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top \leftarrow \texttt{svd}(\mathbf{AB})$
$\mathbf{Q} \leftarrow \mathbf{U}\mathbf{V}^\top$

Return: $\mathbf{Q}$

Figure 2.10. L1-norm Tucker Decomposition algorithm for batch-processing.

to this approach, HOOI optimizes the N bases jointly by means of an iterative algorithm. Given some initial bases $\{\mathbf{Q}_{n,0}\}_{n\in[N]}$, at any iteration $i > 0$ and for $n = 1, 2, \ldots, N$, HOOI computes $\mathbf{Q}_{n,i}$ by solving

$$\underset{\mathbf{Q}\in\mathbb{S}_{D_n,d_n}}{\text{max.}} \ \|\mathbf{A}_{n,i}{}^\top \mathbf{Q}\|_F^2 \tag{2.35}$$

where

$$\mathbf{A}_{n,i} = \texttt{mat}(\boldsymbol{\mathcal{X}} \times_{m\in[n-1]} \mathbf{Q}_{m,i}^\top \times_{k\in[N-n]+n} \mathbf{Q}_{k,i-1}^\top, n). \tag{2.36}$$

2.5.3 L1-Tucker to Combat Outliers

The success of L1-PCA in matrix analysis [3] on outlier corrupted data has prompted its generalization to handle tensor data, resulting in the development of L1-norm Tucker decomposition. Multiple recent works have demonstrated that L1-Tucker performs similar to Tucker with nominal data, while offering significantly better performance on data corrupted with outliers [11–14, 140]. Mathematically, similar to L1-PCA, L1-Tucker derives by simply replacing the outlier-sensitive L2 norm in (2.33) by the outlier-resistant L1-norm, as

$$\underset{\{\mathbf{Q}_n\in\mathbb{S}_{D_n,d_n}\}_{n\in[N]}}{\text{max.}} \ \left\| \boldsymbol{\mathcal{X}} \times_{n\in[N]} \mathbf{Q}_n^\top \right\|_1, \tag{2.37}$$

where $\|\boldsymbol{\mathcal{X}} \times_{n\in[N]} \mathbf{Q}_n^\top\|_1 = \sum_{t\in[T]} \|\boldsymbol{\mathcal{X}}_t \times_{n\in[N]} \mathbf{Q}_n^\top\|_1$. Researchers have developed a robust analog of HOSVD, named L1-HOSVD [12, 15, 141], to solve L1-Tucker in (2.37). L1-HOSVD approximates

the solution to L1-Tucker by N parallel L1-PCA problems as

$$\underset{\mathbf{Q}\in\mathbb{S}_{D_n,d_n}}{\text{max.}} \|\texttt{mat}(\boldsymbol{\mathcal{X}},n)^\top\mathbf{Q}\|_1. \tag{2.38}$$

Although L1-HOSVD is a fast single-shot algorithm it often obtains low L1-Tucker metric. In order to achieve a higher L1-Tucker metric, L1-HOOI was proposed and initialized at the solution of L1-HOSVD [12, 142]. Initialized at the L1-HOSVD bases $\{\mathbf{Q}_{n,0}\}_{n\in[N]}$, L1-HOOI updates $\mathbf{Q}_{n,i}$ at each iteration $i \geq 1$ by solving

$$\underset{\mathbf{Q}\in\mathbb{S}_{D_n,d_n}}{\text{max.}} \|\mathbf{A}_{n,i}{}^\top\mathbf{Q}\|_1, \tag{2.39}$$

where $\mathbf{A}_{n,i}$ is defined in (2.36). A pseudo-code of L1-Tucker, implemented by means of L1-HOOI is offered in Figure 2.10. The work in [12] presents formal convergence guarantees and offers complexity analysis of the L1-HOOI algorithm. Since, L1-Tucker is implemented as a series of L1-PCAs, it may be infeasible to rely on the exact algorithms of L1-PCA due to their high cost [3]. In this case, it is beneficial to leverage the approximate algorithms for L1-PCA instead [4, 6]. In this work, we make use of the L1-BF algorithm of [4] and for the sake of completeness, we present an overview of L1-BF in Section 2.2 and the corresponding pseudo-code is presented in Figure 2.1.

2.5.4 Proposed Algorithm

Motivated by the incremental and adaptive L1-PCA solvers of [24], we develop an algorithm for dynamic L1-Tucker (D-L1-Tucker) in the sequel.

Batch Initialization

In order to obtain an initial set of L1-Tucker estimates $\mathcal{Q}_0 = \{\mathbf{Q}_1^{(0)}, \ldots, \mathbf{Q}_N^{(0)}\}$, we collect an initial batch of $B \ll T$ measurements in $\mathcal{B} = \{\boldsymbol{\mathcal{X}}_1, \ldots, \boldsymbol{\mathcal{X}}_B\}$ and run L1-HOSVD/L1-HOOI on them. In the absence of an initial batch, $\mathcal{Q}_0$ is initialized arbitrarily.

Additionally, we initialize a batch of measurements that acts as the memory of our algorithm, namely, the *memory set* $\mathcal{M}_0 = \Omega(\mathcal{B}, M)$, for some maximum memory size $M \geq 0$, where, for any

ordered set $\mathcal{I}$ and integer $Z \geq 0$, we define

$$\Omega(\mathcal{I}, Z) = \begin{cases} \mathcal{I}, & \text{if} \quad |\mathcal{I}| \leq Z, \\ [\mathcal{I}]_{|\mathcal{I}|-Z+1:|\mathcal{I}|}, & \text{if} \quad |\mathcal{I}| > Z, \end{cases} \tag{2.40}$$

where $|\cdot|$ denotes the cardinality (number of elements in a set) of its input argument. In simple words, $\Omega(\mathcal{B}, M)$ returns the last $\min\{M, B\}$ elements in $\mathcal{B}$. We resort to memoryless processing in the absence of an initial memory batch $\mathcal{B}$.

When a new measurement $\bar{\boldsymbol{\mathcal{X}}}_t \neq \mathbf{0}$, $t \geq 1$ arrives[3], we compute its conformity with respect to the most recently estimates bases $\mathcal{Q}_{t-1}$, using its reliability defined as [24, 59, 121, 143]

$$r_t = \left\| \bar{\boldsymbol{\mathcal{X}}}_t \times_{n\in[N]} \mathbf{Q}_n^{(t-1)^\top} \right\|_F^2 \left\| \bar{\boldsymbol{\mathcal{X}}}_t \right\|_F^{-2} \quad \in [0, 1]. \tag{2.41}$$

Upon further simplification, (2.41) can be rewritten as

$$r_t = \cos^2(\phi(\text{vec}(\bar{\boldsymbol{\mathcal{X}}}_t \times_{n\in[N]} \mathbf{Q}_n^{(t-1)} \mathbf{Q}_n^{(t-1)^\top}), \text{vec}(\bar{\boldsymbol{\mathcal{X}}}_t))), \tag{2.42}$$

where $\phi(\cdot, \cdot)$ returns the angle between its two vector arguments. We note that r_t quantifies the measure of conformity of $\bar{\boldsymbol{\mathcal{X}}}_t$ to the subspace spanned by $\{\mathbf{Q}_n^{(t-1)}\}_{n\in[N]}$, or, the angular proximity of $\text{vec}(\bar{\boldsymbol{\mathcal{X}}}_t \times_{n\in[N]} \mathbf{Q}_n^{(t-1)} \mathbf{Q}_n^{(t-1)^\top})$ to $\text{vec}(\bar{\boldsymbol{\mathcal{X}}}_t)$.

The value of r_t is guaranteed to exist between 0 and 1. If $r_t = 1$, then the bases in $\mathcal{Q}_{t-1}$ perfectly describe $\bar{\boldsymbol{\mathcal{X}}}_t$. However, if $r_t = 0$, then the set $\mathcal{Q}_{t-1}$ does not capture any component of $\bar{\boldsymbol{\mathcal{X}}}_t$. In order to decide if a measurement $\bar{\boldsymbol{\mathcal{X}}}_t$ should be admitted for processing or not, we define a threshold τ and consider $\bar{\boldsymbol{\mathcal{X}}}_t$ for bases update if its reliability $r_t \geq \tau$. Otherwise, $\bar{\boldsymbol{\mathcal{X}}}_t$ is considered to be an outlier and it is rejected. We note that since the most recently updated bases were computed via L1-Tucker, the reliability defined in (2.41) inherits its robustness. In the unlikely event that an outlier passes the reliability check, L1-Tucker will suppress it, and provide a reliable update of bases.

When a new measurement $\bar{\boldsymbol{\mathcal{X}}}_t$ is deemed reliable by the reliability check, it is used to update the bases and memory by first appending it to the existing memory batch to obtain the augmented memory batch $\mathcal{M}' = \Phi(\mathcal{M}_{t-1}, \bar{\boldsymbol{\mathcal{X}}}_t) = \mathcal{M}_{t-1} \cup \bar{\boldsymbol{\mathcal{X}}}_t$. Next, we initialize L1-HOOI at $\mathcal{Q}_{t-1}$ and run it on $\mathcal{M}'$ to obtain the updated bases $\mathcal{Q}_t$. Finally, we discard the oldest measurement in the memory

[3] A bar over a tensor denotes that it is streaming.

batch to maintain a constant batch size as

$$\mathcal{M}_t = \Omega(\mathcal{M}', M). \tag{2.43}$$

Therefore, the cost of the underlying L1-HOOI algorithm employed to update the bases at each update index remains low (mostly constant) as the processed memory batch comprises at most $M + 1$ measurements.

When a measurement $\bar{\boldsymbol{\mathcal{X}}}_t$ is discarded as a result of its failure in the reliability check, we retain the previous bases and memory by setting $\mathcal{Q}_t = \mathcal{Q}_{t-1}$ and $\mathcal{M}_t = \mathcal{M}_{t-1}$, respectively. At this point, it is worth noting that the proposed approach focuses on temporal coherence of streaming measurements. That is, temporally sporadic points from a second nominal source of measurements could be perceived as outliers.

Zero Centering

In some applications, for example in image processing, subspaces of zero-centered data may be needed. To this end, we can modify the proposed algorithm as follows. First, at each update index $(t-1)$, we compute the mean of the memory batch as $\boldsymbol{\mathcal{C}}_{t-1} = (1/M)\sum_{m=1}^{M}[\mathcal{M}_{t-1}]_m$ and then use it for zero-centering as $\bar{\boldsymbol{\mathcal{X}}}_t^c = \bar{\boldsymbol{\mathcal{X}}}_t - \boldsymbol{\mathcal{C}}_{t-1}$. Next, he reliability check is enforced on the zero-centered memory batch and it is used to update the bases as described above, if it passes the reliability check.

Adaptation to Subspace Changes

In many applications of interest, the underlying data subspaces change over time. In such cases, the changing subspaces have to be tracked. To this end, we modify our D-L1-Tucker algorithm to process measurements from changing subspaces. A natural ambiguity that arises when the data subspace changes is on understanding whether the new measurement is to be treated as an outlier or nominal with respect to a different subspace. We address this ambiguity as follows.

First, we introduce a memory buffer for ambiguous measurements, $\mathcal{W}$, with capacity $W > 0$. When a streaming measurement fails the reliability check, instead of discarding it, we store it in $\mathcal{W}$. This procedure is repeated for consecutive unreliable measurements until a measurement passes the reliability check, when $\mathcal{W}$ is emptied. However, if W consecutive streaming measurements are

Proposed Dynamic L1-Tucker Decomposition

Input: $\{\bar{\boldsymbol{\mathcal{X}}}_t\}_{t\in[T]}$, B, M, W, τ, $\mathcal{Q} \leftarrow \cup_{n\in[N]}\mathbf{Q}_n$

$\mathcal{B} \leftarrow \bigcup_{t\in[B]} \bar{\boldsymbol{\mathcal{X}}}_t$

$(\mathcal{Q}, \mathcal{M}, \mathcal{W}) \leftarrow$ `batch-init`$(\mathcal{B}, \mathcal{Q}, M)$

For $t > B$

    $(\mathcal{Q}, \mathcal{M}, \mathcal{W}) \leftarrow$ `online-updates`$(\bar{\boldsymbol{\mathcal{X}}}_t, \mathcal{M}, \mathcal{Q}, \mathcal{W}, \tau)$

Return: $\mathcal{Q} \rightarrow \{\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_N\}$

Function: `batch-init`$(\mathcal{B}, \mathcal{Q}, M)$

$\mathcal{Q} \leftarrow$ `L1-Tucker`$(\mathcal{B}, \mathcal{Q})$

$\mathcal{M} \leftarrow \Omega(\mathcal{B}, M)$

$\mathcal{W} \leftarrow \varnothing$

Return: $\mathcal{Q}, \mathcal{M}, \mathcal{W}$

Function: `online-updates`$(\bar{\boldsymbol{\mathcal{X}}}_t, \mathcal{M}, \mathcal{Q}, \mathcal{W}, \tau)$

$r_t \leftarrow \|\bar{\boldsymbol{\mathcal{X}}}_t \times_{n\in[N]} \mathbf{Q}_n^\top\|_F^2 \|\bar{\boldsymbol{\mathcal{X}}}_t\|_F^{-2}$

If $r_t > \tau$

    $\mathcal{M}' \leftarrow \Phi(\mathcal{M}, \bar{\boldsymbol{\mathcal{X}}}_t)$

    $\mathcal{Q} \leftarrow$ `L1-Tucker`$(\mathcal{M}', \mathcal{Q})$

    $\mathcal{M} \leftarrow \Omega(\mathcal{M}', M)$

    $\mathcal{W} \leftarrow \varnothing$

Else

    $\mathcal{W} \leftarrow \mathcal{W} \cup \bar{\boldsymbol{\mathcal{X}}}_t$

    If $|\mathcal{W}| = W$

        $(\mathcal{Q}, \mathcal{M}, \mathcal{W}) \leftarrow$ `batch-init`$(\mathcal{W}, \mathcal{Q}, M)$

Return: $\mathcal{Q}, \mathcal{M}, \mathcal{W}$

Figure 2.11. The proposed Dynamic L1-norm Tucker Decomposition algorithm.

rejected as outliers, we detect a subspace change. In order to adapt to this change, we empty the existing memory batch, set $\mathcal{B} = \mathcal{W}$, and re-initialize (reset) the bases and memory, to update them as described earlier. A pseudo-code of the proposed D-L1-Tucker algorithm is presented in Figure 2.11.

2.5.5 Experimental Study

Dynamic Video Foreground/Background Separation

Video background/foreground separation is a commonly used pre-processing technique employed in many video processing applications, including security surveillance, foreground tracking, and traffic monitoring. The static background defines a nominal subspace, while any foreground movement constitute intermittent outliers. In this experiment, we employ the proposed D-L1-Tucker to perform background/foreground separation on videos from the CAVIAR database [144]. In order to demonstrate the ability of D-L1-Tucker to adapt to subspace changes, we make use of two videos capturing different scenes, with 100 frames per video, each of size $(D_1 = 173) \times (D_2 = 231)$. We form a single video stream by concatenating the frames of scene 2 behind those of scene 1 as $\mathcal{X} \in \mathbb{R}^{D_1 \times D_2 \times (T=200)}$.

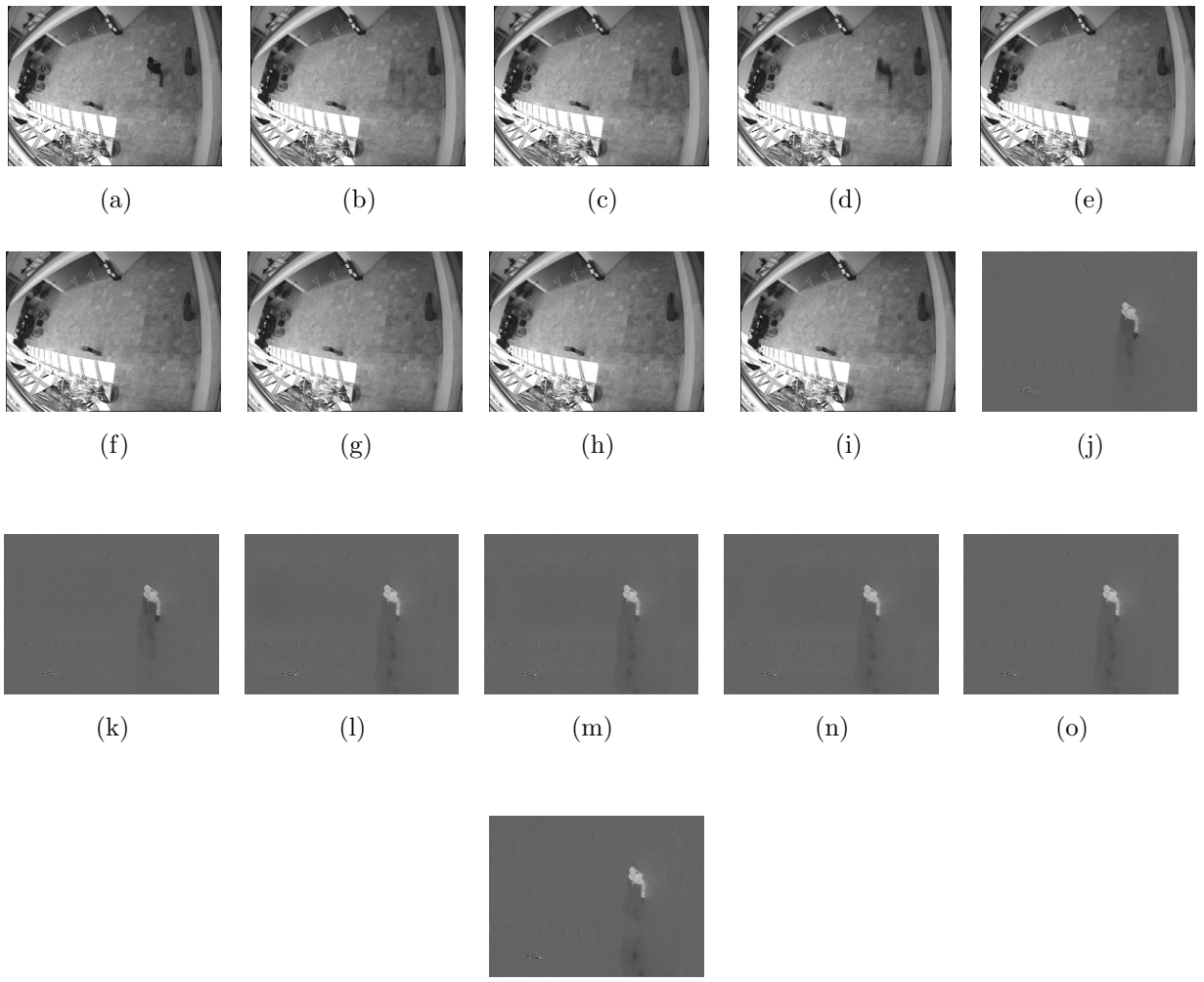


Figure 2.12. Dynamic video foreground/background separation experiment. (a) Original 75-th frame (scene 1). Background extracted by (b) Adaptive Mean ($\lambda = 0.95$), (c) DTA ($\lambda = 0.95$), (d) DTA ($\lambda = 0.7$), (e) LRUT, (f) OSTD, (g) HOOI (increasing memory), (h) L1-HOOI (increasing memory), and (i) D-L1-TUCKER (proposed). Foreground extracted by (j) Adaptive Mean ($\lambda = 0.95$), (k) DTA ($\lambda = 0.95$), (l) DTA ($\lambda = 0.7$), (m) LRUT, (n) OSTD, (o) HOOI (increasing memory), (p) L1-HOOI (increasing memory), and (q) D-L1-TUCKER (proposed).

We set $d_n = d = 3$, $B = 5$, $M = 10$, $W = 20$, τ to the median of the reliabilities of the initial memory batch, and run the proposed algorithm on $\mathcal{X}$. For every $t \in [T - B] + B$, we obtain bases $\mathbf{Q}_1^{(t)}$ and $\mathbf{Q}_2^{(t)}$ and the mean frame $\boldsymbol{\mathcal{C}}_t$. Accordingly, we estimate the background as $\boldsymbol{\mathcal{X}}_t^{\text{BG}} = \mathbf{Q}_1^{(t)} \mathbf{Q}_2^{(t)\top} \left(\bar{\boldsymbol{\mathcal{X}}}_t - \boldsymbol{\mathcal{C}}_t \right) \mathbf{Q}_2^{(t)} \mathbf{Q}_2^{(t)\top} + \boldsymbol{\mathcal{C}}_t$ and the foreground as $\boldsymbol{\mathcal{X}}_t^{\text{FG}} = \boldsymbol{\mathcal{X}}_t - \boldsymbol{\mathcal{X}}_t^{\text{BG}}$.

For the sake of comparison, we employ DTA, LRUT, OSTD, HOOI (increasing batch), and L1-HOOI (increasing batch). We note that HOOI and L1-HOOI are initialized arbitrarily and utilize

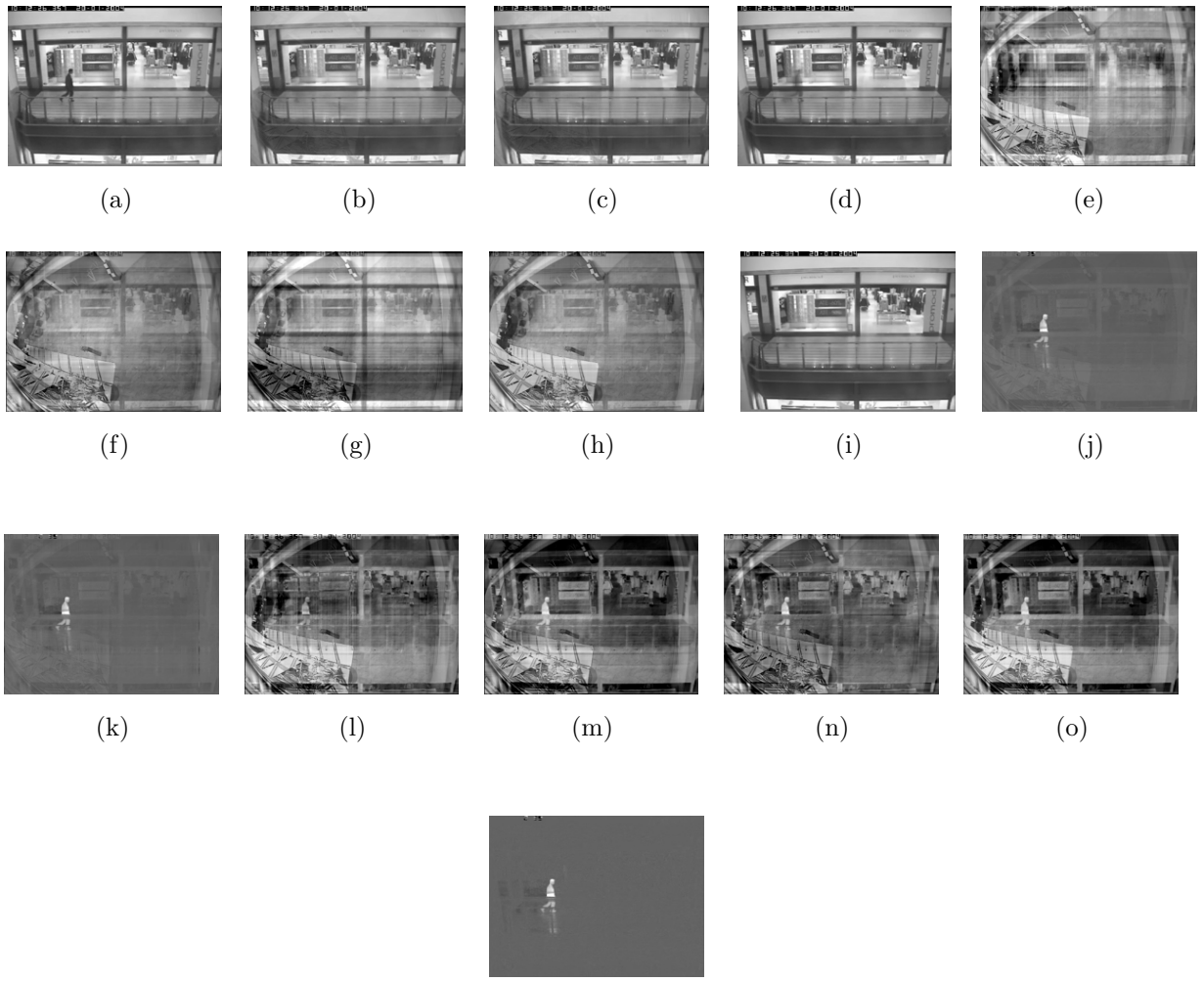


Figure 2.13. Dynamic video foreground/background separation experiment. (a) Original 150-th frame (scene 2). Background extracted by (b) Adaptive Mean ($\lambda = 0.95$), (c) DTA ($\lambda = 0.95$), (d) DTA ($\lambda = 0.7$), (e) LRUT, (f) OSTD, (g) HOOI (increasing memory), (h) L1-HOOI (increasing memory), and (i) D-L1-TUCKER (proposed). Foreground extracted by (j) Adaptive Mean ($\lambda = 0.95$), (k) DTA ($\lambda = 0.95$), (l) DTA ($\lambda = 0.7$), (m) LRUT, (n) OSTD, (o) HOOI (increasing memory), (p) L1-HOOI (increasing memory), and (q) D-L1-TUCKER (proposed).

all frames until frame index t, $\{\bar{\boldsymbol{\mathcal{X}}}_j\}_{j\in[t]}$ to estimate the background/foreground at update index t. Since we rely on zero-centering for background estimation, we modify the adaptive method of DTA to incorporate mean tracking as $\boldsymbol{\mathcal{C}}_t^{\text{DTA}} = \lambda\boldsymbol{\mathcal{C}}_{t-1}^{\text{DTA}} + (1-\lambda)\bar{\boldsymbol{\mathcal{X}}}_t$, for $\boldsymbol{\mathcal{C}}_1^{\text{DTA}} = \bar{\boldsymbol{\mathcal{X}}}_1$. For all other methods, we compute the mean incrementally at any t as $\boldsymbol{\mathcal{C}}_t = \left((t-1)\boldsymbol{\mathcal{C}}_{t-1} + \bar{\boldsymbol{\mathcal{X}}}_t\right)/t$. For DTA, we use two values of forgetting factor, $\lambda = 0.95, 0.7$ and for LRUT we set the number of additional core dimensions to $k_n = D_n - d - 3$.

Figures 2.12 and 2.13 depict the backgrounds and foregrounds obtained by the proposed method

and all other methods under comparison at 75-th frame (scene 1) and 150-th frame (scene 2) respectively. From Figure 2.12, we observe that HOOI (increasing batch), LRUT, and OSTD demonstrate similar performance with a trail of ghostly appearance along the path of the pedestrian in their corresponding foreground frames. OSTD and L1-HOOI (increasing batch) perform better with a smoother trail behind the person in their foreground frames DTA background contains the person, leading to an undesirably smudged foreground estimate for $\lambda = 0.7$. However, DTA is able to obtain a cleaner estimate of the foreground with $\lambda = 0.95$, similar to that of adaptive mean (background estimated by the same adaptive mean that we use for DTA), but their backgrounds contain a ghostly appearance of the person. The proposed method extracts a cleaner background and foreground owing to its outlier rejection capability.

We demonstrate the adaptation capability of the proposed method by presenting the estimated backgrounds and foregrounds after the scene change occurs after frame index $t = 100$. Specifically, we present the background/foreground separating results on frame $t = 150$ in Figure 2.13 and observe that HOOI, L1-HOOI, OSTD, and LRUT demonstrate degraded performance as these methods that are not capable of subspace adaptation. Similar to our observation in Figure 2.12, we note that DTA ($\lambda = 0.95$) outperforms DTA ($\lambda = 0.7$) after scene change, on frame $t = 150$. We observe that DTA ($\lambda = 0.7$) retains the background of the previous scene resulting in an undesirable estimate of the foreground, while DTA ($\lambda = 0.95$) obtains a clean background, and hence a smooth foreground, wherein the person appears slightly blurry. On the other hand, the proposed method demonstrates superior performance by successfully tracking the scene change and consequently obtaining a clean estimate of the background and foreground. To quantify the background/foreground estimation performance, we compute, for every frame, the Peak Signal-to-Noise Ratio (PSNR) defined as $\text{PSNR} = 10\texttt{log}_{10}\left(\frac{255^2}{\text{MSE}}\right)$, where MSE is the mean square error of the estimated background and the ground truth (clean) background. In Figure 2.14, we plot PSNR versus the frame index and observe that all methods begin with high PSNR and their PSNR degrades as foreground movement occurs. We observe that the PSNR of the proposed method is the highest after approximately frame 25. When the scene changes, the PSNR of all methods drops instantaneously. The PSNR values of HOOI, L1-HOOI, LRUT, and OSTD increase at a low rate as they process frames from the new scene. Adaptive mean and DTA with $\lambda = 0.95$ demonstrate better performance with faster PSNR increase. DTA with $\lambda = 0.5$ adapts to the new scene very quickly, but it is affected by foreground movement (depicted by oscillations in its PSNR values). The proposed method adapts to the new scene after it processes $W = 20$ measurements and attains the highest PSNR values across all frame indices thereafter. The PSNR values of the proposed method drop and stay constant for $W = 20$ frames after scene change because the first $W = 20$ points fail the reliability check and do not participate in the subspace update. After

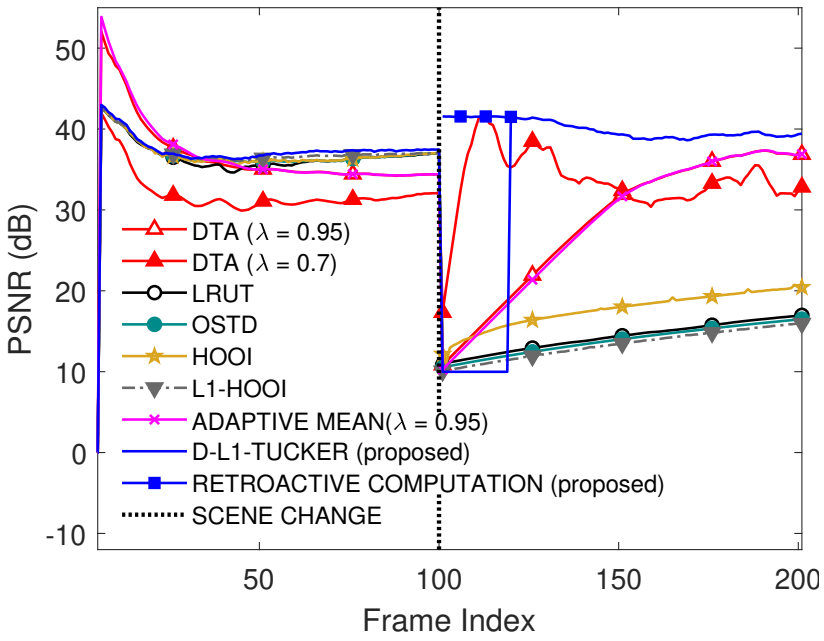


Figure 2.14. Dynamic video foreground/background separation experiment. PSNR (dB) versus frame index.

$W = 20$ frames, the ambiguity memory overflows and the proposed method rapidly adapts to the new scene. Right after adaptation, the proposed method is straightforwardly capable of extracting the background and foreground of all ambiguous frames in $\mathcal{W}$ in a retroactive fashion, leading to a better performance during scene change as shown in Figure 2.14.

2.6 Conclusions

In this chapter, we presented algorithms for reliable data analysis based on the L1-norm. In order to handle matrix data, we presented an algorithmic framework for incremental and adaptive L1-PCA. Our algorithm (L1-APCA) is capable of estimating the sought-after subspace in an online fashion and adapting the L1-PCA solution to changes in the nominal signal subspace, while remaining robust against outliers. Next, for complex-valued streaming data processing, we presented an algorithm for incremental L1-PCA of complex-valued data and applied it for DoA estimation. Our algorithm combines low computational cost with sturdy corruption resistance, allowing for superior DoA estimation in the presence of intermittent jamming signals. Finally, to handle streaming tensor data we proposed an algorithm for dynamic and outlier-resistant Tucker analysis. Our algorithm is successful in multilinear subspace estimation, online outlier identification and rejection, and adaptation to nominal subspace changes. Our numerical studies corroborate the efficacy of our algorithms in terms of remarkable online subspace estimation and superior outlier resistance.

Chapter 3

Robust Stochastic Principal Component Analysis

3.1 Introduction

Modern machine learning applications rely on large volumes of data to achieve impressive generalization performance and in order to perform effective underlying pattern discovery, these data need to be manipulated efficiently. PCA is a ubiquitous method for high dimensional data analysis with a plethora of applications in machine learning and signal processing among many other areas as also motioned earlier [145]. Given a data matrix, PCA strives to estimate a low-rank subspace whereon data variance is maximized by solving the problem in (2.2). Given access to the whole population of data arising from some unknown zero-mean distribution $\mathcal{D}$, PCA returns the subspace on which the expected squared Frobenius-norm of the data projection is maximized by solving

$$\mathbf{Q}_{\mathrm{L2}} = \underset{\mathbf{Q}\in\mathbb{S}_{D,K}}{\operatorname{argmax}} \underset{\mathbf{x}\sim\mathcal{D}}{\mathbb{E}} \left\{ \left\| \mathbf{Q}^\top \mathbf{x} \right\|_F^2 \right\}. \tag{3.1}$$

Defining $\mathbf{C} := \mathbb{E}_{\mathbf{x}}\{\mathbf{x}\mathbf{x}^\top\}$, the objective in (3.1) can be rewritten as $\mathbb{E}_{\mathbf{x}}\{\texttt{trace}(\mathbf{Q}^\top\mathbf{x}\mathbf{x}^\top\mathbf{Q})\} = \texttt{trace}(\mathbf{Q}^\top\mathbf{C}\,\mathbf{Q})$ and it is maximized by the dominant K-eigenvectors of $\mathbf{C}$. In practical applications, $\mathbf{C}$ is unknown and sample-average estimated as $\widehat{\mathbf{C}} := \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i\mathbf{x}_i^\top = \frac{1}{N}\mathbf{X}\mathbf{X}^\top$, based on a size-$N$ batch of coherent data points $\mathbf{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_N]$. Accordingly, substituting $\mathbf{C}$ with $\widehat{\mathbf{C}}$, the metric of (3.1) can be approximated as $\texttt{trace}(\mathbf{Q}^\top\hat{\mathbf{C}}\mathbf{Q}) = \frac{1}{N}\|\mathbf{X}^\top\mathbf{Q}\|_F^2$, where the squared Frobenius-norm $||.||_F^2$ returns the sum of the squared entries of its argument as discussed earlier.

This results in the following sample average approximation of the PCA formulation in (3.1)

$$\widehat{\mathbf{Q}_{\mathrm{L2}}} = \underset{\mathbf{Q}\in\mathbb{S}_{D,K}}{\operatorname{argmax}} \ \frac{1}{N}\left\|\mathbf{X}^\top\mathbf{Q}\right\|_2^2. \tag{3.2}$$

Notice that the objective in (3.2) is simply a scaled version of that in (2.2), therefore, their corresponding maximizing arguments are the same.

It is straightforward to show that for $K = 1$, the objective in (3.1) simplifies to

$$\mathbf{q}_{\mathrm{L2}} = \underset{\mathbf{q}\in\mathbb{S}_D}{\operatorname{argmax}} \ \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}}\{|\mathbf{x}^\top\mathbf{q}|^2\}, \tag{3.3}$$

whose solution is the dominant eigenvector of $\mathbf{C}$. Similarly, the sample approximated PC is obtained by solving

$$\widehat{\mathbf{q}_{\mathrm{L2}}} = \underset{\mathbf{q}\in\mathbb{S}_D}{\operatorname{argmax}} \ \frac{1}{N}||\mathbf{X}^\top\mathbf{q}||_2^2. \tag{3.4}$$

We note that the solutions of (3.2) and (3.4) warrant the computation of the sample approximation of the covariance matrix, which results in cost $\mathcal{O}(D^2N)$ and an additional cost of $\mathcal{O}(D^3)$ to compute the eigenvalue decomposition (EVD). Interestingly, the solution to (3.2) and (3.4) can be obtained directly from the SVD of $\mathbf{X}$ with cost $\mathcal{O}(ND\texttt{min}\{N, D\})$. We note that in our big data era, practical applications come with very large D and/or N, leading to prohibitively high cost of solving (3.2) and (3.4) either using EVD or SVD.

In order to process big data with low cost, the power method was proposed in [17]. Although the power method is faster than SVD in practice (more details in the related work subsection), for massive N and/or D, the power method is rendered restrictive, creating a need for efficient streaming PCA algorithms. To this end, Oja proposed a stochastic PCA algorithm in [18]. Oja's algorithm has gained popularity in the big-data era due to its low cost and convergence guarantees [146–148]. Oja's method can be perceived as a projected stochastic gradient ascent algorithm on the objective of (3.1). A more detailed description of Oja's algorithm and the associated cost is presented in the related work subsection. Despite its recent fame, Oja's algorithm is notoriously sensitive to outliers due to the squared emphasis it places on each datum, including outliers. Moreover, since Oja's method employs stochastic gradient that relies on a single sample at each index, its updates are noisy as also noted in [149].

3.1.1 Contributions

In this chapter, we propose stochastic algorithms for outlier-resistant PCA that can process streaming data with low cost. For the case of $K = 1$, we propose a modified stochastic L1-PCA algorithm and for $K \geq 1$, we propose a generalized framework for stochastic PCA based on the Barron loss [26]. Our generalized framework comes with a tunable parameter for robustness that can be handcrafted to achieve a trade-off between higher robustness and faster convergence. Interestingly, our numerical studies show that when the robustness parameter is chosen to offer outlier resistance, we not only achieve superior robustness against outliers, but also the updates are of low variance. Specifically, our contributions in this line of research can be summarized as follows

1. For $K = 1$, we extend batch L1-PC and propose a novel algorithm for stochastic PC calculation based on mean absolute projection maximization, with formal convergence guarantees.
2. Since the L1-PCA objective is discontinuous and therefore non-differentiable at the origin, we propose an approach to approximate the L1-PCA objective near the origin to achieve differentiablity. Next, we leverage the seminal stochastic approximation theory [150] to develop an algorithm that offers superior outlier resistance while converging quickly to low subspace error.
3. For $K \geq 1$, we propose a generalized framework for Oja-like PCA algorithms by employing the Barron loss [26]. Our framework offers the capability to tune the trade-off between robustness versus speed of convergence by handcrafting the robustness parameter α based on the application.
4. We show in theory that by tuning the robustness parameter α we can vary the step-size of the proposed algorithm, with particular values of α resulting in the coincidence of the proposed algorithm with that of Oja, while other lower values of α yield robust counterparts including the robust stochastic PCA algorithm in [27].
5. We offer a variety of experimental studies on both synthetic and real-world data to demonstrate the efficacy of the proposed methods. Our studies on synthetic data demonstrate the convergence and outlier-resistance of the proposed algorithms. Our real-world studies on images, videos, and biomedical data demonstrate the impressive robustness of the proposed methods compared to state-of-the-art stochastic PCA methods.

The rest of this chapter is organized as follows. We offer an overview of existing stochastic PCA algorithms and their variants in Section 3.2. Next, in Section 3.3, we present our algorithm and

related experimental studies for $K = 1$. Section 3.4 deals with the derivation of our algorithm for $K \geq 1$ based on Barron loss and the corresponding experimental studies. Finally, in Section 3.5, we offer our concluding remarks.

3.2 Related Works

In this section, we provide an overview of existing works on stochastic PCA, wherein the goal is to perform incremental updates to the estimated subspace each time a new data measurement is available. The fundamental approach for stochastic PCA based on the stochastic approximation theory was proposed in 1982 by Oja [18]. Oja's method has been studied and applied widely in practice [149, 151–154]. The popularity of Oja's method can be attributed to its intuitive formulation, simplicity, ease of computation, and guaranteed asymptotic convergence under mild conditions. Mathematically, given data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$, for $K = 1$, Oja's method estimates the dominant PC by performing the following update

$$\mathbf{q}_t = \mathcal{P}(\mathbf{q}_{t-1} + \eta_t \mathbf{x}_t \mathbf{x}_t^\top \mathbf{q}_{t-1}), \tag{3.5}$$

where $t = 1, 2, \ldots, N$, η_t is the step-size (or learning rate), $\mathbf{q}_0$ is arbitrarily chosen on the unit sphere, and for any $\mathbf{g} \in \mathbb{R}^D \setminus \mathbf{0}_D$, $\mathcal{P}(\mathbf{g}) = \frac{\mathbf{g}}{\|\mathbf{g}\|_2}$. The extension of Oja's method to multiple components ($K > 1$) is straightforward and is achieved by updating the estimated subspace as

$$\mathbf{Q}_t = \mathcal{P}(\mathbf{Q}_{t-1} + \eta \mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}), \tag{3.6}$$

where for any $\mathbf{G} \in \mathbb{R}^{D \times K}$, $\mathcal{P}(\mathbf{G})$ projects $\mathbf{G}$ onto the Stiefel manifold $\{\mathbf{Q} \in \mathbb{R}^{D \times K} : \mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_K\}$ by taking the QR-decomposition of $\mathbf{G}$ as $\mathbf{G} \leftarrow \texttt{QR}(\mathbf{G})$. The cost per iteration is $\mathcal{O}(DK)$ to compute $\mathbf{Q}_{t-1} + \eta \mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}$ in (3.6) and $\mathcal{O}(DK^2)$ to compute the orthogonalization step $\mathcal{P}(\cdot)$ [153]. Interestingly, the work in [153] showed that the orthogonalization step is performed purely for computational purposes and excluding it still results in the iterates spanning the same subspace. Therefore, in order to maintain numerical stability, the orthogonalization step may be performed infrequently in practice. On the other hand, the power method performs repeated updates of the form $\mathbf{Q}_t = \mathcal{P}(\widehat{\mathbf{C}}\mathbf{Q}_{t-1})$, with cost $\mathcal{O}(KD^2)$ (assuming $\widehat{\mathbf{C}}$ is available) to compute $\widehat{\mathbf{C}}\mathbf{Q}_{t-1}$ and an additional cost of $\mathcal{O}(DK^2)$ for orthogonalization. Clearly, neglecting the cost of orthogonalization for both methods, the cost of Oja, $\mathcal{O}(DK)$ is significantly lower (linear in D) compared to the cost of power method, $\mathcal{O}(D^2K)$ (quadratic in D), especially in cases where D is large.

