Learning, Inference, and Unmixing of Weak, Structured Signals in Noise

by

Arvind Prasadan

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in the University of Michigan
2020

Doctoral Committee:

Associate Professor Raj Rao Nadakuditi, Chair
Professor Jinho Baik
Professor Jeffrey A. Fessler
Professor Alfred O. Hero III

Arvind Prasadan
prasadan@umich.edu
ORCID iD: 0000-0002-2521-7113

Dedication

> The mystery of life is not a problem to be solved; it is a reality to be experienced [126].

For those who have helped me experience life.

Acknowledgments

I want to thank my advisor, Prof. Raj Nadakuditi. I look back to where I was when I first came to the University of Michigan, and I stand miles away from where I was. Without your direction, ideas, and enthusiasm for all that we do, I would not be here writing this thesis. I am extremely grateful for your flexibility that allowed me to work remotely a lot of the time, and for your generosity with your time. Moreover, being a teaching assistant for your classes, first 551 and then what is now 505 has been an incredibly rewarding experience.

I would also like to thank my committee members, Prof. Jeff Fessler, Prof. Al Hero, and Prof. Jinho Baik, for all of their feedback and time. Your time and efforts have made this a better thesis. In particular, being a teaching assistant for 551 with Prof. Fessler was a very good learning experience, and I found that the organizational skills I picked up that semester were very useful in the following semesters.

I want to thank my fellow teaching assistants, (in roughly chronological order) Raj Tejas Suryaprakash, David Hong, Jonas Kersulis, Travis DePrato, Hao Wu, Yash Bhalgat, Dipak Narayan, Naveen Murthy, Rishi Sonthalia, Cameron Blocker, Claire Lin, and Kyle Gilman. It has been an honor and pleasure to work with all of you, and it was never a chore to hang around after class was over and talk about random things.

I'd also like to thank my fellow research group members, (in roughly chronological order) Nick Asendorf, Raj Tejas Suryaprakash, Himanshu Nayar, Brian Moore, and Hao Wu for many stimulating discussions about research and coursework.

Additionally, I'd like to thank collaborators Prof. Debashis Paul and Asad Lodhia. Prof Paul, I greatly appreciate your generosity with your time and efforts. Asad, I've greatly enjoyed our conversations and collaboration, and thank you for your flexibility that allowed me to work remotely.

I'd also like to thank the staff in the EECS department, especially Judi Jones and Kristen Thornton. Your efforts have made my time here much easier, especially as I finish up.

I'd also like to thank everyone I worked with at MIT Lincoln Laboratory, including Rajmonda Caceres, Cem Sahin, and Vijay Gadepally. Also Lisa and Pat for all of their assistance and making my logistically weird setup work smoothly.

I would like to thank my cohort and other friends from my first two years at Michigan. All of the fun we had went a long way in firming my conviction that I'd made the right choice in coming here. I also thank everyone I met at MATC; my time at the helm taught me many life skills. I'd particularly like to thank Mitchell Bloch and Rajan Bhambroo; I

look back on our time in charge of the club fondly. Also, I would like to thank everyone I saw regularly on Thursday evenings, in particular Harish Ganesh, Blake Charlebois, and Huzefa Khalil. Your company has been incredibly enjoyable, enlightening, and memorable, and because of you, I leave Ann Arbor as a far more cultured man than I was when I came in.

I thank my parents and brother for their encouragement to do all that I am capable of, to pursue what I am passionate about and what I am good at, and all of their help and assistance over the years. From mundane matters like logistics and helping me move to the less tangible matters like making me feel like my room in Pittsburgh was still my room, I am incredibly grateful.

I'd like to thank my (former) roommate Josh Hunt. We've come a long way since that bus ride in 2010. I'm unsure how exactly to convey via writing what your companionship has meant to me over the years.

Finally, I'd like to thank Florica Constantine. You were with me when I applied to schools and are with me now as I finish with school. For me to list or type out why and how I am thankful to you would be too limiting. Your support has meant the world to me.

Table of Contents

List of Figures

List of Tables

Abstract

In this thesis, we study two methods that can be used to learn, infer, and unmix weak, structured signals in noise: the Dynamic Mode Decomposition algorithm and the sparse Principal Component Analysis problem. Both problems take as input samples of a multivariate signal that is corrupted by noise, and produce a set of structured signals. We present performance guarantees for each algorithm and validate our findings with numerical simulations.

First, we study the Dynamic Mode Decomposition (DMD) algorithm. We demonstrate that DMD can be used to solve the source separation problem. That is, we apply DMD to a data matrix whose rows are linearly independent, additive mixtures of latent time series. We show that when the latent time series are uncorrelated at a lag of one time-step then the recovered dynamic modes will approximate the columns of the mixing matrix. That is, DMD unmixes linearly mixed sources that have a particular correlation structure.

We next broaden our analysis beyond the noise-free, fully observed data setting. We study the DMD algorithm with a truncated-SVD denoising step, and present recovery guarantees for both the noisy data and missing data settings. We also present some preliminary characterizations of DMD performed directly on noisy data. We end with some complementary perspectives on DMD, including an optimization-based formulation.

Second, we study the sparse Principal Component Analysis (PCA) problem. We demonstrate that the sparse inference problem can be viewed in a variable selection framework and analyze the performance of various decision statistics. A major contribution of this work is the introduction of False Discovery Rate (FDR) control for the principal component estimation problem, made possible by the sparse structure. We derive lower bounds on the size of detectable coordinates of the principal component vectors, and utilize these lower bounds to derive lower bounds on the worst-case risk.

Chapter 1

Introduction

This thesis studies methods for learning, inferring, and unmixing weak, structured signals in the presence of noise. To illustrate what this sentence and the title of this work mean, we will work with a concrete example: the cocktail party problem [29]. Imagine that we are in a restaurant, and at most of the tables, people are talking; there is also music in the background, maybe noises from the kitchen or cutlery or the street. Now, in this restaurant, there are microphones scattered throughout. Each microphone will pick up a mixture of all of these sounds. For simplicity, we may take the microphone recordings to be synchronized in time and assume that there are several, scattered microphones throughout the venue. These recordings form a dataset that is then given to an analyst, who is given the task of isolating and extracting the individual conversations and voices from the recordings. That is, the goal is to *unmix* the voices from the recordings. Alternatively, we may say that we wish to *learn* or *infer* the individual conversations, and discover, learn, or separate what is content from what is irrelevant. However, perhaps the *noise* level in the restaurant is high, or the microphones are low quality, or the individual voices are soft (i.e., the *signals* that we wish to *unmix* are *weak*). It may be difficult to proceed without assuming something about the *structure* of the voices to be extracted. That is, to get anywhere, we need to make some assumptions, or, it may be the case that unless certain assumptions hold, we cannot get anywhere.

We might imagine other datasets or situations in which we need to learn, infer, or unmix weak signals in the presence of noise. For example, given human biomarker data for both healthy and sick subjects, we may want to locate the biomarkers that correspond to the sickness. Here, one common structural assumption is that only a few biomarkers (out of many) are relevant. For another example, if we are given seismic sensor measurements for a region, we might want to separate what signals are potentially dangerous tremors and what are human activities or sensor noise. Here, we would have several physical assumptions about the structure of real tremors.

In the following section, we formalize this discussion, describe two threads of continuity through this thesis, and introduce the two algorithms that we study: the Dynamic Mode Decomposition (DMD) algorithm and the sparse Principal Component Analysis (PCA) problem.

1.1 Two Threads of Continuity

The first thread of continuity throughout this thesis is the analysis of high dimensional eigenvalue problems. We study two problems that in a very general sense, start with the same setup. In particular, we are in the setting where we receive p-dimensional samples $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ as columns of the $p \times n$ matrix X; i.e.,

$$X = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_n \end{bmatrix}.$$

Given X, we form a second matrix. In the case of the first problem that we study, the Dynamic Mode Decomposition (DMD) algorithm [110], we form

$$\widehat{A} = X_{(1)} X_{(0)}^{+},$$

where

$$X_{(0)} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{n-1} \end{bmatrix} \text{ and } X_{(1)} = \begin{bmatrix} \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_n \end{bmatrix},$$

and $^{+}$ denotes the Moore-Penrose pseudoinverse. Note that the ordering of the samples is important in DMD. In the second problem, sparse Principal Component Analysis (PCA) [61], we form the sample covariance matrix,

$$\widehat{\Sigma} = \frac{1}{n} X X^{H},$$

where ^{H} denotes the conjugate transpose (we use ^{T} for the transpose). For this problem, the sample ordering is not important. After forming these matrices, we take an eigendecomposition, and the resulting eigenvectors are one of the quantities of interest. If Q is the matrix of normalized eigenvectors, in both problems, the columns of

$$\left(Q^{+}X\right)^{H}$$

are the other quantities of interest.

Both of the problems that we study seek to learn or infer latent signals from observations of a linear mixture, i.e., they are unmixing algorithms. However, there are often limitations that preclude perfect recovery or inference, and a second thread of continuity through this thesis comes in the form of the data that is used to solve these eigenvalue problems. When the data are not noisy, are fully observed, fit whatever model we are using, and the number of samples is large relative to the dimension, there are no problems: we expect good results. However, if each sample $\mathbf{x}_i$ is corrupted by additive noise, e.g.,

$$\mathbf{x}_i = \mathbf{y}_i + \mathbf{g}_i,$$

where $\mathbf{y}_i$ is the true, latent signal and $\mathbf{g}_i$ is the part of $\mathbf{x}_i$ that is entirely noise, then the eigendecompositions of $\widehat{A}$ and $\widehat{\Sigma}$ will change [12, 13]. The estimated eigenvectors will be perturbed relative to the truth, where the truth is defined as the output in the noise free setting. A similar issue will occur if the samples are only partially observed, e.g., if instead of observing all entries of X, we instead observe a subset. Finally, both with and without noise, if the number of variables p is very large, forming, storing, and decomposing $\widehat{A}$ and $\widehat{\Sigma}$ may be computationally demanding. Additionally, if the number of variables p is large relative to the number of samples and the data are noisy or partially observed, the problems are compounded and it may be impossible to consistently estimate the true signals or eigenvectors without additional structure [17, 61, 12].

In this thesis, we will provide specific solutions for each algorithm/problem. For both problems, we will first use a low rank plus noise data structure. That is, while we may observe many variables, there are relatively few (latent) variables that actually matter or explain whatever behavior is going on. For DMD, we will show that if the eigenvectors have corresponding temporal variations that are uncorrelated at a lag of one time step while having non-vanishing autocorrelations at the same lag, we have good recovery performance. That is, if the data have a specific temporal structure, DMD works well. For the PCA problem, we impose sparsity: i.e., the principal components are sparse, or, only have a few non-zero coordinates. We will characterize the performance of sparse PCA methods in terms of the size of these non-zero coordinates.

1.2 Organization

In Chapter 2. we study the DMD algorithm. We first present a novel analysis of the algorithm in the noise free setting. In particular, our analysis reveals that DMD solves the blind source separation problem [29]. We present performance bounds for DMD in the noise free setting, and validate them with numerical simulations. As a precursor to Chapter 3, we study the performance of DMD with missing data. In particular, we present performance bounds for DMD performed after a denoising or imputation step, i.e., the truncated singular value decomposition (SVD) [88].

In Chapter 3, we study the DMD algorithm in the presence of additive white noise. We once again present performance bounds for DMD performed after the truncated SVD denoising step. We revisit the missing data setting, and derive some characterizations of DMD on missing *and* noisy data. We also derive some preliminary characterizations of DMD directly performed on a noisy data matrix, and present some conjectures about the behavior of DMD on pure noise.

In Chapter 4, we discuss some complementary viewpoints of DMD. We begin with an optimization-based formulation. That is, in Chapter 2, we found that DMD unmixes signals that are uncorrelated at a lag of one time-step, and we translate this result into the language of convex optimization. We demonstrate how we may use this formulation to introduce sparsity, and present some preliminary results for the convergence of our problem. We next present what we will call Hilbert DMD, that is. DMD performed on Hilbert-transformed signals. We end with an application of DMD on a real dataset.