An approach closely related to Oja for the computation of the dominant principal component was presented by Krasulina in [146, 155, 156]. Krasulina's update at each measurement index t is

$$\mathbf{q}_t = \mathbf{q}_{t-1} + \eta_t \Big(\mathbf{x}_t \mathbf{x}_t^\top - \frac{\mathbf{q}_{t-1}^\top \mathbf{x}_t \mathbf{x}_t^\top \mathbf{q}_{t-1} \mathbf{I}_D}{\|\mathbf{q}_{t-1}\|_2^2} \Big) \mathbf{q}_{t-1}. \tag{3.7}$$

We note that the estimators of Oja and Krasulina in (3.5) and (3.7) respectively converge to the dominant eigenvector of the data covariance matrix $\mathbb{E}\{\mathbf{x}\mathbf{x}^\top\}$, under mild conditions [18, 146, 148] such as

- $\sum_{t=1}^{+\infty} \eta_t = +\infty$ and $\sum_{t=1}^{+\infty} \eta_t^2 < +\infty$.
- $(\lambda_1 - \lambda_2) > 0$ where λ_1 and λ_2 are the top two eigenvectors of $\mathbb{E}\{\mathbf{x}\mathbf{x}^\top\}$.
- Bounded higher-order moment of $\mathbf{x}$.

The recent work of [156] generalized Krasulina's method for $K > 1$ by proposing a new method named "Matrix Krasulina". This work shows both theoretically and experimentally that their method converges exponentially fast to the underlying subspace. In addition, they prove that the gradient variance of the proposed Matrix Krasulina method decays naturally over time and achieves exponential convergence rates even in the presence of noise.

As a consequence of its low computational complexity, Oja's method has attracted significant research interest resulting in thorough analysis of the algorithm for different step-sizes. Multiple works have derived theoretical convergence rates for Oja's method. Some methods propose variants of Oja's algorithm with constant step-size [157–159]. The algorithm in [157] is a variant of the Oja method with an added momentum term for accelerated convergence. The method employs mini-batch processing to achieve optimal convergence rates. Interestingly, this work demonstrates that naively adding momentum to Oja's algorithm usually fails in practice and claims that when the sample variance is bounded, momentum can achieve convergence acceleration. The work in [159] introduces a variant of the Oja's algorithm for time-series data. Data dependency is an issue with time-series data and to overcome it, their algorithm employs down-sampling to generate weakly dependant samples, thereby controlling the bias of the stochastic gradient caused by data dependency. The work in [158] provides an Oja variant for $K = 1$ component with theoretical convergence guarantees under the subgaussian distributional assumption on the data. The algorithm in [160] relies on the step-size format $\eta_t = \frac{c}{t}$ and offers an acceleration scheme for the block variant of Oja's method without the need for apriori information like the eigengap. Convergence analysis is

offered for the spiked covariance model and demonstrates empirically that their algorithm achieves fast convergence even when the original (unaccelerated) algorithms fail to converge.

More recently, the work in [161] proposed a new block variant of Oja's algorithm for streaming PCA that utilizes mini-batches of past data samples and name their method "History PCA". The algorithm is accompanied by formal convergence guarantees. History PCA uses past data measurements to achieve gradient variance reduction leading to faster convergence. Importantly, the step size is of the format $\eta_t = \frac{1}{t}$ and therefore it is hyper-parameter free. The work in [149] strives to achieve variance reduction in Oja's method by leveraging recent variance reduction techniques proposed for stochastic gradients in strongly convex problems [162]. The algorithm is referred to as VR (variance reduced)-PCA and is shown to converges exponentially fast to the optimal solution. The convergence analysis holds for any data distribution. The algorithm employs a fixed step-size and requires multiple passes over the entire dataset making it an offline algorithm. The approach in [163] presents streaming Oja-type algorithms that work well with noisy data. The also propose an algorithms to handle missing and/or outlier data along with noise. These algorithms employ step size format $\eta_t = \frac{c}{\sqrt{t}}$. An algorithm with non-zero approaching adaptive learning rate scheme for global convergence of the Oja method for $K = 1$ was proposed in [151]. Their algorithm relies on the step size $\eta_t = \frac{\zeta}{\mathbf{q}_t^\top \mathbf{x}_t \mathbf{x}_t^\top \mathbf{q}_t}$, where $0 < \zeta \leq 0.8$ is a tunable hyperparameter.

Additionally, the work in [151] identified the challenges of tuning the step-size in Oja-type algorithms including the need for apriori information such as the eigengap of the data covariance matrix and the influence of the correctness of the chosen step-size on convergence. Based on the AdaGrad algorithm [164], the work in [151] proposed an adaptive learning scheme for Oja PCA without the need to tune the step-size and was shown to work well in practice and under noise. The work in [165] proposed an Oja-type algorithm that is hyper-parameter free. The proposed algorithm leverages an implicit averaging scheme to achieve variance reduction and fast convergence, while being efficient. A detailed convergence analysis is offered in [166]. The work in [167] proposes an online approach for setting the learning rate for Oja-type algorithms. The presented method is capable of learning the step-size in an online fashion based on observed data. It relies on a burn-in scheme along with a probabilistic method to choose the step-size.

Block variants of Oja's algorithm have been proposed in [157, 168–173]. These methods leverage mini-batches of data as opposed to single samples to achieve faster convergence and higher noise immunity. However, these algorithms have shown to be computationally intensive [149]. The method of [168] is a block variant of Oja and offers convergence proofs for the spiked covariance model but it is demonstrated to work well in practice for general data. The method of [169] was

shown to work well with noisy data due to the use of mini batches and offers convergence analysis for a more general distribution-free model. The algorithm in [171] is able to select the batch size automatically leading to faster convergence.

Very recently, the work of [174] showed the equivalence between Oja's method and GROUSE [175]. They showed that at any iteration, given a step size for one of the algorithms, it is possible to construct a step size for the other algorithm, resulting in an identical update.

Moreover, multiple works have focused on developing convergence theory for stochastic PCA algorithms [146, 147, 152, 156, 166, 176–179]. The works in [146, 147, 152, 176] focus on $K = 1$, while the works in [152, 156, 177–179] provide convergence analysis for $K > 1$. Authors in [146] provide finite sample convergence rates for Oja and Krasulina and show that Krasulina's update can be seen as performing stochastic gradient descent on the Rayleigh quotient objective. The work of [147] provides an eigengap free convergence guarantee for $K = 1$ Oja PCA, when the step-size is appropriately chosen. The work in [178] proves global theoretical convergence for Oja method and the convergence rate is independent of the eigengap. They also provide an Oja variant algorithm that runs much faster than Oja and name it Oja^{++}. The algorithm in [152] presents an Oja-type algorithm for $K > 1$ and offers a local convergence result that is eigengap dependent. It requires an accurate initialization, with sufficient correlation with the optimal PCs. This happens with low probability if arbitrary initialization is used in high dimensions.

In a different line of research, stochastic optimization was leveraged for kernel PCA [180]. In [181], the authors propose a distributed averaging scheme for stochastic PCA, wherein they make use of multiple machines to estimate the top K-PCs. Since the estimate of each machine is prone to noise, they propose a distributed averaging scheme to average out the noise.

3.3 Contribution 1: Stochastic L1-PCA — One Component

While stochastic PCA has been well studied in the literature, the stochastic formulation of L1-PCA remains to date unexplored, despite its clear robustness in batch processing. Under this contribution, we formulate a novel stochastic version of L1-PCA, for one principal component, based on mean absolute projection maximization. Then, we propose an incremental algorithm for its solution, based on fundamental stochastic approximation theory [150]. Our method is accompanied by formal convergence guarantees and numerical studies that corroborate its corruption resistance. We present a step-by-step derivation of the proposed algorithm in the sequel.

3.3.1 Problem Formulation

First, we note that the metric of (2.3) can be equivalently rewritten as $\frac{1}{N}\|\mathbf{X}^\top\mathbf{q}\|_1$. Next, we note that $\frac{1}{N}\|\mathbf{X}^\top\mathbf{q}\|_1$ tends to $\mathbf{E}_\mathbf{x}\{|\mathbf{x}^\top\mathbf{q}|\}$, as N tends to infinity [182]. Thus, the stochastic L1-PC takes the form

$$\mathbf{q}_{\text{L1}} = \underset{\mathbf{q}\in\mathbb{S}_D}{\text{argmax}}\ \mathbb{E}_\mathbf{x}\{|\mathbf{x}^\top\mathbf{q}|\}. \tag{3.8}$$

Incorporating the norm constraint in the objective function, we can rewrite the stochastic L1-PC in (3.8) as the mean-absolute projection maximization (MaxAP)

$$\mathbf{z}_{\text{MaxAP}} = \underset{\mathbf{z}\in\mathbb{R}^D}{\text{argmax}}\ \mathbb{E}_\mathbf{x}\left\{\frac{|\mathbf{x}^\top\mathbf{z}|}{||\mathbf{z}||_2}\right\}, \tag{3.9}$$

noticing that $\mathcal{P}(\mathbf{z}_{\text{MaxAP}})$ solves (3.8). In order to ensure the definition and continuous differentiability of the function inside the expectation, we modify it as

$$M(\mathbf{x};\mathbf{z},\epsilon) := \frac{\sqrt{|\mathbf{x}^\top\mathbf{z}|^2+\epsilon}}{\sqrt{||\mathbf{z}||_2^2+\epsilon}}, \tag{3.10}$$

for some positive $\epsilon \ll 1$, and rewrite MaxAP as

$$\mathbf{z}_{\text{MaxAP}} = \underset{\mathbf{z}\in\mathbb{R}^D}{\text{argmax}}\ \mathbb{E}_\mathbf{x}\Big\{M(\mathbf{x};\mathbf{z},\epsilon)\Big\}. \tag{3.11}$$

Certainly, for $\epsilon = 0$, (3.11) coincides with (3.9). In the sequel, we focus on solving (3.11).

3.3.2 Proposed MaxAP Maximization

At the extrema of the ϵ-modified MaxAP metric in (3.11), it holds

$$\nabla_{\mathbf{z}^\top}\Big[\mathbb{E}_\mathbf{x}\Big\{M(\mathbf{x};\mathbf{z},\epsilon)\Big\}\Big] = \mathbf{0}. \tag{3.12}$$

In accordance with the regularity conditions on $M(\mathbf{x};\mathbf{z},\epsilon)$ [183], we interchange the gradient and expectation operators in (3.12) as

$$\mathbb{E}_\mathbf{x}\{L(\mathbf{x};\mathbf{z},\epsilon)\} = \mathbf{0}_D, \tag{3.13}$$

Proposed Stochastic MaxAP

Input: $\{\mathbf{x}_t\}_{t=1,2,\dots,N}$, $\{\eta_t\}_{t=1,2,\dots,N}$, $\mathbf{q}_0 \in \mathbb{R}^D$

1: for $t = 1, 2, \dots, N$
2: $\quad \mathbf{z}_t \leftarrow \mathbf{z}_{t-1} + \eta_t L(\mathbf{x}_t; \mathbf{z}_{t-1}, \epsilon)$
3: $\quad \mathbf{q}_t \leftarrow \mathcal{P}(\mathbf{z}_t)$
4: end for

Output: $\mathbf{q}_N$ (as estimate of the stochastic L1-PC $\mathbf{q}_{L_1}$)

Figure 3.1. Proposed method for PC estimation through MaxAP.

where

$$L(\mathbf{x}; \mathbf{z}, \epsilon) := \nabla_{\mathbf{z}^\top} M(\mathbf{x}; \mathbf{z}, \epsilon) \tag{3.14}$$

$$= \frac{\mathbf{x}^\top \mathbf{z}}{\sqrt{(|\mathbf{x}^\top \mathbf{z}|^2 + \epsilon)(\mathbf{z}^\top \mathbf{z} + \epsilon)}} \mathbf{x} - \frac{\sqrt{|\mathbf{x}^\top \mathbf{z}|^2 + \epsilon}}{(\sqrt{\mathbf{z}^\top \mathbf{z} + \epsilon})^3} \mathbf{z}. \tag{3.15}$$

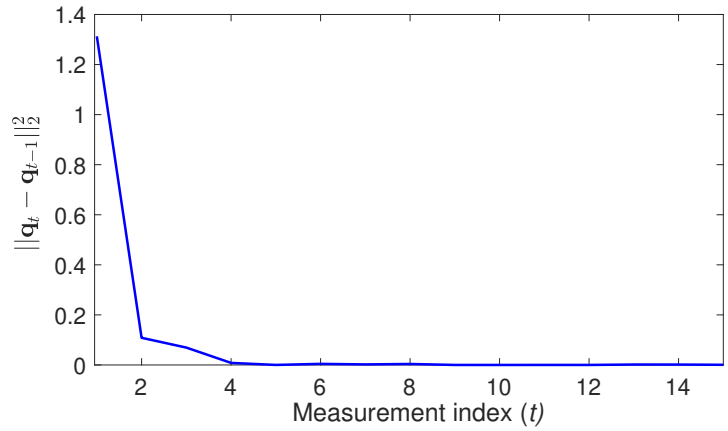
In view of (3.14), we first propose the iteration

$$\mathbf{z}_t = \mathbf{z}_{t-1} + \eta_t L(\mathbf{x}_t; \mathbf{z}_{t-1}, \epsilon), \ t = 1, 2, \dots, \tag{3.16}$$

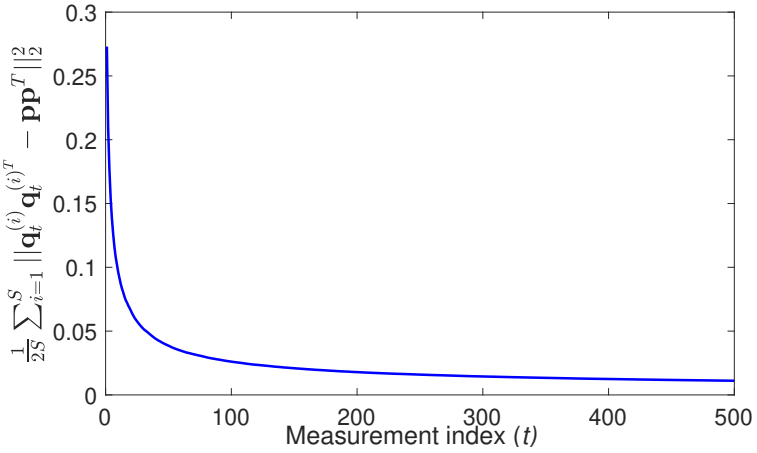
for step sizes $\{\eta_t\}$ that satisfy $\sum_{t=1}^{\infty} \eta_t = \infty$ and $\sum_{t=1}^{\infty} \eta_t^2 < \infty$ (e.g., $\eta_t = \frac{\gamma}{t}$, or $\eta_t = \frac{\gamma}{\sqrt{t}}$ for some constant $\gamma > 0$). First, we note that, due to the step-size conditions, for any given stream of data, the iteration converges in the argument: $\lim_{t\to\infty} \|\mathbf{z}_t - \mathbf{z}_{t-1}\|_2 = 0$. In addition, based on the fundamental stochastic approximation lemma presented by Robbins-Monro in [150], the iteration attains stochastic convergence, in the mean-square (m.s.) sense, to a root of (3.13). In fact, similar to gradient ascent for standard Rayleigh quotient maximization [184], (3.16) appears to stochastically converge to a maximizer of the ϵ-modified MaxAP in (3.11). Accordingly, the dependent PC sequence

$$\mathbf{q}_t = \mathcal{P}(\mathbf{z}_t), \ t = 1, 2, \dots, \tag{3.17}$$

would converge to a solution of stochastic L1-PCA in (3.8). For $\epsilon = 0$, the proposed iteration in (3.16) simplifies to $\mathbf{z}_t = \mathbf{z}_{t-1} + \eta_t \mathbf{P}_t \mathbf{v}_t \text{sgn}(\mathbf{v}_t^\top \mathbf{z}_{t-1})$, $t = 1, 2, \dots$, where $\mathbf{P}_t := \mathbf{I}_D - \mathbf{z}_{t-1} \frac{1}{\|\mathbf{z}_{t-1}\|_2^2} \mathbf{z}_{t-1}^\top = \mathbf{I}_D - \mathbf{q}_{t-1}\mathbf{q}_{t-1}^\top$ is the projection matrix to the nullspace of the previous PC and $\mathbf{v}_t := \mathbf{x}_t \frac{1}{\|\mathbf{z}_{t-1}\|_2}$ is the normalized new sample. A pseudocode for the proposed method is presented in Figure (3.1).



(a)



(b)

Figure 3.2. Convergence of the proposed method: (a) argument convergence in an arbitrary single stream of data; (b) estimated mean-square convergence to the stochastic L1-PC.

3.3.3 Experimental Studies

Synthetic Data

Convergence: We consider $D = 3$ and draw $N = 500$ data points from $\mathcal{N}(\mathbf{0}_3, \mathbf{C})$, where $\mathbf{C} = \begin{bmatrix} 2.05 & 1.05 & 1.08 \\ 1.05 & 0.7 & 0.31 \\ 1.08 & 0.31 & 0.97 \end{bmatrix}$. We set $\epsilon = 0$ and run the proposed iteration with a step size $\eta_t = \frac{1}{t}$. Then, we compute the change magnitude $||\mathbf{q}_t - \mathbf{q}_{t-1}||_2^2$ and plot it in Figure 3.2(a), versus t. An argument convergence to zero-change is observed after just 5 iterations. Then, we notice that, specifically for Gaussian data $\mathbf{x} \sim \mathcal{N}(\mathbf{0}_3, \mathbf{C})$, $|\mathbf{z}^\top \mathbf{x}|$ follows half-normal distribution with mean $\mathbb{E}_{\mathbf{x}}\{|\mathbf{x}^\top \mathbf{z}|\} = \sqrt{\frac{2}{\pi}\mathbf{z}^\top \mathbf{C}\mathbf{z}}$ [185]. Accordingly, the MaxAP metric becomes $\mathbb{E}_{\mathbf{x}}\{M(\mathbf{x}; \mathbf{z}, 0)\} = \sqrt{\frac{2}{\pi}}\sqrt{\frac{\mathbf{z}^\top \mathbf{C}\mathbf{z}}{\mathbf{z}^\top \mathbf{z}}}$ and the stochastic L1-PC that solves (3.8) coincides with the dominant eigenvector of $\mathbf{C}$, denoted here

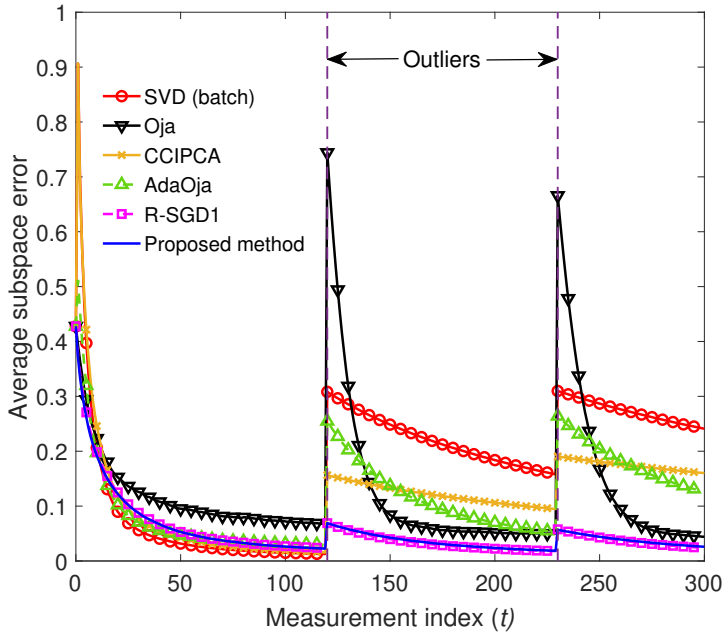
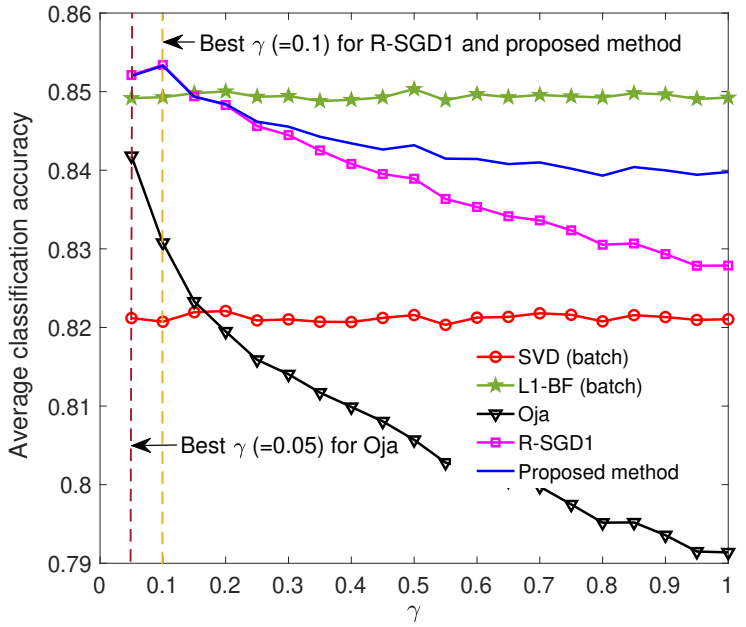


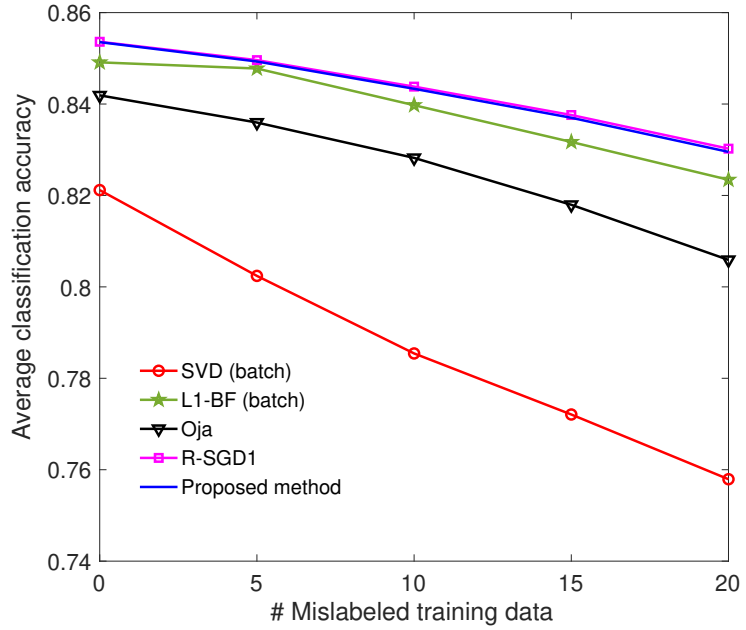
Figure 3.3. Average subspace error versus measurement index (t).

by $\mathbf{p}$. To demonstrate the stochastic convergence of the proposed iteration, we draw $S = 1000$ independent realizations of length-N data streams from the above Gaussian distribution and, for the i-th stream, we compute the proposed PC sequence $\{\mathbf{q}_t^{(i)}\}_{t=1,2,\ldots,N}$ according to (3.17). Then, we average-estimate the m.s. convergence metric as $\frac{1}{2S}\sum_{i=1}^{S}\|\mathbf{q}_t^{(i)}\mathbf{q}_t^{(i)^\top} - \mathbf{p}\mathbf{p}^\top\|_F^2$, for every t, where $\|.\|_F^2$ returns the squared Frobenius-norm of its matrix argument, and plot it in Figure 3.2(b). The stochastic convergence of the proposed iteration is clearly documented.

Subspace estimation: We draw $N = 300$ independent points from $\mathcal{N}(\mathbf{0}_3, \mathbf{C})$, where $\mathbf{C} = \begin{bmatrix} 7 & 7 & 6 \\ 7 & 10 & 9 \\ 6 & 9 & 13 \end{bmatrix}$, and form data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{3\times 300}$. Then, we add benign zero-mean white Gaussian noise (AWGN) from $2\mathcal{N}(0,1)$ to every entry of $\mathbf{X}$. In addition, we corrupt $\mathbf{x}_{120}$ and $\mathbf{x}_{230}$ with outlying corruption drawn from $\mathcal{N}(\mathbf{0}_3, \mathbf{C}_o)$, where $\mathbf{C}_o = \begin{bmatrix} 1585 & 1039 & -76 \\ 1039 & 1311 & -61 \\ -76 & -61 & 104 \end{bmatrix}$, such that the dominant eigenvector of the outlier covariance matrix $\mathbf{C}_o$, $\mathbf{p}_o$, makes an angle of about 45° with the dominant eigenvector of the nominal covariance matrix $\mathbf{C}$, $\mathbf{p}$. Then, we apply the proposed method to estimate $\mathbf{p}$, from the corrupted data stream. We repeat this task for $S = 10^4$ independent data realizations and plot in Figure 3.3 the average-estimated mean subspace error $\frac{1}{\mathrm{S}}\sum_{i=1}^{\mathrm{S}}||\mathbf{q}_t^{(i)}\mathbf{q}_t^{(i)^\top} - \mathbf{p}\mathbf{p}^\top||_F^2$, versus t. Alongside the proposed algorithm, we also plot the performance of SVD (batch processing –joint calculation on $[\mathbf{X}]_{:,1:t}$ at the t-th iteration), Oja's standard stochastic PCA algorithm of [18], candid covariance-free incremental PCA (CCIPCA) [165], AdaOja [151], and R-SGD1 [27]. For fairness we tune the step sizes of Oja, R-SGD1, and the proposed algorithm to $\eta_t = \frac{0.05}{\sqrt{t}}$. We notice that for $t < 120$, all methods perform similarly well and returned subspaces nearly converge to the span of $\mathbf{p}$. For $t \geq 120$, we observe that all L2-based methods



(a)



(b)

Figure 3.4. Average classification accuracy versus (a) γ and (b) number of mislabeled training points.

(SVD, Oja, CCIPCA, and AdaOja) are significantly affected by the corruptions. On the other hand, the robust solver R-SGD1 [27] and the proposed algorithm show similarly good corruption resistance, maintaining low subspace error.

Wisconsin Breast Cancer Dataset

In this experiment, we perform nearest subspace (NS) classification on the Wisconsin breast cancer dataset [186, 187], which contains ($D = 30$)-dimensional measurements on healthy and unhealthy cell nuclei. The features of each sample describe characteristics of the cell nuclei present in the digitized image of a fine needle aspirate (FNA) of a breast mass. The dataset consists of 569 samples in total, with 212 samples belonging to the benign class and 357 samples belonging to the malignant class. We first split the dataset into 85% training and 15% testing data. In order to identify a preferred learning rate for the stochastic methods, we vary $\gamma \in \{0.05, 0.05, 1\}$ and perform NS classification using the final element of each stochastic PC sequence. Specifically, we first estimate for each method the individual PCs of training data from class 1 (healthy) and class 2 (unhealthy). Next, for each testing point, we measure its squared projection error for each PC and assign it to the class with the PC that attained the lowest error. We repeat this experiment over 10^4 independent data splits and, in Figure 3.4(a), we plot the average classification accuracy versus γ (including the batch methods as benchmarks). For Oja's method, $\gamma = 0.05$ is preferred, while for R-SGD1 and the proposed method $\gamma = 0.1$ is preferred. Interestingly, we notice that L1-BF outperforms SVD. In addition, we observe that the proposed method is significantly more robust than its counterparts against inferior choices of γ.

Next, we tune γ to the preferred values found above and mislabel a portion of the training data. We repeat the experiment over 10^3 independent data splits and plot in Figure 3.4(b) the average NS classification accuracy versus number of mislabeled training data points from each class. First, we observe that as the number of mislabelings increases, the performance of SVD drops significantly. On the other hand, batch L1-BF, R-SGD1, and the proposed algorithm exhibit similar robustness against mislabeling.

3.4 Contribution 2: Generalized Framework for Stochastic PCA via Barron Loss — Multiple Components

In this line of research, we extend the stochastic MaxAP algorithm we presented in the previous section for $K = 1$ to multiple components ($K \geq 1$) by employing the robust Barron loss [26]. Specifically, we propose a generalized framework for Oja-type PCA algorithms, which includes a family of algorithms with varying degree of outlier resistance. Our framework includes the fundamental Oja algorithm which is sensitive to outliers and the robust stochastic PCA algorithm of [27] as special cases. We note that although Oja's method is outlier-sensitive, while processing nominal data, it is shown to converge faster to the underlying subspace compared to its robust counterparts. We show in theory and practice that using the Barron loss to derive stochastic PCA algorithms offers the user the flexibility to control the trade-off between faster convergence versus higher outlier resistance by controlling the robustness parameter α. Since our framework is based on the robust Barron loss, we provide a brief overview of Barron loss in the sequel.

A general and adaptive loss function with an in-built robustness parameter was proposed in [26]. Similar to the work in [188], we refer to this loss function as the Barron loss, after the last name of the paper's author. Virtually all machine learning applications require algorithms to be significantly less influenced by outliers compared to nominal data. To this end, multiple robust loss functions have been proposed, including the L1-loss and the Huber loss [189]. We read in [26] that the proposed generalized loss function is a superset of many existing robust and non-robust losses. The expression for Barron loss is given by

$$f(x;\alpha,\beta) = \frac{|\alpha-2|}{\alpha}\left(\left(\frac{(\frac{x}{\beta})^2}{|\alpha-2|}+1\right)^{\frac{\alpha}{2}}-1\right), \tag{3.18}$$

where α is the shape parameter that controls robustness and $\beta > 0$ is the scale parameter that controls the size of the quadratic bowl near $x = 0$. We present a visual representation of the Barron loss curves for different α values in Figure 3.5 (a) and the corresponding derivatives in Figure 3.5 (b). The Barron loss in (3.18) approaches the standard L2-loss in limit as α tends to 2 and for $\alpha = 1$, it simplifies to the smoothed L1-loss or the pseudo-Huber loss. For $\alpha = 0$, it yields the Cauchy loss, $\alpha = -2$ results in the Geman-McClure loss, and for $\alpha = -\infty$, it simplifies to the Welsch loss in limit. Although Barron loss cannot yield the L1-loss for any value of α, it approximates the L1-loss for $\alpha = 1$, when $x >> \beta$, that is, $f(x;1,\beta) \approx \frac{|x|}{\beta} - 1$ [26]. Some properties of Barron loss make it well-suited for use in gradient based optimization. For instance, Barron loss is smooth over its input and parameters, its value is zero at the origin and increases monotonically with respect

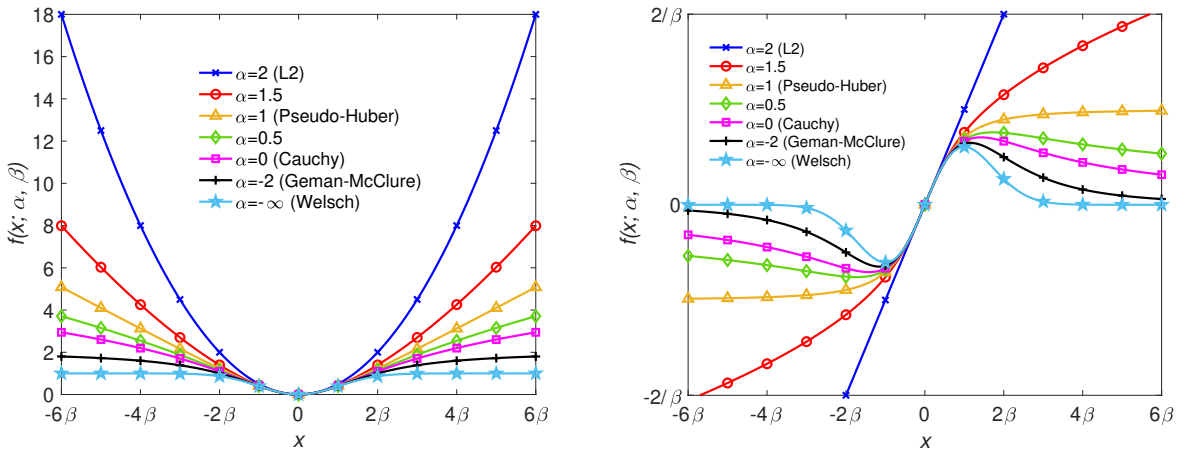


Figure 3.5. (a) Barron loss curves and (b) their corresponding gradients for various values of the robustness parameter α. Note that different values of α show how differently the loss curves and their corresponding gradients evolve. Specific values of α result in the well-known loss functions, for example, $\alpha = 2$ yields the L2 loss and $\alpha = 1$ results in the pseudo-Huber loss.

to $|x|$. Owing to these useful properties, Barron loss has been successfully applied in multiple machine learning applications [26, 188, 190] and in this work, we employ it to derive a generalized framework for stochastic PCA algorithms offering varying robustness against outliers. Specifically, we propose a generalized framework for Oja-type stochastic PCA algorithms which includes a family of algorithms whose outlier-resistance can be controlled via the robustness parameter α. In the following, we present the problem formulation and derivation of our generalized framework.

3.4.1 Barron Loss

3.4.2 Problem Formulation

In our derivation, without the loss of generality, we fix the scale parameter of the Barron loss to $\beta = 1$. Nevertheless, our analysis holds for general β values. Given data measurements $\mathbf{x}$ arriving from a distribution $\mathcal{D}$, we strive to find an orthonormal basis of size $D \times K$ that optimizes

$$\underset{\mathbf{Q}\in\mathbb{S}_{D,K}}{\operatorname{argmax}} \underset{\mathbf{x}\sim\mathcal{D}}{\mathbb{E}} \left\{ f\big(\|\mathbf{Q}^\top\mathbf{x}\|_F; \alpha, \beta = 1\big) \right\}, \tag{3.19}$$

where $f(\|\mathbf{Q}^\top\mathbf{x}\|_F; \alpha, \beta = 1) = \frac{|\alpha-2|}{\alpha}\left(\left(\frac{\|\mathbf{Q}^\top\mathbf{x}\|_F^2}{|\alpha-2|} + 1\right)^{\frac{\alpha}{2}} - 1\right)$ is the Barron loss with objective $\|\mathbf{Q}^\top\mathbf{x}\|_F$. Hereon, for brevity in notation, we drop β in the Barron loss notation for the proposed generalized stochastic PCA framework as it is set to one and does not affect the loss function. We

note that different values of α yield different step-sizes for the Oja algorithm and therefore different amount of robustness. Specifically, lower values of α offer higher outlier-resistance compared to higher α values. For example, when $\alpha = 2$, the objective in (3.19) simplifies to $f(\|\mathbf{Q}^\top \mathbf{x}\|_F; \alpha, \beta = 1) = \frac{1}{2}\|\mathbf{Q}^\top \mathbf{x}\|_F^2$ which is the L2-loss and when $\alpha = 1$, $f(\|\mathbf{Q}^\top \mathbf{x}\|_F; \alpha, \beta = 1) = (\|\mathbf{Q}^\top \mathbf{x}\|_F^2 + 1)^{\frac{1}{2}} - 1$, which is the of pseudo-Huber loss. In the sequel, we present the stochastic gradient of the objective in (3.19) and use it with projected gradient ascent to estimate the $\mathbf{Q} \in \mathbb{S}_{D,K}$ that solves (3.19).

3.4.3 Proposed Generalized Framework

The expected value in (3.19), $\mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\left\{f\big(\|\mathbf{Q}^\top \mathbf{x}\|_F; \alpha, \beta = 1\big)\right\} = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\left\{\frac{|\alpha-2|}{\alpha}\left(\left(\frac{\|\mathbf{Q}^\top \mathbf{x}\|_F^2}{|\alpha-2|} + 1\right)^{\frac{\alpha}{2}} - 1\right)\right\}$ can be sample average approximated as $\frac{1}{N}\sum_{t=1}^{N}\left\{\frac{|\alpha-2|}{\alpha}\left(\left(\frac{\|\mathbf{Q}^\top \mathbf{x}_t\|_F^2}{|\alpha-2|} + 1\right)^{\frac{\alpha}{2}} - 1\right)\right\}$, whose gradient with respect to $\mathbf{Q}$ can be shown to be $\frac{1}{N}\sum_{t=1}^{N}\left(\frac{\|\mathbf{Q}^\top \mathbf{x}_t\|_F^2}{|\alpha-2|} + 1\right)^{\frac{\alpha}{2}-1}\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}$. In a practical streaming application, one may not have access to all the N measurements to begin with as they may arrive incrementally over time. In order to process streaming data efficiently, similar to Oja's method [18], we approximate the batch gradient by the stochastic counterpart that relies on just one sample at each update index t, resulting in the stochastic gradient $\left(\frac{\|\mathbf{Q}^\top \mathbf{x}_t\|_F^2}{|\alpha-2|} + 1\right)^{\frac{\alpha}{2}-1}\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}$. We employ this single-sample approximated gradient in projected gradient descent to obtain the generalized framework

$$\mathbf{Q}_t \leftarrow \mathcal{P}\bigg(\mathbf{Q}_{t-1} + \eta_t\bigg(\frac{\|\mathbf{Q}_{t-1}^\top \mathbf{x}_t\|_F^2}{|\alpha-2|} + 1\bigg)^{\frac{\alpha}{2}-1}\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}_{t-1}\bigg), \tag{3.20}$$

where the step-size (or learning rate) $\eta_t = \frac{\gamma}{t}$, such that $\sum_{t=1}^{+\infty}\eta_t = +\infty$, while $\sum_{t=1}^{+\infty}\eta_t^2 < +\infty$ to ensure convergence [18]. Defining $\tilde{\eta}_t = \eta_t\left(\frac{\|\mathbf{Q}_{t-1}^\top \mathbf{x}_t\|_F^2}{|\alpha-2|} + 1\right)^{\frac{\alpha}{2}-1}$, we obtain a simplified version of (3.20) as

$$\mathbf{Q}_t \leftarrow \mathcal{P}\bigg(\mathbf{Q}_{t-1} + \tilde{\eta}_t\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}_{t-1}\bigg), \tag{3.21}$$

which resembles Oja's algorithm in (3.6) with step size $\tilde{\eta}_t$. That is, the proposed update generalizes Oja's algorithm with a different step-size at each update index t that is a function of γ, $\mathbf{Q}_{t-1}$, $\mathbf{x}_t$, and α. For instance, by setting $\alpha = 2$, the update in (3.20) simplifies to Oja's update exactly, but may be sensitive to outliers due to the squared emphasis on each measurement, including outliers.

Proposed Generalized Framework for Oja-type Stochastic PCA

Input: $\{\mathbf{x}_t\}_{t=1,2,\ldots,N}$, $\gamma \in \mathbb{N}$, $\mathbf{Q}_0 \in \mathbb{S}_{D,K}$, and $\alpha \in \mathbb{R}$

1: for $t = 1, 2, \ldots, N$

2: $\quad \mathbf{Q}_t \leftarrow \mathbf{Q}_{t-1} + \frac{\gamma}{t}\left(\frac{\|\mathbf{Q}_{t-1}^\top \mathbf{x}_t\|_F^2}{|\alpha-2|} + 1\right)^{\frac{\alpha}{2}-1} \mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}$

3: $\quad \mathbf{Q}_t \leftarrow \mathtt{QR}(\mathbf{Q}_t)$

4: end for

Output: $\mathbf{Q}_N$

Figure 3.6. Proposed generalized algorithm for Oja-type stochastic PCA.

Values of $\alpha < 2$, impart robustness by down-weighting the influence of outliers compared to inliers. Specifically, by simply tuning the value of α, the user can control the robustness offered by the update in (3.20) per application. Another way to perceive this is that the proposed framework can impart robustness to Oja's algorithm by choosing a different step-size at each update index. We present a pseudo-code for the proposed generalized Oja-type stochastic PCA algorithm in Figure 3.6.

A previous work in [27] noticed the sensitivity of Oja's method to outliers and proposed a robust counterpart of Oja's algorithm named R-SGD1, with the update $\mathbf{Q}_t \leftarrow \mathcal{P}(\mathbf{Q}_{t-1} + \eta_t \frac{\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}}{\|\mathbf{Q}_{t-1}^\top \mathbf{x}_t\|_F})$. Interestingly, our general algorithm for $\alpha = 1$ obtains the update $\mathbf{Q}_t \leftarrow \mathcal{P}\big(\mathbf{Q}_{t-1} + \eta_t \frac{\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}}{(\|\mathbf{Q}_{t-1}^\top \mathbf{x}_t\|_F^2 + 1)^{\frac{1}{2}}}\big)$ and in practice, it holds that $\|\mathbf{Q}_{t-1}^\top \mathbf{x}_t\|_F^2 >> 1$, and therefore our algorithm further simplifies to $\mathbf{Q}_t \leftarrow \mathcal{P}(\mathbf{Q}_{t-1} + \eta_t \frac{\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}}{\|\mathbf{Q}_{t-1}^\top \mathbf{x}_t\|_F})$, coinciding with R-SGD1 [27] as also verified by our experimental studies. Therefore, our framework includes the fundamental Oja's algorithm as well as the robust counterpart R-SGD1 [27], along with any variants in between. Moreover, for values of $\alpha < 1$, we can obtain algorithms more robust compared to R-SGD1 while processing outlier-corrupted data and for values of $\alpha > 2$, we can obtain algorithms that converge faster on average under nominal conditions. In addition, our experimental studies demonstrate that the variance of the stochastic gradient of Oja's algorithm is high as it takes a larger step in the direction of the noisy gradient compared to robust counterparts as noted in [149, 156, 157]. The size of the step taken by the proposed framework in the direction of the noisy gradient can be controlled via the robustness parameter α and therefore for low values of α, our algorithm's variance is minimal and converges to low subspace error even with a single pass on the data.