In Chapter 5, we discuss the sparse PCA problem. We present a novel perspective, and frame the variable selection problem in the language of multiple hypothesis testing. In the rank-1 setting where the coordinates/loadings are non-negative, we analyze various test statistics (based off a summation, ℓ_2 norm, and ℓ_1 norm) and derive performance bounds and detection limits. Moreover, we relate the problem to the sparse normal means problem, and introduce the idea of False Discovery Rate (FDR) control for the sparse PCA problem [36].

Finally, in Chapter 6, we offer some concluding remarks.

1.2.1 List of Publications

Below, we list the publications that some of the work in this thesis is based on. At the time of writing, [101] and [100] have been published, [103] has been accepted for publication, and [102] is under review. Moreover, the work in Chapter 3 is in preparation to be submitted.

1. DMD for Blind Source Separation, Chapter 2, [102]
   Arvind Prasadan and Raj Rao Nadakuditi. "Time Series Source Separation using Dynamic Mode Decomposition". In: *arXiv preprint arXiv:1903.01310* (2019, In Review)

2. DMD for Blind Source Separation, Chapter 2, [101]
   Arvind Prasadan and Raj Rao Nadakuditi. "The Finite Sample Performance of Dynamic Mode Decomposition". In: *Signal and Information Processing (GlobalSIP), 2018 IEEE Global Conference on*. IEEE. 2018, pp. 1–5

3. Noisy DMD, Chapter 3, [100]
   Arvind Prasadan, Asad Lodhia, and Raj Rao Nadakuditi. "Phase Transitions in the Dynamic Mode Decomposition Algorithm". In: *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2019 IEEE Workshop on*. IEEE. 2019, pp. 1–5

4. Sparse PCA, Chapter 5, [103]
   Arvind Prasadan, Raj Rao Nadakuditi, and Debashis Paul. "Sparse Equisigned PCA: Algorithms and Performance Bounds in the Noisy Rank-1 Setting". In: *arXiv preprint arXiv:1905.09369, to appear, Electronic Journal of Statistics* (2020)

Chapter 2

The Dynamic Mode Decomposition

The Dynamic Mode Decomposition (DMD) extracted dynamic modes are the non-orthogonal eigenvectors of the matrix that best approximates the one-step temporal evolution of the multivariate samples. In the context of dynamical system analysis, the extracted dynamic modes are a generalization of global stability modes. We apply DMD to a data matrix whose rows are linearly independent, additive mixtures of latent time series. We show that when the latent time series are uncorrelated at a lag of one time-step then, in the large sample limit, the recovered dynamic modes will approximate, up to a column-wise normalization, the columns of the mixing matrix. Thus, DMD is a time series blind source separation algorithm in disguise, but is different from closely related second order algorithms such as the Second-Order Blind Identification (SOBI) method and the Algorithm for Multiple Unknown Signals Extraction (AMUSE). All can unmix mixed stationary, ergodic Gaussian time series in a way that kurtosis-based Independent Components Analysis (ICA) fundamentally cannot. We use our insights on single lag DMD to develop a higher-lag extension, analyze the finite sample performance with and without randomly missing data, and identify settings where the higher lag variant can outperform the conventional single lag variant. We validate our results with numerical simulations, and highlight how DMD can be used in change point detection.[1]

[1] This is joint work with Raj Nadakuditi, and has appeared in [102, 101].

2.1 Introduction

The Dynamic Mode Decomposition (DMD) algorithm was invented by P. Schmid as a method for extracting dynamic information from temporal measurements of a multivariate fluid flow vector [110]. The dynamic modes extracted are the non-orthogonal eigenvectors of a non-normal matrix that best linearizes the one-step evolution of the measured vector (to be quantified in what follows).

Schmid showed that the dynamic modes recovered by DMD correspond to the globally stable modes in the flow [110]. The non-orthogonality of the recovered dynamic modes reveals spatial structure in the temporal evolution of the measured fluid flows in a way that other second order spatial correlation based methods, such as the Proper Orthogonal Decomposition (POD), do not [66]. This spurred follow-on work on other applications and extensions of DMD to understanding dynamical systems from measurements.

2.1.1 Previous work on DMD and the analysis of dynamical systems

Early analyses of the DMD algorithm drew connections between the DMD modes and the eigenfunctions of the Koopman operator from dynamical system theory. Rowley et al. and Mezić et al. showed that under certain conditions, the DMD modes approximate the eigenfunctions of the Koopman operator for a given system [109, 81]. Related work in [8] studied the Koopman operator directly, analyzed its spectrum, and compared it against the spectrum of the matrix decomposed in DMD. The work in [109] also explained how the linear DMD modes can elucidate the structure in the temporal evolution in nonlinear fluid flows. The work in [30] provided a further analysis of the Koopman operator and more connections to DMD. More recently, Lusch et al. have shown how deep learning can be combined with DMD to extract modes for a non-linearly evolving dynamical system [74].

There have been several extensions of DMD. The authors in [28] developed a method to improve the robustness of DMD to noise. Jovanovic et al. proposed a sparsity-inducing formulation of DMD that allowed fewer dynamic modes to better capture the dynamical system [64]. Tu et al. developed a DMD variant that takes into account systematic measurement errors and measurement noise [124]; this framework was extended in [51]. A Bayesian, probabilistic variant of DMD was developed in [115], where a Gibbs sampler for the modes and a sparsity-inducing prior were proposed. Another recent extension of DMD includes an online (or streaming) version of DMD [136].

Additionally, there have been applications of DMD to other domains besides computational fluid mechanics. The work in [9] applied DMD to compressed sensing settings. A related work applied DMD to model the background in a streaming video [97]. The authors in [76] applied DMD to finance, by using the predicted modes and temporal variations to forecast future market trends. The authors in [14] brought DMD to the field of robotics, and used DMD to estimate perturbations in the motion of a robot. DMD has also been applied to power systems analysis, where it has been used to analyze transients in large power grids [10]. There are many more applications and extensions, and we point the interested reader to the recent book by Kutz et al. [67].

2.1.2 Our main finding: DMD unmixes lag-1 (or higher lag) uncorrelated time series

We will introduce the general problem and model in Section 2.2, but before proceeding, we will consider a simple, illustrative example. Suppose that we are given multivariate observations $\mathbf{x}_t \in \mathbb{R}^p$ modeled as

$$\mathbf{x}_t = H\,\mathbf{s}_t = QD\,\mathbf{s}_t, \tag{2.1}$$

where t is an integer, $H = QD \in \mathbb{R}^{p\times p}$ is a non-singular mixing matrix, and $\mathbf{s}_t \in \mathbb{R}^p$ is the latent vector of random signals (or sources). The matrix $Q \in \mathbb{R}^{p\times p}$ has unit-norm columns and is related to H by

$$Q = \begin{bmatrix} \mathbf{q}_1 & \dots & \mathbf{q}_p \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{h}_1}{\|\mathbf{h}_1\|_2} & \cdots & \dfrac{\mathbf{h}_p}{\|\mathbf{h}_p\|_2} \end{bmatrix}. \tag{2.2}$$

Setting entries of the diagonal matrix $D = \operatorname{diag}(d_1, \dots, d_p)$ as $d_i = \|\mathbf{h}_i\|_2$ ensures that $H = QD$ as in (2.1). Note that by the phrase 'mixing matrix', we mean that $H\,\mathbf{s}_t$ produces a linear combination of the coordinates of $\mathbf{s}_t$, i.e., a mixing of the coordinates.

In what follows, we will adopt the following notational convention: we shall use boldface to denote vectors such as $\mathbf{s}_t$. Matrices, such as H, will be denoted by non-boldface uppercase letters; and scalars, such as s_{t1}, will be denoted by lower-case symbols.

We assume, without loss of generality, that

$$\mathbb{E}\left[\mathbf{s}_t\right] = \mathbf{0}_p \;\text{ and }\; \mathbb{E}\left[\mathbf{s}_t\,\mathbf{s}_t^T\right] = \mathrm{I}_p\,. \tag{2.3}$$

The lag-τ covariance matrix of $\mathbf{s}_t$ is defined as

$$\mathbb{E}[L_\tau] = \mathbb{E}\left[\mathbf{s}_t\,\mathbf{s}_{t+\tau}^T\right] = \mathbb{E}\left[\mathbf{s}_{t+\tau}\,\mathbf{s}_t^T\right], \tag{2.4}$$

where τ is a non-negative integer.

If we are able to form a reliable estimate $\widehat{H}$ of the mixing matrix H from the n multivariate observations $\mathbf{x}_1, \ldots, \mathbf{x}_n$ then, via Eq. (2.1), we can unmix the latent signals $\mathbf{s}_t$ by computing $\widehat{H}^{-1}\,\mathbf{x}_t$. Inferring Q and computing $\widehat{Q}^{-1}\,\mathbf{x}_t$ will also similarly unmix the signals. Inferring the mixing matrix and unmixing the signals (or sources) is referred to as *blind source separation* [29].

Our key finding is that when the lag-1 covariance matrix $\mathbb{E}[L_1]$ in (2.4) is diagonal, corresponding to the setting where the latent signals are lag-1 uncorrelated weakly stationary time series, and there are sufficiently many samples of $\mathbf{x}_t$, then the DMD algorithm in (2.22) produces a non-normal matrix whose non-orthogonal eigenvectors are reliably good (to be quantified in what follows) estimates of Q in (2.1). In other words, DMD unmixes lag-1 uncorrelated signals and weakly stationary time series.

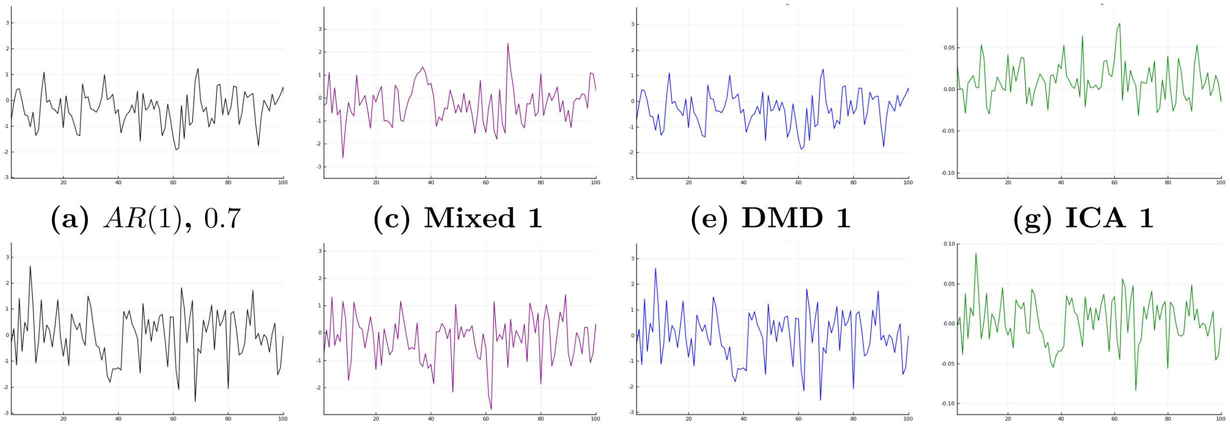
Our findings reveal that a straightforward extension of DMD, described in Section 2.3 and (2.26), allows τ-DMD to unmix lag τ uncorrelated signals and time series. This brings up the possibility of using a higher lag τ to unmix signals that might exhibit a more favorable correlation at larger lag τ than at a lag of one. Indeed, Figure 2.5 provides one such example where 2-DMD provides a better estimate of Q than does 1-DMD.

Our main contribution, which builds on our previous work in [101], is the analysis of the unmixing performance of DMD and τ-DMD (introduced in Section 2.3), when unmixing deterministic signals and random, weakly stationary time series in the finite sample regime and in the setting where there is randomly missing data in the observations $\mathbf{x}_t$.

2.1.3 New insight: DMD can unmix ergodic time series that kurtosis-based ICA cannot

Independent Component Analysis (ICA) is a classical algorithm for blind source separation [70, 84] that is often used for the cocktail party problem of unmixing mixed audio signals. Our analysis reveals that DMD can be succesfully applied to this problem as well because independent audio sources are well modeled as one-lag (or higher lag) uncorrelated (see Figure 2.9).