Outlier-resistance of the proposed framework for $\alpha < 2$. We observe that Oja's iteration (ignoring the orthogonalization) can be rewritten as $\mathbf{Q}_t \leftarrow \mathbf{Q}_{t-1} + \eta_t \frac{\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}}{\|\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}\|_2} \|\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}\|_2$, where $\frac{\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}}{\|\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}\|_2}$ is the gradient direction (with unit magnitude) and $\|\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}\|_2$ is the gradient magnitude, which is upper-bounded by $\|\mathbf{x}_t \mathbf{x}_t^\top\|_2 \|\mathbf{Q}_{t-1}\|_2$. Since $\|\mathbf{Q}_{t-1}\|_2 = 1$, $\|\mathbf{x}_t \mathbf{x}_t^\top \mathbf{Q}_{t-1}\|_2 \leq \|\mathbf{x}_t \mathbf{x}_t^\top\|_2 = \|\mathbf{x}_t\|_2^2$. That is, the magnitude of Oja's step in the direction of the gradient is at most quadratic in the magnitude of $\mathbf{x}_t$, which may yield a large value if $\mathbf{x}_t$ is an outlier, thereby

misleading the updates in the direction of the outlier with a large step. Similarly, the update of the proposed method can be expressed as $\mathbf{Q}_t \leftarrow \mathbf{Q}_{t-1} + \eta_t \frac{\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}_{t-1}}{\|\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}_{t-1}\|_2}\left(\frac{\|\mathbf{Q}_{t-1}^\top\mathbf{x}_t\|_F^2}{|\alpha-2|}+1\right)^{\frac{\alpha}{2}-1}\|\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}_{t-1}\|_2$. We observe that the gradient direction is the same as that of Oja, but the gradient magnitude, $\left(\frac{\|\mathbf{Q}_{t-1}^\top\mathbf{x}_t\|_F^2}{|\alpha-2|}+1\right)^{\frac{\alpha}{2}-1}\|\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}_{t-1}\|_2$, is different. In the sequel, we show that the gradient magnitude of the proposed update is less than that of Oja for $\alpha < 2$, thereby making the proposed method more resistant to outliers. Firstly, we note that for $\alpha < 2$, $\frac{\alpha}{2} - 1 < 0$ and $\|\mathbf{Q}_{t-1}^\top\mathbf{x}_t\|_2^2 >> 1$ in practice, resulting in a gradient magnitude of $\frac{|\alpha-2|^{|\frac{\alpha}{2}-1|}\|\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}_{t-1}\|_2}{\|\mathbf{Q}_{t-1}^\top\mathbf{x}_t\|_2^{2|\frac{\alpha}{2}-1|}}$. As stated earlier, $\|\mathbf{x}_t\mathbf{x}_t^\top\mathbf{Q}_{t-1}\|_2 \leq \|\mathbf{x}_t\|_2^2$ and therefore the denominator, $\|\mathbf{Q}_{t-1}^\top\mathbf{x}_t\|_2^{2|\frac{\alpha}{2}-1|} \leq \|\mathbf{x}_t\|_2^{2|\frac{\alpha}{2}-1|}$. For $\alpha < 2$, $2|\frac{\alpha}{2}-1| > 0$, consequently the influence of $\mathbf{x}_t$ on the gradient magnitude is less than quadratic and this emphasis reduces as α is further decreased. In other words, the magnitude of the gradient step is dampened by $\|\mathbf{Q}_{t-1}^\top\mathbf{x}_t\|_2^{2|\frac{\alpha}{2}-1|}$, thereby imparting robustness against (high magnitude) outliers. For example, if $\alpha = 1$, $\|\mathbf{Q}_{t-1}^\top\mathbf{x}_t\|_2^{2|\frac{\alpha}{2}-1|} \leq \|\mathbf{x}_t\|_2$ and the resulting gradient magnitude is at most linear in $\mathbf{x}_t$, thereby offering robustness against outliers.

To summarize, our generalized framework offers the user the flexibility to choose an appropriate value of α per application to achieve a sweet-spot between faster convergence and higher outlier resistance. Additionally, our algorithm demonstrates intrinsically low gradient variance for smaller values of α assisting in faster convergence under noise. Next, we demonstrate the efficacy of the proposed algorithm in terms of convergence to low subspace error and outlier resistance for various values of α on synthetic and multiple real-world datasets.

3.4.4 Experimental Studies

Synthetic Data Experiments

Convergence. We demonstrate the convergence of the proposed algorithm in terms of subspace estimation by forming a data matrix $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^\top \in \mathbb{R}^{D=3\times N=750}$, where $\mathbf{U} \in \mathbb{S}_{D,D}$, $\Sigma = \texttt{diag}(65, 30, 10)$, $\mathbf{V} \in \mathbb{S}_{N,D}$. We compute the average subspace error

$$\frac{1}{2K}\frac{1}{S}\sum_{i=1}^{S}\|\mathbf{P}\mathbf{P}^\top - \mathbf{Q}_t^{(i)}\mathbf{Q}_t^{(i)\top}\|_2^2, \tag{3.22}$$

and plot is versus measurement index t in Figure 3.7, where $S = 2000$ is the number of realizations over which the average is computed, $K = 2$ is the number of PCs we seek, $\mathbf{P} = \mathbf{U}_{:,1:K}$, and $\mathbf{Q}_t^{(i)}$ is

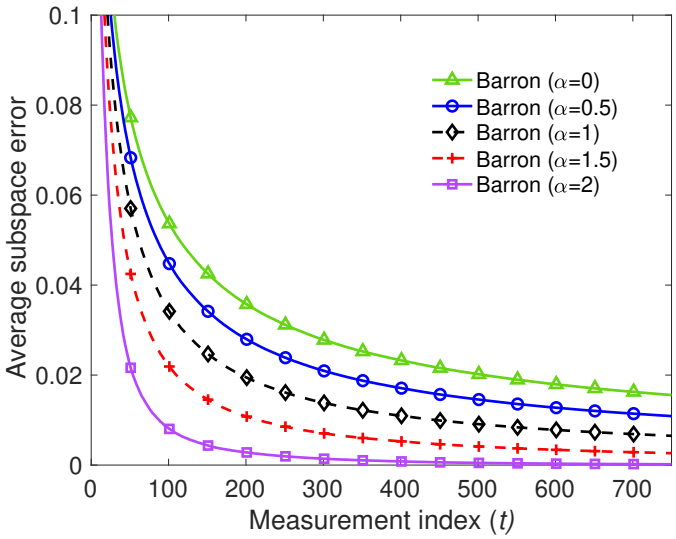


Figure 3.7. Synthetic data. Empirical convergence of the proposed stochastic Barron PCA algorithm for different values of the robustness parameter α.

the estimated subspace at the i-th realization after the t-the measurement is processed.

We set the learning rate $\gamma = 5$, $\beta = 1$, and initialize all algorithms arbitrarily unless specified otherwise. We observe from Figure 3.7 that the proposed algorithm converges to low subspace error for various values of α. We note that the rate of convergence to low subspace error is faster for larger values of α compared to smaller values because smaller α values impart robustness by being reluctant to quicker updates of the algorithm.

Outlier resistance. In this experiment we demonstrate the outlier resistance of the proposed algorithm for various α values compared to state-of-the-art stochastic PCA algorithms, a robust variant (R-SGD1 [27]), and L1-Oja whose update is of the form $\mathbf{Q}_t \leftarrow \mathcal{P}\big(\mathbf{Q}_{t-1} + \eta_t \mathbf{x}_t \mathtt{sign}(\mathbf{x}_t^\top \mathbf{Q}_{t-1})\big)$. Notice that the update of L1-Oja corresponds to the objective $\|\mathbf{Q}^\top \mathbf{x}\|_1$, which is robust against outliers because of the L1-norm. To this end, we form the data matrix $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^\top \in \mathbb{R}^{D=4\times N=1000}$, where $\mathbf{U} \in \mathbb{S}_{D,D}$, $\mathbf{\Sigma} = \mathtt{diag}(100, 95, 1, 0.25)$, and $\mathbf{V} \in \mathbb{S}_{N,D}$. We add additive white Gaussian noise (AWGN) to $\mathbf{X}$ with signal-to-noise-ratio (SNR) 20dB. We corrupt measurements $t = 350, 750$ by replacing nominal data with outliers chosen arbitrarily from the outlier data matrix $\mathbf{X}_o = \mathbf{U}_o\mathbf{\Sigma}_o\mathbf{V}_o^\top$, where $\mathbf{U}_o$, $\mathbf{V}_o$ are drawn from the Stiefel manifold of conforming sizes, and $\mathbf{\Sigma}_o = \mathtt{diag}(600, 850, 1250, 1750)$. This yields an average normalized subspace distance of about 0.3 and an average angle of about 58° between $\mathbf{U}_{:,1:K}$ and $\mathbf{U}_{o_{:,1:K}}$, resulting in the outliers being significantly far from nominal data. We compute the average subspace error for $K = 2$ versus measurement index (t) over 10^4 realizations and plot it in Figure 3.8(a) and observe that when the data is nominal, all methods converge quickly to low subspace error. However, when outliers are encountered at fixed indices $t = 350, 750$, we observe that L2-norm based methods Oja [148]

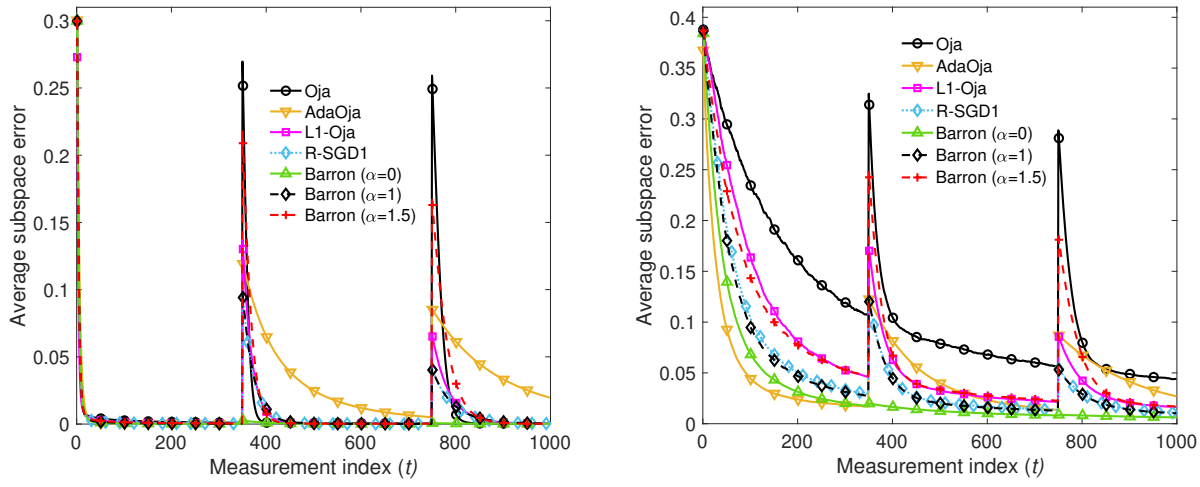


Figure 3.8. Results on synthetic data with fixed outlier indices $t = 350, 750$. Average subspace error versus measurement index t for (a) SNR = 20dB and (b) SNR = 1dB.

and Ada-Oja [151] are significantly affected, while L1-Oja demonstrates more outlier resistance compared to Oja and AdaOja. Proposed method with larger α values are affected more by outliers. For example, for $\alpha = 1.5$, the proposed method is significantly affected by outliers whereas for $\alpha = 0$, the proposed method is unaffected by outliers and maintains the lowest subspace error across the board. Interestingly, as previously derived in theory, the proposed method with $\alpha = 1$ performs very similar to a state-of-the-art robust stochastic PCA method R-SGD1 [27].

We repeat the experiment with low SNR, specifically 1 dB and plot the results in Figure 3.8(b). We observe that Oja converges slowly to low subspace error due to noisy gradients at each update and it is significantly affected by the outliers fixed at indices $t = 350, 750$. On the other hand, we notice that AdaOja quickly converges to low error owing to its adaptive step size that is able to account for lower SNR, however, it is significantly affected by outliers. The proposed method for lower α values converges faster to lower subspace error and demonstrates more resistance against the outliers compared to higher α values. Specifically, for $\alpha = 0$, the proposed method quickly converges to low subspace error and remains there regardless of the occurrence of outliers, whereas for $\alpha = 1.5$, the proposed method's performance approaches that of Oja – slow convergence and high outlier susceptibility.

Next, instead of fixing the outlier indices beforehand, we let each measurement be corrupted by outlier with probability 2.5% and plot the resulting average subspace error versus update index in Figure 3.9(b). We set the SNR to 4dB in this study and observe that Oja's method converges slowly and to higher subspace error because of the probability of each measurement being corrupted by an outlier, albeit with a small probability. AdaOja performs similarly by converging to higher subspace

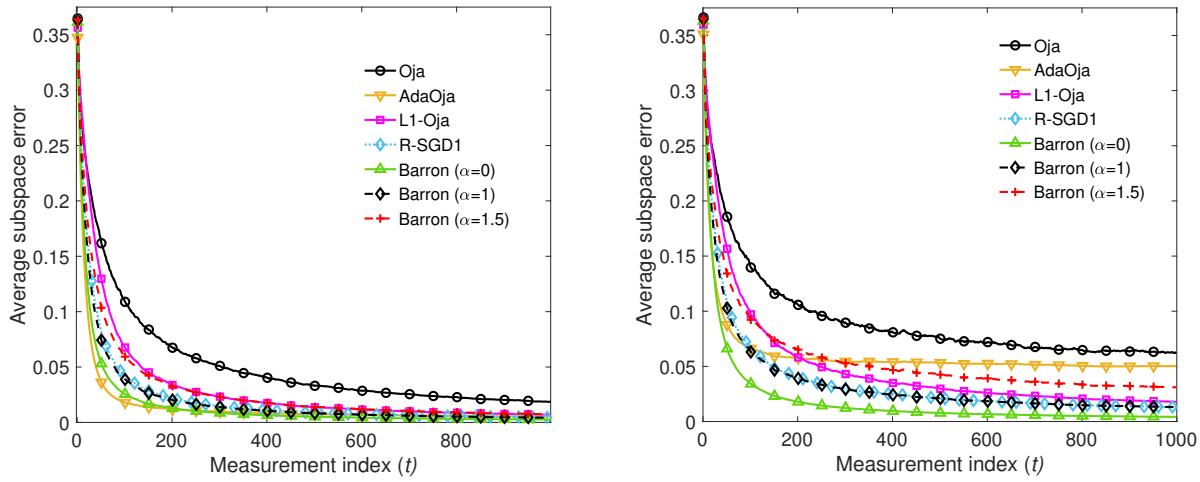


Figure 3.9. Results on synthetic data with SNR=4dB and each measurement corrupted by an outlier with some probability. Average subspace error versus measurement index t for probability of outlier corruption per frame (a) = 0 and (b) = 2.5%.

error on average. L1-Oja is able to achieve low subspace error owing to its outlier resistance, but converges slowly. The proposed method for lower α values converge to lower subspace errors faster compared to higher α values, demonstrating the superior outlier resistance of the proposed method for low α values. For the sake of comparison, we present the performance of all algorithms with zero probability of corruption per measurement in Figure 3.9(a). The comparison between 3.9(a) and 3.9(b) clearly depicts the outlier sensitivity of Oja, AdaOja, and the proposed method for larger α values, while reinforcing the outlier resistance of the proposed method with low α values.

Our synthetic data studies demonstrate the generality of the proposed framework in terms of the speed of convergence to the underlying ground-truth subspace and the level of outlier resistance for different values of α. In appendix A, we offer a thorough comparison of our method with other state-of-the-art online non-Oja-type robust and non-robust algorithms.

Real-world Data Experiments

Glare/shadow artifact removal in face images. In this experiment, we operate on the Extended Yale Face Database B [191]. The dataset consists of face images of 28 subjects captured under 64 varying illuminations in 9 different poses. We select the face images with front pose of individuals under 64 varying illuminations, resize each image to 173×152 and form the data matrix by vectorizing each image and stacking them as the columns of $\mathbf{X}$ of size 26296×64. We note that the face images (with the same pose) form a lower-dimensional subspace while the varying

Figure 3.10. Glare/shadow removal results. Comparison with state-of-the-art Oja type methods. Rows 1 - 3 correspond to subject 05, rows 4 - 6 correspond to subject 09, and rows 7 - 9 correspond to subject 18 respectively. For each subject, we demonstrate the glare/shadow artifact removal at frame indices $t = 40$, 46, and 54.

illumination resulting in glare and shadows on the face constitute sparse outliers [40]. We aim to perform glare/shadow of any image vector $\mathbf{x}_t$, where $t = \{1, 2, \dots 64\}$ by projecting it onto the subspace estimated after processing the t-th image vector as $\mathbf{Q}_t\mathbf{Q}_t^\top\mathbf{x}_t$, with a learning rate $\eta_t = 10^{-3}/t$. We use this learning rate for all out studies on real-world datasets. It is known that

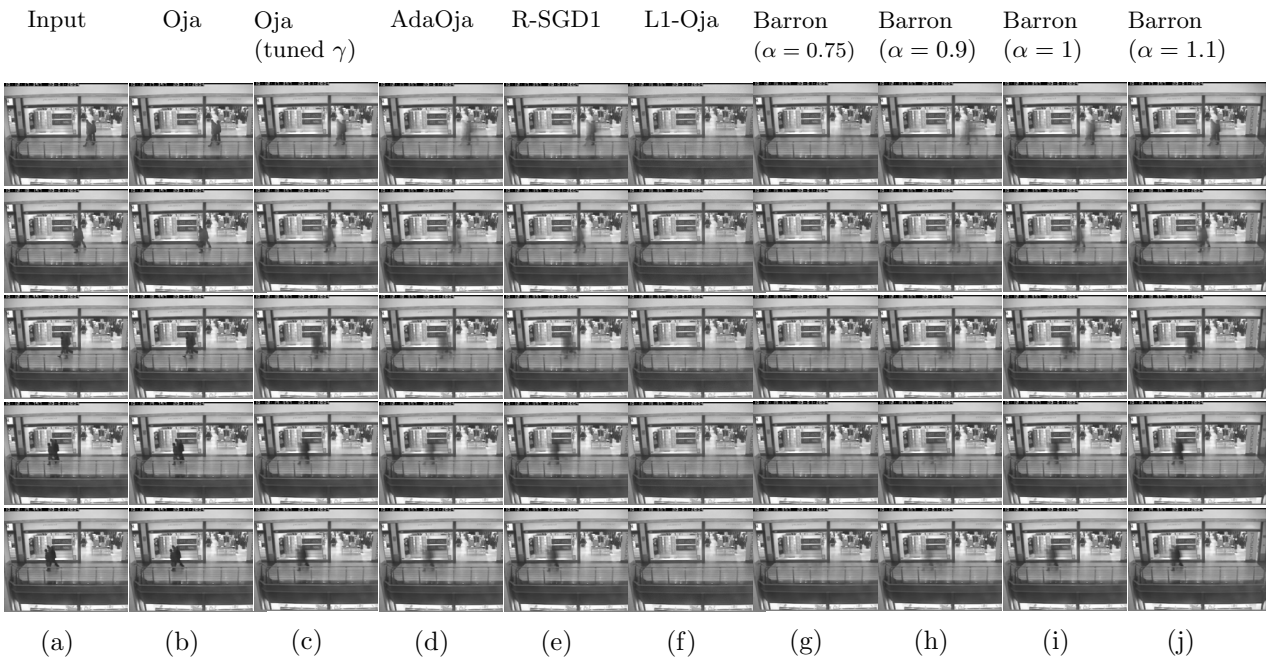


Figure 3.11. Video background/foreground separation. Comparison with state-of-the-art Oja type methods. Rows 1 - 5 correspond to frame indices $t = 100$, 120, 140, 160, and 165.

around a subspace of $K = 9$ well approximates the face images under varying illumination [27, 192] and with this motivation, we use $K = 9$ in our study. We present the glare removal results for frame indices $t = 40$, 46, 54 of subjects 5, 9, and 18 in Figure 3.10 and observe that L2-norm based methods Oja [148] and Ada-Oja [151] retain most of the glare/shadows on the face. The robust counterpart R-SGD1 [27] performs better than Oja and AdaOja, similar to the proposed method with $\alpha = 1$. The proposed method with low α values result in effective glare/shadow removal and smoothen the features of the face. By tuning the α values, we are able to retain more facial features versus more glare. Moreover, in order to demonstrate our claim of making Oja's method resistant to outliers by choosing a different step-size per update, we hand-tune the step-size $\eta_t = 10^{-7}/t$ for Oja's method and observe that the glare/shadow artifact removal performance is much improved.

Foreground/background separation in surveillance videos. Video foreground extraction is an integral part of many applications including real-time gesture/object identification, human-computer interaction, security surveillance, and traffic monitoring [193]. The static background of all frames of the video spans the common nominal subspace, whereas moving components in the foreground such as people and vehicles constitute intermittent outliers. Common foreground/background separation methods first estimate the underlying background of the video and then subtract it from the original frame to obtain any foreground objects. In this experiment, we make use of a video recorded at a shopping center in Portugal, made available in the benchmark CAVIAR dataset [144]. The video captures the frontal video of the mall corridor with stationary background and

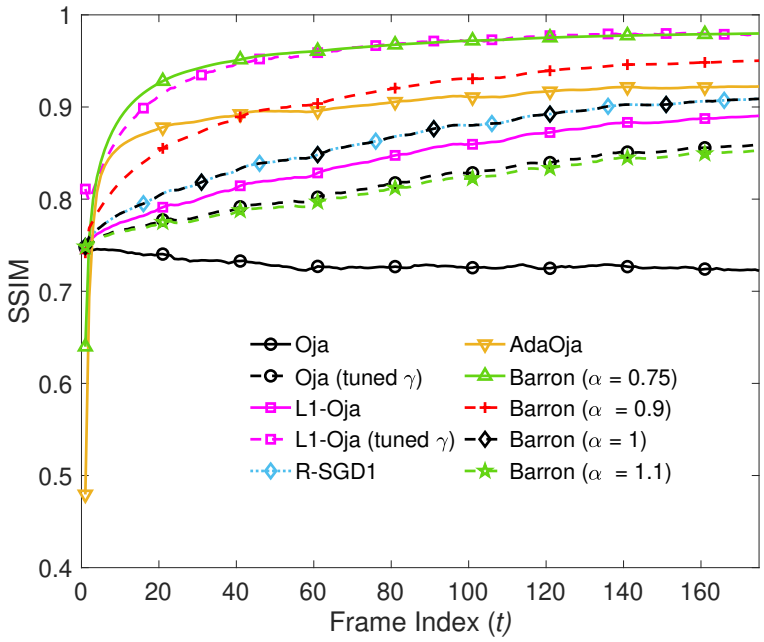


Figure 3.12. Video Background/foreground separation structural similarity index (SSIM) of the estimated background and the ground-truth background versus frame index (t).

sparse foreground movement of shoppers. In order to achieve a manageable computational burden, we resize the original frame size by 50% to 144×192 and operate on $N = 175$ frames. We perform three different types of experiments on this dataset. First, we demonstrate the foreground/background separation performance of each method on the original dataset. Next, we introduce salt & pepper noise with small density to each frame of the video and plot the structural similarity (SSIM) [194] of the estimated background and the ground-truth background. Finally, we introduce occlusions in terms of white patches to simulate outliers and plot the SSIM of the estimated background and the ground-truth background.

Foreground/background separation under nominal conditions. For every frame of the video, we obtain the static background at the particular frame index t by projecting the processed frame onto the estimated subspace as $\mathbf{Q}_t\mathbf{Q}_t^\top\mathbf{x}_t$, with $K = 8$. We present the estimated background frames of frames $t = 100$, 120, 140, 160, 165 in Figure 3.11 and observe that across all frames, Oja retains much of the foreground. AdaOja performs marginally better owing to its adaptive step-size selection per update. Oja with a hand-tuned step-size of $8 \cdot 10^{-8}/t$ performs slightly better than Oja. The robust algorithm R-SGD1 performs better than Oja but retains some part of the foreground, similar to the proposed method for $\alpha = 1$. L1-Oja demonstrates its robustness by eliminating most of the foreground. As expected, the proposed method obtains a cleaner background with lesser foreground component for smaller values of α. Specifically, for $\alpha = 0.75$, the proposed method obtains the best estimate of the background across all frames.

Foreground/background separation under salt & pepper noise. We read in [195–197] that salt & pepper noise is a common occurrence in images and video surveillance due to acquisition and/or transmission errors. In order to evaluate the performance of the proposed method under noise, we introduce salt & pepper noise with a small density of 2%. We evaluate the background estimation performance by computing the structural similarity index (SSIM) of the estimated background frame and the ground-truth background frame, which is defined for any two images $\mathbf{X}$ and $\mathbf{Y}$ as [194]

$$\text{SSIM}(\mathbf{X}, \mathbf{Y}) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_x\sigma_y + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{3.23}$$

where μ_x is the average pixel intensity of $\mathbf{X}$, μ_y is the average pixel intensity of $\mathbf{Y}$, σ_x^2 is the variance of the pixel intensities of $\mathbf{X}$, σ_y^2 is the variance of the pixel intensities of $\mathbf{Y}$, $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ are two variables that prevent the division with a weak denominator, with $k_1 = 0.01$ and $k_2 = 0.03$ by default and L is the dynamic range of the pixel intensities (e.g., 255 for 8-bit gray-scale images). We note that $\text{SSIM}(\mathbf{X}, \mathbf{Y})$ is bounded by one and is equal to one only if $\mathbf{X} = \mathbf{Y}$ and in general a higher value of SSIM signifies better structural similarity between $\mathbf{X}$ and $\mathbf{Y}$. Keeping all the parameters of the experiment the same as before, except the added salt & pepper noise with density 2%, we plot the average SSIM versus frame index t in Figure 3.12 computed over 1000 independent realizations of noise in order to demonstrate the evolution of the proposed algorithm under noise and observe that Oja's method does not improve over t due to the noise in each frame. Hand-tuning the step-size of Oja to $8 \cdot 10^{-8}/t$ (same as before) yields better performance, comparable to the proposed method with $\alpha = 1.1$. AdaOja demonstrates good performance, similar to R-SGD1, and the proposed method with $\alpha = 1$, owing to its adaptive step-size. We note that L1-Oja performs better than Oja with the same step-size, but by setting the step size to $2 \cdot 10^{-5}/t$, it performs the best, very similar to the proposed algorithm with $\alpha = 0.75$. As noted in previous experiments, the proposed method is capable of generalizing the performance of Oja and other robust variants, with lower values of α yielding higher noise immunity and outlier resistance. Next, we vary the noise density from 0% to 5% in steps of 1% and plot the average SSIM of the last 75 frames of the video computed over 1000 independent realizations of noise in Figure 3.13. We observe that Oja is significantly affected by noise and its susceptibility to noise can be controlled by setting its step-size to the same value as before, $8 \cdot 10^{-8}/t$. AdaOja is relatively more robust against noise, similar to our observations in previous experiments. L1-Oja with a step-size of $2 \cdot 10^{-5}/t$ performs well, demonstrating high noise immunity. The proposed method with $\alpha = 0.75$ on the other hand performs the best with the highest noise immunity across the board. Higher values of α results in high noise susceptibility owing to larger steps along the noisy

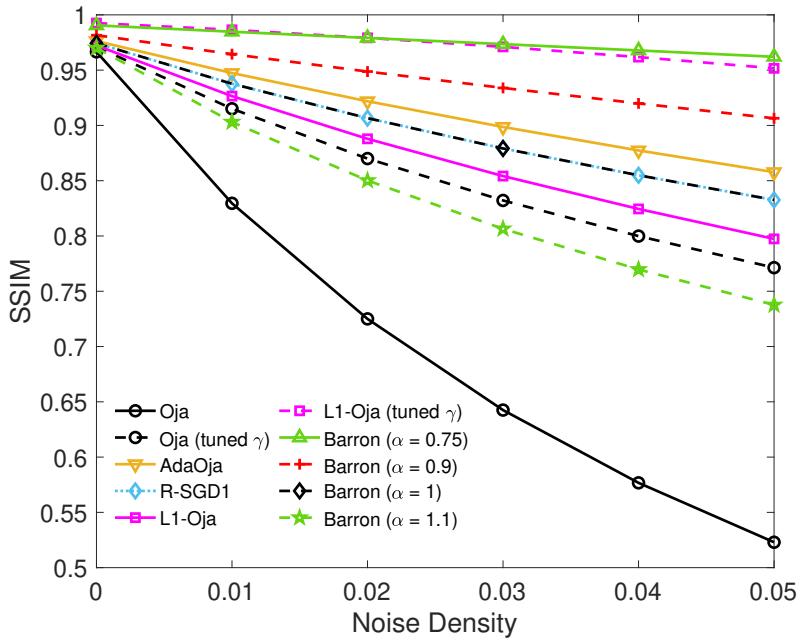


Figure 3.13. Video Background/foreground separation structural similarity index (SSIM) of the estimated background and the ground-truth background versus salt & pepper noise density.

gradient. Finally, we demonstrate the robustness of the proposed method for low α values when the processed data contain outliers.

Foreground/background separation under arbitrary occlusions. In this experiment, we operate with the original dataset (no noise), but we simulate outliers through arbitrary occlusions of a given frame. A corrupted frame contains a square white patch of size 75×75 whose entries are all 255, placed at an arbitrary position in the image. Similar to previous experiment, we report the performance by computing the SSIM of the last 75 frames. We vary the corruption probability per frame in the range $[0 : 10\% : 50\%]$ and compute the SSIM of the estimated background frame with that of the ground-truth frame, computed over 500 realizations of the experiment. We plot the average SSIM versus occlusion probability per frame in Figure 3.14 and observe that Ada Oja is affected the most by outliers in the form of occlusions. Interestingly, L1-Oja shows degraded performance with increasing occlusion probability. Oja's method is also significantly affected by outliers, whereas the proposed method with lower α values demonstrate superior outlier resistance and their outlier resistance vanes as α increases. Similar to previous observations, R-SGD1 and the proposed method for $\alpha = 1$ demonstrate the same performance across all occlusion probabilities.

Classification of Wisconsin Breast Cancer data. In this experiment, we perform nearest subspace (NS) classification on the Wisconsin breast cancer dataset [186, 187] in two parts. Firstly, we compute the F1-measure versus measurement index t using the original dataset. Next, we sim-

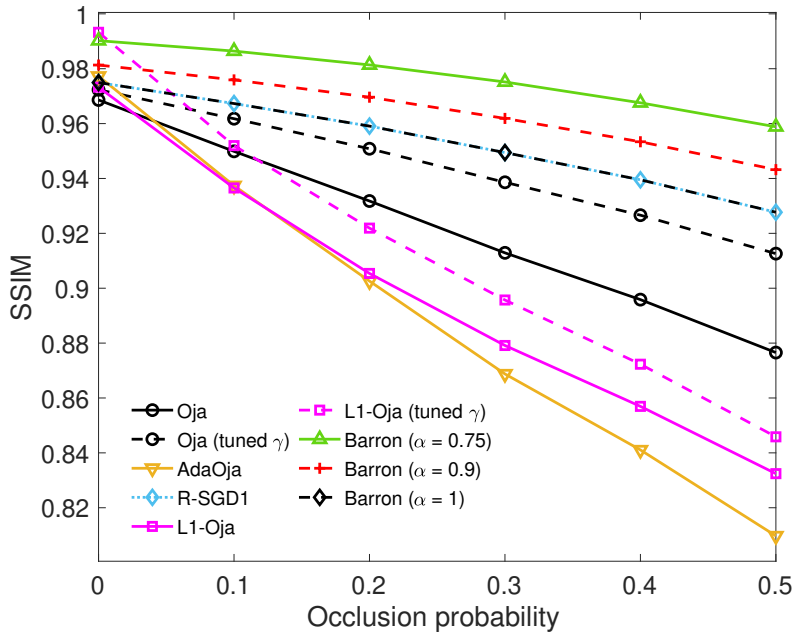


Figure 3.14. Video Background/foreground separation structural similarity index (SSIM) of the estimated background and the ground-truth background versus probability of occlusion per frame.

ulate outliers in the dataset by mislabelling each measurement with some probability. We vary the mislabeling probability per measurement and compute the F1-measure per mislabeling probability. As noted earlier, the dataset consists of 569 samples in total, with 212 samples belonging to the benign class and 357 samples belonging to the malignant class. Out of these measurements, we arbitrarily choose 150 measurements per class for training and 25 measurements per class for testing at each realization of the experiment. In nearest subspace classification, we estimate the subspace of each class using the corresponding training data. We project each held-out test data point onto both subspaces and measure the corresponding projection magnitudes and assign it to the class whose subspace yields the higher projection magnitude. Mathematically, at any update index t, assign the i-th test data $\mathbf{x}_{test}^{(i)}$ to class 1, if $\|\mathbf{Q}_t^{(1)}\mathbf{Q}_t^{(1)^\top}\mathbf{x}_{test}^{(i)}\|_F^2 > \|\mathbf{Q}_t^{(2)}\mathbf{Q}_t^{(2)^\top}\mathbf{x}_{test}^{(i)}\|_F^2$, else assign it to class 2, where $K = 3$, $\mathbf{Q}_t^{(1)}$ and $\mathbf{Q}_t^{(2)}$ are the estimated subspaces of class 1 and class 2 respectively at update index t. The F1-measure is measured as the harmonic mean of precision and recall, that is, $\texttt{F1-measure} = 2\frac{\texttt{precision}\cdot\texttt{recall}}{\texttt{precision}+\texttt{precision}}$, where $\texttt{precision} = \frac{TP}{TP+FP}$, $\texttt{recall} = \frac{TP}{TP+FN}$, TP is the number of true positive predictions, FP is the number of false positive predictions, and FN is the number of false negative predictions.

Evolution of the classification performance. We compute the average precision (as explained earlier) over 4000 realizations of the experiment and plot it versus the measurement index t in Figure 3.15, where we observe that Oja evolves slowly to higher average f1-measure. Hand-tuning the step size to $2.5 \cdot 10^{-6}/t$ results in a better performance for Oja. AdaOja quickly achieves relatively

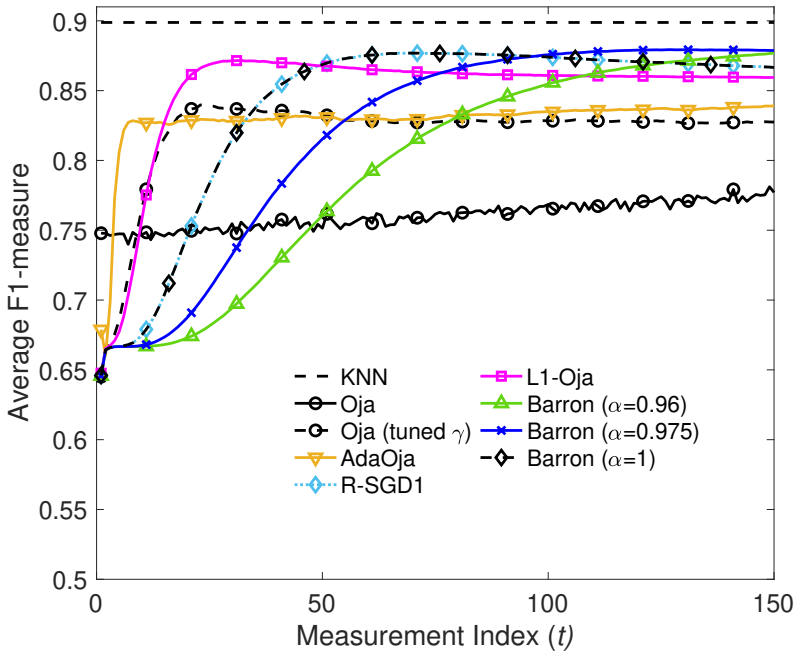


Figure 3.15. Average F1-measure versus measurement index (t) on the Wisconsin breast cancer dataset.

high average f1-measure but converges close to Oja with handpicked learning rate. L1-oja adapts slightly slower but converges to higher average F1-score. We observe that the proposed method with lower α values evolve slower, but converge to higher average F1-scores. This demonstrates the trade-off between faster convergence versus higher outlier-resistance controlled by the value of α. For the sake of baseline comparison, we plot the F1-score obtained by k-nearest neighbor (k-NN) classification with $k = 1$, seen as horizontal black dashed line. Interestingly, we notice that L2-based methods converge to lower F1-scores compared to the robust counterpart, L1-Oja, and the proposed method because the original dataset consists of subspace outliers with high magnitude. We present a detailed discussion of this in Appendix A.

Classification performance under mislabeling. In this experiment, we simulate outliers in the dataset by introducing mislabeling between classes. Specifically, we interchange the labels of each training datum with some probability. We vary the mislabeling probability per measurement and compute the average F1-measure at the last update. The average F1-measure is computed over 5000 realizations and plotted versus mislabeling probability per measurement in Figure 3.16, where we observe that k-NN's performance severely degrades under mislabeling. L2-based approaches Oja, Oja with handcrafted learning rate, and AdaOja are also significantly affected by mislabeling. L1-Oja, R-SGD1, and the proposed method demonstrates remarkable resistance against mislabeling. The proposed method with $\alpha = 0.95$ shows the best performance across increasing probability of

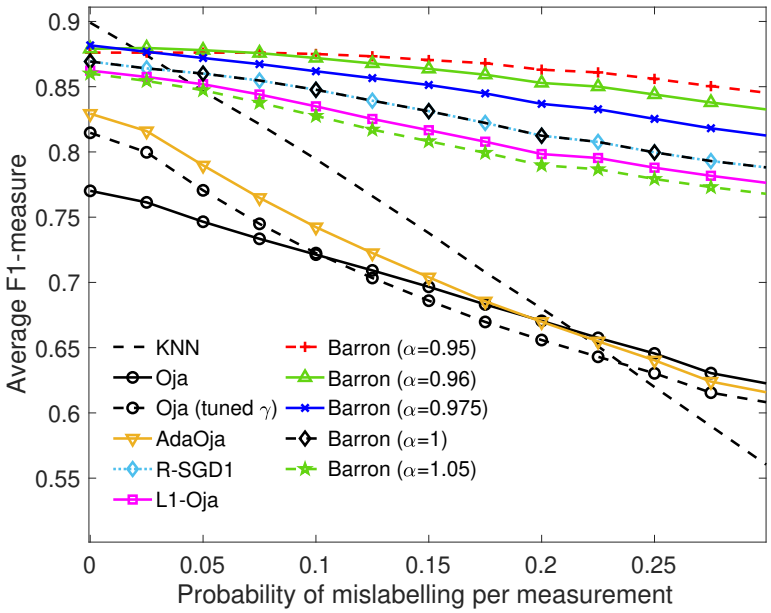


Figure 3.16. Average F1-measure versus probability of mislabeling per frame on the Wisconsin breast cancer dataset.

mislabeling per frame, reinforcing the superior outlier resistance of the proposed method with low α values.

3.5 Conclusions

In this chapter, we propose algorithms for outlier-resistant stochastic PCA. First, we propose a modified L1-norm stochastic PCA for $K = 1$ with formal convergence guarantees. Next, we propose a generalized framework for Oja-type stochastic PCA based on the Barron loss for $K \geq 1$. The proposed framework yields algorithms with varying degree of outlier-resistance depending on the robustness parameter α. We show that the proposed method is the same as Oja update with a different step-size. The proposed method allows the user to choose a trade-off between faster convergence versus higher outlier resistance by tuning the robustness parameter α. Additionally, by appropriately choosing the α value for the proposed framework, we can approximate the performance of Oja, a robust variant R-SGD1, and any other Oja-type algorithm in between. Interestingly, we note that lower values of α not only yield higher outlier resistance, but they also demonstrate lower gradient variance under noisy conditions. Our numerical studies demonstrate the convergence and corroborate the sturdy outlier resistance of the proposed method on synthetic and multiple real-world datasets.

Chapter 4

Asymptotic Theory for Lp-norm Principal Component Analysis

4.1 Introduction

Although PCA enjoys many advantages including a simple solution via SVD [38], many works in the literature have demonstrated the severe outlier susceptibility of PCA, and offered robust counterparts [3–6, 27, 40]. One line of research imparts robustness to PCA by employing the general Lp-norm as opposed to the L2-norm. Specifically it has been observed that for $p < 2$, Lp-norm offers more robustness compared to the L2-norm [19–22]. Motivated by this observation, the formulation of projection maximization PCA was generalized to the Lp-norm, giving rise to Lp-norm principal component analysis (Lp-PCA), with $p = 2$ resulting in standard L2-PCA and $p = 1$ yielding L1-PCA. Early algorithms for general p, that is, for $0 < p \leq 2$ were proposed in [19] and multiple solvers were proposed for $p \leq 1$ in [20–22]. In general, it is known that the algorithms for $p < 1$ are more robust to outliers but converge slower compared to L1-PCA, owing to sub-linear emphasis on each datum, whereas for $p > 1$, they are less robust, but converge faster. In Figure 4.1, we present the loss curves of Lp-norm for various p values in one dimension and observe that the loss curves corresponding to larger values of p evolve much more rapidly compared to smaller values of p as $|x|$ increases. This demonstrates that an outlier with large magnitude results in a significantly large loss for larger values of p.

Despite the fact that Lp-PCA algorithms (for $p < 2$) have demonstrated significant outlier-resistance compared to L2-PCA, they are perceived as "***robust heuristics***", owing to the lack

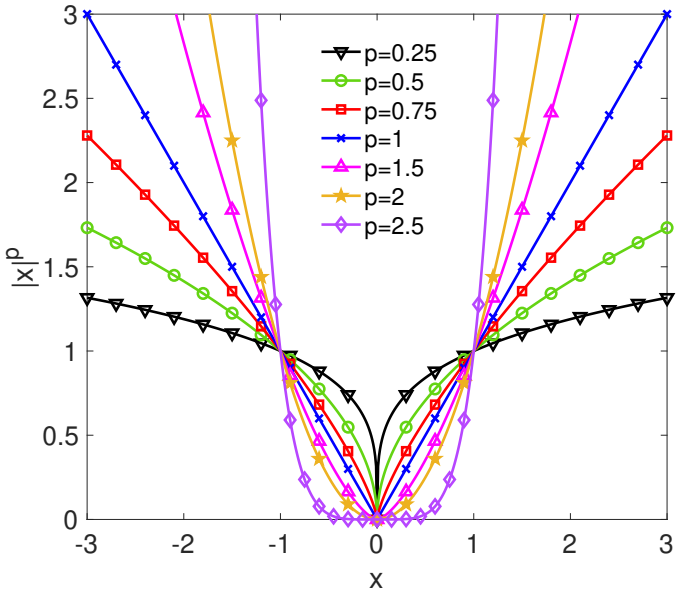


Figure 4.1. Loss curves of Lp-norm, for various p values. x-axis represents the scalar argument x and the y-axis represents the p-th power of absolute value of x, that is, $|x|^p$.

of asymptotic convergence properties. Although relying on Lp-PCA algorithms for $p < 2$ to impart robustness makes intuitive sense, there is a lack of understating of their asymptotic behavior, unlike the case of L2-PCA. For the special case of $p = 1$, the work in [198] proved that if the data is drawn from a zero-mean Gaussian distribution, then the L1-PC coincides asymptotically with the standard PC, therefore to the dominant eigenvector. First, we extend this result for the large family of Elliptical distribution, which includes the Gaussian distribution. Next, we bridge the literature gap in the understanding of asymptotic convergence of Lp-PCA for general values of p and $K \geq 1$ by deriving theoretical convergence guarantees for Lp-PCA as $N \to +\infty$. Our analysis stems from the asymptotic convergence properties of standard PCA and for the sake of completeness, we offer a brief overview of the asymptotic theory for PCA.

4.1.1 Brief Review of Asymptotic PCA

As previously discussed in Section 3.1, it has been shown for PCA that, as the number of data points $N \to +\infty$, the dominant PCs coincide with the dominant eigenvectors of the covariance matrix of the data. That is, assuming the data is drawn from a zero-mean distribution $\mathcal{D}$, with covariance matrix $\mathbf{C}$, the dominant K-PCs obtained by the SVD of $X \in \mathbb{R}^{D \times N}$ coincide asymptotically with the dominant K-eigenvectors of $\mathbf{C}$ [199, 200]. This is a well-known asymptotic property that motivates practitioners to rely on PCA for data analysis. Mathematically, consider independent and identically distributed (i.i.d.) data vectors $\{\mathbf{x}_n\}_{n=1}^N$, drawn from some distribution $\mathcal{D}$. Since

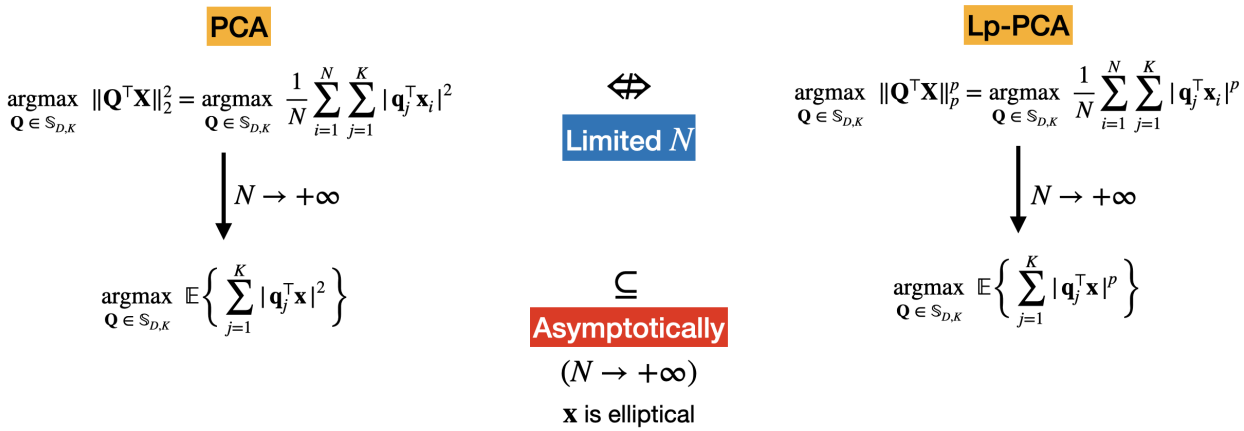


Figure 4.2. Illustration of our proof-sketch for the asymptotic coincidence of the subspaces of Lp-PCA and PCA.

$\frac{1}{N}\|\mathbf{X}^\top\mathbf{Q}\|_F^2 = \frac{1}{N}\texttt{trace}(\mathbf{Q}^\top\mathbf{X}\mathbf{X}^\top\mathbf{Q})$, standard PCA in (2.22) is equivalent to the sample average approximated version in (3.2),

$$\underset{\mathbf{Q}\in\mathbb{S}_{D,K}}{\text{argmax}}\ \texttt{trace}(\mathbf{Q}^\top\widehat{\mathbf{C}}\mathbf{Q}), \tag{4.1}$$

where $\widehat{\mathbf{C}} := \frac{1}{N}\sum_{i=1}^N \mathbf{x}_i\mathbf{x}_i^\top = \frac{1}{N}\mathbf{X}\mathbf{X}^\top$, based on a size-$N$ batch of coherent data points $\mathbf{X} := [\mathbf{x}_1,\ldots,\mathbf{x}_N]$. For $K=1$, the above expression in (4.1) simplifies to

$$\underset{\mathbf{q}\in\mathbb{S}_D}{\text{argmax}}\ \mathbf{q}^\top\widehat{\mathbf{C}}\mathbf{q}. \tag{4.2}$$

By the strong law of large numbers, as N increases asymptotically, $\widehat{\mathbf{C}}$ converges to the distribution covariance matrix $\mathbf{C}$. Accordingly, (4.1) converges to $\texttt{trace}(\mathbf{Q}^\top\mathbf{C}\mathbf{Q}) = \mathbb{E}\left\{\left|\mathbf{Q}^\top\mathbf{X}\right|^2\right\}$, where $\mathbb{E}\{\cdot\}$ returns the expected value of its argument with respect to the data distribution. Thus, the following lemma holds.

Lemma 1: As N increases asymptotically, the solution of (3.2), $\hat{\mathbf{Q}}_{L2}$, maximizes $\mathbf{q}^\top\mathbf{C}\mathbf{q}$ over $\mathbb{S}_D$, coinciding with $\mathbf{Q}_{\text{eig}}$ (the dominant eigenvector subspace of the covariance matrix $\mathbf{C}$), spanning the subspace corresponding to the K-highest eigenvalues of the distribution $\mathcal{D}$ [199, 200].

Corollary 1: For $K=1$, as N increases asymptotically, the solution of (3.4), $\hat{\mathbf{q}}_{L2}$, maximizes $\mathbf{q}^\top\mathbf{C}\mathbf{q}$ over $\mathbb{S}_D$, coinciding with $\mathbf{q}_{\text{eig}}$, which is the dominant eigenvector of the covariance matrix $\mathbf{C}$, spanning the highest-variance line (rank-1 subspace) of distribution $\mathcal{D}$.

4.1.2 Brief Overview of Proof-sketch

We offer a brief overview of our proof-sketch in the following. We assume that the data is independent and identically distributed (i.i.d.) from a zero-mean Elliptical distribution. Firstly, we note that for limited N, the solutions of Lp-PCA [19] and PCA can be very different. Next, we show that as $N \to +\infty$, the sample average approximations of PCA and Lp-PCA tend almost surely (a.s.) to their population counterparts. Finally, we prove that the subspace spanned by Lp-PCA coincides with that of PCA asymptotically. An illustration of our proof-sketch is presented in Figure 4.2.

4.1.3 Contributions

Our contributions in this chapter are as follows:

1. We offer the proof of asymptotic coincidence of the L1-PC and the standard PC for $K = 1$.
2. We show that L1-PCA is as good an estimator of the maximum-variance line (dominant eigenvector) as PCA asymptotically, while remaining robust against outliers in the limited data scenario.
3. We present the proof of asymptotic convergence of the Lp-PCA subspace to the L2-PCA subspace and therefore to the dominant eigensubspace for $K \geq 1$.
4. Although the dominant asymptotic Lp-PCs span the same subspace as that of the dominant asymptotic L2-PCs, we show that the asymptotic Lp-PCs are specific rotated versions of the dominant eigenvectors of the covariance matrix. That is, we show that asymptotic Lp-PCA solves asymptotic L2-PCA, but not every solution of asymptotic L2-PCA solves asymptotic Lp-PCA.
5. We offer an algorithm to derive the specific rotation matrix that obtains the asymptotic Lp-PCs from the dominant eigenvectors.
6. We demonstrate the derived convergence theory in practice through our experimental studies on data drawn from multiple members of the elliptical distribution.
7. Finally, we show that the proposed theory can be leveraged to initialize the iterative algorithms of Lp-PCA to obtain faster and better convergence on synthetic and real-world datasets.

We organize the rest of this chapter as follows. In Section 4.2.1, we offer a background on Elliptical distribution, its probability density function, some relevant properties, and highlight its application in modeling real-world data. Followed by Section 4.3.1, where we present the coincidence of L1-PCA and PCA for $K = 1$ and offer experimental studies. Finally, in Section 4.4, we generalize this result to $K \geq 1$ and Lp-PCA, where we prove that the subspaces obtained by Lp-PCA and PCA coincide asymptotically. Moreover, we show that the Lp-PCs are specific rotated versions of the standard PCs as $N \to +\infty$ and offer a wide variety of experimental studies to demonstrate the proposed theory empirically.

4.2 Background on Multivariate Elliptical Distribution

4.2.1 Density

In this work, we focus on zero-mean Elliptical distributions that have a well-defined probability density function. We denote by $\mathbf{x} \sim \mathcal{E}_D(\mathbf{0}_D, \mathbf{\Sigma}, g_D)$ a zero-mean Elliptically distributed multivariate random variable $\mathbf{x}$, where $\mathbf{\Sigma} \in \mathbb{R}^{D \times D}$ is a positive-definite scatter matrix and $g_D(\cdot)$ is a non-negative function known as the density generator such that $\int_0^{+\infty} u^{\frac{D}{2}-1} g_D(u) du < +\infty$ for any positive u [201, 202]. The density of $\mathbf{x} \sim \mathcal{E}_D(\mathbf{0}_D, \mathbf{\Sigma}, g_D)$ is defined as $f(\mathbf{x}) = \frac{c_D}{|\mathbf{\Sigma}|^{\frac{1}{2}}} g_D\left(\mathbf{x}^\top \mathbf{\Sigma}^{-1} \mathbf{x}\right)$, where $|\cdot|$ returns the determinant of its matrix argument and c_D is a normalizing constant such that $\int_{-\infty}^{+\infty} f(\mathbf{x}) d\mathbf{x} = 1$ [201, 202]. For any elliptically distributed random variable, if $\int_0^{+\infty} g_1(u) du < +\infty$, then the mean vector is given by $\mu := \mathbb{E}\{\mathbf{x}\}$ and if $|\psi'(0)| < +\infty$, then the covariance matrix is given by $\mathbf{C} := \mathbb{E}\left\{\mathbf{x}\mathbf{x}^\top\right\} = -2\psi'(0)\mathbf{\Sigma}$, where $\psi(\cdot)$ is known as the characteristic generator function and $\psi'(\cdot)$ is the first derivative of $\psi(\cdot)$ [202, 203]. The density generator function for some members of the Elliptical distribution is presented in Table 4.1 below, where κ for the Power Exponential distribution is kurtosis parameter that defines the shape of the distribution, for example $\kappa = 1$ yields the Gaussian distribution, while $\kappa = 0.5$ yields the Laplace distribution [204]. For Student's t distribution, ν signifies the number of degrees of freedom.

Elliptical Family Member	**Density Generator** $g_D(u)$
Power Exponential (Generalized Gaussian)	$\texttt{exp}(-\frac{u^\kappa}{2})$
Gaussian	$\texttt{exp}(-\frac{u}{2})$
Laplace	$\texttt{exp}(-\frac{u^{0.5}}{2})$
Logistic	$\frac{\texttt{exp}(-u)}{(1+\texttt{exp}(-u))^2}$
Student's t	$(1+\frac{1}{\nu}u)^{-\frac{\nu+D}{2}}$

Table 4.1: Important members of the Elliptical Distribution and their density generator functions.

Importantly, for any $\mathbf{x} \sim \mathcal{E}_D(\mathbf{0}_D, \mathbf{\Sigma}, g_D)$ and $\mathbf{q} \in \mathbb{R}^D$, it holds that $Y = \mathbf{q}^\top \mathbf{x} \sim \mathcal{E}_1(0, \mathbf{q}^\top \mathbf{\Sigma} \mathbf{q}, g_1)$ [201, 202]. Thus, the probability density function of Y is

$$f_Y(y) = \frac{c_1}{(\mathbf{q}^\top \mathbf{\Sigma} \mathbf{q})^{\frac{1}{2}}} g_1 \left(\frac{y^2}{\mathbf{q}^\top \mathbf{\Sigma} \mathbf{q}} \right). \tag{4.3}$$

4.2.2 Applications of Elliptical Data

Elliptical distributions are commonly used to model financial data and consequently, they are applied in the areas of portfolio theory and risk management [205]. In addition, they are used in astronomy, physics, radioimmunoassay, signal processing, machine learning, and pattern recognition [206]. Members of Elliptical distribution such as Gaussian, Laplace, Student's t, Logistic, and Generalized Gaussian are extensively used in practice.

For example, Gaussian distribution being one of the most widely encountered distribution in practice finds application in many areas including in image segmentation and anomaly detection [207], face detection [208], wireless communications [209], cyber attack detection [210], financial economics [211], and biology [212], to name just a few. Laplace distribution is used to model AC coefficients generated by the discrete cosine transform (DCT) in JPEG image compression [213], discrete Fourier transform (DFT) coefficient priors in speech recognition [214], and wind-shear experienced by airplanes [215]. Student's t distribution is commonly used for cluster analysis, regression analysis, discriminant analysis, missing data imputation, and to model stock returns [216–219]. Logistic distribution is used in applications of population growth, bioassay, medical diagnosis, public health and social sciences [220]. Generalized Gaussian (Power Exponential) distribution [221–223] is a notable member of the Elliptical distribution used to model Gabor coefficeint features for face recognition in [224], multiple access interference in ultra wide-band wireless communication systems [225], synthetic aperture radar (SAR) images [226], and echocardiographic (ECG) images [227]. In signal processing, the generalized Gaussian distribution finds many applications as stated in [228], including: (i) for $p \leq 1$ to model interference for high to moderate signal-to-noise ratio in ultra-wide band systems with time hopping, (ii) modeling atmospheric noises (for $0.1 < p < 0.6$), and (iii) modeling the underwater acoustic channel (for $p = 2.22$, generalized Gaussian models well the ship-transit noise, while for $p = 1.6$, it models the sea surface agitation noise).

4.3 Contribution 1: Asymptotic Theory for L1-PCA — One Component

In this section, we present the coincidence of L1-PCA and PCA for $K = 1$. We provide a detailed proof sketch that proves the equivalence and offer experimental studies to corroborate the proposed theory.

4.3.1 Proposed Theorem

We start with Lemma 2 that expresses L1-PCA in an equivalent form, in terms of its maximization argument.

Lemma 2: The L1-PC $\hat{\mathbf{q}}_{L1}$, solution of (2.3), also solves

$$\max_{\mathbf{q}\in\mathbb{S}_D} \frac{1}{N}\sum_{i=1}^{N}|\mathbf{q}^\top\mathbf{x}_i|. \tag{4.4}$$

Considering again i.i.d. points $\{\mathbf{x}_n\}_{n=1}^N$, drawn from $\mathcal{D}$, random variables $\{|\mathbf{q}^\top\mathbf{x}_i|\}_{n=1}^N$ are also i.i.d. and, by the strong law of large numbers, Lemma 2 below holds true.

Lemma 3: As N increases asymptotically, $\frac{1}{N}\sum_{i=1}^{N}|\mathbf{q}^\top\mathbf{x}_i|$ converges to $\mathbb{E}\left\{|\mathbf{q}^\top\mathbf{x}|\right\}$.

As stated in Section 4.2.1, for any $\mathbf{x}\sim\mathcal{E}_D(\mathbf{0}_D,\mathbf{\Sigma},g_D)$ and any given $\mathbf{q}\in\mathbb{R}^D$, $\mathbf{q}^\top\mathbf{x}\sim\mathcal{E}_1(\mathbf{0},\mathbf{q}^\top\mathbf{\Sigma}\mathbf{q},g_1)$. Accordingly, it holds that [229]

$$\mathbb{E}\left\{\left|\mathbf{q}^\top\mathbf{x}\right|\right\} = \int_{-\infty}^{+\infty}|y|\,\frac{c_1}{(\mathbf{q}^\top\mathbf{\Sigma}\mathbf{q})^{\frac{1}{2}}}g_1\left(\frac{y^2}{\mathbf{q}^\top\mathbf{\Sigma}\mathbf{q}}\right)dy. \tag{4.5}$$

Since $g_1(\cdot)$ is symmetric about 0 and $|\cdot|$ is an even function,

$$\mathbb{E}\left\{\left|\mathbf{q}^\top\mathbf{x}\right|\right\} = \frac{2c_1}{(\mathbf{q}^\top\mathbf{\Sigma}\mathbf{q})^{\frac{1}{2}}}\int_0^{+\infty} y\,g_1\left(\frac{y^2}{\mathbf{q}^\top\mathbf{\Sigma}\mathbf{q}}\right)dy. \tag{4.6}$$

Setting $u = \frac{y^2}{\mathbf{q}^\top\mathbf{\Sigma}\mathbf{q}}$, we get $2ydy = \mathbf{q}^\top\mathbf{\Sigma}\mathbf{q}\,du$ and

$$\mathbb{E}\left\{\left|\mathbf{q}^\top\mathbf{x}\right|\right\} = c_1\,(\mathbf{q}^\top\mathbf{\Sigma}\mathbf{q})^{\frac{1}{2}}\int_0^{+\infty} g_1(u)\,du. \tag{4.7}$$

Interestingly, $\mathbb{E}\{|y|\}$ is finite only if $\beta = \int_0^{+\infty} g_1(u)\,du < +\infty$, which holds true by the definition of $g_1(\cdot)$. Based on the above it follows that

$$\mathbb{E}\left\{\left|\mathbf{q}^\top \mathbf{x}\right|\right\} = c_1\,\beta\,(\mathbf{q}^\top \mathbf{\Sigma} \mathbf{q})^{\frac{1}{2}}. \tag{4.8}$$

Thus, the solution to (4.8), $\mathbf{q}_{L1}$, maximizes $(\mathbf{q}^\top \mathbf{\Sigma} \mathbf{q})^{\frac{1}{2}}$.

The above hold true for any elliptical distribution. For zero-mean elliptical distributions it holds that

$$\mathbf{C} = \mathbb{E}\{\mathbf{x}\mathbf{x}^\top\} = \alpha \mathbf{\Sigma}, \tag{4.9}$$

where $\alpha = -2\psi'(0) > 0$. By (4.8), (4.9), and the monotonicity of $(\cdot)^{1/2}$, the following Lemma 3 holds.

Lemma 4: For elliptical $\mathbf{x} \sim \mathbb{E}_D(\mathbf{0}_D, \mathbf{\Sigma}, g_D)$, $\mathbb{E}\left\{|\mathbf{q}^\top \mathbf{x}|^2\right\} = \mathbf{q}^\top \mathbf{R} \mathbf{q}$ and $\mathbb{E}\left\{|\mathbf{q}^\top \mathbf{x}|\right\}$ in (4.8), are maximized at the same $\mathbf{q} \in \mathbb{S}_D$.

Finally, by Lemmas 1-4 the following Theorem 1 holds.

Theorem 1: Consider i.i.d. data points $\{\mathbf{x}_n\}_{n=1}^N$ from zero-mean Elliptical distribution $\mathcal{E}_D(\mathbf{0}_D, \mathbf{\Sigma}, g_D)$. As N increases asymptotically, the L1-PC $\hat{\mathbf{q}}_{L1}$ in (2.3) coincides with $\mathbf{q}_{L1}$ in (3.8) and $\mathbf{q}_{L2}$ in (3.3).

4.3.2 Experimental Studies

In this section, we demonstrate the asymptotic coincidence of L1-PCA and PCA empirically. In our first experiment, we show that the L1-PCA objective tends to a scaled version of the PCA objective asymptotically, whereas for limited data, L1-PCA objective is generally not a scaled version of the PCA objective. To that end, we draw $N = 5$ (non-asymptotic case) data points from $\mathcal{N}(\mathbf{0}_2, \mathbf{C})$, where $\mathbf{C} = \left[\begin{smallmatrix} 2.2501 & 1.3170 \\ 1.3170 & 1.7299 \end{smallmatrix}\right]$, to form the data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{2 \times N}$. Next, we estimate the L1-PCA objective as $\frac{1}{N}||\mathbf{q}(\theta)^\top \mathbf{X}||_1$, where $\mathbf{q}(\theta) = [\texttt{cos}(\theta), \texttt{sin}(\theta)]^\top$, for θ varied from $-90°$ to $+90°$ in steps of 10^{-3}, and plot it versus θ in Figure 4.3(a). In the same figure, we also plot the estimated PCA objective $\frac{1}{N}||\mathbf{q}(\theta)^\top \mathbf{X}||_2^2$ versus θ and observe that the L1-PCA objective is non-smooth, with multiple local-maxima. Moreover, the maximum value of the L1-PCA objective does not coincide with that of PCA.

Next, we repeat the experiment for $N = 10^5$ (asymptotic case) and plot the resulting L1-PCA and PCA objectives versus θ in Figure 4.3(b). We observe that the L1-PCA objective is (i) smooth and

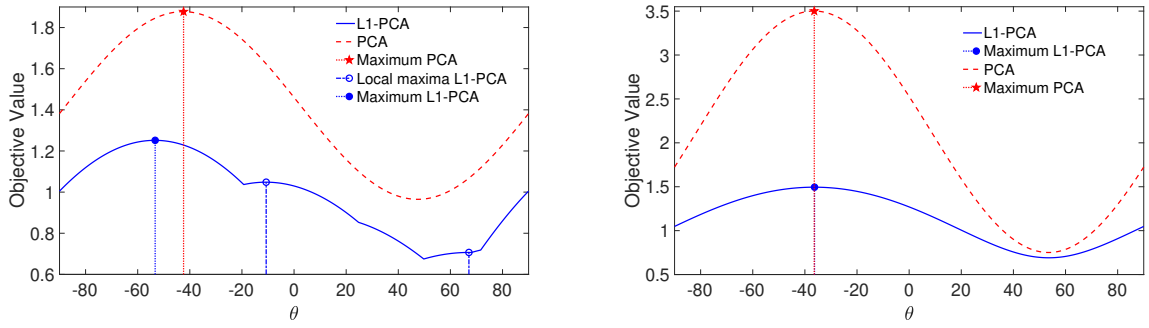


Figure 4.3. Objective value $\left(\frac{1}{N}||\mathbf{q}(\theta)^\top \mathbf{X}||_1 \text{ for L1-PCA and } \frac{1}{N}||\mathbf{q}(\theta)^\top \mathbf{X}||_2^2 \text{ for PCA}\right)$ versus θ for (a) $N = 5$ (b) $N = 10^5$.

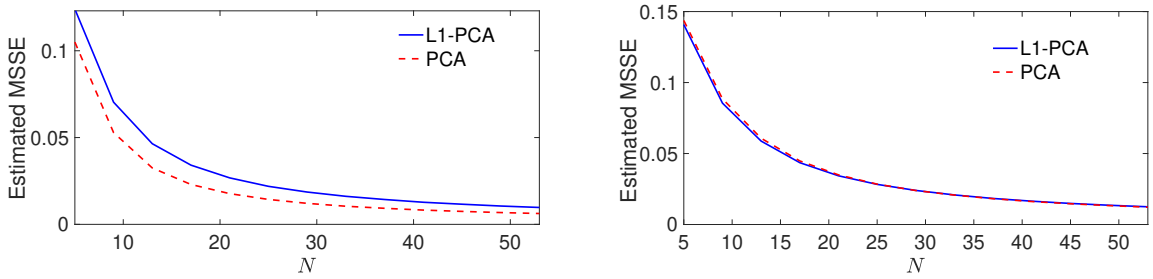


Figure 4.4. Estimated MSSE versus number of processed measurements, N for (a) Gaussian and (b) Laplace data distributions.

(ii) proportional to the square root of the PCA objective, thereby making its global maximum value also proportional to the square root of that of PCA. Importantly, the maximizer of the L1-PCA metric coincides with that of PCA.

In our next experiment, we demonstrate that for data drawn from members of the Elliptical distribution, the optimal L1-PC tends to the dominant eigenvector asymptotically, as explained in the following. We draw $N \in \{5, 9, 13, \ldots, 54\}$ independent data points from $\mathcal{N}(\mathbf{0}_2, \mathbf{C})$ (a member of the Elliptical distribution family), where $\mathbf{C} = \left[\begin{smallmatrix} 4.0172 & 3.4740 \\ 3.4740 & 7.9828 \end{smallmatrix}\right]$, to form the data matrix $\mathbf{X} \in \mathbb{R}^{2 \times N}$. Next for every N, we estimate the mean squared subspace error (MSSE) over $S = 10^6$ independent realizations as $\frac{1}{S}\sum_{i=1}^{S} ||\mathbf{q}_i \mathbf{q}_i^\top - \mathbf{q}_{\text{eig}} \mathbf{q}_{\text{eig}}^\top||_F^2$, where $\mathbf{q}_i$ is the PC obtained after processing N data points at the i-th realization and $\mathbf{q}_{\text{eig}}$ is the dominant eigenvector of $\mathbf{C}$. In Figure 4.4 (a), we plot the estimated MSSE attained by the dominant L1-PC obtained by the optimal L1-PCA algorithm [3] and the dominant PC obtained by SVD (PCA) versus N. We observe that PCA starts at a lower error but as N increases, the MSSE of both L1-PCA and PCA jointly approach zero, coinciding with $\mathbf{q}_{\text{eig}}$.

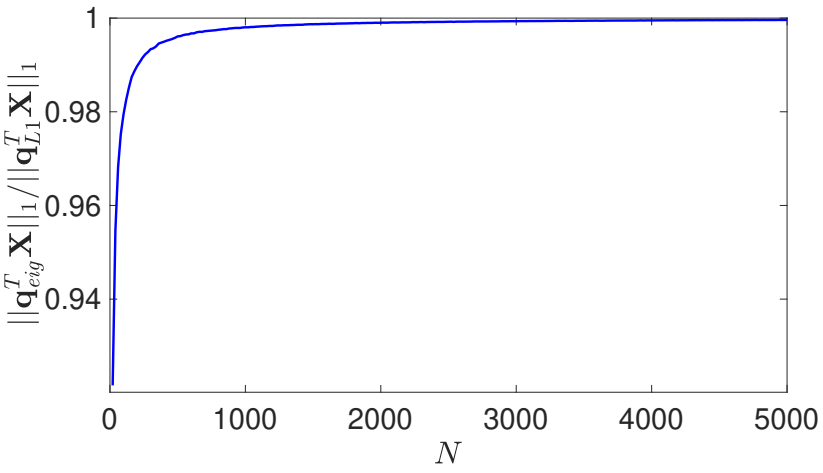


Figure 4.5. $\frac{\|\mathbf{q}_{\text{eig}}^\top \mathbf{X}\|_1}{\|\mathbf{q}_{\text{L1}}^\top \mathbf{X}\|_1}$ versus number of processed measurements, N for Gaussian data distribution.

Subsequently, we repeat the experiment with data drawn from the Laplace distribution (another member of the Elliptical distribution family), $\mathcal{L}(\mathbf{0}_2, \mathbf{C})$, where $\mathbf{C} = \left[\begin{smallmatrix} 5.6950 & 3.9884 \\ 3.9884 & 6.3050 \end{smallmatrix}\right]$. From Figure 4.4 (b), we observe that both L1-PCA and standard PCA demonstrate similar performance across N and appear to jointly coincide with $\mathbf{q}_{\text{eig}}$ for large N.

In addition, our final experiment demonstrates the asymptotic coincidence of the approximate maximizer of the L1-PCA metric in (2.3) and the value of the metric obtained when $\mathbf{q} = \mathbf{q}_{\text{eig}}$, thereby showing that the L1-PC asymptotically coincides with $\mathbf{q}_{\text{eig}}$. To this end, we plot in Figure 4.5, the ratio $r = \frac{\|\mathbf{q}_{\text{eig}}^\top \mathbf{X}\|_1}{\|\mathbf{q}_{\text{L1}}^\top \mathbf{X}\|_1}$ versus $N \in \{20, 40, 60, \ldots, 5000\}$, where $\mathbf{q}^*$ is the dominant eigenvector of the covariance matrix $\mathbf{C} = \left[\begin{smallmatrix} 6.0541 & 0.9985 \\ 0.9985 & 5.9459 \end{smallmatrix}\right]$, $\mathbf{X} \in \mathbb{R}^{2 \times N}$ contains entries drawn from $\mathcal{N}(\mathbf{0}_2, \mathbf{C})$, $\mathbf{q}_{\text{L1}}$ is the maximizer of $\|\mathbf{q}(\theta)^\top \mathbf{X}\|_1$, $\mathbf{q}(\theta) = [\texttt{cos}(\theta), \texttt{sin}(\theta)]^\top$, for $\theta \in \Theta = \{-90^\cdot, -89.75^\cdot, \ldots, 90^\cdot\}$. Since it is infeasible to search all possible $\mathbf{q} \in \mathbb{S}_2$ to find the maximizer $\mathbf{q}_{\text{L1}}$ of $\|\mathbf{q}^\top \mathbf{X}\|_1$, for large values of N, we resort to a fine-approximation of it by the maximizer of $\|\mathbf{q}(\theta)^\top \mathbf{X}\|_1$, over $\theta \in \Theta = \{-90^\cdot, -89.75^\cdot, \ldots, 90^\cdot\}$. We observe that the ratio r starts at a lower value initially and increases with N. Finally for large values of $N \sim 4000$, the value of r converges to 1, signifying that the maximum L1-PCA metric $\|\mathbf{q}_{\text{L1}}^\top \mathbf{X}\|_1$ coincides with the value of $\|\mathbf{q}_{\text{eig}}^\top \mathbf{X}\|_1$.

Therefore, our numerical studies corroborate Theorem 1 by demonstrating empirically, the asymptotic coincidence of L1-PCA and PCA for Elliptical distribution possessing a well-defined mean.

4.4 Contribution 2: Asymptotic Theory for Lp-PCA — Multiple Components

In this section, we generalize the convergence theory presented in the previous section to $K \geq 1$ and Lp-PCA. We offer a detailed proof sketch that proves the equivalence of Lp-PCA and PCA. Specifically, we prove that as $N \to +\infty$, the subspaces spanned by Lp-PCs and the standard PCs coincide. Moreover, we show that the Lp-norm principal components (Lp-PCs) are specific rotated versions of L2-PCs. We present an algorithm to derive the rotation matrix that rotates the asymptotic PCs to obtain the asymptotic Lp-PCs. We leverage the proposed theory to intelligently initialize the iterative algorithms of Lp-PCA, leading to faster and/or better convergence in terms of the Lp-PCA metric. We conclude by offering a wide variety of experimental studies to empirically demonstrate the convergence theory.

Problem formulation of Lp-PCA. Given a data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$, comprising of N measurements drawn from a zero-mean D-dimensional distribution, the Lp-PCs, arranged as the K columns of $\widehat{\mathbf{Q}}_{Lp}$ are defined as the solution to

$$\underset{\mathbf{Q} \in \mathbb{S}_{D,K}}{\operatorname{argmax}} \ \|\mathbf{Q}^\top \mathbf{X}\|_p^p. \tag{4.10}$$

The above problem in (4.10) is equivalent to the following problem

$$\underset{\mathbf{Q} \in \mathbb{S}_{D,K}}{\operatorname{argmax}} \ \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{K} |\mathbf{q}_j^\top \mathbf{x}_i|^p. \tag{4.11}$$

Substituting $p = 2$ in the above problem formulation results in the following standard PCA formulation on (3.2).

4.4.1 Proposed Theorem

Considering i.i.d. data points $\{\mathbf{x}_i\}_{i=1}^N$ drawn from any distribution $\mathcal{D}$, we formalize the convergence of the Lp-PCA objective of (4.11) as $N \to +\infty$ in Lemma 5, similar to Lemma 2 in Section 4.3.1.

Lemma 5: The objective value of Lp-PCA in (4.11), $\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{K} |\mathbf{q}_j^\top \mathbf{x}_i|^p$ converges almost surely to $\mathbb{E}\big\{ \sum_{j=1}^{K} |\mathbf{q}_j^\top \mathbf{x}|^p \big\}$ according to the strong law of large numbers.

Consequently, we obtain the following problem formulation that solves for the asymptotic Lp-PCs

$$\mathbf{Q}_{Lp} = \underset{\mathbf{Q}\in\mathbb{S}_{D,K}}{\operatorname{argmax}} \ \mathbb{E}\left\{\sum_{j=1}^{K} |\mathbf{q}_j^\top \mathbf{x}|^p\right\}. \tag{4.12}$$

Moreover, owing to the linearity of expectation and summation, we can interchange their order without the loss of generality in the objective of (4.12), that is, $\mathbb{E}\left\{\sum_{j=1}^{K} |\mathbf{q}_j^\top \mathbf{x}|^p\right\} = \sum_{j=1}^{K} \mathbb{E}\{|\mathbf{q}_j^\top \mathbf{x}|^p\}$.

Lemma 6: For i.i.d. data points $\{\mathbf{x}_i\}_{i=1}^N$ drawn from a zero-mean Elliptical distribution with scatter matrix $\mathbf{\Sigma}$, we have $\sum_{j=1}^{K} \mathbb{E}\left\{|\mathbf{q}_j^\top \mathbf{x}|^p\right\} = c\sum_{j=1}^{K} \left(\mathbf{q}_j^\top \mathbf{\Sigma}\mathbf{q}_j\right)^{\frac{p}{2}}$, where $0 < c < +\infty$

Similar to Lemma 3, as stated in Section 4.2.1, for any $\mathbf{x}\sim\mathcal{E}_D(\mathbf{0}_D, \mathbf{\Sigma})$ and any given $\mathbf{q} \in \mathbb{R}^D$, $y = \mathbf{q}^\top \mathbf{x}\sim\mathcal{E}_1(\mathbf{0}, \mathbf{q}^\top \mathbf{\Sigma}\mathbf{q})$. Moreover, it holds that [229]

$$\mathbb{E}\left\{\left|\mathbf{q}^\top \mathbf{x}\right|^p\right\} = \int_{-\infty}^{+\infty} |y|^p \, \frac{c_1}{(\mathbf{q}^\top \mathbf{\Sigma}\mathbf{q})^{\frac{1}{2}}} g_1\left(\frac{y^2}{\mathbf{q}^\top \mathbf{\Sigma}\mathbf{q}}\right) dy. \tag{4.13}$$

Since $g_1(\cdot)$ is symmetric about 0 and $|\cdot|^p$ is an even function,

$$\mathbb{E}\left\{\left|\mathbf{q}^\top \mathbf{x}\right|^p\right\} = \frac{2c_1}{(\mathbf{q}^\top \mathbf{\Sigma}\mathbf{q})^{\frac{1}{2}}} \int_0^{+\infty} y^p \, g_1\left(\frac{y^2}{\mathbf{q}^\top \mathbf{\Sigma}\mathbf{q}}\right) dy. \tag{4.14}$$

Setting $u = \frac{y^2}{\mathbf{q}^\top \mathbf{\Sigma}\mathbf{q}}$, we get $2ydy = \mathbf{q}^\top \mathbf{\Sigma}\mathbf{q}\, du$ and

$$\mathbb{E}\left\{\left|\mathbf{q}^\top \mathbf{x}\right|^p\right\} = c_1 \, (\mathbf{q}^\top \mathbf{\Sigma}\mathbf{q})^{\frac{1}{2}} \int_0^{+\infty} y^{p-1} g_1(u) \, du. \tag{4.15}$$

Plugging back $y = \mathbf{q}^\top \mathbf{x}$ in the above expression, we obtain

$$\mathbb{E}\left\{\left|\mathbf{q}^\top \mathbf{x}\right|^p\right\} = c_1 \, (\mathbf{q}^\top \mathbf{\Sigma}\mathbf{q})^{\frac{1}{2}} \int_0^{+\infty} u^{\frac{p-1}{2}} g_1(u) \, du. \tag{4.16}$$

Since $u^{\frac{p-1}{2}}$ and $g_1(u)$ are positive [230], $\int_0^{+\infty} u^{\frac{p-1}{2}} g_1(u) \, du > 0$. Interestingly, $\mathbb{E}\{|y|\}$ is finite only if $\beta = \int_0^{+\infty} u^{\frac{p-1}{2}} g_1(u) \, du < +\infty$. Based on the above it follows that

$$\mathbb{E}\left\{\left|\mathbf{q}^\top \mathbf{x}\right|^p\right\} = c_1 \, \beta \, (\mathbf{q}^\top \mathbf{\Sigma}\mathbf{q})^{\frac{p}{2}}. \tag{4.17}$$

Finally, setting $c = c_1\beta$ and using the above expression, we obtain $\sum_{j=1}^{K} \mathbb{E}\{|\mathbf{q}_j^\top \mathbf{x}|^p\} = c\sum_{j=1}^{K} \left(\mathbf{q}_j^\top \mathbf{\Sigma}\mathbf{q}_j\right)^{\frac{p}{2}}$.

Accordingly, from Lemmas 5 and 6, as $N \to +\infty$, we can formulate Lp-PCA for Elliptical data as

$$\underset{\mathbf{Q} \in \mathbb{S}_{D,K}}{\operatorname{argmax}} \sum_{j=1}^{K} \left(\mathbf{q}_j^\top \mathbf{\Sigma} \mathbf{q}_j \right)^{\frac{p}{2}}. \tag{4.18}$$

Assuming $\mathbf{\Sigma}$ admits eigenvalue decomposition $\mathbf{E}\mathbf{\Lambda}\mathbf{E}^\top$ and setting $\mathbf{Q} = \mathbf{E}\mathbf{R}$ in (4.18), we obtain the following equivalent problem that solves for the optimal rotation matrix

$$\mathbf{R}^* = \underset{\mathbf{R} \in \mathbb{S}_{D,K}}{\operatorname{argmax}} \sum_{j=1}^{K} \left(\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j \right)^{\frac{p}{2}}. \tag{4.19}$$

Lemma 7: For $0 < p < 2$ and $\mathbf{R} \in \mathbb{S}_{D,K}$, the objective in (4.19) is upper-bounded,

$$\sum_{j=1}^{K} \left(\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j \right)^{\frac{p}{2}} \leq \left(\sum_{j=1}^{K} \mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j \right)^{\frac{p}{2}} K^{\frac{2-p}{2}}, \tag{4.20}$$

and achieves equality if and only if $\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j$ is constant $\forall j \in \{1, 2, \ldots, K\}$. The proof follows from the Holder's inequality [231] and is presented in Appendix B.

Lemma 8: The upper-bound in (4.20) (ignoring the scaling) is further upper-bounded as

$$\left(\sum_{j=1}^{K} \mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j \right)^{\frac{p}{2}} \leq \left(\sum_{j=1}^{K} \lambda_j \right)^{\frac{p}{2}}, \tag{4.21}$$

where λ_j is the j-th eigenvalue (diagonal entry) of $\mathbf{\Lambda}$.

Corollary 1: If $D = K$, the above inequality simplifies to equality $\forall \mathbf{R} \in \mathbb{S}_{K,K}$.

Corollary 2: If $D > K$, the upper-bound is achieved in (4.21) if and only if the bottom $D-K$ rows of $\mathbf{R}$ are zeros row vectors. That is, the optimum $\mathbf{R}^* = \begin{bmatrix} \tilde{\mathbf{R}} \in \mathbb{S}_{K,K} \\ \mathbf{0}_{D-K \times K} \end{bmatrix}$, accordingly, the solution to (4.18) is of the form $\mathbf{E}\mathbf{R}^* = \mathbf{E}_{:,1:K}\tilde{\mathbf{R}}$, where $\mathbf{E}_{:,1:K}$ consist of the top-K eigenvectors of $\mathbf{C}$, since $\mathbf{C}$ is a positive scaling of $\mathbf{\Sigma}$ per (4.9), and we refer to $\tilde{\mathbf{R}}$ as the *connecting rotation matrix*. Therefore, asymptotically, the solution of Lp-PCA spans the same subspace as that of the dominant eigenvectors (and the standard PCs). The proof is presented in Appendix B. Moreover, we note that any rotation of the eigenvectors is a valid solution to the asymptotic L2-PCA problem, but a specific rotation of the eigenvectors is a solution to the asymptotic Lp-PCA problem. that is,

Pseudocode for the Connecting Rotation Matrix Computation

Input: $\mathbf{\Lambda}$

$\tilde{\mathbf{\Lambda}} = \mathbf{\Lambda}$, $K = \texttt{size}(\mathbf{\Lambda}, 1)$, $\tilde{\mathbf{R}} = \mathbf{I}_K$, and $\delta = 10^{-10}$
while(true)
  if $\left|\tilde{\mathbf{\Lambda}}(end) - \texttt{mean}(\texttt{diag}(\mathbf{\Lambda}))\right| < \delta$
    break
  end
  $(\mathbf{A}, i, j) \leftarrow \texttt{submatrix}(\tilde{\mathbf{\Lambda}})$
  $\mathbf{R}' \leftarrow \texttt{rotmatrix}(\mathbf{A}, i, j, K)$
  $\tilde{\mathbf{\Lambda}} = \mathbf{R}'^{\top} \tilde{\mathbf{\Lambda}} \mathbf{R}'$
  $\tilde{\mathbf{R}} = \tilde{\mathbf{R}} \mathbf{R}'$

Output: $\tilde{\mathbf{R}} \in \mathbb{S}_{K,K}$.