It is known that kurtosis- or cumulant-based ICA (hereafter refered to as ICA) fails when more then one of the independent, latent signals is normally distributed [56, Ch. 7]. A consequence of this is that ICA will fail to unmix mixed independent, ergodic time series with Gaussian marginal distributions: each latent signal will have a kurtosis of zero. Our analysis, culminating in Theorem 2.2, reveals that DMD will succeed in this setting, even as ICA fails; see Figure 2.1 for an illustration where ICA fails to unmix two mixed, independent Gaussian AR(1) processes while DMD succeeds. Note that these are two independent realizations of AR(1) processes, and that there is no averaging over several realizations. Thus, DMD can and should be used by practitioners to re-analyze multivariate time series data for which the use of ICA has not revealed any insights.



(a) $AR(1)$, 0.7 (c) **Mixed 1** (e) **DMD 1** (g) **ICA 1**

(b) $AR(1)$, 0.2 (d) **Mixed 2** (f) **DMD 2** (h) **ICA 2**

Figure 2.1: We generate two AR(1) signals of length $n = 1000$, with coefficients 0.2 and 0.7 respectively. We mix them orthogonally, and compare the performance of ICA and DMD at unmixing them. We observe that the squared error, defined in (2.39), of ICA is 0.41, whereas that from DMD is 0.0055. Indeed, ICA fails because the marginal distribution of each AR(1) process is Gaussian. In these plots, for ease of visualization we plot the first 100 samples.

2.1.4 New insight: DMD can unmix mixed Fourier series that PCA cannot

Principal Component Analysis (PCA) is a standard, linear dimensionality reduction method [61] that can be expressed in terms of the singular value decomposition (SVD) of a data matrix. The eigenwalker model, described in [123], is a linear model for human motion. The model is a linear combination of vectors, via

$$\mathbf{x}_t = \sum_{i=1}^{k} \mathbf{q}_i \cos\left(\omega_i t + \phi_i\right). \tag{2.5}$$

The vectors $\mathbf{q}_i$ are the modes of the motion, and each has a sinusoidal temporal variation. We generate our model as follows:

$$\mathbf{x}_t = \mathbf{q}_1 \cos\left(2t\right) + \mathbf{q}_2 \cos\left(t/4\right),$$

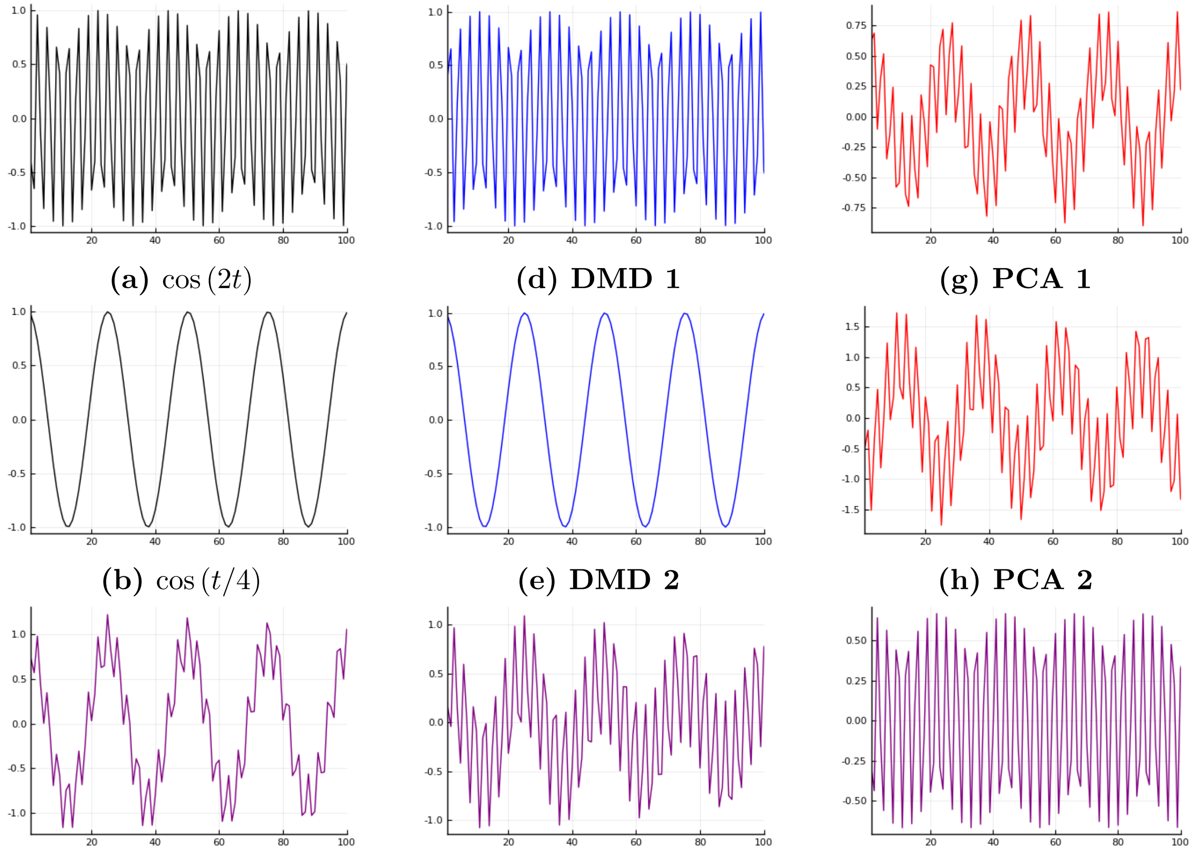
for $t = 1$ to 1000, where

$$Q = \begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 \end{bmatrix} = \begin{bmatrix} 1/3 & 2/\sqrt{5} \\ 2/3 & 1/\sqrt{5} \\ 2/3 & 0 \end{bmatrix}.$$

This model has been decomposed with ICA, and used for video motion editing and analysis [111]. Here, we apply PCA and compare it to DMD. In Figure 2.2, we display the results of unmixing with PCA and with DMD. We observe that DMD successfully unmixes the cosines, while PCA fails: note that unless the $\mathbf{q}_i$ are orthogonal, there is no hope of a successful unmixing. Moreover, the estimation of of Q from PCA fails, as we find that

$$\widehat{Q}_{PCA} = \begin{bmatrix} -0.686895 & 0.624695 \\ -0.623497 & -0.243983 \\ -0.373399 & -0.741774 \end{bmatrix},$$

which has a squared error of 0.81, while the estimate from DMD has a squared error of 2.9×10^{-7}, where the error is computed according to (2.33a).



(a) $\cos(2t)$ **(d) DMD 1** **(g) PCA 1**

(b) $\cos(t/4)$ **(e) DMD 2** **(h) PCA 2**

(c) Mixed 1 **(f) Mixed 2** **(i) Mixed 3**

Figure 2.2: We generate data according to the eigenwalker model (2.5), and use DMD and PCA to recover the cosine signals. We observe that DMD recovers the signals, while PCA does not. Indeed, we observe that the squared error for the recovered cosines, defined in (2.39), from PCA is 1.97**, whereas that from DMD is** 4.57×10^{-7}**. For ease of visualization, we zoom in on the first** 100 **samples.**

2.1.5 Connection with other algorithms for time series blind source separation

Let $H = U\Sigma V^T$ be the singular value decomposition (SVD) of H. Then, we have that $\mathbb{E}[\mathbf{x}_t] = \mathbf{0}_p$ and

$$\Sigma_{\mathbf{x}\mathbf{x}} = \mathbb{E}[\mathbf{x}_t \mathbf{x}_t^T] = HH^T = U\Sigma^2 U^T. \tag{2.6}$$

Given Σ_{xx} and $\mathbf{x}_t$, we can compute the whitened vector

$$\mathbf{w}_t = \Sigma_{\mathbf{x}\mathbf{x}}^{-1/2} \mathbf{x}_t, \tag{2.7}$$

whose covariance matrix is given by $\mathbb{E}[\mathbf{w}_t\,\mathbf{w}_t^T] = \mathrm{I}_p$. Then from (2.1) and (2.6) we have that

$$\mathbf{w}_t = (UV^T)\,\mathbf{s}_t, \tag{2.8}$$

where the mixing matrix UV^T is an orthogonal matrix because the U and V matrices, which correspond to the left and right singular vector matrices of H in (2.1) are orthogonal.

Equation (2.8) reveals that we can solve the blind source separation problem and unmix $\mathbf{s}_t$ from observations of $\mathbf{w}_t$ if we can infer the orthogonal mixing matrix UV^T from data. To that end, we note that

$$\mathbb{E}\left[\mathbf{w}_t\,\mathbf{w}_{t+\tau}^T\right] = (UV^T)\,\mathbb{E}\left[\mathbf{s}_t\,\mathbf{s}_{t+\tau}^T\right](UV^T)^T = (UV^T)\,\mathbb{E}[L_\tau](UV^T)^T. \tag{2.9}$$

Equation (2.9) reveals that when the latent signals $\mathbf{s}_t$ are lag-1 uncorrelated, *i.e.*, $\mathbb{E}[L_1]$ is a diagonal matrix, then the lag-1 covariance matrix of the whitened vector $\mathbf{w}_t$ will be diagonalized by the orthogonal matrix UV^T. The sample lag-1 covariance matrix computed from finite data will, in general, not be symmetric and so we might infer UV^T from the eigenvectors of the symmetric part: this leads to the AMUSE (Algorithm for Multiple Unknown Signals Extraction) method [121].

A deeper inspection of (2.9) reveals that if $\mathbf{s}_t$ are second order, weakly stationary time series that are uncorrelated for multiple values of τ (corresponding to multiple lags), then we can infer (UV^T) (which, incidentally corresponds to the polar part of the polar decomposition of the mixing matrix H in (2.1)) by posing it as joint-diagonalization of $\mathbb{E}\left[\mathbf{w}_t\,\mathbf{w}_{t+\tau_i}^T\right]$ for l lags corresponding to $\tau_1, \ldots, \tau_l$. This is the basis of the Second-Order Blind Identification (SOBI) method [11] where the joint diagonalization problem is addressed by finding the orthogonal matrix Γ that minimizes the sums-of-squares of the off-diagonal entries of $\Gamma^T\,\mathbb{E}\left[\mathbf{w}_t\,\mathbf{w}_{t+\tau_i}^T\right]\Gamma$. Numerically, this problem is solvable via the JADE method [26, 82, 83].

Miettinen et al analyze the performance of a symmetric variant of the SOBI method in [83] and the problem of determining the number of latent signals that are distinct from white noise in [77]. Their results for the performance are asymptotic and distributional. That is, the limiting distribution of the estimated matrix Γ is computed, when the input signals are realizations of some time series, with zero mean and diagonal autocorrelations at every lag $\tau \in \{0, 1, 2, \ldots\}$. As will be seen in what follows, these assumptions are very similar to those that we impose on DMD. Our analysis for the missing data setting is new and has no counter-part in the SOBI or AMUSE performance analysis literature.

In Table 2.1, we summarize the various algorithms for unmixing of stationary time series. Table 2.1 brings into sharp focus the manner in which DMD and τ-DMD are similar to and different from the AMUSE and SOBI algorithms. All algorithms diagonalize a matrix; SOBI and AMUSE estimate orthogonal matrices while DMD and τ-DMD estimate non-orthogonal matrices. The SOBI and AMUSE algorithms diagonalize cross-covariance matrices formed from whitened time series data while DMD and τ-DMD works on the time series data directly. Thus SOBI and AMUSE explicitly whiten the data while DMD implicitly whitens the data. SOBI and DMD exhibit similar performance (see Figure. 2.8) – a more detailed theoretical study comparing their performance in the noisy setting is warranted.