Function: $(\mathbf{A}, i, j) \leftarrow \texttt{submatrix}(\mathbf{B})$

$(i, j) \leftarrow \text{argmax}_{\tilde{i},\tilde{j} \in \{1,2,\ldots,K\}} \left|B_{\tilde{i},\tilde{i}} - B_{\tilde{j},\tilde{j}}\right|$
$\mathbf{A} = \begin{bmatrix} B_{i,i}, B_{i,j} \\ B_{j,i}, B_{j,j} \end{bmatrix}$
Return $\mathbf{A}, i, j$

Function: $\mathbf{R}' \leftarrow \texttt{rotmatrix}(\mathbf{A}, i, j, K)$

$\mathbf{R}' = \mathbf{I}_K$
$\alpha = \frac{A_{2,1}+A_{1,2}}{A_{1,1}-A_{2,2}}$
$\beta = 0.5\texttt{cot}^{-1}(\alpha)$
$R'_{i,i} = \texttt{cos}(\beta)$, $R'_{j,j} = \texttt{cos}(\beta)$, $R'_{j,i} = \texttt{sin}(\beta)$, $R'_{i,j} = \texttt{-sin}(\beta)$
Return $\mathbf{R}'$

Figure 4.6. Algorithm to compute the connecting rotation matrix that rotates the dominant eigenvectors of the covariance matrix, $\mathbf{C}$, to align with the asymptotic Lp-PCs.

Lp-PCA solves L2-PCA asymptotically, however, not every asymptotic L2-PCA solution solves asymptotic Lp-PCA.

As an outcome of Lemmas 1, 5-8, the following theorem holds true.

Theorem 2: For $0 < p < 2$ and i.i.d. data points drawn from a zero-mean Elliptical distribution with covariance matrix $\mathbf{C}$, the subspace spanned by the asymptotic Lp-PCs in (4.12) coincides with the maximum-variance subspace of the distribution and consequently, with the dominant eigensubspace (or the asymptotic L2-PCs) of $\mathbf{C}$. Moreover, the asymptotic Lp-PCs are rotated versions of the dominant eigenvectors.

Connecting rotation matrix estimation algorithm. Equality in Lemma 7 is achieved if and only if $\forall j \in \{1, 2, \ldots, K\}$, $\mathbf{r}_j^{\top} \mathbf{\Lambda} \mathbf{r}_j$ is a constant. Moreover, per Corollary 2 of Lemma 8, for $D > K$, the bottom $D - K$ rows of the optimal rotation matrix must contain zeros. In other words, for any

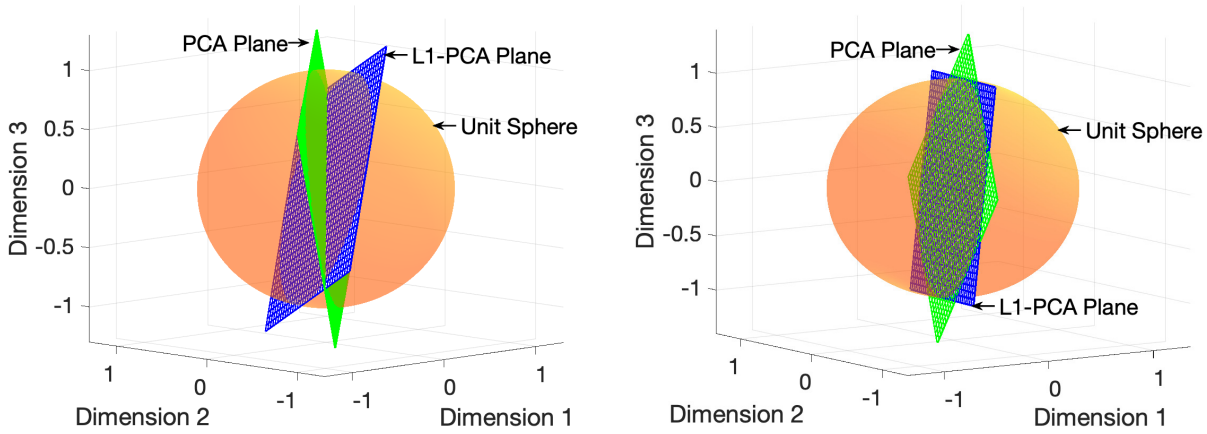


Figure 4.7. Graphical illustration of the asymptotic coincidence of the subspaces of L1-PCA and PCA for Gaussian data and $K = 2$. (a) $N = 10$, the L1-PCA plane does not coincide with that of PCA. (b) $N = 10^8$, planes of L1-PCA and PCA coincide. The angle of rotation between the L1-PCs and the PCs is 45°.

K, we seek a rotation matrix of the form

$$\mathbf{R}^* = \begin{bmatrix} \tilde{\mathbf{R}} \in \mathbb{S}_{K,K} \\ \mathbf{0}_{D-K \times K} \end{bmatrix}, \tag{4.22}$$

such that $\mathbf{R}^{*\top} \mathbf{\Lambda} \mathbf{R}^*$ results in a matrix with equal diagonal entries, where $\tilde{\mathbf{R}} \in \mathbb{S}_{K,K}$ is the connecting rotation matrix between the dominant eigenvectors, $\mathbf{E}_{:,1:K}$, and the asymptotic Lp-PCs, $\mathbf{Q}_{Lp}$. That is, we seek a $K \times K$ rotation matrix, such that $\mathbf{R}^{*\top} \mathbf{\Lambda} \mathbf{R}^*$ achieves unitary similarity to a matrix with equal diagonals [232]. In order to compute the optimal rotation matrix, we need access to the true (distribution) eigen (or singular) values. However, in a practical setting, since the population eigen (or singular) values are unavailable, we approximate them by their sample estimates. For any K, given the top K singular (or eigen) values, we compute the optimal rotation matrix that rotates the dominant eigenvectors to coincide with the Lp-PCs using the algorithm in Figure 4.6 [232].

Toy example for the visual representation of asymptotic coincidence of PCA and L1-PCA. First, we note that for $K = 2$, regardless of the eigen (singular) values, the optimal 2×2 sub-rotation matrix is always $\tilde{\mathbf{R}} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$ and $\mathbf{R}^*$ is formed by arranging $\tilde{\mathbf{R}}$ on top with the bottom $D - K$ rows being zeros as in (4.22). That is, the asymptotic Lp-PCs are 45° rotated versions of asymptotic PCs. We demonstrate this visually in Figure 4.7 for Gaussian data. We form a data matrix $\mathbf{X} \in \mathbb{R}^{3 \times N}$, whose entries are drawn from a multivariate zero-mean Gaussian

distribution. Next for $N = 10$, we estimate the PCs using SVD and L1-PCs using the algorithm of [6]. We observe from Figure 4.7(a) that the PCs that span the PCA plane and the L1-PCs that span the L1-PCA plane do not coincide. That is, the L1-PCs and standard PCs span different subspaces. However, when we set $N = 10^8$ and repeat the experiment, we observe from Figure 4.7(b) that the L1-PCA plane and PCA plane coincide exactly and span the same 2-dimensional subspace. Moreover, the subspace angle (angle between the PCs and L1-PCs) is 45° as predicted in theory.

4.4.2 Experimental Studies

In this section, we demonstrate empirically the asymptotic coincidence of Lp-PCA and PCA on data drawn from multiple members of the Elliptical distribution. Moreover, we leverage the proposed theory to offer *intelligent* initialization for the iterative algorithms of Lp-PCA, thereby resulting in faster and/or convergence to higher objective values on both synthetic and real-world datasets following the Elliptical distribution. Additionally, we demonstrate the general applicability of the proposed intelligent initialization scheme by employing it on multiple real-world non-elliptical datasets and achieving much faster and better convergence. First, we present our result on synthetic datasets.

Studies on Synthetic Data

Asymptotic coincidence. In this experiment we demonstrate the asymptotic convergence of Lp-PCA to the underlying dominant eigensubsapce for various p values and data distributions. We set $D = 3$, $K = 2$, to form the covariance matrix $\mathbf{C} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^\top$, where $\mathbf{E} \in \mathbb{S}_{D\times D}$ and $\mathbf{\Lambda} = \texttt{diag}(200, 45, 25)$. We draw data from zero-mean Gaussian, Laplace, and Power Exponential (or Generalized Gaussian) distribution. The Power Exponential distribution is a generalization of the Gaussian distribution and is parameterized on a shape parameter κ. For $\kappa = 1$ and $\kappa = 0.5$, it coincides with Gaussian and Laplace distribution respectively. Power Exponential distribution allows for tails that are heavier and lighter than that of Gaussian for $\kappa < 1$ and $\kappa > 1$ respectively. In our studies, we use $\kappa = 0.75$ to obtain tails heavier than Gaussian, $\kappa = 1.5$ and $\kappa = 2.5$ to obtain tails lighter than Gaussian. We form the data matrix $\mathbf{X} \in \mathbb{R}^{D\times N}$ for various values of $N \in [10, 50, 10^2, 10^3, 10^4, 10^5]$ and $p \in [0.5, 0.75, 1, 1.5]$ and employ the iterative Lp-PCA algorithm of [19] to obtain the respective subspaces. Next, we compute the proximity of the computed subspace $\mathbf{Q} \in \mathbb{S}_{D,K}$ to that of the underlying optimal eigenbasis $\mathbf{E} \in \mathbb{S}_{D,K}$ by estimating the mean

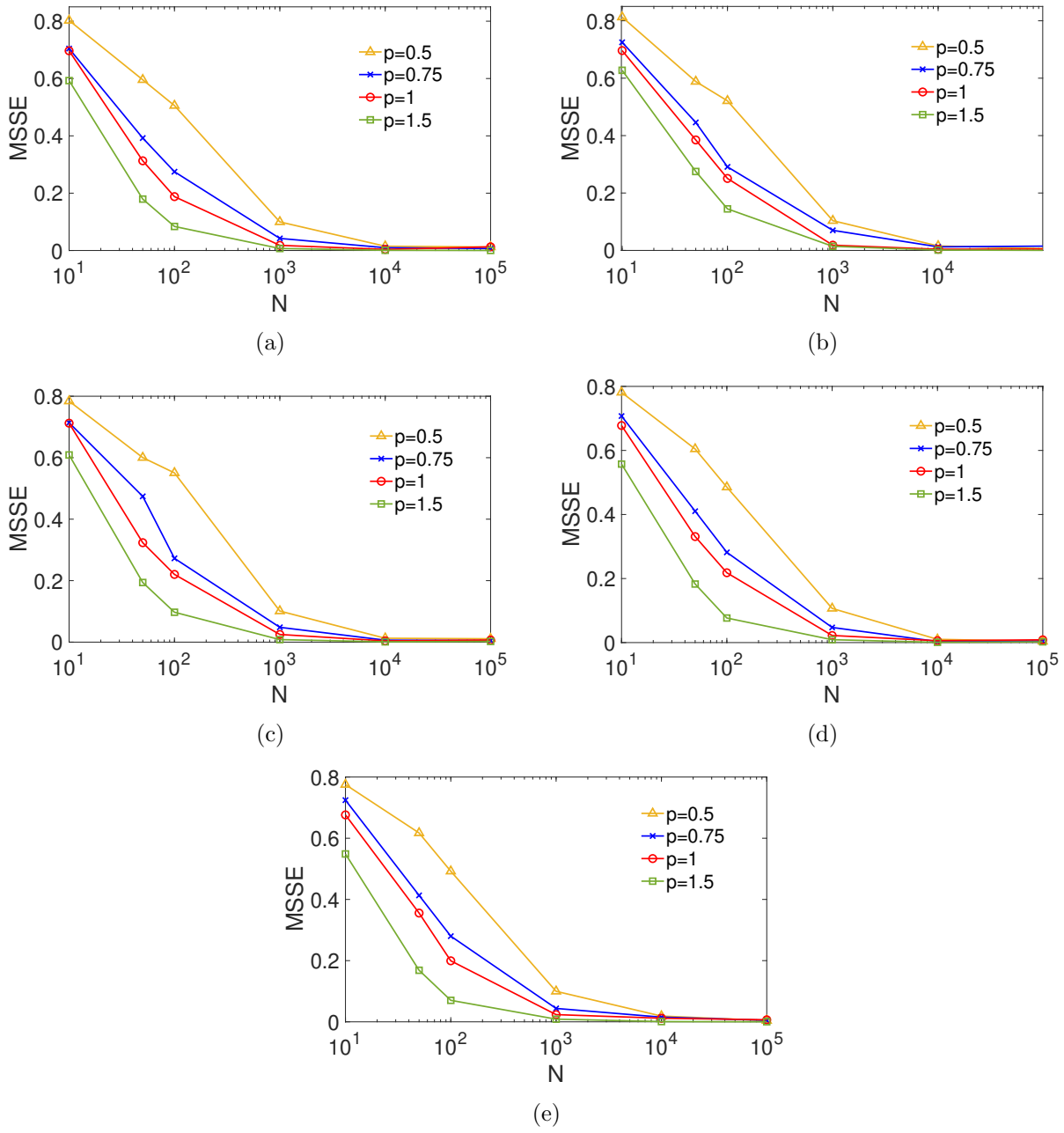


Figure 4.8. MSSE versus N for various values of p, with data drawn from (a) Gaussian, (b) Laplace, (c) Power Exponential ($\kappa = 0.75$), (d) Power Exponential ($\kappa = 1.5$), (e) Power Exponential ($\kappa = 2.5$).

subspace error (MSSE) as

$$\text{MSSE} = \frac{1}{S}\sum_{i=1}^{S} \|\mathbf{E}_{:,1:K}\mathbf{E}_{:,1:K}^{\top} - \mathbf{Q}_i\mathbf{Q}_i\|_2^2, \tag{4.23}$$

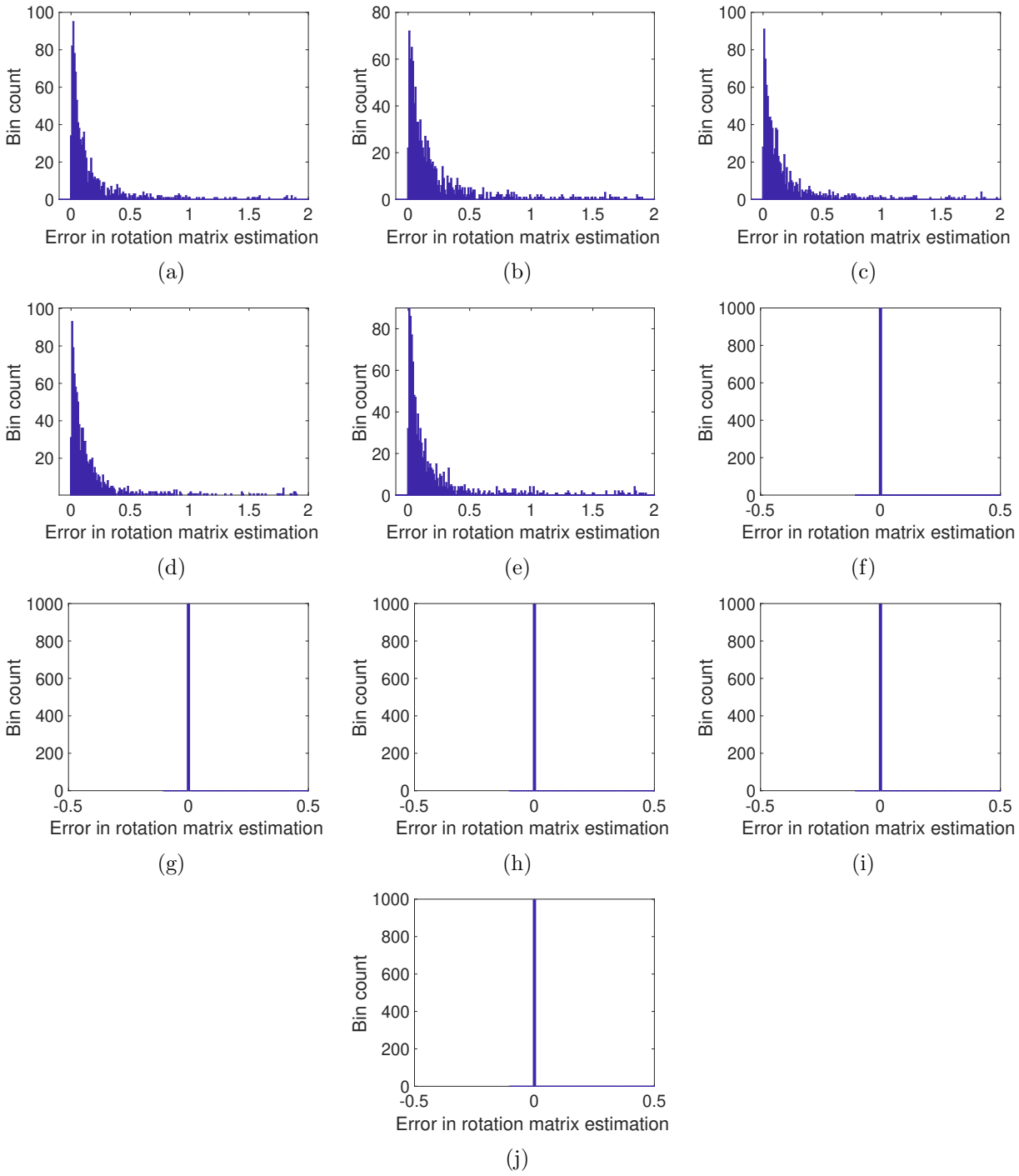


Figure 4.9. Histogram of rotation matrix estimation error for $p = 0.75$, $N = 10$ with data drawn from (a) Gaussian, (b) Laplace, (c) Power Exponential ($\kappa = 0.75$), (d) Power Exponential ($\kappa = 1.5$), (e) Power Exponential ($\kappa = 2.5$). Histogram of rotation matrix estimation error for $p = 0.75$, $N = 10^6$ with data drawn from (f) Gaussian, (g) Laplace, (h) Power Exponential ($\kappa = 0.75$), (i) Power Exponential ($\kappa = 1.5$), (j) Power Exponential ($\kappa = 2.5$).

where S is the number of independent realizations of the experiment, set to 10^3 in this experiment and $\mathbf{Q}_i$ is the subspace computed at the i-th realization. We plot the MSSE versus N for Gaussian, Laplace, Power Exponential with $\kappa = 0.75$, 1, and 1.5 in Figures 4.8 (a), (b), (c), (d), and (e) respectively. We observe that consistently across data distributions and various values of p, the MSSE starts at a non-zero value for small N values, while converging to zero for large values of N (around $N = 10^4$), demonstrating asymptotic convergence. Interestingly, we note that larger values of p achieve faster convergence to lower MSSE.

Next, we show that not only, the Lp-PCs span the same subspace as the PCs, they are specific rotated version of the PCs by computing the histogram of rotation matrix estimation error as $\|\mathbf{R}^* - \hat{\mathbf{R}}\|_2^2$, where $\mathbf{R}^* = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 \end{bmatrix}$ as predicted by theory for $K = 2$, $D = 3$, and $\hat{\mathbf{R}} = \mathbf{E}^\top \widehat{\mathbf{Q}_{Lp}}$, where $\widehat{\mathbf{Q}_{Lp}}$ consists of the Lp-PCs estimated by the iterative algorithm of [19]. We plot the histogram of the rotation matrix estimation error computed over 1000 realization on data drawn from Gaussian, Laplace, Power Exponential with $\kappa = 0.75$, 1, and 1.5 data for $N = 10$ and $p = 0.75$ in Figures 4.9 (a)-(e) and observe that the estimated rotation between Lp-PCs and PCs and the expected rotation predicted by theory do not coincide. However, for $N = 10^6$, we observe that the histogram shows a spike at zero with a magnitude of 1000, signifying that the estimated rotation and the expected rotation coincide exactly across all realizations. We repeat the experiment for $p = 1$ in Figure 4.10 and make similar observations.

Intelligent initialization for large N**.** In the next set of experiments, we leverage the proposed theory to intelligently initialize the iterative algorithms of Lp-PCA. Traditionally, Lp-PCA (and L1-PCA) algorithms are initialized randomly or at the estimated dominant eigenvectors of the covariance matrix [4, 6, 19, 22, 233]. We demonstrate that such initialization schemes are outperformed by the proposed intelligent initialization, wherein we initialize at specific rotation of the dominant eigenvectors as obtained by the algorithm in Figure 4.6. We observe that the proposed initialization consistently outperforms other initialization schemes across different algorithms and datasets. Specifically, since we initialize at the expected solution (rotated asymptotic PCs), the iterative algorithms converge at the initialization for large N and converge faster to better L1-PCA objective value in very few iterations for small N. Firstly, we plot the L1-PCA objective value $\frac{1}{N}\|\mathbf{Q}^\top \mathbf{X}\|_1$ versus the number of iterations of the algorithm in [6], where $\mathbf{Q}$ is the L1-subspace estimated by the algorithm of [6], $\mathbf{X} \in \mathbb{R}^{D=3 \times N=15000}$ is the data matrix with entries drawn from different distributions in Figure 4.11. Results for data drawn from zero-mean Gaussian, Laplace, Logistic, Power Exponential with $\kappa = 0.75$, 1.5, and 2.5, and Student's t with degress of

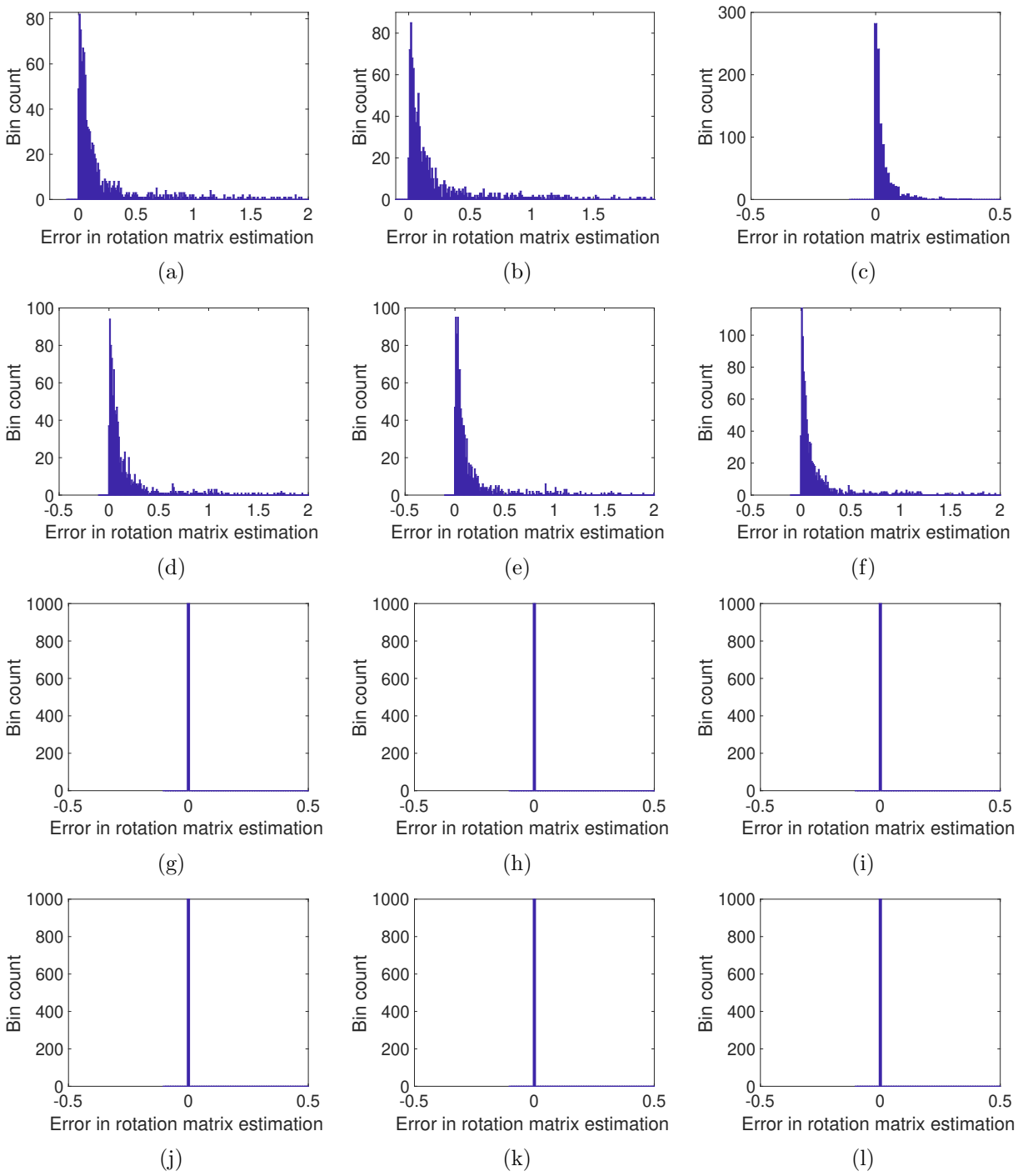


Figure 4.10. Histogram of rotation matrix estimation error for $p = 1$, $N = 10$ with data drawn from (a) Gaussian, (b) Laplace, (c) Logistic, (d) Power Exponential ($\kappa = 0.75$), (e) Power Exponential ($\kappa = 1.5$), (f) Power Exponential ($\kappa = 2.5$). Histogram of rotation matrix estimation error for $p = 1$, $N = 10^6$ with data drawn from (a) Gaussian, (b) Laplace, (c) Logistic, (d) Power Exponential ($\kappa = 0.75$), (e) Power Exponential ($\kappa = 1.5$), (f) Power Exponential ($\kappa = 2.5$).

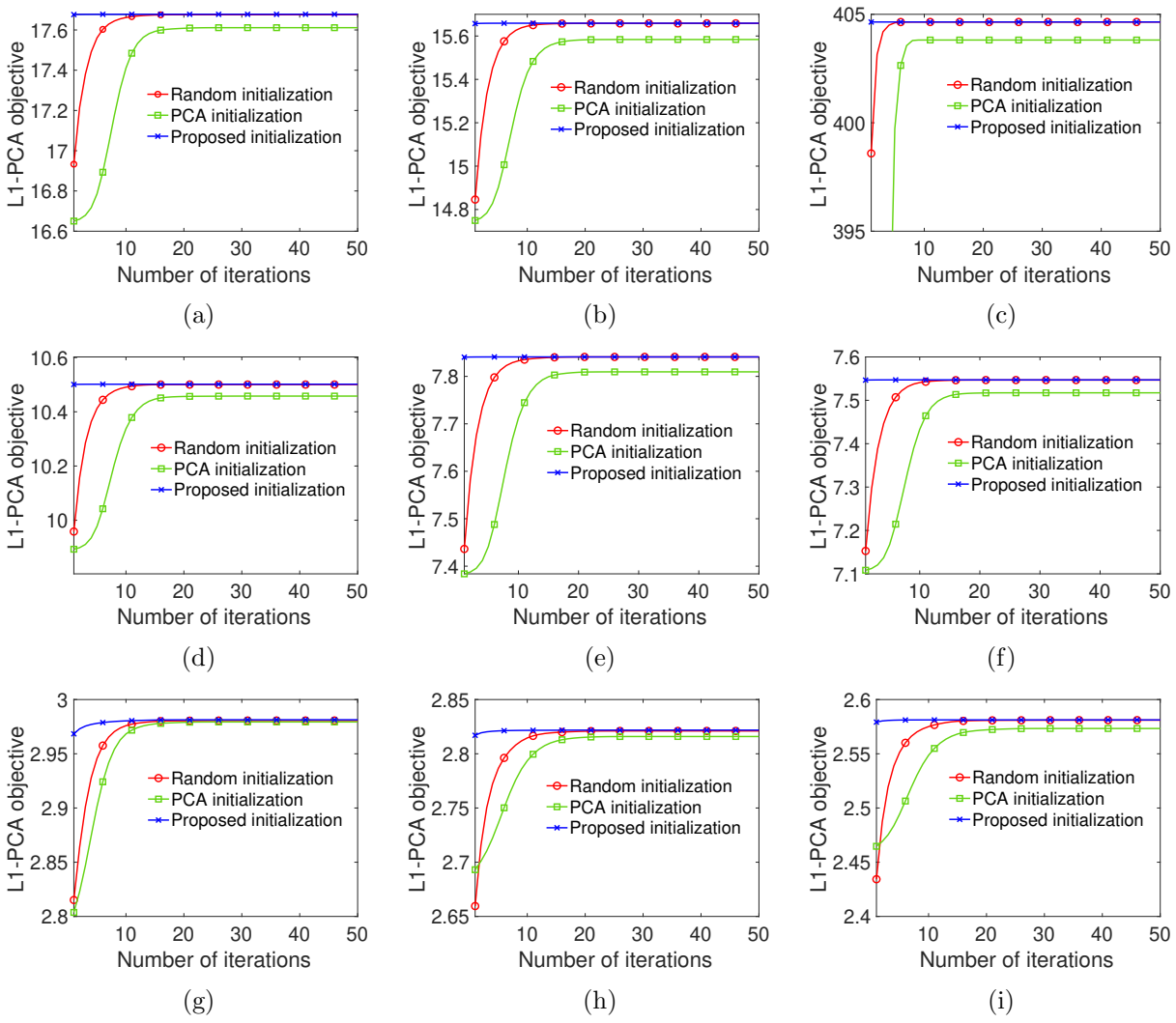


Figure 4.11. L1-PCA objective versus number of iterations for $p = 1$, $N = 15000$, with different initializations and data drawn from (a) Gaussian, (b) Laplace, (c) Logistic, (d) Power Exponential ($\kappa = 0.75$), (e) Power Exponential ($\kappa = 1.5$), (f) Power Exponential ($\kappa = 2.5$), (g) Student's t ($\nu = 2.25$), (h) Student's t ($\nu = 2.5$), and (i) Student's t ($\nu = 3$).

freedom $\nu = 2.25$, 2.5, and 3 are presented in Figure 4.11 (a)-(i) respectively. We observe that across the board, the proposed initialization leads to quick convergence to the highest objective value. Random initialization achieves the objective value achieved by the proposed initialization but consumes more iterations, however, initialization to eigenvectors (or L2-PCs) takes the longest to converge and at convergence, it achieves lower L1-PCA objective value. We offer more details on why PCA initialization fails in Appendix B. We repeat the experiment with L1-BF [4] in Figure 4.12 for $N = 1500$, wherein we observe that random initialization and PCA initialization require significantly more iterations to converge, although random initialization is slightly better compared to PCA initialization. Interestingly, a special initialization scheme proposed in [4], namely the

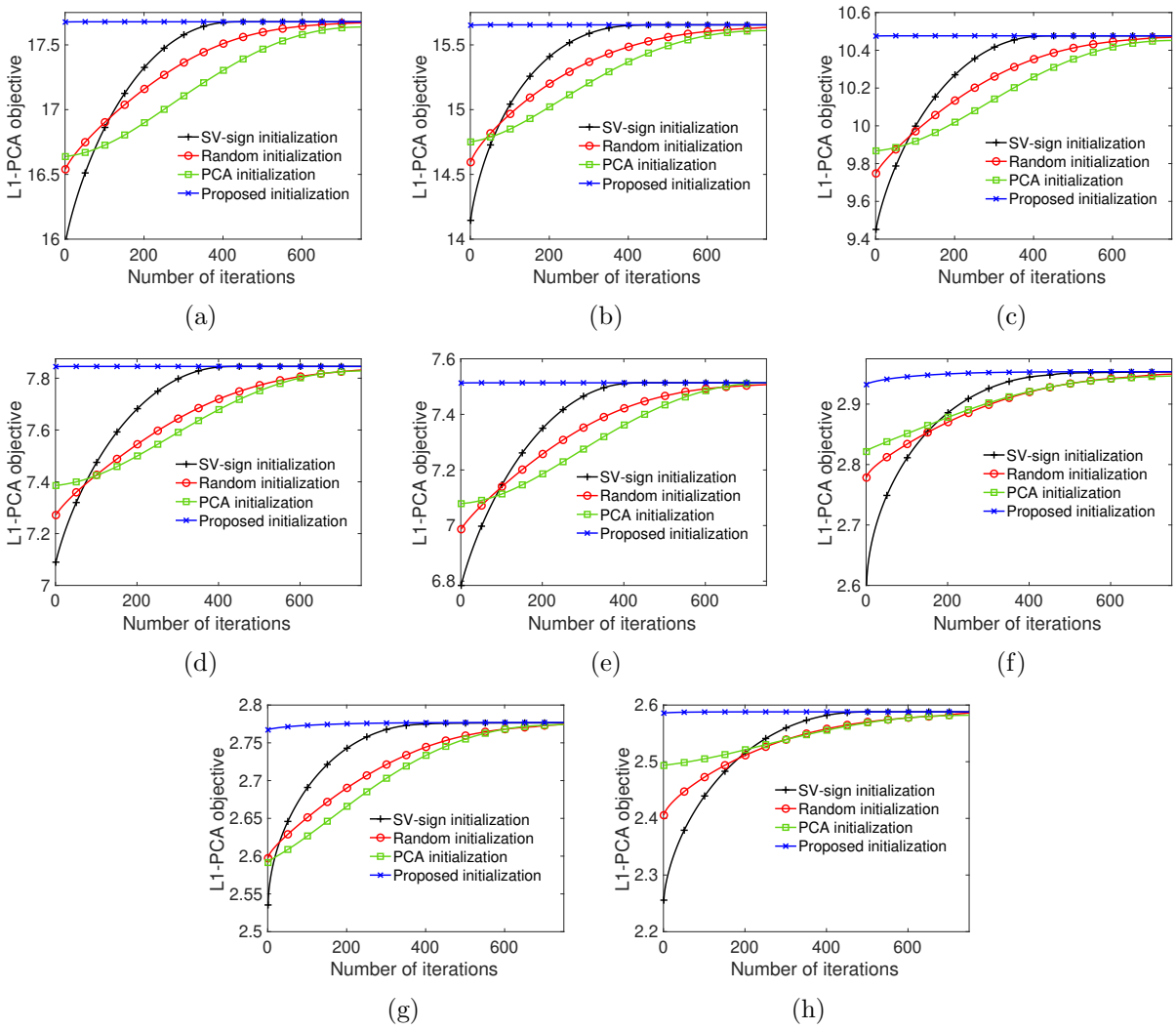


Figure 4.12. L1-PCA objective versus number of iterations of the L1-BF algorithm for $p = 1$, $N = 1500$, with different initializations and data drawn from (a) Gaussian, (b) Laplace, (c) Power Exponential ($\kappa = 0.75$), (d) Power Exponential ($\kappa = 1.5$), (e) Power Exponential ($\kappa = 2.5$), (f) Student's t ($\nu = 2.25$), (g) Student's t ($\nu = 2.5$), and (h) Student's t ($\nu = 3$).

SV-sign initialization achieves much faster convergence compared to random and PCA initialization. However, the proposed initialization scheme outperforms all other initialization schemes by converging extremely quickly to the highest metric value across the board.

Intelligent initialization for small N. In this experiment, we demonstrate the benefit of the proposed initialization scheme on datasets with small values of N. Specifically, we plot the L1-PCA objective achieved by L1-BF [4] versus the number of iterations of the algorithm for various initializations on Gaussian, Laplace, and Logistic data with $N = 100$ in Figures 4.13 (a)-(c) respectively. We observe that the proposed initialization converges closest to the optimal L1-PCA metric obtained by the optimal algorithm of [3] across all datasets. The SV-sign initialization

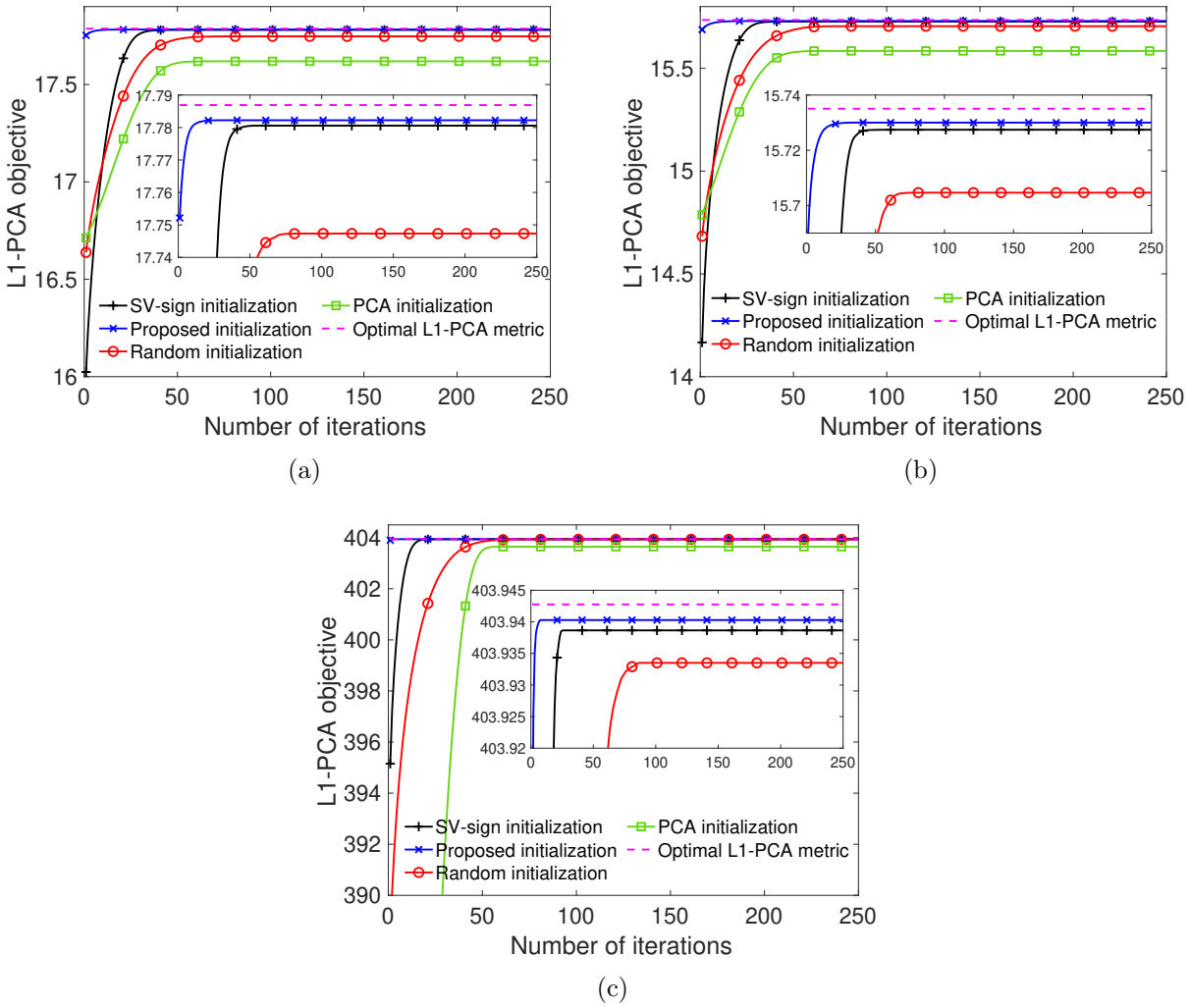


Figure 4.13. L1-PCA objective versus number of iterations of the L1-BF algorithm for $p = 1$, $N = 100$, with different initializations and data drawn from (a) Gaussian, (b) Laplace, and (c) Logistic.

proposed in [4] demonstrates the next best performance followed by random initialization, and PCA initialization.

This concludes our experiments on synthetic data. In summary, we have empirically demonstrated the proposed convergence theory on various members of Elliptical distribution for different values of p. Moreover, we illustrate the superior advantage of the proposed intelligent initialization scheme for faster and/or better convergence in terms of the Lp-PCA metric.

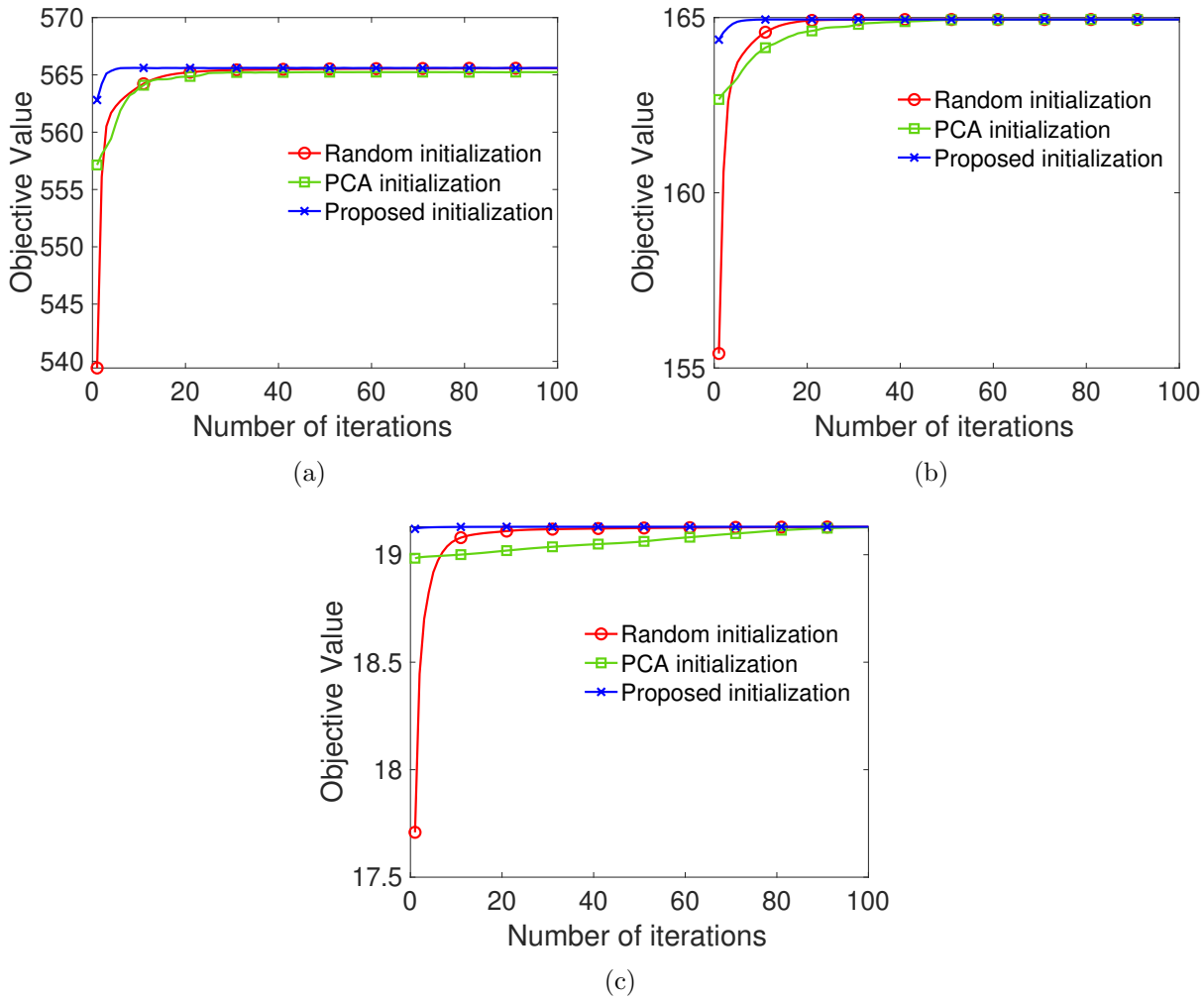


Figure 4.14. L1-PCA objective versus number of iterations on stock return data of Microsoft, Boeing, and Ford collected from Jan. 2000 to Feb. 2020 for (a) $p = 0.75$, (b) $p = 1$, and (c) $p = 1.5$.

Studies on Real-world Data

In this section, we take advantage of the proposed theory to initialize the iterative algorithms of Lp-PCA intelligently. Across our experiments, we observe that the proposed initialization results in faster convergence to higher objective value on both real-world elliptical and non-elliptical data.

Stock returns data (Elliptical). In our first experiment on real-world data, we operate on stock return data obtained from Yahoo Finance [234]. We read in [217–219, 235] that stock returns can be modeled well by a multivariate Student's t distribution, a member of the Elliptical distribution. In our experiments we make use of two datasets. First dataset consists of stock returns of Microsoft, Boeing, and Ford from January 2000 to February 2020 arranged as the columns of

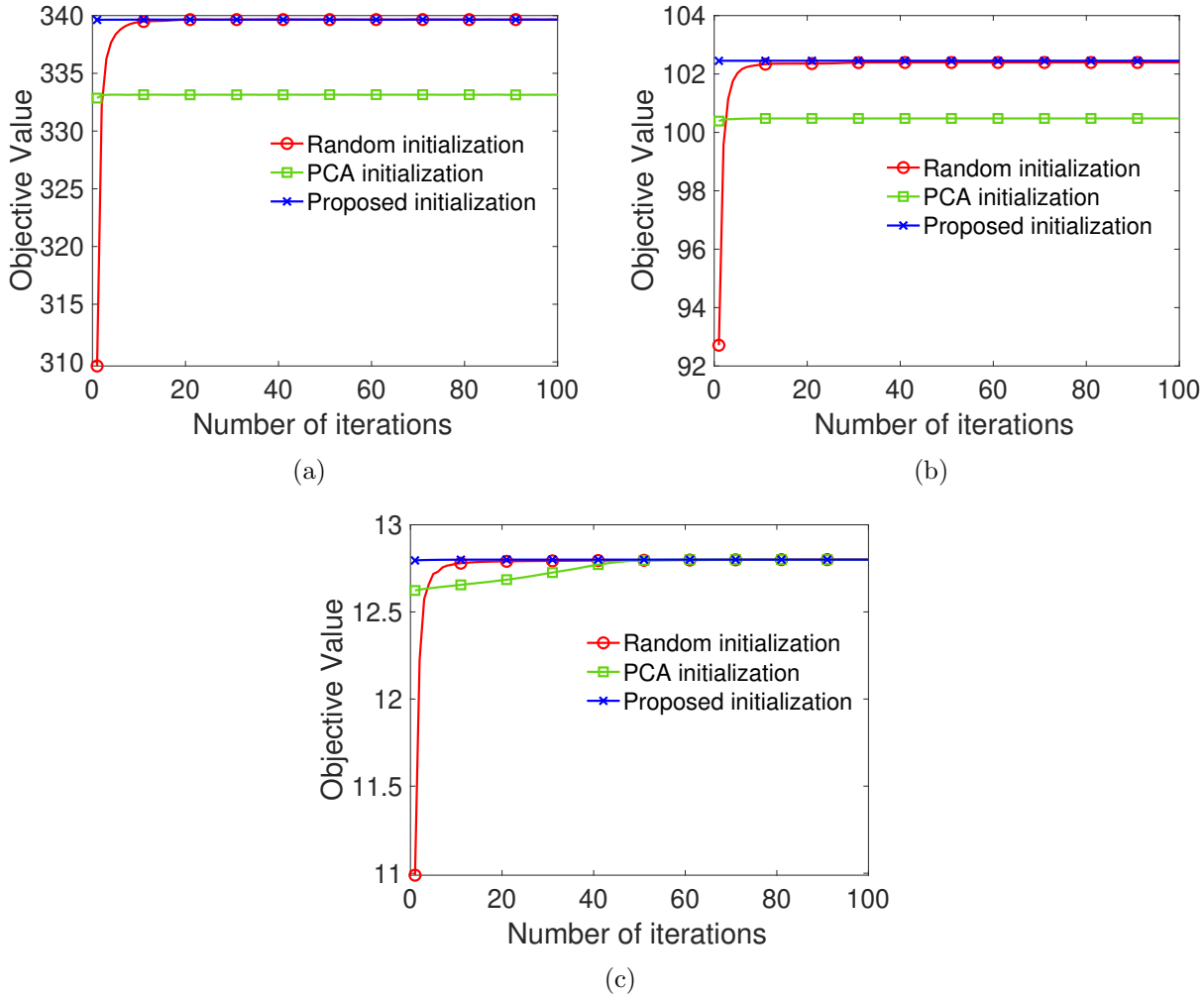


Figure 4.15. L1-PCA objective versus number of iterations on stock return data of Microsoft, Amazon, and Netflix collected from Jan. 2010 to Dec. 2020 for (a) $p = 0.75$, (b) $p = 1$, and (c) $p = 1.5$.

$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3] \in \mathbb{R}^{D=3 \times N=5070}$, where $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$ are the stock returns of Microsoft, Boeing, and Ford respectively. We zero-center the data matrix $\mathbf{X}$ and set $K = 2$ (chosen to capture at least 75% variance of the dataset in all our studies on real-world data). We plot the Lp-PCA objective value versus the number of iterations of the algorithm in [19] for $p = 0.75$, 1, and 1.5 in Figures 4.14 (a)-(c), wherein we observe that the proposed initialization quickly converges to the highest objective value across p values, whereas random initialization and PCA initialization converge much slower. In all our experiments involving the proposed initialization scheme, we compute the rotation matrix, $\mathbf{R}^*$, using the algorithm in Figure 4.6 and then obtain the proposed initialization as $\widehat{\mathbf{E}}\mathbf{R}^*$, where $\widehat{\mathbf{E}}$ is the estimated eigenvectors of the covariance matrix (or the SVD of $\mathbf{X}$).

Our second dataset $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3] \in \mathbb{R}^{D=3 \times N=2768}$, where $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$ are the stock returns of Microsoft, Amazon, and Netflix respectively from January 2010 to December 2020. We make

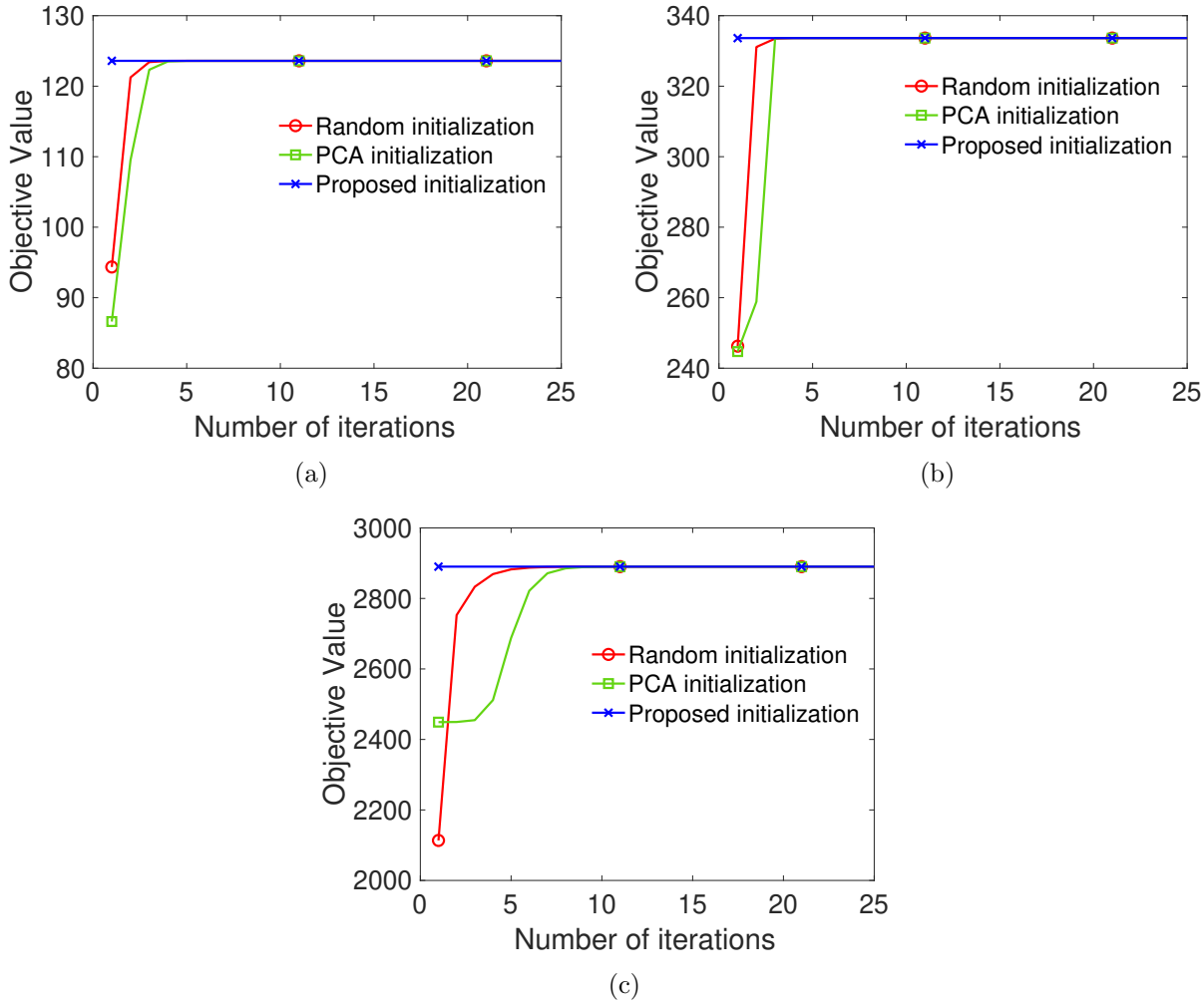


Figure 4.16. L1-PCA objective versus number of iterations on body measurements data from Kaggle for (a) $p = 0.75$, (b) $p = 1$, and (c) $p = 1.5$.

similar observations for the proposed initialization and random initialization as before – fast convergence to the highest metric for proposed initialization and slower convergence to possibly lower metric value when random initialization is employed. Interestingly, we observe that for $p = 0.75$ and 1, PCA initialization converges to much lower metric without any improvement. For $p = 1.5$ however, it converges to the metric value obtained by the proposed initialization after about 50 iterations. This observation is consistent with that in [19] on the existence of multiple local optima for $p \leq 1$ and the inability of the algorithm to converge to the global solution frequently. Therefore, the above studies demonstrate that the proposed initialization is successful in achieving a high objective value, thereby signifying convergence close to (or to) the global optimum.

Body measurements dataset (Elliptical). In this experiment, we operate on the Kaggle body measurements dataset [236]. The dataset contains body measurements of 250 individuals, measur-

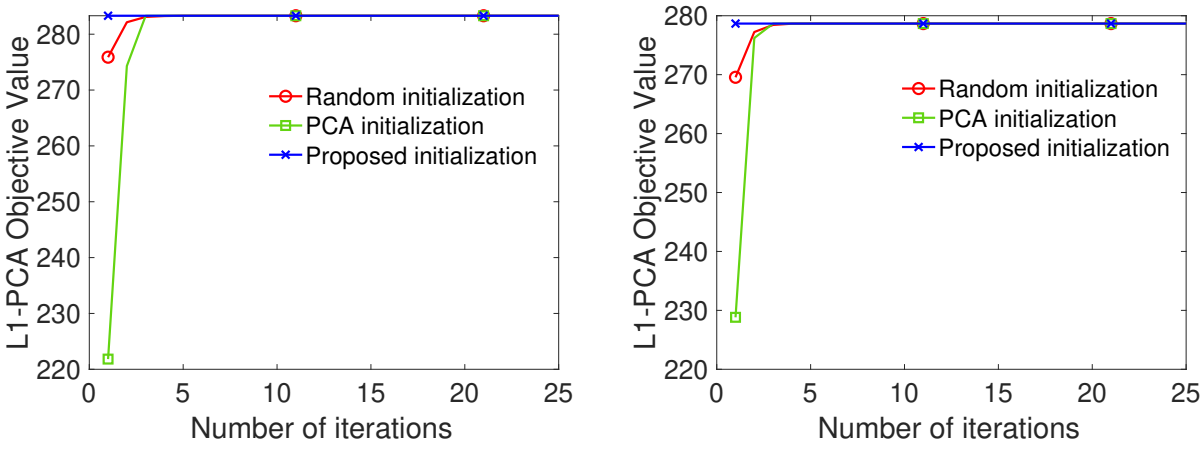


Figure 4.17. L1-PCA objective versus number of iterations for $p = 1$ on texture images of (a) tree bark and (b) leaves.

ing their age in years, height in inches, weight in pounds, chest circumference in cm, abdominal circumference in cm, and fat percentage. We form the data matrix $\mathbf{X}$ of size 5×250 with each row corresponding to the body measurements mentioned above except age. This dataset follows a multivariate Gaussian distribution. We compute the objective value for $p = 0.75$, 1, and 1.5 using the algorithm of [19] and plot it versus the number of iterations of the algorithm in Figures 4.16 (a)-(c). We observe that across p values, the proposed initialization converges in a single iteration of the algorithm, while PCA and random initialization converge slightly slower.

Texture images (Elliptical). The wavelet statistics of texture images such as those of a tree bark, leaves, grass, sand, among others follow a Generalized Gaussian distribution [223]. We make use of texture images from the VisTex dataset [237] and form the data matrix $\mathbf{X} \in \mathbb{R}^{3\times 262144}$, whose columns are obtained by filtering the red, green , and blue channels of a 512×512 sized image using the stationary wavelet transform with db4 wavelet as suggested in [223]. We perform two experiments, one on a tree bark image and another on a leaves image. We plot the L1-PCA objective value versus the number of iterations of [6] in Figures 4.17 and observe that the proposed initialization converges in a single iteration while random and PCA initialization converge after a few iterations.

Convolutional layer weights of a CNN (non-Elliptical). In this experiment, we operate on the convolutional layer weights of convolutional neural networks (CNNs). In specific, we operate on the convolutional weights of layer 14 of AlexNet [238], layer 337 of ResNet-101 [239], and layer

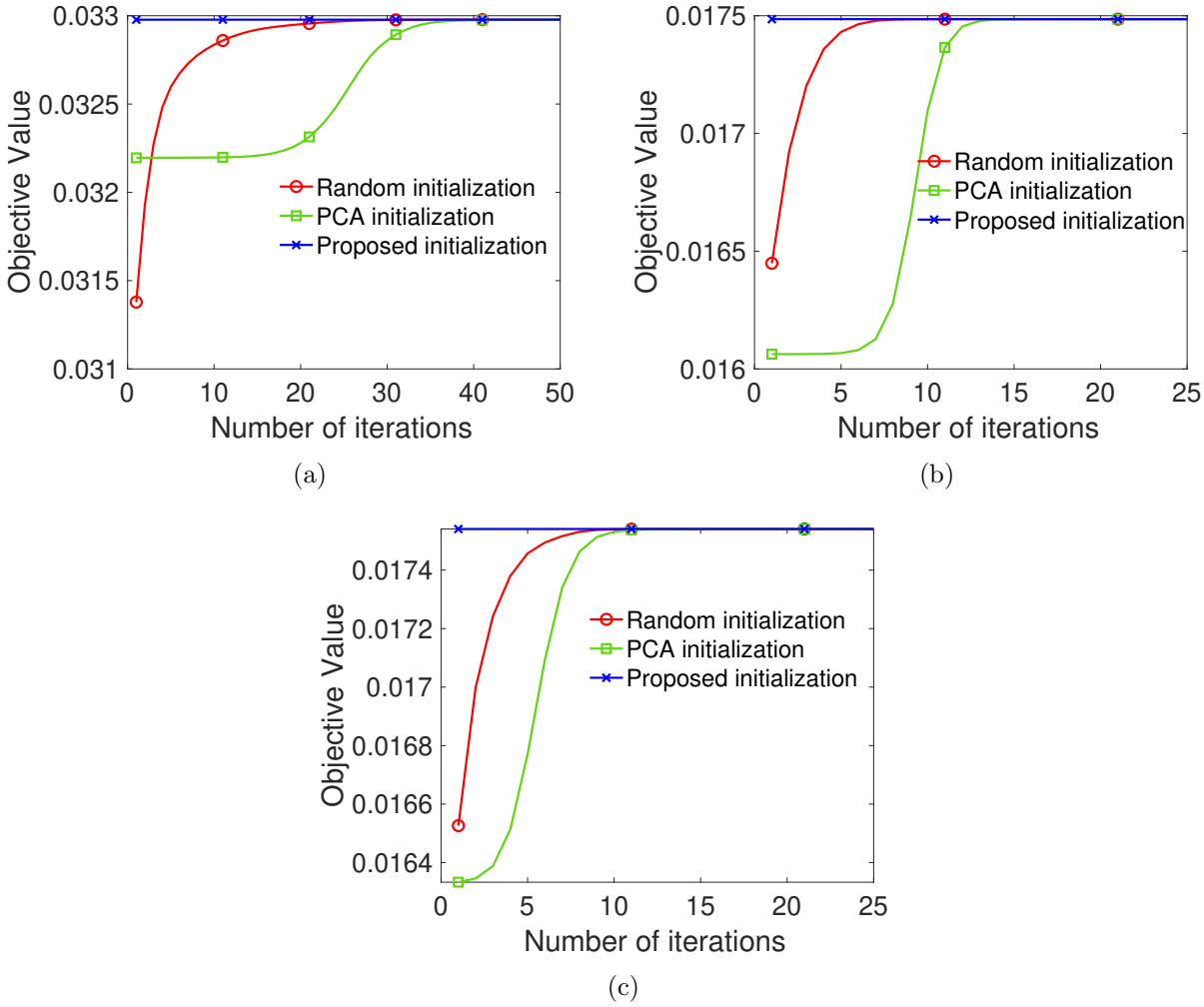


Figure 4.18. L1-PCA objective versus number of iterations for $p = 1$ on convolutional layer weights of (a) layer 14 of AlexNet, (b) layer 337 of ResNet-101, and (b) layer 16 of VGG-19.

16 of VGG-19 [240]. For AlexNet, the 14-th layer is a convolution layer of size $3 \times 3 \times 128 \times 192$, which is reshaped into a matrix $\mathbf{X}$ of size 3×73728 by arranging the mode-1 fibers of the original convolutional layer as the columns of $\mathbf{X}$. Similarly, we reshape the 337-th layer of ResNet-101 of size $3 \times 3 \times 512 \times 512$ into a matrix $\mathbf{X}$ of size 3×786432. Finally, the 16-th layer of VGG-19 of size $3 \times 3 \times 256 \times 256$ into a matrix $\mathbf{X}$ of size 3×196608. We plot the L1-PCA objective value versus the number of iterations of the algorithm of [6] in Figures 4.18 (a)-(b) obtained on the weights of AlexNet, ResNet-101, and VGG-19 respectively. In consistent with our observations on elliptical data, we observe that the proposed initialization converges at the very first iteration. Random initialization converges slower than the proposed but faster than PCA initialization, which takes significantly higher algorithm iterations to converge.

Fully-connected layer weights of a CNN (non-Elliptical). We operate on the fully-connected

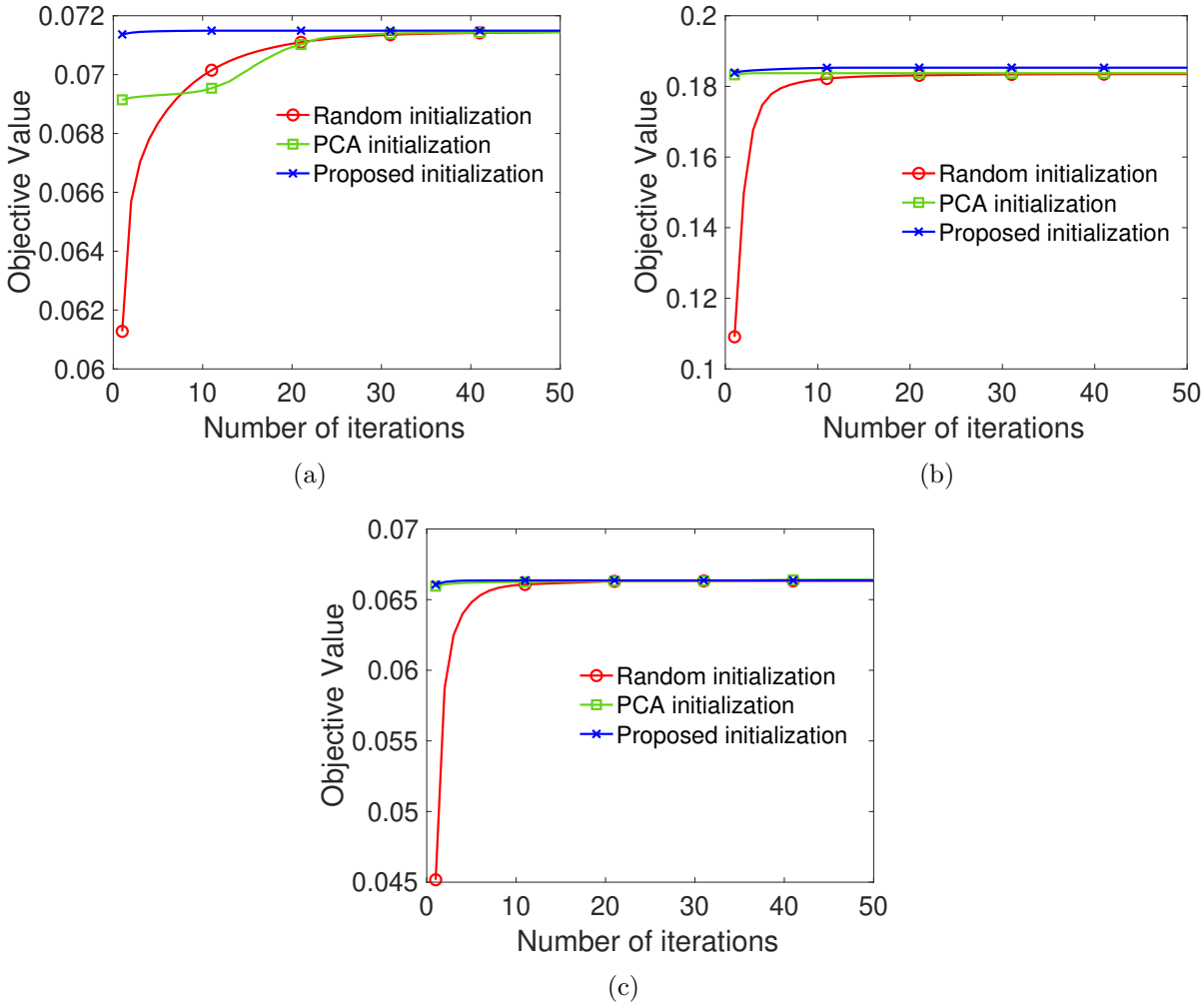


Figure 4.19. L1-PCA objective versus number of iterations for $p = 1$ on full-connected layer weights of (a) layer 20 of AlexNet, (b) layer 345 of ResNet-101, and (b) layer 45 of VGG-19.

layer weights of the same CNNs as in the previous experiment. Specifically, we operate on the weights of layers 20, 345, and 45 of AlexNet, ResNet-101, and VGG-19 respectively. Layer 20 of AlexNet is a fully-connected layer of size 4096×4096, layer 345 of ResNet-101 is a fully-connected layer of size 1000×2048, and layer 45 of VGG-19 is a fully-connected layer of size 1000×4096. We compute the L1-PCA objective value of each of these layers using the algorithm in [6] with $K = 2$ and plot it versus the number of iterations in Figures 4.19(a)-(c). We observe that again, the proposed initialization leads to very quick convergence. PCA initialization achieves the next fastest convergence, followed by random initialization.

Human activity detection dataset (non-Elliptical). In this experiment, we operate on the human activity detection dataset [241]. The dataset consists of five human activities, namely, sitting, standing, walking, running, and dancing. The dataset is of size 60×24075. We compute

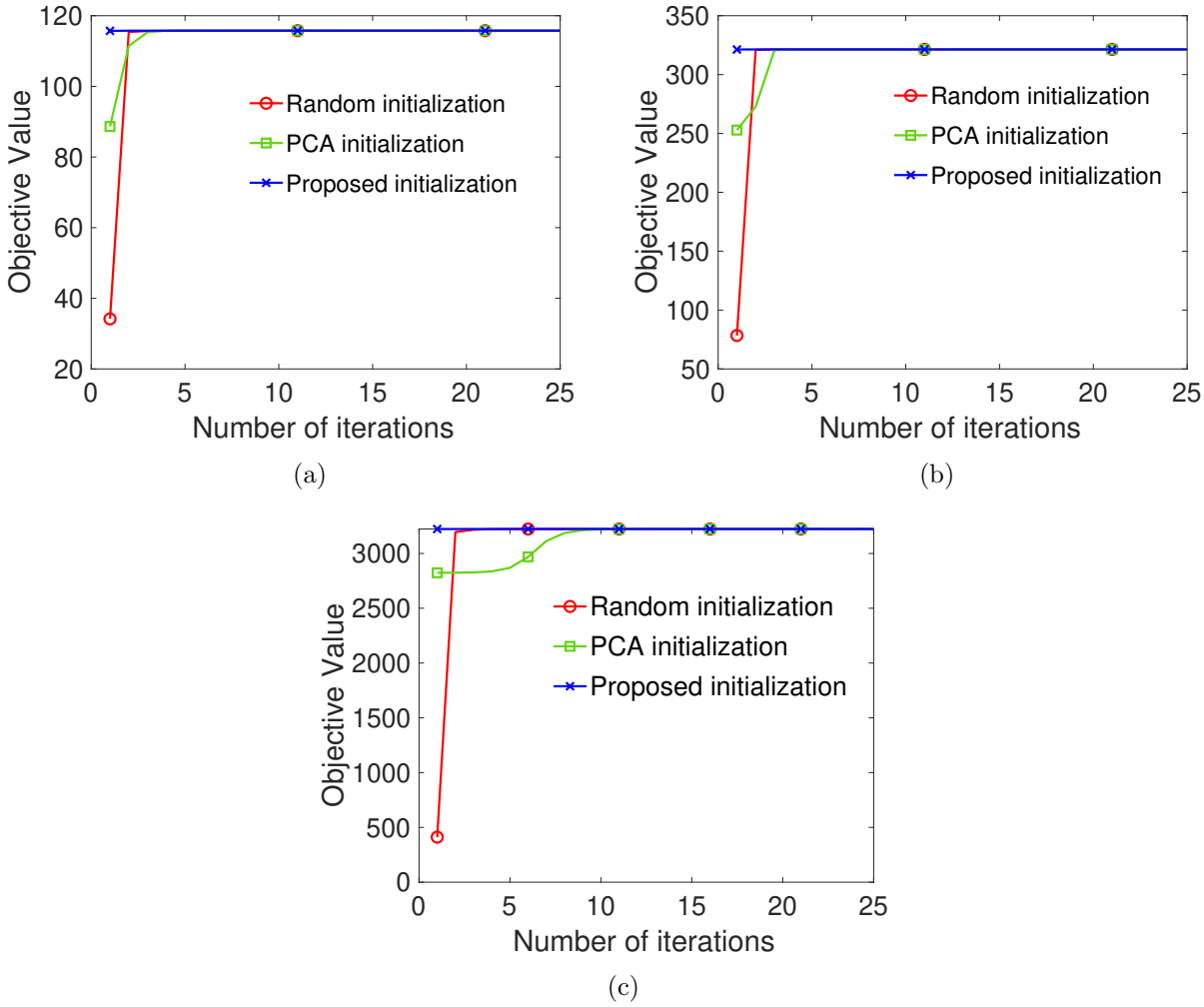


Figure 4.20. L1-PCA objective versus number of iterations on the human activity dataset for (a) $p = 0.75$, (b) $p = 1$, and (c) $p = 1.5$.

the objective value of Lp-PCA for $p = 0.75$, 1, and 1.5, obtained using the algorithm of [5] with $K = 2$ and plot it versus the number of iterations in Figures 4.20(a)-(c). We observe that the proposed initialization scheme archives the highest metric value in the least number of iterations, while random and PCA initializations converge slower.

Fisher Iris dataset (non-Elliptical). We operate on the Fisher Iris dataset [242] in this experiment. The dataset consists of Iris flower measurements such as the sepal length, sepal width, petal length, and petal width in cm corresponding to 3 classes of flowers, namely Setosa, Versicolor, and Virginica. The dataset is of size 4×150 and we set $K = 2$ to compute the Lp-PCA objective using the algorithm in [19]. We plot the Lp-PCA objective value versus the number of iterations of the algorithm for $p = 0.75$, 1, and 1.5 in Figures 4.21 (a)-(c) and observe that PCA initialization converges the slowest. Random initialization converges faster. The proposed initialization yields

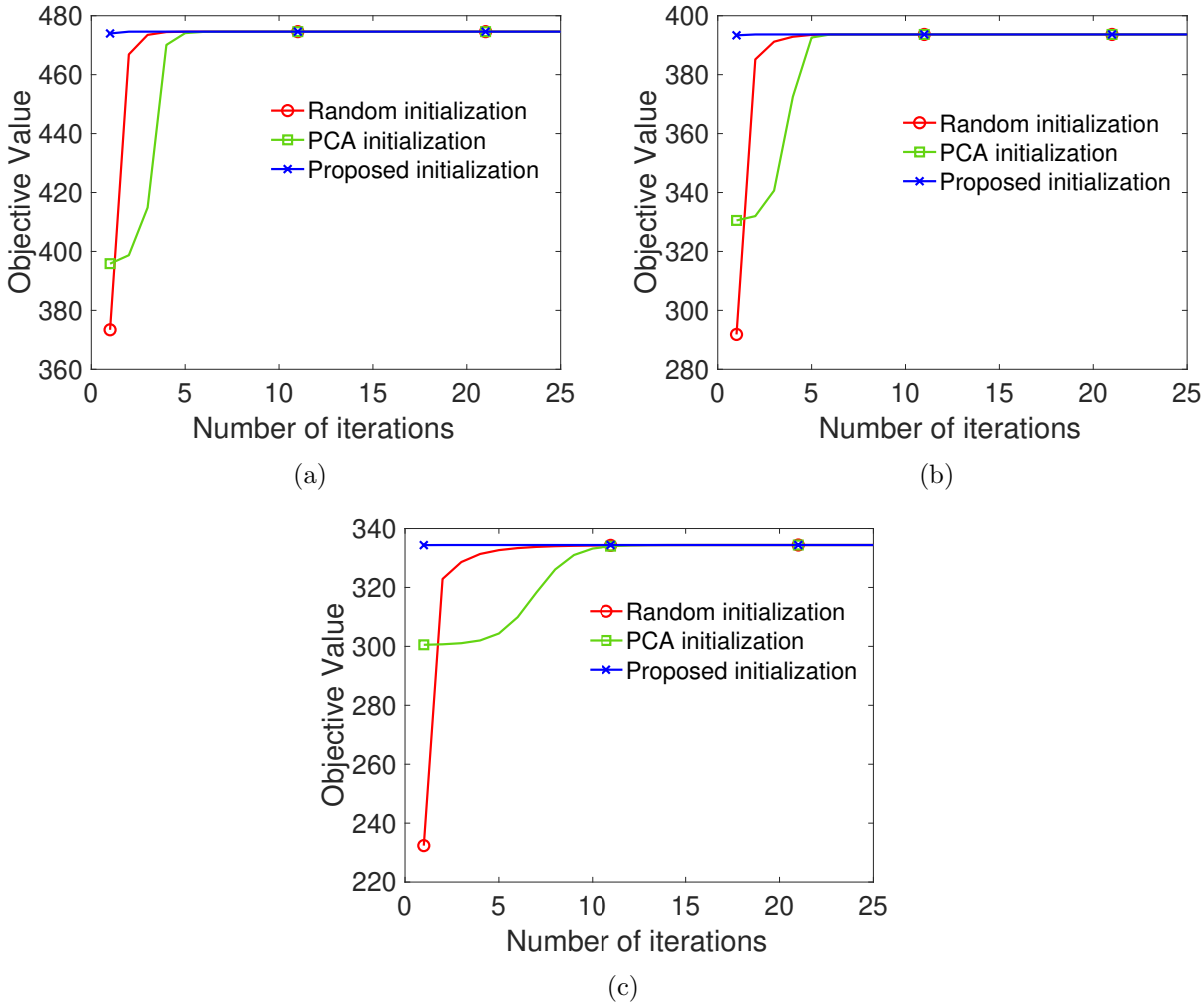


Figure 4.21. L1-PCA objective versus number of iterations on the Fisher Iris dataset for (a) $p = 0.75$, (b) $p = 1$, and (c) $p = 1.5$.

instantaneous convergence to the highest metric value across the board.

Ionosphere dataset (non-Elliptical). In this experiment, we make use of the Ionosphere dataset from the UCI Machine Learning Repository [243]. This data consists of radar returns collected in Goose Bay, Labrador, Canada. The radar system system consists of a phased array of 16 high-frequency antennas with a total transmitted power on about 6.4 kilowatts. The transmitted signal passes through the ionosphere, hits the free electrons in it, and returns back to the Earth. Any received signal is labelled *good* if it shows evidence of structure in the ionosphere or *bad* if it does not. The received radar returns are processed via an auto-correlation function with a pulse number of 17. Each instance is complex valued and therefore there are 34 entries per measurement and a total of 351 measurements, resulting in a data matrix of size 34×351. We compute the L1-PCA objective value using the algorithm in [6], setting $K = 10$ and plot it versus the number of iterations

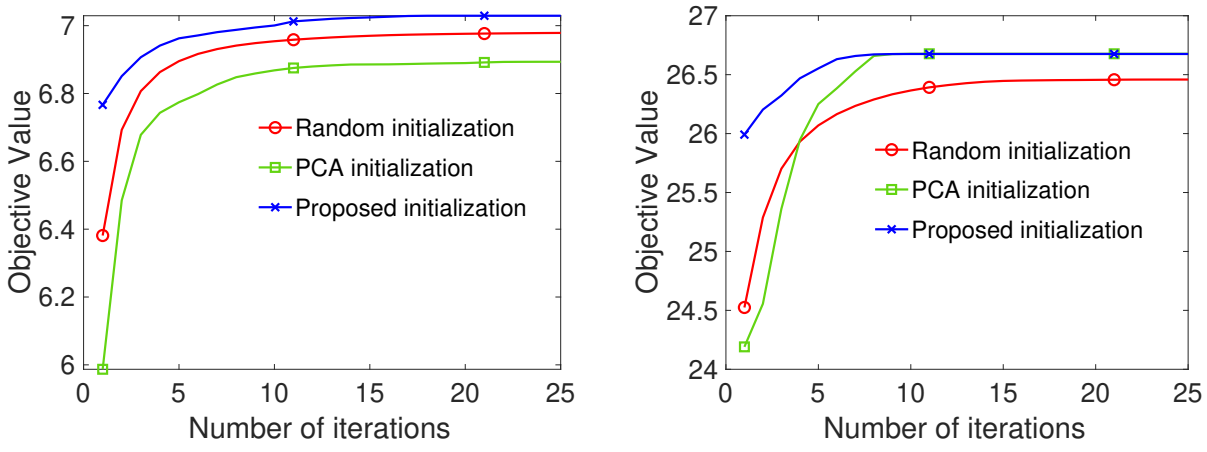


Figure 4.22. L1-PCA objective versus number of iterations on (a) the Ionosphere dataset and (b) the Exam Grades dataset for $p = 1$.

in Figure 4.22(a). We observe that the proposed initialization scheme achieves the highest L1-PCA objective, followed by random initialization. PCA initialization achieves convergence to a lower objective value.

Exam grades dataset (non-Elliptical). In this experiment, we operate on the Exam Grades dataset [244], consisting of the grades of 120 students in 5 subjects on a scale of 0-100. The dataset is of size 5×120. We set $K = 4$ and compute the L1-PCA objective value using the algorithm in [6] and plot it versus number of iterations in Figure 4.22(b). We observe that the proposed initialization begins at and converges the fastest to the highest objective value. PCA initialization starts with a low objective value but recovers to the objective value achieved by the proposed initialization scheme. Random initialization however yields convergence to lower objective value.

This concludes our studies on real-world data. Throughout our studies, we have observed that the proposed asymptotic theory for Lp-PCA can be leveraged to initialize the iterative algorithms of Lp-PCA intelligently to achieve superior performance in terms of faster convergence and convergence to high objective value on both elliptical and non-elliptical real-world datasets. This indicates that the proposed initialization scheme initializes the iterative algorithms of Lp-PCA close to the optimum solution. We note that each iteration of the existing Lp-PCA algorithms can be computationally intensive for large N and/or D, and enabling faster convergence (to a better solution) by leveraging the proposed intelligent initialization scheme results in significant overall computational cost savings.

4.5 Conclusions

In this chapter, we offer novel asymptotic theory for Lp-PCA when the data is drawn from the broad family of Elliptical distributions. First, we prove the asymptotic coincidence of the L1-PCA and the standard PC for $K = 1$. We show that L1-PCA is as good an estimator of the maximum-variance line (dominant eigenvector) as PCA asymptotically, while remaining robust against outliers in the limited data scenario. Next, for $K \geq 1$, we present a proof of asymptotic convergence of the Lp-PCA subspace to the L2-PCA subspace and therefore to the dominant eigensubspace. Moreover, we show that the asymptotic Lp-PCs are specific rotated versions of the dominant eigenvectors of the covariance matrix and offer an algorithm to derive the specific rotation matrix to obtain the asymptotic Lp-PCs from the dominant eigenvectors. Finally, we offer a wide variety of experimental studies to demonstrate the derived convergence theory in practice, including experiments on leveraging the proposed theory for intelligent initialization of iterative Lp-PCA algorithms to achieve faster and better convergence on both synthetic and real-world datasets belonging to Elliptical and non-elliptical distributions.

Chapter 5

Future Work

We dedicate this chapter to include the limitations and possible future extensions of this dissertation. First, we identify some shortcomings and then propose future extensions to address them in the following.

5.1 Limitations of Existing Work

In Chapter 3's Section 3.3, for the case of $K = 1$, we note that the L1-PCA objective is discontinuous and therefore its derivative is undefined at the origin. We overcome this problem by smoothening the L1-norm at the origin and thereby enabling the definition of the gradient. This smoothened stochastic L1-PCA formulation in (3.11) can be solved using projected stochastic gradient ascent as in Figure 3.1. Next, we extend our research for $K \geq 1$ by employing the Barron loss with a tunable robustness parameter α [26]. Notice that we rely on a robust loss function as opposed to generalizing to multiple components ($K \geq 1$) using Lp-norm. Although this is not a limitation, a future research direction may include the development of stochastic Lp-PCA algorithms for multiple components.

On the other hand, in Chapter 4, we proposed asymptotic theory for Lp-PCA, for $p \in (0, 2)$. Interesting recent work of [245] has shown that $p > 2$ in Lp1-PCA can be beneficial for spectral clustering. A limitation of our theory is that it does not cover this regime of $2 < p < +\infty$.