Algorithm	Key Matrix	Fit for Key Matrix	Numerical Method
DMD	$\widehat{A} = X_{(1)} \left[X_{(0)}\right]^{+}$	QL_1Q^{+}, Q non-orthogonal	Non-Symmetric Eig.
τ-DMD	$\widehat{A}_\tau = X^{\tau}_{(1)} \left[X^{\tau}_{(0)}\right]^{+}$	$QL_\tau Q^{+}$, Q non-orthogonal	Non-symmetric Eig.
AMUSE	$\widehat{A}_\tau = Y^{\tau}_{(1)} \left[Y^{\tau}_{(0)}\right]^{T}$	$\Gamma L_\tau \Gamma^T$, Γ orthogonal	Eig. of Symmetric part
SOBI	$\widehat{A}_{\tau_i} = Y^{\tau_i}_{(1)} \left[Y^{\tau_i}_{(0)}\right]^{T}$, $i \in \{1, 2, \ldots l\}$	$\Gamma L_{\tau_i} \Gamma^T$, Γ orthogonal	Joint Diagonalization

Table 2.1: Comparison of the various second order algorithms for time series blind source separation. Here $Y = \left[XX^T\right]^{-1/2} X$, is the whitened data matrix and $Y^{\tau}_{(0)}$ and $Y^{\tau}_{(1)}$ are defined analogous to $X^{\tau}_{(0)}$ and $X^{\tau}_{(1)}$, as in (2.21), (2.25), and (2.29), respectively.

2.1.6 Organization

The remainder of this chapter is organized as follows. In Section 2.2, we introduce the time series data matrix model and describe the DMD algorithm in Section 2.2.1. We describe a higher lag extension of DMD, which we call τ-DMD, in Section 2.3. We provide a DMD performance guarantee for unmixing deterministic signals in Section 2.4; a corollary of that result in Section 2.4.3 explains why DMD is particularly apt for unmixing multivariate mixtures of Fourier series such as the "eigen-walker" model. We extend our analysis to stationary, ergodic time series data in Section 2.4.4. In Section 2.5, we provide results for the estimation error of the latent signals. We analyze the setting where the time series data matrix has randomly missing data in Section 2.6. We validate our theoretical results with numerical simulations in Section 2.7. In Section 2.8, we describe how a time series matrix can be factorized using DMD to obtain a Dynamic Mode Factorization (DMF) involving the product of the DMD estimate of the (column-wise normalized) mixing matrix and the coordinates, which represent the unmixed latent signals. We show how DMF can be applied to the cocktail party problem in [29] in Section 2.8 and how unmixing the latent series via DMF can help improve time series change point detection in Section 2.8.2. We offer some concluding remarks in Section 2.9. The proofs of our results are deferred to the appendices.

Summary of Theorems

A contribution of this work is a non-asymptotic finite sample performance analysis for the DMD and τ-DMD algorithm in the setting where the mixed deterministic signals or stationary, ergodic time series are approximately (or exactly) one- or higher lag uncorrelated. Our main results will concern the estimation errors of the mixing matrices. Theorem 2.1 presents a general result with bounds for deterministic signals and all lags $\tau \geq 1$. Corollary 2.1 presents bounds for the lag-one, deterministic case where the latent signals are cosines. Theorem 2.2 generalizes Theorem 2.1 to the setting where the latent signals are realizations of a stationary, ergodic time series. We present results for the estimation of the latent signals in Theorem 2.3, and extend the results to missing data in 2.4.

2.2 Model and Setup

Suppose that, at time t, we are given a p dimensional time series vector

$$\mathbf{x}_t = \begin{bmatrix} x_{1t} & x_{2t} & \dots & x_{pt} \end{bmatrix}^T,$$

where an individual entry x_{jt}, for $j = 1, 2, \dots, p$, of $\mathbf{x}_t$ is modeled as

$$x_{jt} = \sum_{i=1}^{k} b_{ij} c_{it}, \tag{2.10}$$

and b_{ij} is the j^{th} entry of a p dimensional vector $\mathbf{b}_i$. Each c_{it} is the t^{th} entry of an n dimensional vector $\mathbf{c}_i$, and the c_{it} are samples of a time series. Equation (2.10) can be succinctly written in vector form as

$$\mathbf{x}_t = \sum_{i=1}^{k} \mathbf{b}_i \, c_{it} = B \begin{bmatrix} c_{1t} \\ \vdots \\ c_{kt} \end{bmatrix}, \tag{2.11}$$

where the $p \times k$ matrix B is defined as $B = \begin{bmatrix} \mathbf{b}_1 & \cdots & \mathbf{b}_k \end{bmatrix}$. We are given samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ corresponding to uniformly spaced time instances $t_1, \ldots t_n$. In what follows, without loss of generality, we assume that $t_i = i$. Let X be the $p \times n$ matrix defined as

$$X = \begin{bmatrix} \mathbf{x}_1 & \cdots & \mathbf{x}_n \end{bmatrix}. \tag{2.12}$$

We define the $n \times k$ matrix C with columns $\mathbf{c}_1, \ldots, \mathbf{c}_k$ as

$$C^T = \left\{ \begin{bmatrix} & c_{1t} & \\ \ldots & \vdots & \ldots \\ & c_{kt} & \end{bmatrix} \right\}_{t=1}^{n}. \tag{2.13}$$

Consequently, we have that

$$X = B\, C^T, \tag{2.14}$$

where C^T is the “latent time series” matrix given by (2.13). Equation (2.14) reveals that the multivariate time series matrix X is a linear combination of rows of the latent time series matrix.

Suppose that for $i = 1, \ldots, k$,

$$\mathbf{q}_i = \frac{\mathbf{b}_i}{\| \mathbf{b}_i \|_2} \text{ and } \mathbf{s}_i = \frac{\mathbf{c}_i}{\| \mathbf{c}_i \|_2}, \tag{2.15}$$

and the matrices

$$Q = \begin{bmatrix} \mathbf{q}_1 & \cdots & \mathbf{q}_k \end{bmatrix} \text{ and } S = \begin{bmatrix} \mathbf{s}_1 & \cdots & \mathbf{s}_k \end{bmatrix}. \tag{2.16}$$

Then, from (2.14), and from the definition of Q and S, it can be shown that

$$X = Q\, DS^T \tag{2.17}$$

where, for $i = 1, \ldots, k$,

$$D = \operatorname{diag}\left(\ldots, \|\, \mathbf{b}_i \,\|_2 \cdot \|\, \mathbf{c}_i \,\|_2, \ldots\right). \tag{2.18}$$

We will define

$$d_i = \|\, \mathbf{b}_i \,\|_2 \cdot \|\, \mathbf{c}_i \,\|_2, \tag{2.19}$$

and assume that, without loss of generality, the d_i and hence the $\mathbf{b}_i$, $\mathbf{c}_i$, $\mathbf{q}_i$, and $\mathbf{s}_i$ are ordered so that

$$d_1 \geq d_2 \geq \ldots \geq d_k > 0. \tag{2.20}$$

Note that by construction, in (2.17), the k columns of the matrices Q and S have unit norm. In what follows, we assume that Q and S have linearly independent columns, that $k \leq p \leq n - 1$, that the columns of S have zero mean, and that the columns of Q are canonically *non-random* and *non-orthogonal*. Our goal in what follows is to estimate the columns of the matrices Q and S.

2.2.1 Dynamic Mode Decomposition (DMD)

From (2.11), we see that the columns of X represent a multivariate time series. We first partition the matrix X into two $p \times n - 1$ matrices

$$X_{(0)} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{n-1} \end{bmatrix} \text{ and } X_{(1)} = \begin{bmatrix} \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_n \end{bmatrix}. \tag{2.21}$$

We then compute the $p \times p$ matrix $\widehat{A}$ via the solution of the optimization problem

$$\widehat{A} = \operatorname*{argmin}_{A \in \mathbb{R}^{p \times p}} \left\| X_{(1)} - A X_{(0)} \right\|_F. \tag{2.22}$$

The minimum norm solution to (2.22) is given by

$$\widehat{A} = X_{(1)} X_{(0)}^{+}, \tag{2.23}$$

where the superscript $^+$ denotes the Moore-Penrose pseudoinverse. Note that $\widehat{A}$ will be a non-symmetric matrix with a rank of at most k because X, from which $X_{(1)}$ and $X_{(0)}$ are derived, has rank k from the construction in (2.17). Let

$$\widehat{A} = \widehat{Q} \widehat{\Lambda} \widehat{Q}^{+}, \tag{2.24}$$

be its eigenvalue decomposition. In (2.24), $\widehat{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_k)$ is a $k \times k$ diagonal matrix, where the λ_i, ordered as $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_k| > 0$, are the, possibly complex, eigenvalues of $\widehat{A}$ and $\widehat{Q}$ is a $p \times k$ matrix of, generically non-orthogonal, unit-norm eigenvectors, denoted by $\widehat{\mathbf{q}}_i$.

In what follows, we will refer to the computation of (2.23) and the subsequent decomposition (2.24) as the DMD algorithm and we will show that under certain conditions, $\widehat{\mathbf{q}}_i$ is close to $\mathbf{q}_i$.

2.3 A Natural Generalization: $\tau-$DMD

We have just described the DMD algorithm at a lag of 1. That is, we let $X_{(0)}$ and $X_{(1)}$ differ by one time-step. However, we might easily allow $X_{(0)}$ and $X_{(1)}$ to differ by τ time steps, and in certain settings, it may be advantageous to use $\tau > 1$.

From (2.11), we recall that the columns of X represent a multivariate time series. We first partition the matrix X into two $p \times n - \tau$ matrices:

$$X_{(0)}^{\tau} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{n-\tau} \end{bmatrix} \text{ and } X_{(1)}^{\tau} = \begin{bmatrix} \mathbf{x}_{1+\tau} & \mathbf{x}_{2+\tau} & \cdots & \mathbf{x}_{n} \end{bmatrix}. \tag{2.25}$$

At this point, the procedure is identical to the DMD algorithm: we compute the $p \times p$ matrix $\widehat{A}(\tau)$ via the solution of the optimization problem

$$\widehat{A}_\tau = \underset{A \in \mathbb{R}^{p \times p}}{\mathrm{argmin}} \left\| X_{(1)}^{\tau} - A X_{(0)}^{\tau} \right\|_F, \tag{2.26}$$

and the minimum norm solution to (2.26) is given by

$$\widehat{A}_\tau = X_{(1)}^{\tau} \left(X_{(0)}^{\tau} \right)^{+}. \tag{2.27}$$

Once again, let

$$\widehat{A}_\tau = \widehat{Q} \widehat{\Lambda} \widehat{Q}^{+}, \tag{2.28}$$

be its eigenvalue decomposition. In (2.28), $\widehat{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_k)$ is a $k \times k$ diagonal matrix, where $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_k| \geq 0$ are the (possibly complex) eigenvalues of $\widehat{A}_\tau$ and $\widehat{Q}$ is a $p \times k$ matrix of, generically non-orthogonal, unit-norm eigenvectors that are denoted by $\widehat{\mathbf{q}}_i$.

In what follows, we will refer to the computation of (2.27) and the subsequent decomposition (2.28) as the τ-DMD algorithm. Note that the DMD algorithm is a special case of the τ-DMD algorithm, and when we say 'DMD' we mean the $\tau = 1$ setting.

2.4 Performance Guarantee

The central object governing the performance of the τ-DMD algorithm is the lag-τ cross covariance matrix. Let the $k \times k$ lag-τ covariance matrix L_τ defined as

$$[L_\tau]_{ij} = \sum_{l=1}^{n} S_{i,l} S_{j,[l+\tau] \bmod n}. \tag{2.29}$$

Note that we can succinctly express L_τ as $L_\tau = S^T(P^\tau S)$ where P is the matrix formed by taking the $n \times n$ identity matrix and circularly right shifting the columns by one.

2.4.1 Technical Assumptions

We will require the following set of technical assumptions on the data.

1. Assume that k is fixed, with

$$k \leq \min\{p, n - \tau\} \tag{2.30a}$$

2. Assume that the $\mathbf{q}_i$ are linearly independent, so that $\sigma_1(Q)/\sigma_k(Q)$ is a finite quantity:

$$1 \leq \frac{\sigma_1(Q)}{\sigma_k(Q)} < \infty. \tag{2.30b}$$

   Here, $\sigma_i(Q)$ denotes the i^{th} singular value of Q. Essentially, the conditioning of the $\mathbf{q}_i$ is independent of n and p. Moreover, the $\mathbf{q}_i$ are canonically *non-random* and *not necessarily orthogonal.*

3. Assume that

$$\lim_{n\to\infty} \frac{d_1}{d_k} \nrightarrow \infty, \tag{2.30c}$$

   i.e., that the limit of the ratio is finite.