Moreover, our theory proves the asymptotic convergence of the subspaces of Lp-PCA and dominant eigenvectors, but does not provide a closed form expression for the dominant eigenvector subspace

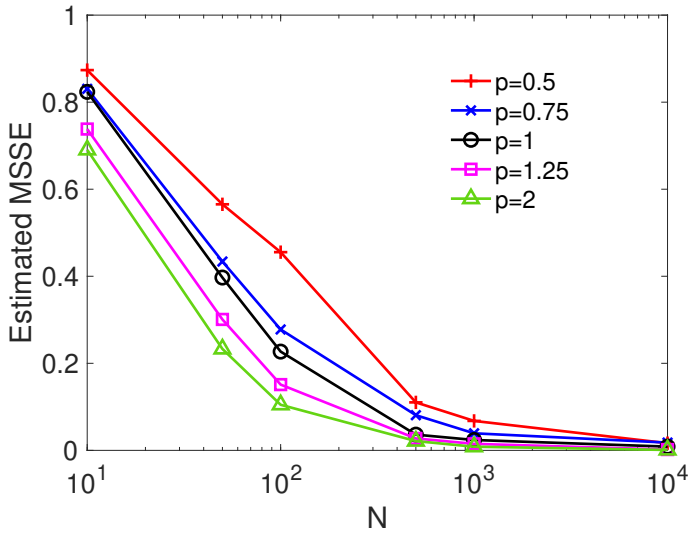


Figure 5.1. stimated mean subspace error versus N for various p values.

estimation error of Lp-PCA for limited N.

5.2 Possible Extensions of Existing Work

Based on the limitations identified in the section above, we proposed the following potential future extensions:

- **Stochastic Lp-PCA algorithms for multiple ($K \geq 1$) components.** Instead of developing stochastic PCA algorithms using Barron loss, it would be interesting to do so with the Lp-norm. Specifically, since Lp-norm for $p \leq 1$ has an undefined derivative at the origin, one may smoothen it similar to our approach in Chapter 3's Section 3.3, and develop a projected stochastic gradient ascent algorithm to solve the corresponding problem.

- **Explore asymptotic theory for Lp-PCA for $p > 2$.** The asymptotic theory we have derived in Chapter 4 is for $p \in (0, 2)$. However, the case of $p > 2$ is unexplored to date and maybe of interest in some cases, for example in spectral clustering [245].

- **Develop a closed form expression of the dominant eigensubspace estimation error of Lp-PCA for limited number of data points N.** We observe empirically that Lp-PCA for larger p values converges faster to lower subspace error. That is, for a given covariance matrix $\mathbf{C}$ and number of measurements N, Lp-PCA for a larger p appears to have a lower subspace error compared to that of smaller p values. We present an illustration of this effect in Figure 5.1. Specifically, given data drawn from a distribution with covariance matrix $\mathbf{C}$

that admits EVD $\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^\top$ and some $p \in (0, 2)$, the goal would be to derive a closed from expression for

$$\text{MSSE} = \mathbb{E}\left\{\left\|\mathbf{E}_{:,1:K}\mathbf{E}_{:,1:K}^\top - \mathbf{Q}_{Lp}\mathbf{Q}_{Lp}\right\|_2^2\right\}, \tag{5.1}$$

where $\mathbf{Q}_{Lp}$ consists of the K Lp-PCs. Since a closed form expresison for the above is not known at this time, we approximate the expectation in the above expression using the sample average computed over $S = 10^4$ independent realizations and plot it versus N in Figure 5.1, where $\mathbf{C} = \mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^\top$, $\boldsymbol{\Lambda} = \texttt{diag}(50,\ 25,\ 15)$, and $K = 2$ using the expression below

$$\text{Estimated MSSE} = \frac{1}{S}\sum_{i=1}^{S}\left\|\mathbf{E}_{:,1:K}\mathbf{E}_{:,1:K}^\top - \mathbf{Q}_{Lp}\mathbf{Q}_{Lp}\right\|_2^2. \tag{5.2}$$

Similar to Figure 4.8, we observe that larger p values converge to lower estimated MSSE values quicker.

Bibliography

[1] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, 2nd ed.* New York, NY: John Wiley & Sons, 2012.

[2] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *J. Ed. Psych.*, vol. 24, pp. 417–441, Sep. 1933.

[3] P. P. Markopoulos, G. N. Karystinos, and D. A. Pados, "Optimal algorithms for L1-subspace signal processing," *IEEE Trans. Signal Process.*, vol. 62, pp. 5046–5058, Oct. 2014.

[4] P. P. Markopoulos, S. Kundu, S. Chamadia, and D. Pados, "Efficient L1-Norm Principal-Component Analysis via Bit Flipping," *IEEE Trans. Signal Process.*, vol. 65, pp. 4252–4264, Aug. 2017.

[5] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, pp. 1672–1680, Sep. 2008.

[6] F. Nie, H. Huang, C. Ding, D. Luo, and H. Wang, "Robust principal component analysis with non-greedy L1-norm maximization," in *Proc. Int. Joint Conf. Art. Intell. (IJCAI),* Barcelona, Spain, Jul. 2011, pp. 1433–1438.

[7] M. McCoy and J. A. Tropp, "Two proposals for robust PCA using semidefinite programming," *Electron. J. Statist.*, vol. 5, pp. 1123–1160, Jun. 2011.

[8] A. Anandkumar, R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky, "Tensor decompositions for learning latent variable models," *J. Machine Learn. Res. (JMLR)*, vol. 15, pp. 2773–2832, Aug. 2014.

[9] A. Cichocki, D. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and H. A. Phan, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 145–163, 2015.

[10] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, Aug. 2009.

[11] P. P. Markopoulos, D. G. Chachlakis, and E. E. Papalexakis, "The exact solution to rank-1 L1-norm tucker2 decomposition," *IEEE Signal Process. Lett.*, vol. 25, no. 4, pp. 511–515, Apr. 2018.

[12] D. G. Chachlakis, A. Prater-Bennette, and P. P. Markopoulos, "L1-norm tucker tensor decomposition," *IEEE Acc.*, vol. 7, pp. 178 454–178 465, Nov. 2019.

[13] D. G. Chachlakis and P. P. Markopoulos, "Robust decomposition of 3-way tensors based on L1-norm," in *Proc. SPIE Def. Commer. Sens.,* Orlando, FL, vol. 10658. International Society for Optics and Photonics, May 2018, pp. 1 065 807–1–1 065 807–13.

[14] ——, "Novel algorithms for exact and efficient L1-norm-based TUCKER2 decomposition," in *IEEE Int. Conf. Acoust. Speech Signal Process.(ICASSP), Calgary, Canada*, Apr. 2018, pp. 6294–6298.

[15] P. P. Markopoulos, D. G. Chachlakis, and A. Prater-Bennette, "L1-norm higher-order singular-value decomposition," in *IEEE Glob. Conf. Signal Inform. Process. (GlobalSIP)* Anaheim, CA, Nov. 2018, pp. 1353–1357.

[16] Y. Pang, X. Li, and Y. Yuan, "Robust tensor analysis with l1-norm," *IEEE Trans. Circ. Syst. Vid. Tech.*, vol. 20, no. 2, pp. 172–178, Apr. 2009.

[17] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.

[18] E. Oja, "Simplified neuron model as a principal component analyzer," *J. Math. Bio.*, vol. 15, pp. 267–273, Nov. 1982.

[19] N. Kwak, "Principal component analysis by Lp-norm maximization," *IEEE Trans. Cybern.*, vol. 44, no. 5, pp. 594–609, Jun. 2013.

[20] D. G. Chachlakis and P. P. Markopoulos, "Combinatorial search for the Lp-norm principal component of a matrix," in *Proc. Asilomar Conf. Signal. Syst. Comput.,* Pacific Grove, CA, Nov. 2019, pp. 1611–1615.

[21] ——, "Novel algorithms for Lp-Quasi-norm principal-component analysis," in *Proc. IEEE Eur. Signal Process. Conf. (EUSIPCO),* Amsterdam, Netherlands,, Jan. 2021, pp. 1045–1049.

[22] B. Minnehan, N. Nagananda, and A. Savakis, "Grlp-PCA: Grassmann iterative p-norm principal component analysis," *IEEE Open J. Signal Process.*, vol. 1, pp. 90–98, Jun. 2020.

[23] M. Dhanaraj, "Incremental and adaptive L1-norm principal component analysis: Novel algorithms and applications," M.S. thesis, Dept. Elect. and Microelect. Engg., Rochester Institute of Technology, Rochester, NY, USA, Jul. 2018. [Online]. Available: https://www.proquest.com/docview/2104054727?pq-origsite=gscholar&fromopenview=true

[24] P. P. Markopoulos, M. Dhanaraj, and A. Savakis, "Adaptive L1-Norm Principal-Component Analysis With Online Outlier Rejection," *IEEE J. Select. Topics Signal Process.*, vol. 12, pp. 1131–1143, Dec. 2018.

[25] M. Dhanaraj and P. P. Markopoulos, "Novel algorithm for incremental L1-norm principal-component analysis," in *Proc. IEEE Eur. Signal Process. Conf. (EUSIPCO),* Rome, Italy, Sep. 2018, pp. 2020–2024.

[26] J. T. Barron, "A general and adaptive robust loss function," in *IEEE Conf. Comput. Vis. Pattern Rec. (CVPR)* Long Beach, CA, Jun. 2019, pp. 4331–4339.

[27] J. Goes, T. Zhang, R. Arora, and G. Lerman, "Robust Stochastic Principal Component Analysis," in *Proc. Artificial Intell. Statist. (AISTATS),* Reykjavik, Iceland, Apr. 2014, pp. 266–274.

[28] I. T. Jolliffe, *Principal Component Analysis.* New York, NY. Springer, 1986.

[29] M. E. Wall, A. Rechtsteiner, and L. M. Rocha, "Singular value decomposition and principal component analysis," in *Pract. App. Micr. Data Anal.*, 2003, pp. 91–109.

[30] O. Edfors, M. Sandell, J.-J. Van de Beek, S. K. Wilson, and P. O. Borjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. Commun.*, vol. 46, pp. 931–939, Jul. 1998.

[31] H. Q. Ngo and E. G. Larsson, "EVD-based channel estimation in multicell multiuser mimo systems with very large antenna arrays," in *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP),* Kyoto, Japan, Mar. 2012, pp. 3249–3252.

[32] M. B. Christopher, *Pattern Recognition and Machine Learning.* New York, NY: Springer-Verlag, 2016.

[33] C. Ding and X. He, "K-means clustering via principal component analysis," in *Proc. ACM Int. Conf. Mach. Learn.,* Alberta, Canada, Jul. 2004, pp. 29–36.

[34] J. Yang, D. Zhang, A. F. Frangi, and J. Y. Yang, "Two-dimensional PCA: a new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, pp. 131–137, Jan. 2004.

[35] S. Sanei and J. A. Chambers, *EEG Signal Processing*. John Wiley and Sons, 2013.

[36] F. Castells, P. Laguna, L. Sörnmo, A. Bollmann, and J. M. Roig, "Principal component analysis in ECG signal processing," *EURASIP J. App. Signal Process.*, vol. 2007, pp. 98–98, Dec. 2007.

[37] K. Y. Yeung and W. L. Ruzzo, "Principal component analysis for clustering gene expression data," *Bioinformatics*, vol. 17, pp. 763–774, Sep. 2001.

[38] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *J. Psychometrika*, vol. 1, pp. 211–218, Sep. 1936.

[39] V. Barnett and T. Lewis, *Outliers in statistical data*. New York, NY: Wiley, 1994.

[40] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust Principal Component Analysis?" *J. ACM*, vol. 58, pp. 1–39, May 2011.

[41] N. Vaswani, T. Bouwmans, S. Javed, and P. Narayanamurthy, "Robust subspace learning: Robust PCA, robust subspace tracking, and robust subspace recovery," *IEEE Signal Process. Mag.*, pp. 32–55, Jul. 2018.

[42] G. Mateos and G. B. Giannakis, "Robust PCA as bilinear decomposition with outlier-sparsity regularization," *IEEE Trans. Signal Process.*, vol. 60, pp. 5176–5190, Oct. 2012.

[43] H. Xu, C. Caramanis, and S. Sanghavi, "Robust PCA via outlier pursuit," *IEEE Trans. Info. Theory*, vol. 58, pp. 3047–3064, May 2012.

[44] M. Chen, A. Ganesh, Z. Lin, Y. Ma, J. Wright, and L. Wu, "Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix," *J. Coord. Sci. Lab. Rep. no. UILU-ENG-09-2214*, Aug. 2009.

[45] F. De la Torre and M. J. Black, "Robust principal component analysis for computer vision," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV),* Venice, Italy, Jul. 2001, pp. 362–369.

[46] F. De La Torre and M. J. Black, "A framework for robust subspace learning," *Int. J. Comput. Vis.*, vol. 54, pp. 117–142, Aug. 2003.

[47] Q. Ke and T. Kanade, "Robust L1-norm factorization in the presence of outliers and missing data by alternative convex programming," in *IEEE Conf. Comput. Vis. Pattern Rec. (CVPR)* San Diego, CA, vol. 1, Jun. 2005, pp. 739–746.

[48] J. P. Brooks, J. H. Dulá, and E. L. Boone, "A pure L1-norm principal component analysis," *Elsevier Comput. Stat. Data Anal.*, vol. 61, pp. 83–98, May 2013.

[49] N. Tsagkarakis, P. P. Markopoulos, and D. A. Pados, "On the L1-norm approximation of a matrix by another of lower rank," in *Proc. IEEE Conf. Mach. Learn. and App.,* Anaheim, CA, Dec. 2016, pp. 768–773.

[50] C. Ding, D. Zhou, X. He, and H. Zha, "R1-PCA: rotational invariant L1-norm principal component analysis for robust subspace factorization," in *Proc. 23rd ACM Int. Conf. Mach. Lear.*, Jun. 2006, pp. 281–288.

[51] R. He, B.-G. Hu, W.-S. Zheng, and X.-W. Kong, "Robust principal component analysis based on maximum correntropy criterion," *IEEE Trans. Imag. Process.*, vol. 20, pp. 1485–1494, Jun. 2011.

[52] Q. Wang, Q. Gao, X. Gao, and F. Nie, "L2,p-norm based PCA for image recognition," *IEEE Trans. Image Process.*, pp. 1336–1346, Mar. 2016.

[53] F. Nie, J. Yuan, and H. Huang, "Optimal mean robust principal component analysis," in *Int. Conf. Mach. Learn. (ICML)* Beijing, China, Jan. 2014, pp. 1062–1070.

[54] M. Luo, F. Nie, X. Chang, Y. Yang, A. Hauptmann, and Q. Zheng, "Avoiding optimal mean robust PCA/2DPCA with non-greedy L1-norm maximization," in *25th Int. Joint Conf. Artif. Intell.*, Jul. 2016, pp. 1802–1808.

[55] M. Luo, F. Nie, X. Chang, Y. Yang, A. G. Hauptmann, and Q. Zheng, "Avoiding optimal mean l2,1-norm maximization-based robust PCA for reconstruction," *J. Neu. Comput.*, vol. 29, pp. 1124–1150, Apr. 2017.

[56] P. P. Markopoulos, D. A. Pados, G. N. Karystinos, and M. Langberg, "L1-norm principal-component analysis in L2-norm-reduced-rank data subspaces," in *Proc. SPIE,* Anaheim, CA, vol. 10211, May 2017, pp. 04:1–04:10.

[57] N. Tsagkarakis, P. P. Markopoulos, G. Sklivanitis, and D. A. Pados, "L1-norm principal-component analysis of complex data," *IEEE Trans. Signal Process.*, vol. 66, pp. 3256–3267, Apr. 2018.

[58] S. Chamadia and D. A. Pados, "Optimal sparse L1-norm principal-component analysis," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP),* New Orleans, LA, Mar. 2017, pp. 2686–2690.

[59] P. P. Markopoulos, S. Kundu, and D. A. Pados, "L1-fusion: Robust linear-time image recovery from few severely corrupted copies," in *Proc. IEEE Int. Conf. Image Process. (ICIP),* Quebec, Canada, Sep. 2015, pp. 1225–1229.

[60] M. Johnson and A. Savakis, "Fast L1-eigenfaces for robust face recogntion," in *Proc. IEEE West. New York Image Signal Process. Workshop (WNYISPW),* Rochester, NY, Nov. 2014, pp. 1–5.

[61] F. Maritato, Y. Liu, S. Colonnese, and D. A. Pados, "Cloud-assisted individual L1-PCA face recognition using wavelet-domain compressed images," in *Proc. IEEE Euro. Workshop Vis. Info. Process. (EUVIP),* Marseille, France, Oct. 2016, pp. 1–6.

[62] ——, "Face recognition with L1-norm subspaces," in *Proc. SPIE,* Baltimore, MD, vol. 9857, May 2016, pp. 0L:1–0L:8.

[63] Y. Liu and D. A. Pados, "Compressed-sensed-domain L1-PCA video surveillance," *IEEE Trans. Multimed.*, vol. 18, pp. 351–363, Mar. 2016.

[64] M. Pierantozzi, Y. Liu, D. A. Pados, and S. Colonnese, "Video background tracking and foreground extraction via L1-subspace updates," in *Proc. SPIE,* Baltimore, MD, vol. 9857, May 2016, pp. 08:1–08:16.

[65] D. G. Chachlakis, P. P. Markopoulos, R. J. Muchhala, and A. Savakis, "Visual Tracking with L1-Grassmann Manifold Modeling," in *Proc. SPIE,* Anaheim, CA, vol. 10211, Apr. 2017, pp. 02:1–02:10.

[66] P. P. Markopoulos, "Reduced-rank filtering on L1-norm subspaces," in *Proc. IEEE Workshop Sens. Array Multichannel Signal Process. (SAM),* Rio de Janeiro, Brazil, Jul. 2016, pp. 1–5.

[67] N. Tsagkarakis, P. P. Markopoulos, and D. A. Pados, "Direction finding by complex L1-principal-component analysis," in *Proc. IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC),* Stockholm, Sweden, Jun. 2015, pp. 475–479.

[68] P. P. Markopoulos, N. Tsagkarakis, D. A. Pados, and G. N. Karystinos, "Direction-of-arrival estimation by L1-norm principal components," in *Proc. IEEE Int. Symp. Phased Array Syst. Tech. (PAST),* Waltham, MA, Oct. 2016, pp. 1–6.

[69] P. P. Markopoulos and F. Ahmad, "Indoor human motion classification by L1-norm subspaces of micro-doppler signatures," in *Proc. IEEE Radar Conf.,* Seattle, WA, May 2017, pp. 1807–1810.

[70] Y. Liu, D. A. Pados, S. N. Batalama, and M. J. Medley, "Iterative re-weighted L1-norm principal-component analysis," in *Proc. IEEE Asilomar Conf. Signals, Syst., Comput.,* Pacific Grove, CA, Oct. 2017, pp. 425–429.

[71] E. E. Papalexakis, C. Faloutsos, and N. D. Sidiropoulos, "Tensors for data mining and data fusion: Models, applications, and scalable algorithms," *ACM Trans. Intell. Syst. Technol.*, vol. 8, pp. 16:1–16:44, Jan. 2017.

[72] P. P. Markopoulos. L1-PCA code repository. [Online]. Available: https://github.com/RIT-MILOS-LAB/Efficient-L1-Norm-Principal-Component-Analysis-via-Bit-Flipping

[73] F. Shang, J. Cheng, Y. Liu, Z.-Q. Luo, and Z. Lin, "Bilinear factor matrix norm minimization for robust PCA: Algorithms and applications," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 14, Sep. 2017.

[74] C. Wang, Y. Wang, Z. Lin, A. L. Yuille, and W. Gao, "Robust estimation of 3D human poses from a single image," in *Proc. IEEE Conf. Comput. Vis. Patt. Recogn. (CVPR),* Columbus, OH, Jun. 2014, pp. 2361–2368.

[75] T. Bouwmans, "Subspace learning for background modeling: A survey," *J. Rec. Patents Computer Sci.*, vol. 2, pp. 223–234, Nov. 2009.

[76] L. Wang, L. Wang, Q. Zhuo, H. Xiao, and W. Wang, "Adaptive eigenbackground for dynamic background modeling," in *Intell. Comput. Signal Process. Pattern Recogn.* Springer, *Berlin, Heidelberg*, 2006, pp. 670–675.

[77] J. Zhang and Y. Zhuang, "Adaptive weight selection for incremental eigen-background modeling," in *Proc. IEEE Int. Conf. Mult. Expo., (ICME),* Beijing, China, Jul. 2007, pp. 851–854.

[78] T. Bouwmans and E. H. Zahzah, "Robust PCA via principal component pursuit: A review for a comparative evaluation in video surveillance," *Elsevier Comput. Vis. Image Underst.*, vol. 122, pp. 22–34, May 2014.

[79] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face Recognition: A Literature Survey," *J. ACM Comput. Surveys*, vol. 35, pp. 399–458, Dec. 2003.

[80] P. Narayanamurthy and N. Vaswani, "Provable dynamic robust PCA or robust subspace tracking," in *Proc. IEEE Int. Symp. Inf. Theory,* Vail, CO, Jun. 2018, pp. 1–5.

[81] J. Zhan, B. Lois, H. Guo, and N. Vaswani, "Online (and offline) robust PCA: Novel algorithms and performance guarantees," in *Proc. Art. Intell. Stat.,* Cadiz, Spain, May 2016, pp. 1488–1496.

[82] N. Vaswani and P. Narayanamurthy, "Static and dynamic robust PCA via low-rank + sparse matrix decomposition: A review," *arXiv preprint arXiv:1803.00651*, Mar. 2018.

[83] H. Guo, C. Qiu, and N. Vaswani, "An online algorithm for separating sparse and low-dimensional signal sequences from their sum," *IEEE Trans. Signal Process.*, pp. 4284–4297, Aug. 2014.

[84] J. Feng, H. Xu, S. Mannor, and S. Yan, "Online PCA for contaminated data," in *Proc. Adv. Neu. Inform. Process. Syst. (NIPS),* Lake Tahoe, NV, 2013, pp. 764–772.

[85] J. Feng, H. Xu, and S. Yan, "Online robust PCA via stochastic optimization," in *Proc. Adv. Neu. Inform. Process. Syst. (NIPS),* Lake Tahoe, NV, Dec. 2013, pp. 404–412.

[86] M. Mardani, G. Mateos, and G. B. Giannakis, "Dynamic anomalography: Tracking network anomalies via sparsity and low rank," *IEEE J. Select. Topics Signal Process.*, vol. 7, pp. 50–66, Feb. 2013.

[87] ——, "Recovery of low-rank plus compressed sparse matrices with application to unveiling traffic anomalies," *IEEE Trans. Info. Theory*, vol. 59, pp. 5186–5205, Aug. 2013.

[88] R. Otazo, E. Candès, and D. K. Sodickson, "Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components," *J. Magn. Reson. Med.*, vol. 73, pp. 1125–1136, Mar. 2015.

[89] P. Netrapalli, U. Niranjan, S. Sanghavi, A. Anandkumar, and P. Jain, "Non-convex robust PCA," in *Proc. Adv. Neu. Inform. Process. Syst. (NIPS),* Montreal, Canada, Dec. 2014, pp. 1107–1115.

[90] P. Rodriguez and B. Wohlberg, "Incremental principal component pursuit for video background modeling," *J. Math. Imag. Vis.*, vol. 55, pp. 1–18, May 2016.

[91] Y. Li, L. Xu, J. Morphett, and R. Jacobs, "An integrated algorithm of incremental and robust PCA," in *Proc. IEEE Int. Conf. Image Process. (ICIP),* Barcelona, Spain, Sep. 2003, pp. 245–248.

[92] X. Yi, D. Park, Y. Chen, and C. Caramanis, "Fast algorithms for robust PCA via gradient descent," in *Proc. Adv. Neu. Info. Process. Syst. (NIPS),* Barcelona, Spain, Dec. 2016, pp. 4152–4160.

[93] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the grassmannian for online foreground and background separation in subsampled video," in *Proc. IEEE Conf. Comput. Vis. Patt. Recog. (CVPR), Providence, RI*, Jun. 2012, pp. 1568–1575.

[94] Y. Li, "On incremental and robust subspace learning," *Elsevier J. Patt. Recogn.*, vol. 37, pp. 1509–1518, Jul. 2004.

[95] D. Skočaj and A. Leonardis, "Incremental and robust learning of subspace representations," *Elsevier J. Imag. Vis. Comput.*, vol. 26, pp. 27–38, Jan. 2008.

[96] S. Liwicki, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "Euler principal component analysis," *Int. J. Computer Vis.*, vol. 101, pp. 498–518, Feb. 2013.

[97] J. R. Bunch and C. P. Nielsen, "Updating the singular value decomposition," *J. Numerische Mathematik*, vol. 31, pp. 111–129, Jun. 1978.

[98] Context aware vision using image-based active recognition (CAVIAR). [Online]. Available: http://homepages.inf.ed.ac.uk/rbf/CAVIAR/.Accessedon:Jul.2020.

[99] X. Wang, M. Amin, F. Ahmad, and E. Aboutanios, "Interference DOA estimation and suppression for GNSS receivers using fully augmentable arrays," *IET Radar, Sonar & Nav.*, vol. 11, no. 3, pp. 474–480, Mar. 2017.

[100] M. G. Amin, X. Wang, Y. D. Zhang, F. Ahmad, and E. Aboutanios, "Sparse arrays and sampling for interference mitigation and DOA estimation in GNSS," *Proc. IEEE*, vol. 104, no. 6, pp. 1302–1317, Jun. 2016.

[101] A. Yassin, Y. Nasser, M. Awad, A. Al-Dubai, R. Liu, C. Yuen, R. Raulefs, and E. Aboutanios, "Recent advances in indoor localization: A survey on theoretical approaches and applications," *IEEE Comm. Surv. & Tut.*, vol. 19, no. 2, pp. 1327–1346, Nov. 2016.

[102] F. Engels, P. Heidenreich, A. M. Zoubir, F. K. Jondral, and M. Wintermantel, "Advances in automotive radar: A framework on computationally efficient high-resolution frequency estimation," *IEEE Signal Process. Mag.*, vol. 34, no. 2, pp. 36–46, Mar. 2017.

[103] J. Sheinvald, M. Wax, and A. J. Weiss, "On maximum-likelihood localization of coherent signals," *IEEE Trans. Signal Process.*, vol. 44, no. 10, pp. 2475–2482, Oct. 1996.

[104] D. H. Johnson, "The application of spectral estimation methods to bearing estimation problems," *Proc. IEEE*, vol. 70, no. 9, pp. 1018–1028, Sep. 1982.

[105] M. Carlin, P. Rocca, G. Oliveri, F. Viani, and A. Massa, "Directions-of-arrival estimation through bayesian compressive sensing strategies," *IEEE Trans. Antennas Propag.*, vol. 61, no. 7, pp. 3828–3838, Jul. 2013.

[106] Y. Yu, A. P. Petropulu, and H. V. Poor, "MIMO radar using compressive sampling," *IEEE J. Sel. Topics Signal Process.,*, vol. 4, no. 1, pp. 146–163, Feb. 2010.

[107] R. Grover, D. A. Pados, and M. J. Medley, "Subspace direction finding with an auxiliary-vector basis," *IEEE Trans. Signal Process.*, vol. 55, no. 2, pp. 758–763, Feb. 2007.

[108] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antenna Prop.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[109] H. Abeida and J.-P. Delmas, "Efficiency of subspace-based doa estimators," *Signal Process.*, vol. 87, no. 9, pp. 2075–2084, Sep. 2007.

[110] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, Jul. 2017.

[111] D. Tao, X. Li, W. Hu, S. Maybank, and X. Wu, "Supervised tensor learning," in *IEEE Int. Conf. Data Mining*, Houston, TX, Nov. 2005, pp. 8–pp.

[112] J. Kossaifi, A. K. Z. C. Lipton, A. Khanna, T. Furlanello, , and A. Anandkumar, "Tensor regression networks," *J. Mach. Learn. Res.*, vol. 21, no. 123, pp. 1–21, Jul. 2020.

[113] M. B. Amin, W. Zirwas, and M. Haardt, "HOSVD-based denoising for improved channel prediction of weak massive MIMO channels," in *IEEE Vehicular Technol. Conf.*, Sydney, Australia, Jun. 2017, pp. 1–5.

[114] D. C. Araújo, A. L. De Almeida, J. P. Da Costa, and R. T. De Sousa, "Tensor-based channel estimation for massive MIMO-OFDM systems," *IEEE Access*, vol. 7, pp. 42 133–42 147, Mar. 2019.

[115] R. Ballester-Ripoll, P. Lindstrom, and R. Pajarola, "TTHRESH: Tensor compression for multidimensional visual data," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 9, pp. 2891–2903, Mar. 2019.

[116] H. Ben-Younes, R. Cadene, M. Cord, and N. Thome, "Mutan: Multimodal tucker fusion for visual question answering," in *Proc. IEEE Int. Conf. Comput. Vision*, Venice, Italy, Oct. 2017, pp. 2612–2620.

[117] A. Koochakzadeh and P. Pal, "On canonical polyadic decomposition of overcomplete tensors of arbitrary even order," in *Proc. IEEE Int. Workshop Comput. Advances Multi-Sensor Adaptive Process.*, Curaçao, Dutch Antilles, Dec. 2017, pp. 1–5.

[118] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, pp. 1253–1278, Apr. 2000.

[119] D. Goldfarb and Z. Qin, "Robust low-rank tensor recovery: Models and algorithms," *SIAM J. Matrix Anal. Appl.*, vol. 35, no. 1, pp. 225–253, 2014.

[120] X. Cao, X. Wei, Y. Han, and D. Lin, "Robust face clustering via tensor decomposition," *IEEE Trans. Cybern.*, vol. 45, no. 11, pp. 2546–2557, 2015.

[121] K. Tountas, D. G. Chachlakis, P. P. Markopoulos, and D. A. Pados, "Iteratively re-weighted L1-PCA of tensor data," in *IEEE Asilomar Conf. Signals Syst. Comput.*, Pacific Grove, CA, Nov. 2019, pp. 1658–1661.

[122] J. Sun, D. Tao, S. Papadimitriou, P. S. Yu, and C. Faloutsos, "Incremental tensor analysis: Theory and applications," *ACM Trans. Knowl. Discovery Data*, vol. 2, no. 3, pp. 1–37, Oct. 2008.

[123] J. Sun, D. Tao, and C. Faloutsos, "Beyond streams and graphs: Dynamic tensor analysis," in *ACM Int. Conf. Knowl. Discovery Data mining*, Philadelphia, PA, Aug. 2006, pp. 374–383.

[124] R. Yu, D. Cheng, and Y. Liu, "Accelerated online low rank tensor learning for multivariate spatiotemporal streams," in *Proc. Int. Conf. Machine Learn.*, Lille, France, Jul. 2015, pp. 238–247.

[125] S. Papadimitriou, J.Sun, and C. Faloutsos, "Streaming pattern discovery in multiple time-series," in *Int. Conf. Very Large Databases*, Trondheim, Norway, Aug. 2005, pp. 697–708.

[126] Y. Liu, K. Tountas, D. A. Pados, S. N. Batalama, and M. J. Medley, "L1-subspace tracking for streaming data," *Pattern Recogn.*, vol. 97, pp. 106 992:1–13, Jan. 2020.

[127] M. Baskaran, M. H. Langston, T. Ramananandro, D. Bruns-Smith, T. Henretty, J. Ezick, and R. Lethin, "Accelerated low-rank updates to tensor decompositions," in *Proc. IEEE High Perf. Extreme Comput. Conf.*, Waltham, MA, Sep. 2016, pp. 1–7.

[128] O. A. Malik and S. Becker, "Low-rank tucker decomposition of large tensors using tensorsketch," in *Proc. Advances Neural Inf. Process. Syst.*, Montreal, Canada, Dec. 2018, pp. 10 096–10 106.

[129] R. Pagh, "Compressed matrix multiplication," *ACM Trans. Comput. Theory*, vol. 5, no. 3, pp. 1–17, Aug. 2013.

[130] Y. Sun, Y. Guo, C. Luo, J. Tropp, and M. Udell, "Low-rank Tucker approximation of a tensor from streaming data," *arXiv preprint arXiv:1904.10951*, 2019.

[131] W. Hu, X. Li, X. Zhang, X. Shi, S. Maybank, and Z. Zhang, "Incremental tensor subspace learning and its applications to foreground segmentation and tracking," *Int. J. Comput. Vision*, vol. 91, no. 3, pp. 303–327, Feb. 2011.

[132] X. Li, W. Hu, Z. Zhang, X. Zhang, and G. Luo, "Robust visual tracking based on incremental tensor subspace learning," in *IEEE Int. Conf. Comput. Vision*, Rio de Janeiro, Brazil, Oct. 2007, pp. 1–8.

[133] A. Sobral, C. G. Baker, T. Bouwmans, and E.-h. Zahzah, "Incremental and multi-feature tensor subspace learning applied for background modeling and subtraction," in *Int. Conf. Image Anal. Recogn.*, Algarve, Portugal, Oct. 2014, pp. 94–103.

[134] A. Sobral, S. Javed, S. Ki Jung, T. Bouwmans, and E.-h. Zahzah, "Online tensor decomposition for background subtraction in multispectral video sequences," in *IEEE Int. Conf. Comput. Vision*, Santiago, Chile, Dec. 2015, pp. 106–113. [Online]. Available: https://github.com/andrewssobral/ostd

[135] S. Smith, K. Huang, N. D. Sidiropoulos, and G. Karypis, "Streaming tensor factorization for infinite data sources," in *Proc. SIAM Int. Conf. Data Mining*, San Diego, CA, May 2018, pp. 81–89.

[136] Y. Du, Y. Zheng, K.-c. Lee, and S. Zhe, "Probabilistic streaming tensor decomposition," in *Proc. IEEE Int. Conf. Data Mining*, Singapore, Nov. 2018, pp. 99–108.

[137] M. Najafi, L. He, and S. Y. Philip, "Outlier-robust multi-aspect streaming tensor completion and factorization." in *Int. Joint Cond. Artifical Intell.*, Macao, China, Aug. 2019, pp. 3187–3194.

[138] P. Li, J. Feng, X. Jin, L. Zhang, X. Xu, and S. Yan, "Online robust low-rank tensor modeling for streaming data analysis," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 30, no. 4, pp. 1061–1075, Aug. 2018.

[139] D. G. Chachlakis, A. Prater-Bennette, and P. P. Markopoulos, "L1-norm Tucker tensor decomposition," *IEEE Access*, vol. 7, pp. 178 454–178 465, Nov. 2019.

[140] Y. Pang, X. Li, and Y. Yuan, "Robust tensor analysis with L1-norm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, pp. 172–178, Feb. 2010.

[141] D. G. Chachlakis, M. Dhanaraj, A. Prater-Bennette, and P. P. Markopoulos, "Options for multimodal classification based on L1-tucker decomposition," in *Proc. SPIE Defense Commercial Sens.*, Apr. 2019, pp. 109 890O:1–109 890O:13.

[142] D. G. Chachlakis, A. Prater-Bennette, and P. P. Markopoulos, "L1-norm higher-order orthogonal iterations for robust tensor analysis," in *IEEE Int. Conf. Acoust. Speech Signal Process.*, Barcelona, Spain (online), May 2020, pp. 4826–4830.

[143] K. Tountas, G. Sklivanitis, D. A. Pados, and M. J. Medley, "Tensor data conformity evaluation for interference-resistant localization," in *IEEE Asilomar Conf. Signals Syst. Comput.*, Pacific Grove, CA, Nov. 2019, pp. 1582–1586.

[144] CAVIAR: Context Aware Vision using Image-based Active Recognition. [Online]. Available: https://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/.

[145] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2006.

[146] A. Balsubramani, S. Dasgupta, and Y. Freund, "The Fast Convergence of Incremental PCA," in *Adv. Neural Inf. Process. Syst. (NeurIPS),* Lake Tahoe, NV, Dec. 2013, pp. 3174–3182.

[147] O. Shamir, "Convergence of stochastic gradient descent for pca," in *Proc. Int. Conf. Mach. Learn. (ICML),* New York, New York, Jun. 2016.

[148] E. Oja and J. Karhunen, "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix," *J. Math. Anal. Appl.*, vol. 106, pp. 69–84, Feb. 1985.

[149] O. Shamir, "A Stochastic PCA and SVD algorithm with an Exponential Convergence Rate," in *Proc. Int. Conf. Mach. Learn. (ICML),* Lille, France, Jun. 2015, pp. 144–152.

[150] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, pp. 400–407, Sep. 1951.

[151] A. Henriksen and R. Ward, "AdaOja: Adaptive Learning Rates for Streaming PCA," *arXiv preprint arXiv:1905.12115*, May 2019.

[152] O. Shamir, "Fast stochastic algorithms forSVD and PCA: Convergence properties and convexity," in *Int. Conf. Mach. Learn. (ICML),* New York, NY, Jun. 2016, pp. 248–256.

[153] R. Arora, A. Cotter, K. Livescu, and N. Srebro, "Stochastic Optimization for PCA and PLS," in *Proc. Annu. Conf. Commun. Control Comput. (Allerton)*, Oct. 2012, pp. 861–868.

[154] R. Arora, A. Cotter, and N. Srebro, "Stochastic Optimization of PCA with Capped MSG," in *Adv. Neural Inf. Process. Syst. (NeurIPS),* Lake Tahoe, NV, Dec. 2013, pp. 1815–1823.

[155] T. Krasulina, "The method of stochastic approximation for the determination of the least eigenvalue of a symmetrical matrix," *USSR Comput. Math. & Math. Phy.*, vol. 9, no. 6, pp. 189–195, Jan. 1969.

[156] C. Tang, "Exponentially convergent stochastic k-pca without variance reduction," in *Adv. Neural Inf. Process. Syst. (NeurIPS),* Vancouver, Canada, vol. 32, Dec. 2019.

[157] C. De Sa, B. He, I. Mitliagkas, C. Ré, and P. Xu, "Accelerated Stochastic Power Iteration," *Proc. Mach. Learn. Res. (PMLR*, vol. 84, p. 58, Jun. 2018.

[158] C. J. Li, M. Wang, H. Liu, and T. Zhang, "Near-optimal stochastic approximation for online principal component estimation," *Math. Prog.*, vol. 167, pp. 75–97, Jan. 2018.

[159] M. Chen, L. Yang, M. Wang, and T. Zhao, "Dimensionality reduction for stationary time series via stochastic nonconvex optimization," in *Adv. Neural Inf. Process. Syst. (NeurIPS),* Montreal, Canada, vol. 31, Dec. 2018.

[160] S. Alakkari and J. Dingliana, "An Acceleration Scheme for Memory Limited, Streaming PCA," *arXiv preprint arXiv:1807.06530*, Jul. 2018.

[161] P. Yang, C.-J. Hsieh, and J.-L. Wang, "History pca: A new algorithm for streaming pca," *arXiv preprint arXiv:1802.05447*, Feb. 2018.

[162] R. Johnson and T. Zhang, "Accelerating stochastic gradient descent using predictive variance reduction," in *Adv. Neural Inf. Process. Syst. (NeurIPS),* Lake Tahoe, NV, vol. 26, Dec. 2013.

[163] T. V. Marinov, P. Mianjy, and R. Arora, "Streaming Principal Component Analysis in Noisy Settings," in *Proc. Int. Conf. Mach. Learn. (ICML),* Stockholm, Sweden, Jul. 2018, pp. 3410–3419.

[164] R. Ward, X. Wu, and L. Bottou, "Adagrad stepsizes: Sharp convergence over nonconvex landscapes, from any initialization," *arXiv preprint arXiv:1806.01811*, Jun. 2018.

[165] J. Weng, Y. Zhang, and W.-S. Hwang, "Candid covariance-free incremental principal component analysis," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 25, pp. 1034–1040, Aug. 2003.

[166] Y. Zhang and J. Weng, "Convergence analysis of complementary candid incremental principal component analysis," *Mich. Sta. Uni., East Lansing, MI, USA, Tech. Rep. MSU-CSE-01-23*, Aug. 2001.

[167] S. Chrétien, C. Guyeux, and Z.-W. O. Ho, "Average performance analysis of the stochastic gradient method for online PCA," in *Int. Conf. Mach. Lear. Optimi. Data Sci.,* Volterra, Tuscany, Italy, Sep. 2018, pp. 231–242.

[168] I. Mitliagkas, C. Caramanis, and P. Jain, "Memory Limited, Streaming PCA," in *Adv. Neural Inf. Process. Syst. (NeurIPS),* Lake Tahoe, NV, Dec. 2013, pp. 2886–2894.

[169] M. Hardt and E. Price, "The Noisy Power Method: A Meta Algorithm with Applications," in *Adv. Neural Inf. Process. Syst. (NeurIPS),* Montreal, Canada, Dec. 2014, pp. 2861–2869.

[170] J. C. Lv, Z. Yi, and K. K. Tan, "Global convergence of Oja's PCA learning algorithm with a non-zero-approaching adaptive learning rate," *J. Theory Comput. Sci.*, vol. 367, no. 3, pp. 286–307, Dec. 2006.

[171] C.-L. Li, H.-T. Lin, and C.-J. Lu, "Rivalry of two families of algorithms for memory-restricted streaming pca," in *Proc. Artificial Intell. Statist. (AISTATS),* Cadiz, Spain, May 2016, pp. 473–481.

[172] M.-F. Balcan, S. S. Du, Y. Wang, and A. W. Yu, "An improved gap-dependency analysis of the noisy power method," in *Proc. Conf. Learn. Theory (COLT),* New York, NY, Jun. 2016, pp. 284–309.