4. Assume that columns of S (the $\mathbf{s}_i$) each have zero mean (the sum of each column is zero), and that they are linearly independent. Moreover, assume that there exists an

$\alpha > 0$ such that

$$\max_{i,j} |S_{ij}| \leq O\left(\frac{1}{n^{\alpha}}\right). \tag{2.30d}$$

I.e., the $\mathbf{s}_i$ are not too sparse.

5. Assume that τ is small relative to n; i.e., that

$$\tau n^{-2\alpha} \nrightarrow \infty \text{ and } n - \tau \approx n \text{ for large } n. \tag{2.30e}$$

Remark 2.1. *Conditions 1, 2, and the first part of 4 are required for the data matrix to actually have rank k. I.e., if there are k latent signals, we need the columns of Q to be linearly independent and we need the signals to be linearly independent to recover all k signals and the k columns of Q and not linear combinations thereof. We need at least as many linear combinations and samples as there are signals to recover the signals. Moreover, the linear independence and full column rank conditions yield that Q and S are unique, and hence can (in principle) be estimated uniquely up to a sign or phase shift. Note that for a rank k matrix, there are many different possible factorizations, but our results here will identify when the specific Q and S matrices can be recovered. Condition 3 ensures that, in the limit, we can recover all k signals. Intuitively, if the ratio (2.30c) diverged, the data matrix would eventually have a numerical rank smaller than k, and the smallest signal would look like noise relative to the largest. Finally, the second part of condition 4 ensures that the latent signals are sufficiently dense, or that they are not very transient. That is, the signals are not something like a spike. Condition 4 is purely technical and is needed for the proofs of the performance bounds. Finally, condition 5 is technical, and ensures that each of $X_{(0)}^{\tau}$ and $X_{(1)}^{\tau}$ contain enough information.*

To avoid an alphabet soup of constants, we use the notational shorthand $x \leq O(f(n))$ to mean that there exists a universal constant C independent of n such that x is bounded by $Cf(n)$, and we will write $O(f(n))$ instead of $Cf(n)$.

2.4.2 Deterministic Signals

We now establish a recovery condition for the setting where $\mathbf{c}_i$ in (2.13) are deterministic.

Remark 2.2. *In the following result and in all subsequent results, there is an ambiguity or mismatch between the ordering of the $\mathbf{q}_i$, $\mathbf{s}_i$, d_i, and $[L_\tau]_{ii}$ with that of the $\widehat{\mathbf{q}}_j$ and*

λ_j. Formally, there exists a permutation $\sigma(i)$ that reorders the $\widehat{\mathbf{q}}_j$ and λ_j to correspond to the $\mathbf{q}_i$ and other quantities, such that the error is minimal. In the statement of our results, without loss of generality, we will assume that $\sigma(i) = i$, i.e., that it is the identity permutation.

Theorem 2.1 (τ-lag DMD)**.** *For X as in (2.17) and L_τ defined as in (2.29), suppose that the conditions in (2.30) hold. Further suppose that*

$$\lim_{n\to\infty} |[L_\tau]_{ii}| \nrightarrow 0. \tag{2.31a}$$

Moreover, assume that for $i \neq j$ we have that

$$\left|[L_\tau]_{ij}\right| \leq O(f(n)) \text{ and } \left|\mathbf{s}_i^T\,\mathbf{s}_j\right| \leq O(f(n)) \tag{2.31b}$$

for some $f(n)$ such that $\lim_{n\to\infty} f(n) = 0$.
a) Then, assuming that p_i is given by

$$p_i = sign\left(\widehat{\mathbf{q}}_i^T\,\mathbf{q}_i\right), \tag{2.32}$$

we have that

$$\sum_{i=1}^{k} \|\widehat{\mathbf{q}}_i - p_i\,\mathbf{q}_i\|_2^2 \leq O\left(\left[\frac{d_1}{d_k}\right]^2 \cdot \frac{k^7}{\delta_L^2} \cdot \left[f^2(n) + \tau n^{-2\alpha}\right]\right), \tag{2.33a}$$

where δ_L is given by

$$\delta_L = \min_{i\neq j}\left|[L_\tau]_{ii} - [L_\tau]_{jj}\right|. \tag{2.33b}$$

b) Moreover, for each $[L_\tau]_{ii}$, we have that

$$|[L_\tau]_{ii} - \lambda_i|^2 \leq O\left(\left[\frac{d_1}{d_k}\right]^2 \cdot k^6 \cdot \left[f^2(n) + \tau n^{-2\alpha}\right]\right). \tag{2.33c}$$

Note that the bound (2.33a) depends on δ_L: if two of the signals have identical lag-τ autocorrelations, the bound becomes trivial and the signals may not be able to be unmixed.

Moreover, this result is entirely in terms of the latent signals, $\mathbf{s}_i$: $f(n)$ is the lag-1 cross correlation decay rate, α governs the sparsity/density of the signals, and d_i is the magnitude of each signal. We have specified conditions on the latent signals such that they may be unmixed. Of course, without knowledge of the latent signals, these bounds are not computable. Noting that δ_L is a function of τ, we anticipate that some values of τ would lead to better results than others: we will demonstrate this behavior numerically in Section 2.7.

2.4.3 Application of Theorem 2.1: DMD Unmixes Multivariate Mixed Fourier Series

Consider the setting where c_{it} in (2.10) is modeled as

$$c_{it} = \cos\left(\omega_i t + \phi_i\right). \tag{2.34}$$

The x_{it} is thus a linear mixture of Fourier series. This model frequently comes up in many applications such as the eigenwalker model for human motion: [122, Equations (1) and (2)], [123] and [125, Equations (1) and (2)].

This model fits into the framework of Theorem 2.1 via an application of Corollary 2.1 below. This implies the DMD modes will correctly correspond to the non-orthogonal mixing modes. Using PCA on the data matrix in this setting would recover orthogonal modes that would be linear combinations of the latent non-orthogonal dynamic modes.

Corollary 2.1 (Mixtures of Cosines)**.** *Assume that the $\mathbf{c}_i$ are given by (2.34) and that we apply DMD with $\tau = 1$. Then we have that*

$$\sum_{i=1}^{k} \|\widehat{\mathbf{q}}_i - p_i\, \mathbf{q}_i\|_2^2 \leq O\left(\left[\frac{d_1}{d_k}\right]^2 \cdot \frac{k^7}{\delta_L^4} \cdot \frac{1}{n}\right), \tag{2.35a}$$

where

$$\delta_L = \min_{i \neq j} \left|\cos\omega_i - \cos\omega_j\right|, \tag{2.35b}$$

and that for each ω_i, we have that

$$\left|\cos\omega_i - \lambda_i\right|^2 \leq O\left(\left[\frac{d_1}{d_k}\right]^2 \cdot \frac{k^6}{n}\right). \tag{2.35c}$$

Corollary 2.1 explains why DMD successfully unmixes the eigenwalker data in Figure 2.2. In that setting, PCA does not succeed because it returns an orthogonal matrix as an estimate of the non-orthogonal mixing matrix. The ability of DMD to reliably unmix non-orthogonally mixed multivariate Fourier series, and the fact that the eigenvalues are cosines of the frequencies, provides some context for the statement that DMD is a spectral algorithm where the eigen-spectra reveal information on Fourier spectra [109].

Note that by Theorem 2.1, we require that the lag-1 autocorrelations are distinct. In this case, it is equivalent to requiring that the cosines have distinct frequencies. In the notation of Theorem 2.1, we have that $\alpha = 1/2$ and $f(n) = 1/\sqrt{n}$.

2.4.4 Extensions of Theorem 2.1: Stationary, Ergodic Time Series

We now consider the setting where c_{it} are elements of a stationary, ergodic time series and the $\mathbf{c}_i$, thus formed; we say that a process is stationary and ergodic when its statistical properties do not change over time, and when they can be estimated from a sufficiently long realization. We point the reader to [63, Ch. 2.3, 15.4] for formal definitions of these terms. Consider the matrix

$$\mathbb{E}\left[L_\tau\right]_{ij} = \mathbb{E}\left[S_{i,l} S_{j,[l+\tau] \bmod n}\right]. \tag{2.36}$$

When $\mathbb{E}\, L_\tau$ is diagonal, then τ-DMD asymptotically unmixes the time series, as expressed in the Theorem below. We will require the assumptions from (2.30), with the following updates:

1. Assume that the $\mathbf{b}_i$, $\mathbf{c}_i$, $\mathbf{q}_i$, and $\mathbf{s}_i$ are ordered so that

$$\mathbb{E}\, d_1 \geq \mathbb{E}\, d_2 \geq \ldots \geq \mathbb{E}\, d_k > 0, \tag{2.37a}$$

   where $\mathbb{E}\, d_i = \|\, \mathbf{b}_i \,\|_2 \cdot \mathbb{E} \,\|\, \mathbf{c}_i \,\|_2$.

2. Assume that

$$\lim_{n\to\infty} \frac{\mathbb{E}\, d_1}{\mathbb{E}\, d_k} \nrightarrow \infty, \tag{2.37b}$$

   i.e., that the limit of the ratio is finite.

Theorem 2.2 (Stationary, Ergodic Time Series at Lag τ). *Suppose that the conditions in*

(2.37) hold, in addition to conditions (1, 2, 4, 5) from (2.30).

$$1 \leq \tau \leq n^{\frac{r}{2(r-2)}}, \tag{2.38a}$$

for some value of $r \geq 4$.

Let the $\mathbf{c}_i$ be as described above, and let $\mathbb{E}\, L(\tau)$ be as defined in (2.36). Assume that $\mathbb{E}\,[L_\tau]_{ii} \neq 0$, $\mathbb{E}\,[L_\tau]_{ij} = 0$, and $\mathbb{E}\,\mathbf{s}_i^T\,\mathbf{s}_j = 0$. Then, we have that
a) For some $\epsilon > 0$ and $r \geq 4$, we have that

$$f(n) \leq o\left((\log n)^{2/r}\,(\log\log n)^{(1+\epsilon)2/r}\,n^{-1/2}\right). \tag{2.38b}$$

Then,

$$\left|[L_\tau]_{ij}\right| \leq O(f(n)) \text{ and } \left|\mathbf{s}_i^T\,\mathbf{s}_j\right| \leq O(f(n)) \tag{2.38c}$$

with probability at least

$$1 - O\left(\left[\log n\,(\log\log n)^{1+\epsilon}\right]^{-1}\right). \tag{2.38d}$$

b) Then we have that $|d_i - \mathbb{E}\,d_i| \leq f(n)[1 + o(1)]$ for $i = 1, \ldots, k$, with probability (2.38d).
c) For p_i given by (2.32), we have that

$$\sum_{i=1}^{k} \|\widehat{\mathbf{q}}_i - p_i\,\mathbf{q}_i\|_2^2 \leq O\left(\left[\frac{\mathbb{E}\,d_1}{\mathbb{E}\,d_k}\right]^2 \cdot \frac{k^7}{\delta_L^2} \cdot \left[f^2(n) + \tau n^{-2\alpha}\right]\right), \tag{2.38e}$$

where δ_L is given by

$$\delta_L = \min_{i \neq j} \left|\mathbb{E}\,[L_\tau]_{ii} - [L_\tau]_{jj}\right|, \tag{2.38f}$$

with probability (2.38d).

d) Moreover, for each $\mathbb{E}\,L_{ii}(\tau)$, we have that

$$|\mathbb{E}\,[L_\tau]_{ii} - \lambda_i|^2 \leq O\left(\left[\frac{\mathbb{E}\,d_1}{\mathbb{E}\,d_k}\right]^2 \cdot k^6 \cdot \left[f^2(n) + \tau n^{-2\alpha}\right]\right), \tag{2.38g}$$

with probability (2.38d).

If the $\mathbf{c}_i$ are samples from a stationary, ergodic ARMA process, we may simplify the results of Theorem 2.2 slightly.

Corollary 2.2 (ARMA Processes at Lag τ)**.** *Assume that the $\mathbf{c}_i$ are samples from an*

ARMA process. Then (2.38a) may be replaced with $1 \leq \tau \leq [\log n]^a$*, for some* $a > 0$*, and (2.38b) may be replaced with* $f(n) \leq o\left((\log\log n/n)^{1/2}\right)$.