[173] C. Kim and D. Klabjan, "Stochastic variance-reduced algorithms for pca with arbitrary mini-batch sizes," in *Proc. Artificial Intell. Statist. (AISTATS),* Virtual only, Jun. 2020, pp. 4302–4312.

[174] L. Balzano, "On the equivalence of oja's algorithm and GROUSE," in *Proc. Artificial Intell. Statist. (AISTATS),* Virtual only, Mar. 2022, pp. 7014–7030.

[175] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Proc. Annual Allerton Conf. Comm. Control Comput.* Allerton, IA, Sep. 2010, pp. 704–711.

[176] P. Jain, C. Jin, S. M. Kakade, P. Netrapalli, and A. Sidford, "Streaming PCA: Matching Matrix Bernstein and Near-Optimal Finite Sample Guarantees for Oja's Algorithm," in *Proc. Conf. Learn. Theory (COLT),* New York, NY, Jun. 2016, pp. 1147–1164.

[177] De Sa, Christopher and Olukotun, Kunle and Ré, Christopher, "Global convergence of stochastic gradient descent for some non-convex matrix problems," in *Int. Conf. Mach. Learn. (ICML)*, Jul. 2015, pp. 2332–2341.

[178] Z. Allen-Zhu and Y. Li, "First efficient convergence for streaming k-pca: a global, gap-free, and near-optimal rate," in *Proc. IEEE Ann. Symp. Found. Comput. Sci. (FOCS)* Berkeley, California, Oct. 2017, pp. 487–492.

[179] D. Huang, J. Niles-Weed, and R. Ward, "Streaming k-pca: Efficient guarantees for oja's algorithm, beyond rank-one updates," in *Conference on Learning Theory*, Jul. 2021, pp. 2463–2498.

[180] L. Zhang, T. Yang, J. Yi, R. Jin, and Z.-H. Zhou, "Stochastic optimization for kernel pca," in *AAAI Conf. Artif. Intell.* Pheonix, Arizona, Mar. 2016.

[181] A. Bhaskara and P. M. Wijewardena, "On distributed averaging for stochastic k-PCA," in *Adv. Neural Inf. Process. Syst. (NeurIPS),* Vancouver, Canada, vol. 32, Dec. 2019.

[182] R. Martin-Clemente and V. Zarzoso, "On the Link Between L1-PCA and ICA," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, pp. 515–528, Jun. 2016.

[183] S. Kim, "Gradient-based simulation optimization," in *Proc. 2006 Wint. Simul. Conf.* IEEE, Dec. 2006, pp. 159–167.

[184] R. Mahony, U. Helmke, and J. Moore, "Gradient algorithms for principal component analysis," *The ANZIAM Journal*, vol. 37, no. 4, pp. 430–450, Apr. 1996.

[185] F. Leone, L. Nelson, and R. Nottingham, "The folded normal distribution," *J. Technometrics*, vol. 3, pp. 543–550, Nov. 1961.

[186] W. N. Street, W. H. Wolberg, and O. L. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," in *Proc. IS&T/SPIE Int. Symp. Electronic Imaging: Sci. Technol.* San Jose, CA, Jun. 1993, pp. 861–870.

[187] O. L. Mangasarian, W. N. Street, and W. H. Wolberg, "Breast cancer diagnosis and prognosis via linear programming," *Operations Res.*, vol. 43, no. 4, pp. 570–577, Aug. 1995.

[188] M. Mozaffari and P. P. Markopoulos, "Robust barron-loss tucker tensor decomposition," in *Proc. IEEE Asilomar Conf. Signals Syst. Comput.* Online only, Oct. 2021, pp. 1651–1655.

[189] P. J. Huber, "Robust estimation of a location parameter," in *Breakthroughs in statistics.* Springer, 1992, pp. 492–518.

[190] R. Garg, N. Wadhwa, S. Ansari, and J. T. Barron, "Learning single camera depth estimation using dual-pixels," in *IEEE Conf. Comput. Vis. Pattern Rec. (CVPR)* Long Beach, CA, Jun. 2019, pp. 7628–7637.

[191] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, pp. 643–660, Jun. 2001.

[192] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 218–233, Feb. 2003.

[193] B. Lei and L.-Q. Xu, "Real-time outdoor video surveillance with robust foreground extraction and object tracking via multi-state transition management," *Pattern Recogn. Lett.*, vol. 27, no. 15, pp. 1816–1825, Nov. 2006.

[194] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trams. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[195] A. C. Bovik, *Handbook of image and video processing.* Academic press, Jul. 2010.

[196] Y. Zhang, Y. Liu, X. Li, and C. Zhang, "Salt and pepper noise removal in surveillance video based on low-rank matrix recovery," *Comput. Visual Med.*, vol. 1, no. 1, pp. 59–68, Mar. 2015.

[197] A. Joshi, A. K. Boyat, and B. K. Joshi, "Impact of wavelet transform and median filtering on removal of salt and pepper noise in digital images," in *Proc. IEEE Int. Conf. Iss. Chall. Intell. Comput. Tech. (ICICT)* Ghaziabad, India., Feb. 2014, pp. 838–843.

[198] S. Hauberg, A. Feragen, and M. J. Black, "Grassmann averages for scalable robust PCA," in *Proc. IEEE Conf. Comput. Vis. Pattern Rec. (CVPR),* Columbus, OH, Jun. 2014, pp. 3810–3817.

[199] T. W. Anderson, "Asymptotic theory for principal component analysis," *Ann. Math. Statist.*, vol. 34, no. 1, pp. 122–148, Jul. 1963.

[200] X. Mestre, "Improved estimation of eigenvalues and eigenvectors of covariance matrices using their sample estimates," *IEEE Trans. Inform. Theory*, vol. 54, no. 11, pp. 5113–5129, Oct. 2008.

[201] K.-T. Fang, S. Kotz, and K. N. Wang, *Symmetric Multivariate and Related Distributions.* CRC Press, Jan. 2018.

[202] Z. M. Landsman and E. A. Valdez, "Tail conditional expectations for elliptical distributions," *Nor. Amer. Actu. J.*, vol. 7, pp. 55–71, Oct. 2003.

[203] D. Kelker, "Distribution theory of spherical distributions and a location-scale parameter generalization," *Sankhyā: The Ind. J. Statist., Ser. A*, vol. 32, pp. 419–430, Dec. 1970.

[204] T. Zhang, A. Wiesel, and M. S. Greco, "Multivariate generalized Gaussian distribution: Convexity and graphical models," *IEEE Trans. Signal Process.*, vol. 61, pp. 4141–4148, Jun. 2013.

[205] A. K. Gupta, T. Varga, and T. Bodnar, *Elliptically contoured models in statistics and portfolio theory*. Springer, 2013.

[206] G. R. Ducharme and P. L. de Micheaux, "A goodness-of-fit test for elliptical distributions with diagnostic capabilities," *J. Mult. Anal.*, vol. 178, p. 104602, Jul. 2020.

[207] G. G. Hazel, "Multivariate gaussian mrf for multispectral scene segmentation and anomaly detection," *IEEE Trans. Geo. Rem. Sens.*, vol. 38, no. 3, pp. 1199–1211, May 2000.

[208] H. Jin, Q. Liu, H. Lu, and X. Tong, "Face detection using improved lbp under bayesian framework," in *Proc. Int. Conf. Image Gra. (ICIG),* Hong Kong, China, Dec. 2004, pp. 306–309.

[209] G. G. Raleigh and V. Jones, "Multivariate modulation and coding for wireless communication," *IEEE J. Sel. Areas Comm.*, vol. 17, no. 5, pp. 851–866, May 1999.

[210] Y. An and D. Liu, "Multivariate Gaussian-based false data detection against cyber-attacks," *IEEE Acc.*, vol. 7, pp. 119 804–119 812, Aug. 2019.

[211] O. E. Barndorff-Nielsen and N. Shephard, "Econometric analysis of realized covariation: High frequency based covariance, regression, and correlation in financial economics," *Econometrica*, vol. 72, no. 3, pp. 885–925, May 2004.

[212] C. Biernacki, G. Celeux, and G. Govaert, "Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models," *Comput. Statist. & Data Anal.*, vol. 41, no. 3-4, pp. 561–575, Jan. 2003.

[213] J. Minguillon and J. Pujol, "JPEG standard uniform quantization error modeling with applications to sequential and progressive operation modes," *J. Electron. Imag.*, vol. 10, pp. 475–486, Apr. 2001.

[214] T. Eltoft, T. Kim, and T.-W. Lee, "On the multivariate Laplace distribution," *IEEE Signal Process. Lett.*, vol. 13, pp. 300–303, Apr. 2006.

[215] S. Kotz, T. Kozubowski, and K. Podgorski, *The Laplace distribution and generalizations: a revisit with applications to communications, economics, engineering, and finance.* Springer Science & Business Media, Dec. 2012.

[216] S. Kotz and S. Nadarajah, *Multivariate t-distributions and their applications.* Cambridge University Press, 2004.

[217] R. Kan and G. Zhou, "Modeling non-normality using multivariate t: Implications for asset pricing," Rotman School of Management, University of Toronto, Toronto, Canada, Tech. Rep., Dec. 2003.

[218] W. Hu and A. N. Kercheval, "Portfolio optimization for student t and skewed t returns," *Quant. Fin.*, vol. 10, no. 1, pp. 91–105, Jan. 2010.

[219] F. J. Fabozzi, P. N. Kolm, D. A. Pachamanova, and S. M. Focardi, "Robust portfolio optimization," *J. Port. Manag.*, vol. 33, no. 3, pp. 40–48, Apr. 2007.

[220] N. Balakrishnan, *Handbook of the logistic distribution.* CRC Press, 2013.

[221] S. Nadarajah, "A generalized normal distribution," *J. Appl. Statist.*, vol. 32, pp. 685–694, Sep. 2005.

[222] M. Novey, T. Adali, and A. Roy, "A complex Generalized Gaussian Distribution—Characterization, Generation, and Estimation," *Trans. Signal Process.*, vol. 58, pp. 1427–1433, Nov. 2009.

[223] F. Pascal, L. Bombrun, J.-Y. Tourneret, and Y. Berthoumieu, "Parameter estimation for multivariate generalized Gaussian distributions," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 5960–5971, Sep. 2013.

[224] D. González-Jiménez, F. Pérez-González, P. Comesana-Alfaro, L. Pérez-Freire, and J. L. Alba-Castro, "Modeling Gabor coefficients via generalized Gaussian distributions for face recognition," in *Proc. Int. Conf. Imag. Process.,* San Antonio, TX, Oct. 2007, pp. 485–488.

[225] Q. Z. Ahmed, K.-H. Park, and M.-S. Alouini, "Ultrawide bandwidth receiver based on a multivariate generalized gaussian distribution," *IEEE Trans. Wireless Commu.*, vol. 14, no. 4, pp. 1800–1810, Sep. 2014.

[226] M. S. Davis, P. Bidigare, and D. Chang, "Statistical modeling and ML parameter estimation of complex SAR imagery," in *Proc. Asilomar Conf. Signal. Syst. Comput.,* Pacific Grove, CA, Nov. 2007, pp. 500–502.

[227] O. Bernard, J. D'hooge, and D. Fribouler, "Statistical modeling of the radio-frequency signal in echocardiographic images based on generalized Gaussian distribution," in *Proc. 3rd Int. Symp. Bio. Imag.: Nano to Macro.,* Arlington, VA, Apr. 2006, pp. 153–156.

[228] A. Dytso, R. Bustin, H. V. Poor, and S. Shamai, "Analytical properties of generalized Gaussian distributions," *J. Statist. Dist. Appl.*, vol. 5, pp. 1–40, Dec. 2018.

[229] J. K. Blitzstein and J. Hwang, *Introduction to Probability (1st ed.).* Chapman and Hall., 2014.

[230] A. Batsidis and K. Zografos, "A necessary test of fit of specific elliptical distributions based on an estimator of song's measure," *J. Mult. Anal.*, vol. 113, pp. 91–105, Jan. 2013.

[231] L. J. Rogers, "An extension of a certain theorem in inequalities," *Messenger Math.*, vol. 17, pp. 145–150, Mar. 1888.

[232] R. A. Horn and C. R. Johnson, *Matrix Analysis.* Cambridge University Press, Oct. 2012.

[233] B. Minnehan and A. Savakis, "Grassmann Manifold Optimization for Fast L1-Norm Principal Component Analysis," *IEEE Signal Process. Lett.*, vol. 26, pp. 242–246, Dec. 2018.

[234] Yahoo Finance. [Online]. Available: https://finance.yahoo.com.Accessed:Nov.2021.

[235] M. Hofert, "On sampling from the multivariate t distribution." *R J.*, vol. 5, no. 2, p. 129, Dec. 2013.

[236] Kaggle Body Measurement Dataset. [Online]. Available: https://www.kaggle.com/datasets/tombutton/body-measurements.Accessed:Jan.2022.

[237] MIT Visual and Modeling Group, Visual Texture Dataset. [Online]. Available: https://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html.Accessed:Dec.2021.

[238] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Dec. 2012, pp. 1097–1105.

[239] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Rec. (CVPR)* Las Vegas, NV, Jun. 2016, pp. 770–778.

[240] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, Sep. 2014.

[241] MATLAB Human Activity Detection Dataset. [Online]. Available: https://www.mathworks.com/help/stats/sample-data-sets.html.Accessed:Mar.2022.

[242] UCI Machine Learning Repository. Fisher Iris Dataset. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Iris.

[243] UCI Machine Learning Repository. Ionosphere Dataset. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Ionosphere.Accessed:Mar.2022.

[244] MATLAB Exam Grades Dataset. [Online]. Available: https://www.mathworks.com/help/stats/sample-data-sets.html.Accessed:Mar.2022.

[245] M. Krol, "Low-rank clustering via lp1-pca," M.S. thesis, Dept. Elect. and Microelect. Engg., Rochester Institute of Technology, Rochester, NY, USA, May 2022. [Online]. Available: https://scholarworks.rit.edu/theses/11201/

[246] A. Björck and G. H. Golub, "Numerical methods for computing angles between linear subspaces," *Math. Comput*, vol. 27, no. 123, pp. 579–594, Sep. 1973.

Appendices

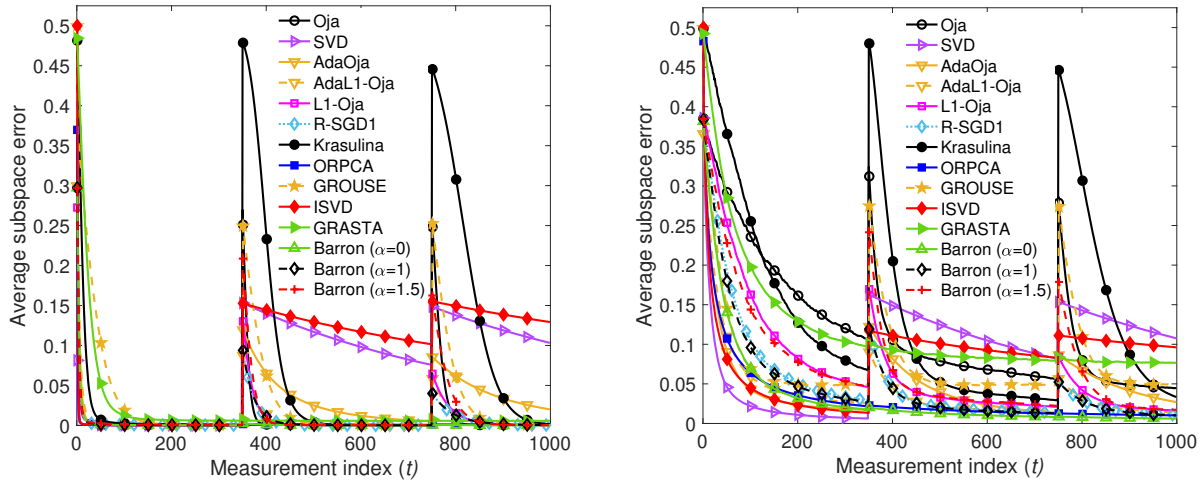
Appendix A

Chapter 3

A.0.1 Experiments on Outlier Resistance

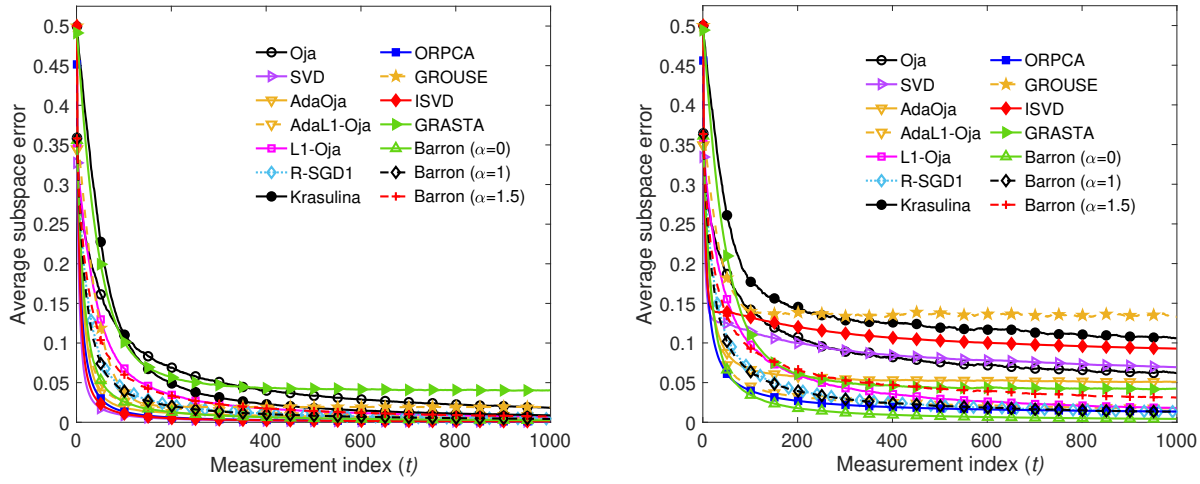
We repeat the experiment associated with Figure 3.8 by maintaining the same setting of Section 3.4.4 and estimate the performance of other non-Oja type algorithms including ORPCA [85], AdaOja [151], ISVD [97], GRASTA [93], GROUSE [175], and Krasulina [155]. We compute the average subspace error for $K = 2$ versus measurement index (t) over 10^4 realizations and plot it in Figure A.1(a). We observe that all methods process nominal data similarly and achieve low subspace error. However, when outliers are encountered at measurement indices $t = 350, 750$, methods relying on the L2-norm, namely SVD (obtained as $\mathbf{Q}_t^{(\text{svd})} \leftarrow \texttt{SVD}\big([\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t]\big)$), Oja, ISVD, and GROUSE are significantly affected. Robust methods like L1-Oja, AdaL1-Oja (L1-Oja with adaptive step size scheme as in [151]), GRASTA, and ORPCA offer relatively better outlier resistance. The proposed method offers high outlier resistance, especially for lower values of α. For $\alpha = 0$, the proposed method quickly converges to low subspace error and stays there by completely disregarding the outliers.

We repeat the above experiment with low SNR (=1 dB) and plot the results in Figure A.1(b). We observe that L2 methods including Oja, and Krasulina converge slowly to low subspace error due to noisy gradients at each update and it is significantly affected by the outliers fixed at indices $t =$ 350, 750. Interestingly, Krasulina achieves lower subspace error much quicker than Oja. GRASTA, although robust against outliers, converges slowly under noise. SVD converges the fastest because it has access to t measurements are at update index, as opposed to just one measurement used by other methods. Despite fast convergence initially, SVD experience performance degradation when it



(a) (b)

Figure A.1. Results on synthetic data with fixed outlier indices $t = 350, 750$. Average subspace error versus measurement index t for (a) SNR = 20dB and (b) SNR = 20dB.



(a) (b)

Figure A.2. Results on synthetic data with SNR=4dB and each measurement corrupted by an outlier with some probability. Average subspace error versus measurement index t for probability of outlier corruption per frame (a) = 0 and (b) = 2.5%.

encounters outliers. Similar performance is demonstrated by ISVD. ORPCA depicts resistance and noise. AdaOja is able to achieve high noise immunity owing to its adaptive step-size selection, but it is misled by outliers. AdaL1-Oja leverages the adaptive step size scheme to achieve low subspace error under noise and its reliance on the L1-norm imparts outlier resistance. The proposed method on the other hand, converges faster to lower subspace error and demonstrates more resistance against the outliers for lower α values.

In our subsequent experiment, we let each measurement be corrupted by outlier with probability 2.5% instead of fixing the outlier indices beforehand and plot the resulting average subspace error versus update index in Figure A.2(b). We set the SNR to 4dB in this study and as expected, L2 methods like GROUSE, OJA, SVD, Krasulina, AdaOja, and Krasulina are misguided by outliers and they converge to higher subspace error. Robust methods such as AdaL1-Oja, RSGD1, GRASTA, and ORPCA demonstrate clear resistance against outliers by converging to lower subspace errors. The proposed method demonstrates sturdy outlier resistance for lower α values converge to lower subspace errors faster compared to higher α values. Next, we re-run the experiment by setting the probability of corruption per measurement to zero and plot the subspace estimation performances in Figure A.2(a). Comparing A.2(a) and A.2(b) demonstrated the susceptibility of L1-based methods to outliers, while the robust counterparts offer significant immunity against them.

A.0.2 Glare/shadow Artifact Removal in Face Images

We repeat the experiment corresponding to Figure 3.10, with the same data and experimental setting on other stat-of-the-art methods including SVD, ISVD, Krasulina, GRASTA, GROUSE, and ORPCA. We present the glare/shadow artifact removal in Figure A.3 and observe that SVD, ISVD, Krasulina, and GROUSE retain most of the glare and shadow specularities on the reconstructed face. ORPCA, although a robust method, performs similar to L2-based methods. GRASTA, performs well be removing most of the glare and shadow artifacts in the image. The proposed method for $\alpha = 0.8$ demonstrates the best performance by getting obtaining a smooth image of the face free of any glare/shadow, even on face images that are mostly dark on one side, for example, rows 3, 6, and 9.

A.0.3 Foreground/background Separation in Surveillance Videos

Finally, we repeat the experiment of Figure 3.11 with the same setting and present the performance of other state-of-the-art methods such as SVD, ISVD, Krasulina, GRASTA, and ORPCA. We note from Figure A.4 that SVD, ISVD, Krasulina retain the foreground in the frame. GROUSE obtains a frame with ghostly appearance of the foreground in the estimated background. GRASTA, although a robust method obtains an estimate of the background with some foreground in it. ORPCA and the proposed method for $\alpha = 0.75$ are able to obtain a clean estimate of the background frame without any foreground component in it.

Figure A.3. Glare/shadow removal results. Comparison with other state-of-the-art methods. Rows 1 - 3 correspond to subject 05, rows 4 - 6 correspond to subject 09, and rows 7 - 9 correspond to subject 18 respectively. For each subject, we demonstrate the glare/shadow artifact removal at frame indices $t = 40$, 46, and 54.

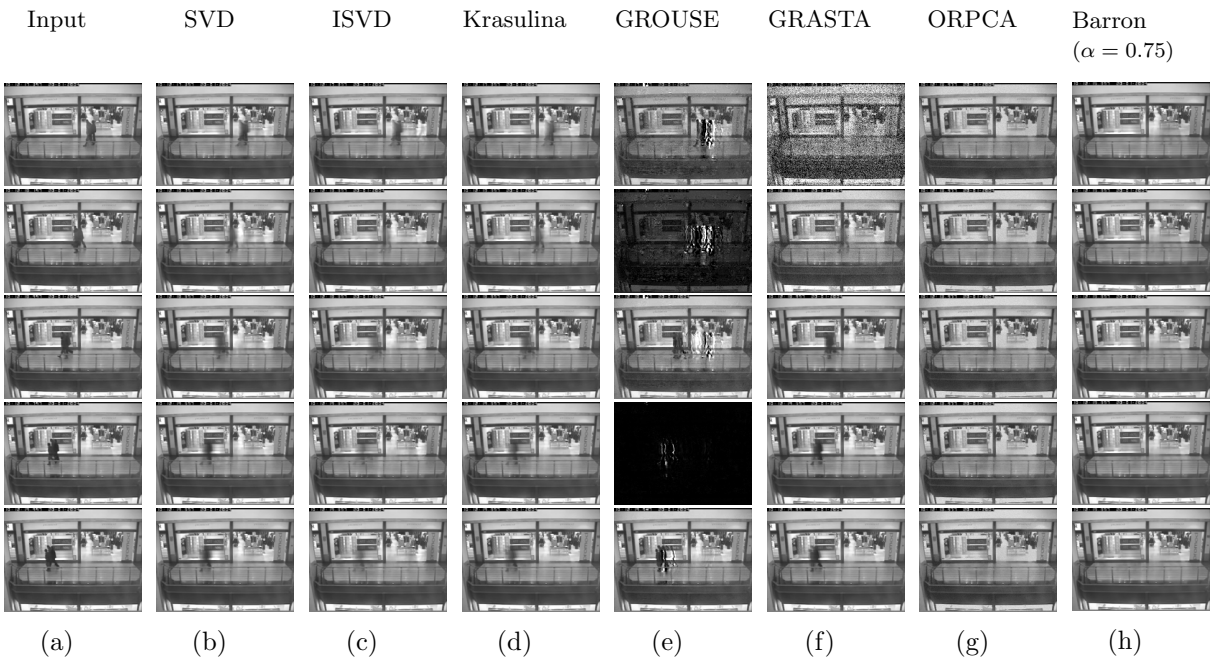


Figure A.4. Video background/foreground separation. Comparison with other state-of-the-art methods. Rows 1 - 5 correspond to frame indices $t = 100$, 120, 140, 160, and 165.

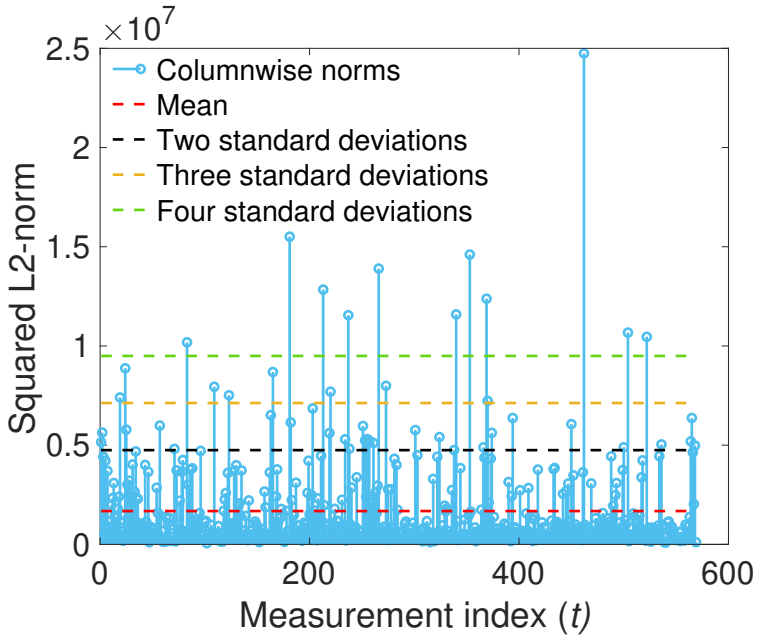


Figure A.5. Squared column-wise L2-norms versus measurement index of the Wisconsin breast cancer dataset.

A.0.4 Analysis of the Wisconsin Breast Cancer Dataset

Firstly, we note that the size of the dataset is 30×569 and compute the squared L2-norm of each measurement. We plot it versus the measurement index in Figure A.5 and observe that some

measurements have significantly higher norm compared to all the others. In order to quantify how large the norms of some measurements are, we plot the mean, two standard deviations, three standard deviations, and four standard deviations as reference points. We notice that the norm of most measurements are around the mean, however, some measurements have norm more than two, three, or four standard deviations from the mean, signifying much higher norm compared to most measurements and such measurements can be deemed norm outliers. In order to check if these measurements are subspace outliers, we follow the following procedure. We consider the whole dataset belonging to class 1 (benign class) $\mathbf{X}1$ and compute its top K PCs, $\mathbf{U}1 \in \mathbb{S}_{D=30,K=3}$ via SVD. Next, we store measurements that have norm less than 3 standard deviations from the mean in matrix $\tilde{\mathbf{X}}1$ and compute its K PCs, $\tilde{\mathbf{U}}1 \in \mathbb{S}_{D=30,K=3}$. Finally, we compute the subspace angle [246] between $\mathbf{U}1$ and $\tilde{\mathbf{U}}1$ and find it to be about $60°$, signifying that the measurements with large magnitude are also subspace outliers for class 1. For class 2, although some measurements are norm outliers, we find them not to be subspace outliers.

Appendix B

Chapter 4

B.0.1 Proof of Lemma 7

For the sake of continuity, we restate Lemma 7 here in the following. For $0 < p < 2$ and $\mathbf{R} \in \mathbb{S}_{D,K}$, the objective in (4.19) is upper-bounded,

$$\sum_{j=1}^{K} \left(\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j\right)^{\frac{p}{2}} \leq \left(\sum_{j=1}^{K} \mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j\right)^{\frac{p}{2}} K^{\frac{2-p}{2}}, \tag{B.1}$$

and achieves equality if and and if $\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j$ is constant $\forall j \in \{1, 2, \ldots, K\}$. We recall Holder's inequality [231], which states that for any two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^D$,

$$|\mathbf{x}^\top \mathbf{y}| \leq \|\mathbf{x}\|_m \|\mathbf{y}\|_n, \tag{B.2}$$

where $1 < m, n \leq +\infty$, such that $\frac{1}{m} + \frac{1}{n} = 1$, with equality if and only if $\left(\frac{x_j}{\|\mathbf{x}\|_m}\right)^m = \left(\frac{y_j}{\|\mathbf{y}\|_n}\right)^n$, $\forall j$. Now, set $\mathbf{y} = \mathbf{1}_K$ (vector of all ones of size K) and $x_j = (\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j)^{p/2}$. From (B.2), we get $\sum_{j=1}^{K} (\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j)^{p/2} \leq \{\sum_{j=1}^{K} \{(\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j)^{p/2}\}^m\}^{1/m} \left(\sum_{j=1}^{K} 1^n\right)^{1/n}$. Setting $m = 2/p$, we get

$$\sum_{j=1}^{K} (\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j)^{p/2} \leq \{\sum_{j=1}^{K} \mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j\}^{p/2} K^{1/n}. \tag{B.3}$$

Since $m = 2/p$ and $m > 1$, p takes values between 0 and 2, that is, $0 < p < 2$ and $n = 2/(2-p)$. We note that the equality in (B.3) is achieved if and only if $\left(\frac{x_j}{\|\mathbf{x}\|_m}\right)^m = \left(\frac{y_j}{\|\mathbf{y}\|_n}\right)^n = \frac{1}{K}$. We have,

$\left(\frac{x_j}{\|\mathbf{x}\|_m}\right)^m = \frac{(\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j)^{mp/2}}{\sum_{j=1}^K (\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j)^{mp/2}} = \frac{\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j}{\sum_{j=1}^K \mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j}$. Therefore, $\left(\frac{x_j}{\|\mathbf{x}\|_m}\right)^m = \frac{\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j}{\sum_{j=1}^K \mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j} = \frac{1}{K} = \left(\frac{y_j}{\|\mathbf{y}\|_n}\right)^n$ if and only if $\mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j$ is equal (constant) $\forall j$. This concludes our proof.

B.0.2 Proof of Corollary 2 of Lemma 8

Lemma 8 states that the upper-bound in (4.20) (ignoring the scaling) is further upper-bounded as

$$\left(\sum_{j=1}^K \mathbf{r}_j^\top \mathbf{\Lambda} \mathbf{r}_j\right)^{\frac{p}{2}} \leq \left(\sum_{j=1}^K \lambda_j\right)^{\frac{p}{2}}. \tag{B.4}$$

The corresponding corollary 2 claims that if $D > K$, the upper-bound in (4.21) if and only if the bottom $D-K$ rows of $\mathbf{R}$ are zeros row vectors. That is, the optimum $\mathbf{R}^* = \begin{bmatrix} \tilde{\mathbf{R}} \in \mathbb{S}_{K,K} \\ \mathbf{0}_{D-K \times K} \end{bmatrix}$, accordingly, the solution to (4.18) is of the form $\mathbf{E}\mathbf{R}^* = \mathbf{E}_{:,1:K}\tilde{\mathbf{R}}$, where $\mathbf{E}_{:,1:K}$ consist of the top-K eigenvectors of $\mathbf{\Sigma}$.

We recall that the objective in problem (3.1) can be rewritten as $\texttt{trace}(\mathbf{Q}^\top \mathbf{C} \mathbf{Q})$ to obtain the corresponding PCA problem

$$\underset{\mathbf{Q} \in \mathbb{S}_{D,K}}{\operatorname{argmax}} \texttt{trace}(\mathbf{Q}^\top \mathbf{C} \mathbf{Q}). \tag{B.5}$$

The above problem in (B.5) can be recast as a problem of estimating the optimal rotation matrix $\mathbf{R} \in \mathbb{R}_{D,K}$, by setting $\mathbf{Q} = \mathbf{E}\mathbf{R}$ and solving for

$$\mathbf{R}^* = \underset{\mathbf{R} \in \mathbb{S}_{D,K}}{\operatorname{argmax}} \texttt{trace}(\mathbf{R}^\top \mathbf{\Lambda} \mathbf{R}) \tag{B.6}$$

Clearly, the $\mathbf{R}^*$ that maximizes the above problem in (B.6) also maximizes the quantity on the left hand side of (B.4).

Now, we note that (B.5) is solved by $\mathbf{Q}$ of the form $\mathbf{Q} = \mathbf{E}_{:,1:K}\tilde{\mathbf{R}}$, where $\mathbf{E}\mathbf{\Lambda}\mathbf{E}^\top \leftarrow \texttt{EVD}(\mathbf{C})$, $\mathbf{E}_{:,1:K}$ consists of the dominant eigenvectors of $\mathbf{C}$, and $\tilde{\mathbf{R}} \in \mathbb{S}_{K,K}$. At the optimum, the metric in (B.5) achieves the value of $\sum_{j=1}^k \lambda_k$, where λ_k is the k-th eigenvalue of $\mathbf{C}$. Therefore, since $\mathbf{I}_D \mathbf{\Lambda} \mathbf{I}_D \leftarrow \texttt{EVD}(\mathbf{\Lambda})$, the optimal rotation matrix, $\mathbf{R}^*$ takes the form $\mathbf{R}^* = \mathbf{I}_{D_{[:,1:K]}}\tilde{\mathbf{R}} = \begin{bmatrix} \tilde{\mathbf{R}} \in \mathbb{S}_{K,K} \\ \mathbf{0}_{D-K \times K} \end{bmatrix}$. Setting $\mathbf{R} = \mathbf{R}^*$ achieves equality in (B.4). This concludes the proof.

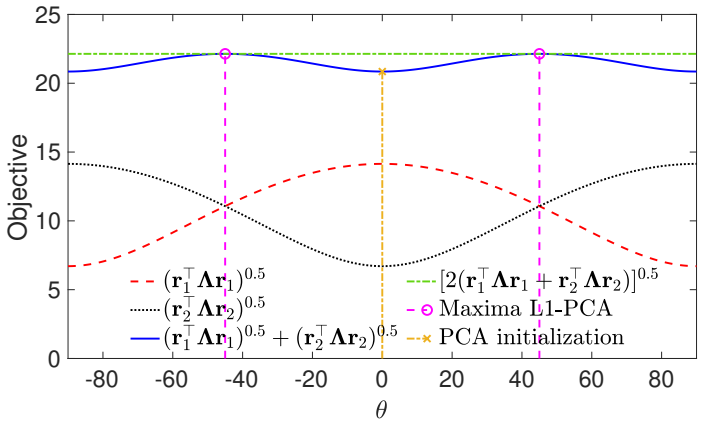


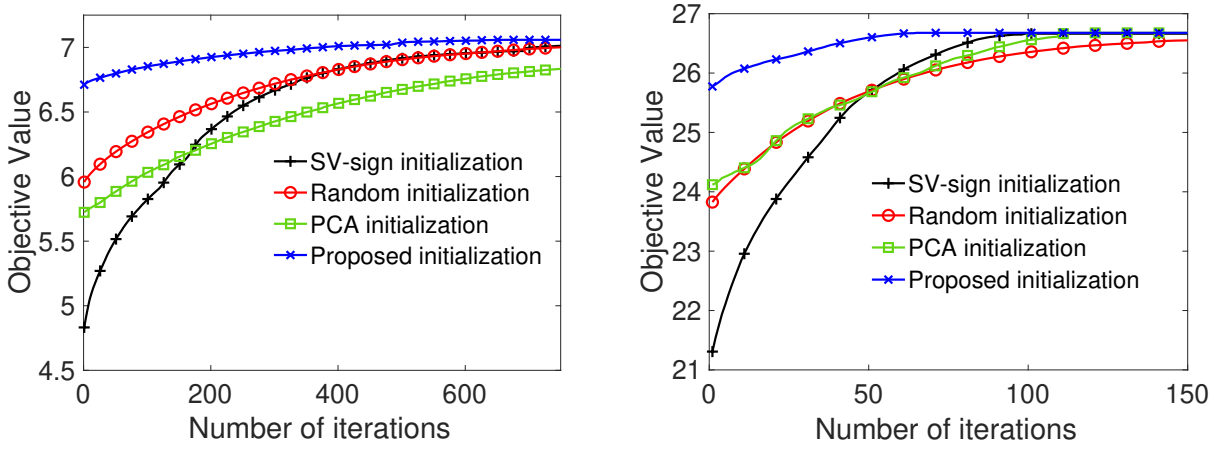
Figure B.1. PCA initialization occurs at the minimum of the L1-PCA objective.

B.0.3 Underperformance of PCA Initialization on Elliptical data

We offer an insight onto why PCA initialization underperforms on Elliptical data. We set $D = K = 2$ in (4.20) and plot the quantities, $(\mathbf{r}_1^\top \mathbf{\Lambda} \mathbf{r}_1)^{0.5}$, $(\mathbf{r}_2^\top \mathbf{\Lambda} \mathbf{r}_2)^{0.5}$, $(\mathbf{r}_1^\top \mathbf{\Lambda} \mathbf{r}_1)^{0.5} + (\mathbf{r}_2^\top \mathbf{\Lambda} \mathbf{r}_2)^{0.5}$, and $(\mathbf{r}_1^\top \mathbf{\Lambda} \mathbf{r}_1 + \mathbf{r}_2^\top \mathbf{\Lambda} \mathbf{r}_2)^{0.5}$, where $\mathbf{r}_1 = [\texttt{cos}(\theta), \texttt{sin}(\theta)]^\top$, $\mathbf{r}_2 = [\texttt{-sin}(\theta), \texttt{cos}(\theta)]^\top$, and $\mathbf{\Lambda} = \begin{bmatrix} 200 & 0 \\ 0 & 45 \end{bmatrix}$, versus $\theta \in \{-90°, -89.9°, \ldots, +90°\}$ in Figure B.1. We observe that the L1-PCA objective metric in terms of the rotation matrix on the left hand side of (4.20), $(\mathbf{r}_1^\top \mathbf{\Lambda} \mathbf{r}_1)^{0.5} + (\mathbf{r}_2^\top \mathbf{\Lambda} \mathbf{r}_2)^{0.5}$ meets its upper, bound $(\mathbf{r}_1^\top \mathbf{\Lambda} \mathbf{r}_1 + \mathbf{r}_2^\top \mathbf{\Lambda} \mathbf{r}_2)^{0.5}$ when $\theta = \pm 45°$. Additionally, the metric, $(\mathbf{r}_1^\top \mathbf{\Lambda} \mathbf{r}_1)^{0.5} + (\mathbf{r}_2^\top \mathbf{\Lambda} \mathbf{r}_2)^{0.5}$ is minimum when $\theta = 0°$. Note that PCA initialization coincides with $\theta = 0°$, therefore corresponding to the minimum value of the metric which occurs the farthest from the maximum value. Such initialization to a minimum value of the metric may make it infeasible for the iterative L1-PCA algorithms to converge to better metric values, especially to obtain near-optimal or optimal solutions.

B.0.4 Experimental Studies on the Ionosphere and Exam Grades Dataset (non-Elliptical)

We repeat the experiments corresponding to the Ionosphere and Exam grades dataset in Section 4.4.2 by replacing the L1-PCA algorithm of [6] by L1-BF [4]. It was shown in [4] that L1-BF obtains higher metric compared to other sub-optimal algorithms, especially when using the SV-sign initialization proposed in that paper. We note that for the Ionosphere dataset, the proposed initialization scheme begins at the highest objective value and converges faster than other initializations, whereas other initialization methods take a long time to converge to smaller metric values. We note that PCA initialization demonstrates the worst performance. Comparing Figure B.2 (a) to Figure



(a) (b)

Figure B.2. L1-PCA objective versus number of iterations of the L1-BF algorithm on (a) the Ionosphere dataset and (b) the Exam Grades dataset for $p = 1$.

(a) (b)

Figure B.3. Illustration of the images from the VisTex dataset used in our studies on real-world data in Section 4.4.2. (a) presents the Bark image and (b) presents the Leaves image.

4.22 (a), we observe that L1-BF is able to achieve higher L1-PCA objective values for random and SV-sign initialization.

Finally, for the Exam grades dataset, we observe from Figure B.2 (b) that the proposed initialization converges to the highest L1-PCA objective quickly. Other initialization methods converge much slower. Random initialization is also able to achieve the maximum metric value obtained by the proposed initialization scheme, as opposed to the maximum metric value obtained by the algorithm of [6] using random initialization in Figure 4.22 (b).

B.0.5 Illustration of the Bark and Leaves Images of the VisTex Dataset

We present illustrations of the texture images we used in our experimental study on real-world datasets in Section 4.4.2 in Figure B.3 (a) and (b).