The iterated logarithmic rate in our error bounds and accompanying probability, are consequences of the classical time series results in [52]. Here, we have stated a result that is similar in spirit to that for SOBI, given in [83]. Our result says that time series $\mathbf{s}_i$ that are uncorrelated at lags 1 and 0 can be unmixed, provided that they are not sparse. The result for SOBI requires uncorrelatedness at all integral lags, and states an asymptotic distributional result; our result relies on looser assumptions, and is a finite sample guarantee. It should be noted that at the expense of using a single lag, our result is slightly weaker than the $1/\sqrt{n}$ convergence described in [83, Theorem 1].

2.5 Estimating the temporal behavior: S

We now establish a recovery condition for deterministic $\mathbf{s}_i$.

Theorem 2.3 (Extending the bounds to S)**.** *Assume that the conditions of Theorem 2.1 hold for a lag* τ *with a bound* $\epsilon_{d,v}^2$ *for the squared estimation error of the* $\mathbf{q}_j$*. Moreover, assume that* $kd_1^2\epsilon_{d,v}^2 < d_k^2$*. Then, given an estimate of the top* k *left eigenvectors of* $\widehat{A}$*, denoted by the rows of the matrix* $\widehat{Q^+}$*, let* $\widehat{S}$ *be formed by normalizing the columns of* $\left(\widehat{Q^+}X\right)^T$*. The columns of* $\widehat{S}$ *are denoted by* $\widehat{\mathbf{s}}_i$*, and let* $p_i = sign\left(\mathbf{s}_i^T\,\widehat{\mathbf{s}}_i\right)$*. Then, we have that*

$$\sum_{i=1}^{k} \|\widehat{\mathbf{s}}_i - p_i\,\mathbf{s}_i\|_2^2 \leq O\left(k\left[\frac{d_1}{d_k}\right]^2 \epsilon_{d,v}^2\right). \tag{2.39}$$

This result translates the results for the mixing matrix Q to the estimation of the signals S. For the practitioner intending to estimate the latent *signals* instead of the mixing matrix, this final result has a greater utility.

2.5.1 Applications of Theorem 2.3: Cosines

As we did for Theorem 2.1, we may restate Theorem 2.3 for the cosine model.

Corollary 2.3 (Cosines)**.** *Assume that the* $\mathbf{c}_i$ *are given by (2.34) and that we apply DMD with* $\tau = 1$*. Then we have that*

$$\sum_{i=1}^{k} \|\widehat{\mathbf{s}}_i - p_i\,\mathbf{s}_i\|_2^2 \leq O\left(\left[\frac{d_1}{d_k}\right]^4 \cdot \frac{k^8}{\delta_L^4} \cdot \frac{1}{n}\right), \tag{2.40}$$

where $\delta_L = \min_{i \neq j} |\cos \omega_i - \cos \omega_j|$.

2.6 Missing Data Analysis

We now consider the randomly missing data setting. We assume that the data is modeled as

$$\widetilde{X} = X \odot M = \left(QDS^T\right) \odot M, \tag{2.41}$$

where M is a masking matrix, whose entries are drawn uniformly at random:

$$M_{i,j} = \begin{cases} 1 & \text{with probability } q, \\ 0 & \text{with probability } 1-q. \end{cases} \tag{2.42}$$

The notation $\odot$ represents the Hadamard or element-wise matrix product. Essentially, we replace unknown entries with zeros, as is done in the compressed sensing literature [25, 104, 88].

2.6.1 The tSVD-DMD algorithm

A natural, and perhaps the simplest, choice to 'fill-in' the missing entries in $\widetilde{X}$ is to use a low-rank approximation, also known as a truncated SVD [32, 38]. That is, given $\widetilde{X}$, we compute the SVD $\widetilde{X} = \widehat{U}\widehat{\Sigma}\widehat{V}^T$, and then the rank-$k$ truncation

$$\widehat{X}_k = \sum_{i=1}^{k} \widehat{\sigma}_i \widehat{\mathbf{u}}_i \widehat{\mathbf{v}}_k^T, \tag{2.43}$$

where the columns of $\widehat{U}$ and $\widehat{V}$ are the $\widehat{\mathbf{u}}_i$ and $\widehat{\mathbf{v}}_i$, respectively, and the $\widehat{\sigma}_i$ are the non-zero entries of $\widehat{\Sigma}$. In what follows, $\mathbf{u}_i$, $\mathbf{v}_i$, and σ_i will denote the singular vectors and values of X. We assume that the number of sources k is known apriori.

After 'filling-in' the missing entries of $\widetilde{X}$ and computing $\widehat{X}_k$, we may apply the τ-DMD algorithm to $\widehat{X}_k$. If $\widehat{X}_k$ has columns $\widehat{X}_k = \begin{bmatrix} \widehat{\mathbf{x}}_1 & \widehat{\mathbf{x}}_2 & \cdots & \widehat{\mathbf{x}}_n \end{bmatrix}$, we may define

$$\widehat{X}^{\tau}_{(0)} = \begin{bmatrix} \widehat{\mathbf{x}}_1 & \widehat{\mathbf{x}}_2 & \cdots & \widehat{\mathbf{x}}_{n-\tau} \end{bmatrix} \text{ and } \widehat{X}^{\tau}_{(1)} = \begin{bmatrix} \widehat{\mathbf{x}}_{1+\tau} & \widehat{\mathbf{x}}_{2+\tau} & \cdots & \widehat{\mathbf{x}}_n \end{bmatrix}. \tag{2.44}$$

We have dropped the k-dependence for clarity. Then, we may define

$$\widetilde{A}_\tau = \widehat{X}^\tau_{(1)} \left(\widehat{X}^\tau_{(0)}\right)^+, \tag{2.45}$$

and take an eigenvalue decomposition:

$$\widetilde{A}_\tau = \widehat{Q}\widehat{\Lambda}\widehat{Q}^+. \tag{2.46}$$

For the sake of naming consistency, we will refer to this procedure as the tSVD-DMD algorithm.

2.6.2 Assumptions

We now provide a DMD recovery performance guarantee. Before stating the result, we require some definitions and further conditions. In addition to the previous assumptions about S, the d_i, the relative values of k, n, p, and τ, and the linear independence of the $\mathbf{q}_i$, we require the following conditions that augment (2.30). For clarity and conciseness in what follows, we define the constant

$$\gamma = \frac{n^{2\alpha}p^{2\beta}}{d_1^2k^2}, \tag{2.47a}$$

and the quantities

$$g(n,p,k,q) = O\Bigg(\sqrt[4]{q(1-q)}d_1k \times \max\left\{n^{1/4-\alpha}p^{1/4-\beta}, n^{-\alpha}, p^{-\beta}\right\}\Bigg), \tag{2.47b}$$

$$\delta_{\sigma,q} = \min_{i=1,2,\ldots,k-1}\left\{q\sigma_k, q^2\sigma_k^2, q^2\sigma_i(\sigma_i-\sigma_{i+1}), q\left(\sigma_i-\sigma_{i+1}\right)\right\}, \tag{2.47c}$$

and

$$\delta_\sigma = \min_{i=1,2,\ldots,k-1}\left\{\sigma_k, \sigma_k^2, \sigma_i(\sigma_i-\sigma_{i+1}), \sigma_i-\sigma_{i+1}\right\}. \tag{2.47d}$$

The quantity $g(n,p,k,q)$ comes from bounding the size of $\left(\widetilde{X} - \mathbb{E}\,\widetilde{X}\right)$, motivated by the approach taken in [88] for handling missing data. The quantities δ_σ and $\delta_{\sigma,q}$ come from applications of the results in [92, Corollary 20, Theorem 23]. The details of how these quantities arise and are used are deferred to the proof of Theorem 2.4, given in Appendix 2.F.

Then, we require:

1. Assume that there is a $\beta > 0$ such that

$$\max_{1 \le i \le p, 1 \le j \le k} |Q_{i,j}| = O\left(p^{-\beta}\right). \tag{2.48a}$$

I.e., the $\mathbf{q}_i$ are not too sparse; this condition is exactly analogous to that for the $\mathbf{s}_i$, where we used the parameter α.

2. Assume that as p and n grow,

$$\frac{1}{\delta_{\sigma,q}}, \frac{q\sigma_1}{\delta_{\sigma,q}}, \frac{1}{\gamma\delta_{\sigma,q}} \nrightarrow \infty. \tag{2.48b}$$

3. Assume that

$$\lim_{p,n\to\infty} d_1 \cdot \max\left\{n^{1/4-\alpha}p^{1/4-\beta}, n^{-\alpha}, p^{-\beta}\right\} = 0, \tag{2.48c}$$

but that

$$\lim_{p,n\to\infty} g(n,p,k,q)^2\gamma \neq 0. \tag{2.48d}$$

Condition (2.48a), along with the analogous condition for the $\mathbf{s}_i$ given in (2.30d), corresponds to the low coherence condition in the matrix completion literature [32, Section 5.2]. I.e., we require that the data matrix is sufficiently dense. Moreover, (2.48c) and (2.48d) imply that the $\mathbf{s}_i$ and $\mathbf{q}_i$ have values of α and β that are at least 1/4 (and less than 1/2, by definition). For example, if we generate a matrix Q by uniformly drawing k vectors from the sphere in $\mathbb{R}^p$ and setting these as the columns, and let S be comprised of cosines as in (2.34), we would anticipate that $\alpha = \beta = 1/2$. In this case, if d_1 is not increasing, we would have that $g(n,p,k,q) = O\left(\sqrt{q}k/\sqrt[4]{pn}\right)$.

Given these assumptions, if we apply the tSVD-DMD algorithm to $\widetilde{X}$, we have the following result for the estimation of the eigenvectors $\mathbf{q}_j$ and eigenvalues λ_i.

2.6.3 Main result

Theorem 2.4 (Missing Data Recovery Guarantee). *Let the assumptions of Theorem 2.2 hold, with a bound $\epsilon_{d,v}^2$ for the squared estimation error of the $\mathbf{q}_i$ and a bound $\epsilon_{d,e}^2$ for the squared error for the individual eigenvalues. Let the conditions in (2.48) hold, let $a > 1$, and let $c_0 > 0$ be some universal constant.*

a) Then, if L_τ is defined in (2.29), δ_L is defined in (2.33b), and p_i is defined in (2.32),

$$\sum_{i=1}^{k} \|\widehat{\mathbf{q}}_i - p_i\,\mathbf{q}_i\|_2^2 \le O\left(\frac{\tau}{q^2} a^2 \left(g(n,p,k,q)\right)^2 \frac{\sigma_1^2}{\delta_\sigma^2}\frac{k^8}{\delta_L^2} + \epsilon_{d,v}^2\right), \tag{2.49}$$

with probability at least

$$1 - O\left(k^2 \cdot 81^k \exp\left(-\left(1-\frac{1}{a}\right)^2 c_0\gamma \frac{\tau\left(g(n,p,k,q)\right)^2}{16}\right)\right) - O\left(k^2\cdot 9^k \exp\left(-c_0\gamma\frac{\delta_{\sigma,q}}{64}\right)\right). \tag{2.50}$$

b) For each $[L_\tau]_{ii}$, we have that

$$|[L_\tau]_{ii} - \lambda_i|^2 \le O\left(\frac{\tau}{q^2} a^2\left(g(n,p,k,q)\right)^2 \frac{\sigma_1^2}{\delta_\sigma^2} k^7 + \epsilon_{d,e}^2\right), \tag{2.51}$$

with probability at least (2.50).

Note that Theorem 2.4 indicates that the dependence of the squared estimation error on q is $O(q^{-3/2})$ for q close to 0. Moreover, for data such that d_1, σ_1, δ_σ and δ_L are not changing with n; Q has dense, linearly independent columns; and such that k and p are fixed, the right-hand sides of (2.49) and (2.51) behave like $O\left(q^{-3/2}n^{1/2-2\alpha}\right)$ with probability at least $1 - O\left(\exp\left(-c_1\sqrt{n}\right)\right) - O\left(\exp\left(-c_2 nq\right)\right)$, for some constants c_1 and c_2. Indeed, if the $\mathbf{c}_i$ are cosines, given by (2.34), we have that $\alpha = 1/2$, so that we have a rate of $O\left(q^{-3/2}n^{-1/2}\right)$.

2.7 Numerical simulations

In this section, we provide numerical verifications of the theorems we have presented. We recall that one of the contributions of this work and the intention of this work is to demonstrate that DMD is a source separation algorithm in disguise. Our goals are not to compete with the state-of-the art in source separation, rather, this work seeks to provide a new analysis and understanding of the DMD algorithm.

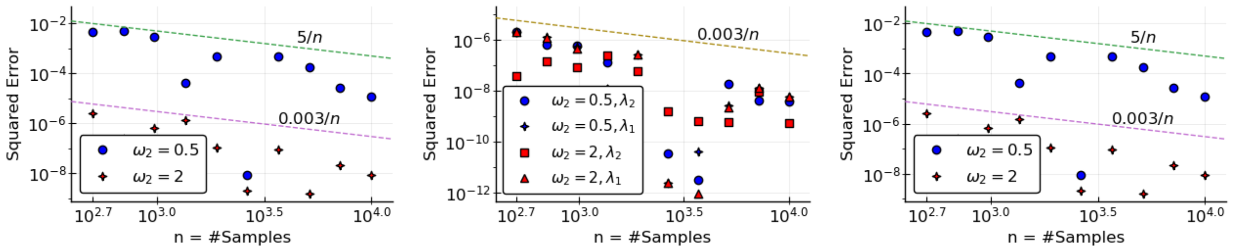
There are two main objects of interest: the error in estimating the eigenvectors $\mathbf{q}_i$, and the error in estimating the eigenvalues λ_i. In the deterministic, fully observed setting, the error in estimating $\mathbf{s}_i$ is also of interest. In what follows, unless otherwise noted, we fix $p = 100$ and $k = 2$, and vary n. We fix the mode magnitudes at $d_1 = d_2 = 1$. We also generate dense, non-orthogonal $\mathbf{q}_i$ by sampling from the sphere in $\mathbb{R}^p$. Equivalently, we sample from the multivariate normal distribution $\mathcal{N}(\mathbf{0}_p, \mathrm{I}_p)$ and normalize the resulting vector to have unit ℓ_2 norm.

We first verify the deterministic error bounds for the cosine model with the DMD algorithm: i.e., Theorem 2.1 and Corollary 2.1, as well as Theorem 2.3 and Corollary 2.3. These verifications are presented in Figure (2.3). We let the columns of C be equal to $\mathbf{c}_{i,t} = \cos(\omega_i t)$. We consider two sets of frequencies: $\omega_1 = 0.25$ and $\omega_2 = 0.5$, as well as $\omega_1 = 0.25$ and $\omega_2 = 2$. We see that as expected, the squared estimation errors for the eigenvalues λ_i, eigenvectors $\mathbf{q}_i$, and the $\mathbf{s}_i$ are bounded by $O(1/n)$. Moreover, the role of δ_L (defined in (2.35b)) is visible, as $\omega_2 = 2$ leads to a lower error relative to $\omega_2 = 0.5$ when estimating the $\mathbf{q}_i$ and $\mathbf{s}_i$. As expected, the non-zero eigenvalues are equal to $\cos\omega_i$.

We also look at the dependence on the rank k in Corollary 2.3. We fix $p = 100$ and $n = 2000$, and vary the rank k; we space the frequencies such that $\cos\omega_i$ is uniform on $[-1, 1]$, and the magnitudes d_i so that $d_1/d_k = 10$. We recall that the predicted rate for the squared estimation errors is $O(k^7)$, and suspect that this may not be tight. In Figure 2.4, we see that the empirical rate is $O(k^5)$. Additionally, there is a 'saturation' as k approaches p, and the growth rate slows down. One important fact to note is that the rate depends on the separation of the autocorrelations, which are bounded in the range $[-1, 1]$; if there are k signals, the minimum spacing between two autocorrelations decays like $O(1/k)$, so that there is yet another implicit factor of k. Importantly, we see that more signals lead to a higher error that grows faster than linear in the number of signals.

We next consider the τ-DMD algorithm, and verify Theorems 2.1 and 2.2, as well as Corollary 2.2. We generate the columns of C as independent, length n realizations of AR(2) processes. That is, $\mathbf{c}_1$ is a realization of an AR(2) process with parameters $(0.2, 0.7)$, and $\mathbf{c}_2$ is also a realization of an AR(2) process with parameters $(0.3, 0.5)$. We compare operating at lags $\tau = 1$ and $\tau = 2$, and average over 200 realizations. Our results appear in Figure (2.5). Note that for a given lag, the non-zero eigenvalues are expected to equal the autocorrelation of the $\mathbf{c}_i$ at that lag; invoking the role of δ_L once again, we observe that the $\mathbf{q}_i$ are better estimated at a lag of $\tau = 2$, as the lag-2 autocorrelations are higher and more separated than the lag-1 values. As expected, the squared estimation errors are bounded by $O(\log\log n/n)$.

Finally, we consider the tSVD-DMD algorithm in the presence of missing data, and verify Theorem 2.4. Here, we fix $p = 500$ and let $d_1 = 2$ and $d_2 = 1$. We let the columns of C be equal to $\mathbf{c}_{i,t} = \cos(\omega_i t)$, for $\omega_1 = 0.25$ and $\omega_2 = 2.0$. Our results are averaged over 200 trials. We consider the effects of varying the entry-wise observation probability q (for $n = 10^4$) in Figure (2.6), and the effects of varying n (for $q = 0.1$) in Figure (2.7). As expected, we see that the squared estimation error decays like $O(1/\sqrt{n})$ for fixed q and like $O(q^{-3/2})$ for fixed n when using the truncated SVD as a preprocessing step. Note that the error of DMD without the SVD is orders of magnitude larger than it is with the SVD, and does not exhibit significant decay with increasing n or q.



(a) The squared estimation error of $\widehat{Q}$ as in (2.35a).

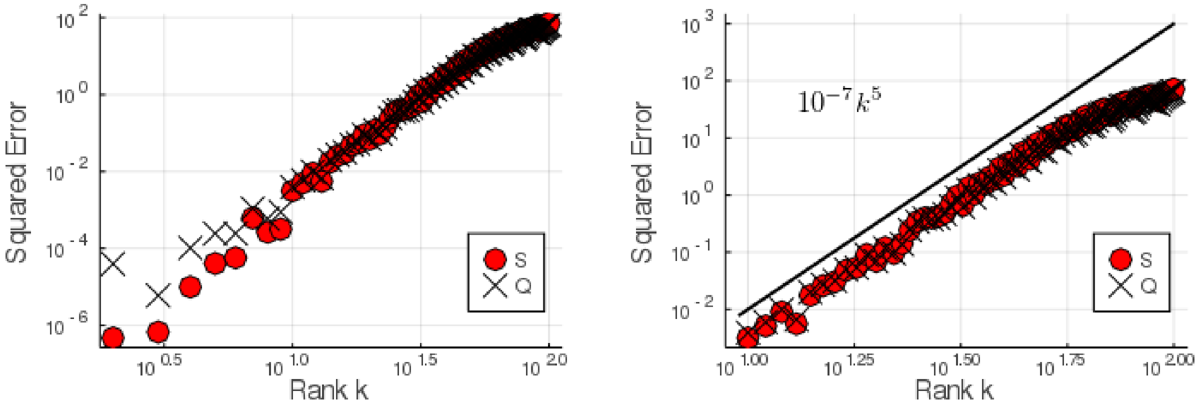
(b) The squared estimation error of the eigenvalues $\widehat{\lambda}_i$ as in (2.28).

(c) The squared estimation error of $\widehat{S}$ as in (2.39).

Figure 2.3: Here, we verify Theorem 2.1 and Corollary 2.1, as well as Theorem 2.3 and Corollary 2.3. We simulate from model (2.11) with a rank 2 cosine signal, first using $\omega_1 = 0.25$ and $\omega_2 = 0.5$, and second using $\omega_2 = 2$. We fix $p = 100$ and use a non-orthogonal Q, and apply DMD with $\tau = 1$. Note that as ω_1 is fixed, $\omega_2 = 2$ leads to a lower error relative to $\omega_2 = 0.5$, due to the greater separation of the frequencies: the error is proportional to $\frac{1}{|\omega_1 - \omega_2|}$. We also plot lines above the samples indicating that the error is bounded by $O(1/n)$.

2.7.1 Comparison with AMUSE/SOBI

We end this section with a comparison of DMD with the AMUSE/SOBI method for source separation [83]. Once again we simulate from model (2.11) with a rank $k = 2$ cosine signal, using $\omega_1 = 0.25$ and $\omega_2 = 2$. We fix $p = 500$, use a Q with non-orthogonal columns, and $d_1 = 2$ and $d_2 = 1$. We use a lag of 1 for the SOBI algorithm (in this case, it is the AMUSE algorithm as we use a single lag). We present these results in Figure 2.8, where we observe that DMD outperforms AMUSE. We note that with some tuning/lag selection, it is possible that SOBI may do better than DMD, but as DMD uses a single lag, SOBI/AMUSE with a single lag is perhaps a fairer comparison. Note that we perform the comparison on a deterministic signal.



(a) The squared estimation error of $\widehat{Q}$ and $\widehat{S}$ as in (2.49).

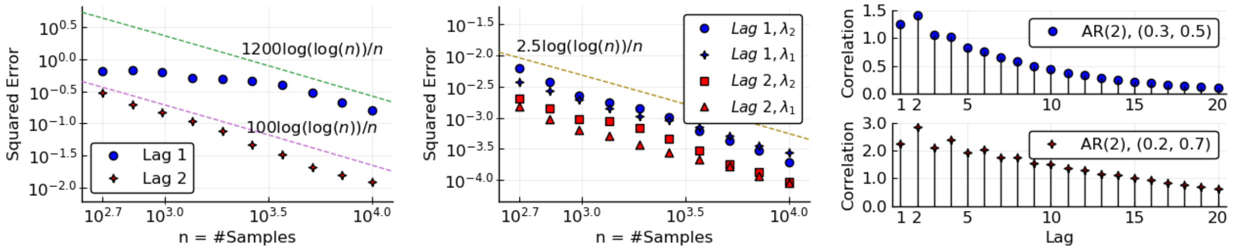
(b) The Figure in (a) zoomed in to $k \geq 10$.

Figure 2.4: Here, we study the dependence on the rank k in Corollary 2.1. We fix $p = 100$ and $n = 2000$, $d_1/d_k = 10$, space the frequencies such that $\cos \omega_i$ is uniform on $[-1, 1]$. We suspect that the factor of k^7 is not tight, and see that empirically, k^5 is a tight rate. We plot lines above the samples indicating the k^5 rate for the squared estimation errors for both $\widehat{Q}$ and $\widehat{S}$.

The theoretical results for SOBI and AMUSE are asymptotic consistency statements, i.e., in the large sample limit, if the latent signals are statistically independent, we may consistently (in a statistical sense) recover them [83, 121]. Other Independent Component Analysis (ICA) methods for this problem have similar statements [27]. It is important to note that here, we have a much weaker assumption (uncorrelatedness at two lags as opposed to independence) and that our results are finite sample bounds.

2.8 Dynamic Mode Factorization of a Time Series Data Matrix

We present the Dynamic Mode Factorization (DMF) algorithm for real data in Algorithm 1. We take the data matrix X and a lag τ as inputs, and return a factorization of X. Our goal is to write $X = QC^T$, where the columns of Q have unit norm. If the matrix has missing entries then we fill in the missing entries with zeroes and then compute the rank k (assumed known) truncated SVD approximation of the matrix as suggested by the analysis



(a) The squared estimation error of $\widehat{Q}$ as in (2.38e).

(b) The squared estimation error of the eigenvalues $\widehat{\lambda}_i$ as in (2.38g).

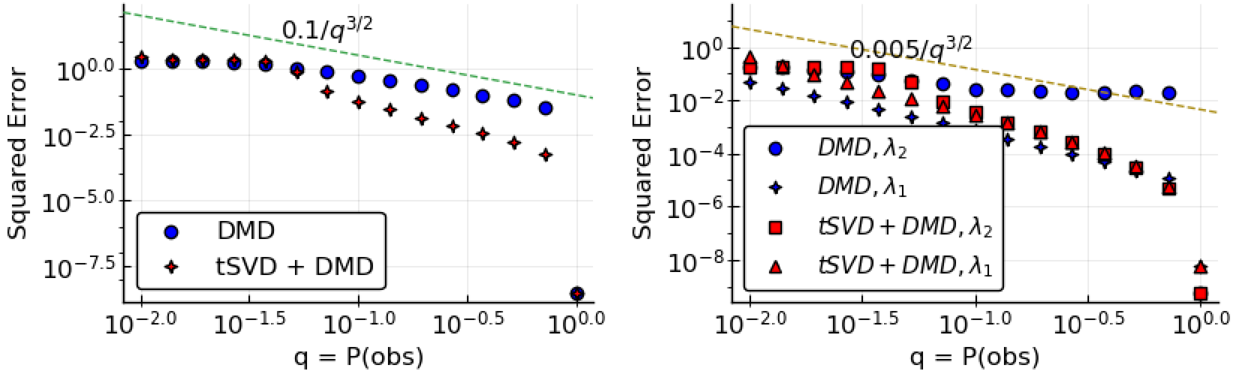
(c) The autocorrelation function of the processes in C. Note that the autocorrelation at lag-2 is higher than that at lag-1 for both signals.

Figure 2.5: Here, we verify Theorems 2.1 and 2.2, as well as Corollary 2.2. We simulate from model (2.11) with a rank 2 signal. The signals in C are drawn as realizations from AR(2) processes, the first with parameters $[0.3, 0.5]$ and the second with parameters $[0.2, 0.7]$. We fix $p = 100$ and use a non-orthogonal Q. The lag-2 DMD algorithm leads to a lower eigenvector loss, as expected, since the autocorrelations at lag-2 are higher than that at lag-1 for both signals, and the difference is also higher at a lag of 2 than at a lag of 1. We also plot lines above the samples indicating that the error is bounded by $O(\log\log n/n)$.

in Section 2.6. We assume henceforth that we are working with this filled-in matrix. If the data matrix has zero mean columns, then we estimate the column-wise mean of X and subtract it to form $\overline{X}$:

$$\widehat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \text{ so that } \overline{X} = X - \widehat{\boldsymbol{\mu}}\,\mathbf{1}_n^T\,. \tag{2.52}$$

Next, we define $\overline{X}^{\tau}_{(0)}$ and $\overline{X}^{\tau}_{(1)}$ analogously to (2.25), and form $\widehat{A}_\tau = \overline{X}^{\tau}_{(1)}\left[\overline{X}^{\tau}_{(0)}\right]^+$. The eigenvectors of $\widehat{A}_\tau$ are the columns of $\widehat{Q}$, so that $\widehat{C}^T = \widehat{Q}^{-1}\widehat{\boldsymbol{\mu}}\,\mathbf{1}_n^T + \widehat{Q}^{-1}\overline{X}$. Note that for a real dataset, we care about C rather than S: the scale of our data matters, as does the mean.



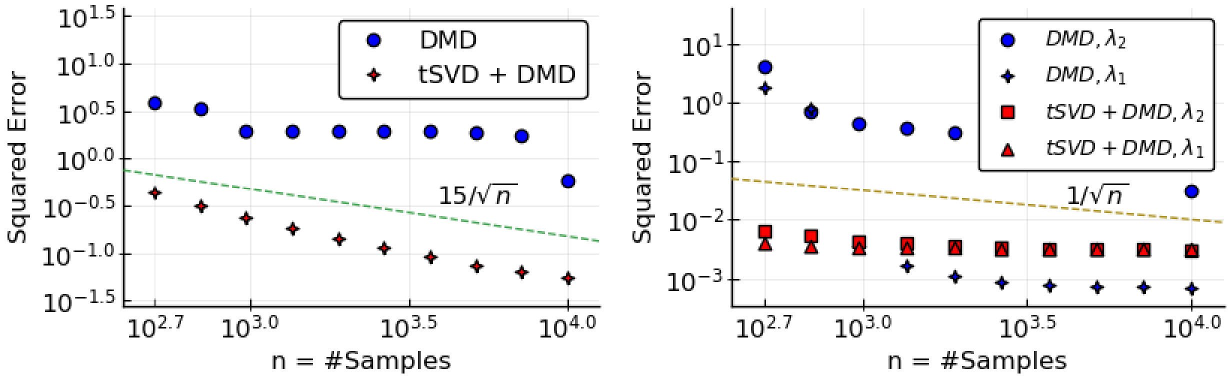
(a) The squared estimation error of $\widehat{Q}$ as in (2.49).

(b) The squared estimation error of the eigenvalues $\widehat{\lambda}_i$ as in (2.51).

Figure 2.6: Here, we verify Theorem 2.4. We fix the sample size $n = 10^4$, and vary the observation probability. We simulate from model (2.11) with a rank 2 cosine signal, using $\omega_1 = 0.25$ and $\omega_2 = 2$. We fix $p = 500$ and use a non-orthogonal Q. We fix $d_1 = 2$ and $d_2 = 1$. We plot the error for the rank-2 truncated SVD (tSVD) followed by DMD, and for just DMD (both with a lag of 1). The results show that the truncated SVD offers a tangible benefit over vanilla DMD. We also plot lines above the samples indicating that the error from the rank-2 tSVD + DMD algorithm is bounded by $O(1/q^{3/2})$.

2.8.1 Application: Source Separation

Next we illustrate that Algorithm 1 can unmix mixed audio signals. The first signal contains the sound of a police siren, and the second contains a music segment. The two signals have $n = 50000$ samples taken at 8 kHz, for a duration of 6.25 seconds each. We de-mean and scale the signals to the range $[-1, 1]$, and form an $n \times 2$ matrix C with these scaled signals as columns. We mix the signals with $Q = \frac{1}{\sqrt{5}} \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}$, and generate a $2 \times n$ data matrix $X = \widehat{Q} C^T$ of the mixed signals, as in (2.17). Note that the Q matrix does not have orthogonal columns. Figures (2.9-e) and (f) show the estimates $\widehat{C} = (Q^{+} X)^T$ produced by the DMF algorithm with a lag of $\tau = 1$, when X is the input as in Figures (2.9-c) and (d). Employing PCA on X does not work well here because the mixing matrix Q is not orthogonal. Figures (2.9-g) and (h) show that PCA fails where the DMD algorithm succeeds. For completeness, in Figures (2.9-i) and (j) we also display the results from using kurtosis-based ICA to unmix the signals. We observe that ICA performs well, but not as



(a) The squared estimation error of $\widehat{Q}$ as in (2.49).

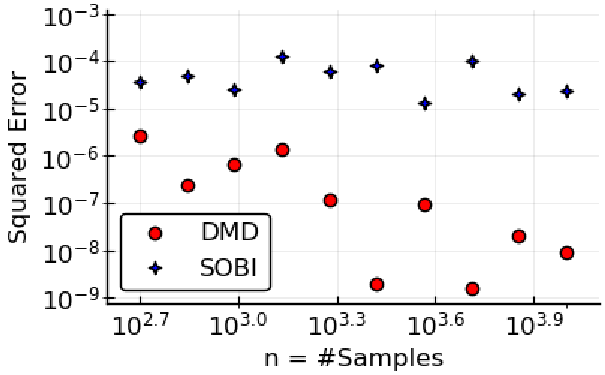
(b) The squared estimation error of the eigenvalues $\widehat{\lambda}_i$ as in (2.51).

Figure 2.7: Here, we verify Theorem 2.4. We fix the observation probability $q = 0.1$, and vary the sample size n. We simulate from model (2.11) with a rank 2 cosine signal, using $\omega_1 = 0.25$ and $\omega_2 = 2$. We fix $p = 500$ and use a non-orthogonal Q. We fix $d_1 = 2$ and $d_2 = 1$. We plot the error for the rank-2 truncated SVD (tSVD) followed by DMD, and for just DMD (both with a lag of 1). The results show that the truncated SVD offers a tangible benefit over vanilla DMD. We also plot lines above the samples indicating that the error from the rank-2 tSVD + DMD algorithm is bounded by $O(1/\sqrt{n})$.

well as DMF (or as quickly).

2.8.2 Application: Changepoint Detection

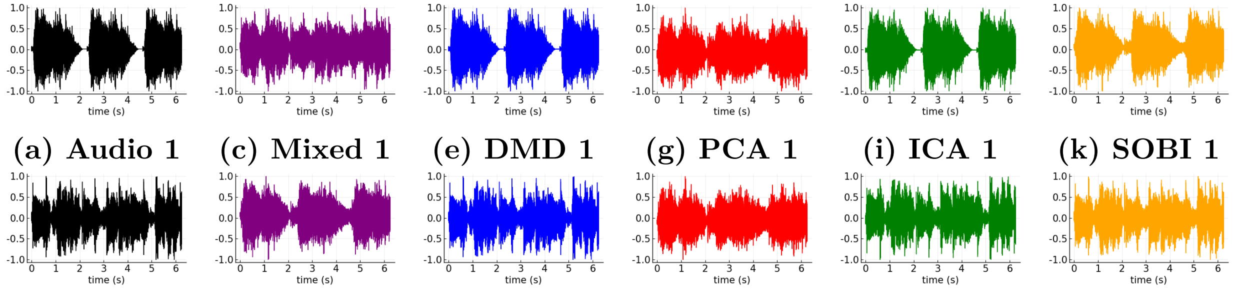
Often, real time series contain one or more changepoints. That is, there are points in time at which the distribution or characteristics of the signal changes. In the context that we are working in, perhaps the data may exhibit a transition between modes; we consider such an example in Figure 2.10. In this setting, we fix $p = 4$, $k = 4$, and use $Q = \frac{1}{\sqrt{5}} \begin{bmatrix} 1 & 0 & 0 & 2 \\ 2 & 1 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 2 & 1 \end{bmatrix}$. We fix $n = 1000$, and generate C as follows. The first 500 samples of $\mathbf{c}_1$ are a realization of an AR(2) process with parameters $(0.2, 0.7)$, and the remaining 500 samples are identically zero. The first 500 samples of $\mathbf{c}_2$ are identically zero, and the remaining 500 are a realization of an AR(2) process with parameters $(0.3, 0.5)$. The first



(a) The squared estimation error of $\widehat{Q}$ as in (2.35a).

(b) The squared estimation error of $\widehat{S}$ as in (2.39).

Figure 2.8: Here, we present results for DMD and AMUSE/SOBI. We simulate from model (2.11) with a rank 2 cosine signal, using $\omega_1 = 0.25$ and $\omega_2 = 2$. We fix $p = 500$ and use a Q with non-orthogonal columns. We fix $d_1 = 2$ and $d_1 = 1$. We plot the estimation error of $\widehat{Q}$ and $\widehat{S}$ and compare the performance of DMD with AMUSE/SOBI for a lag of 1. With a lag of 1, DMD outperforms AMUSE/SOBI.



(a) Audio 1 **(c) Mixed 1** **(e) DMD 1** **(g) PCA 1** **(i) ICA 1** **(k) SOBI 1**

(b) Audio 2 **(d) Mixed 2** **(f) DMD 2** **(h) PCA 2** **(j) ICA 2** **(l) SOBI 2**

Figure 2.9: We mix two audio signals (a police siren and a music segment), and observe that DMD successfully unmixes the signals. The squared estimation error for the unmixed signals is 2.978×10^{-5}. However, we observe that the SVD cannot unmix the signals: the squared estimation errors for the unmixed signals is 1.000. We also display the results of ICA, which has a squared estimation errors for the unmixed signals of 0.0015, and SOBI, which has an error of 0.00125.

500 samples of $\mathbf{c}_3$ are generated as $\cos 2t$, and the remaining 500 are identically zero. The first 500 samples of $\mathbf{c}_4$ are identically zero, and the remaining 500 are generated as $\cos t/2$.

We hope that our algorithm estimates Q and S with low error, and that our estimated S correctly captures the changepoints. That is, we hope to *visually* be able to pick out

Algorithm 1 Dynamic Mode Factorization

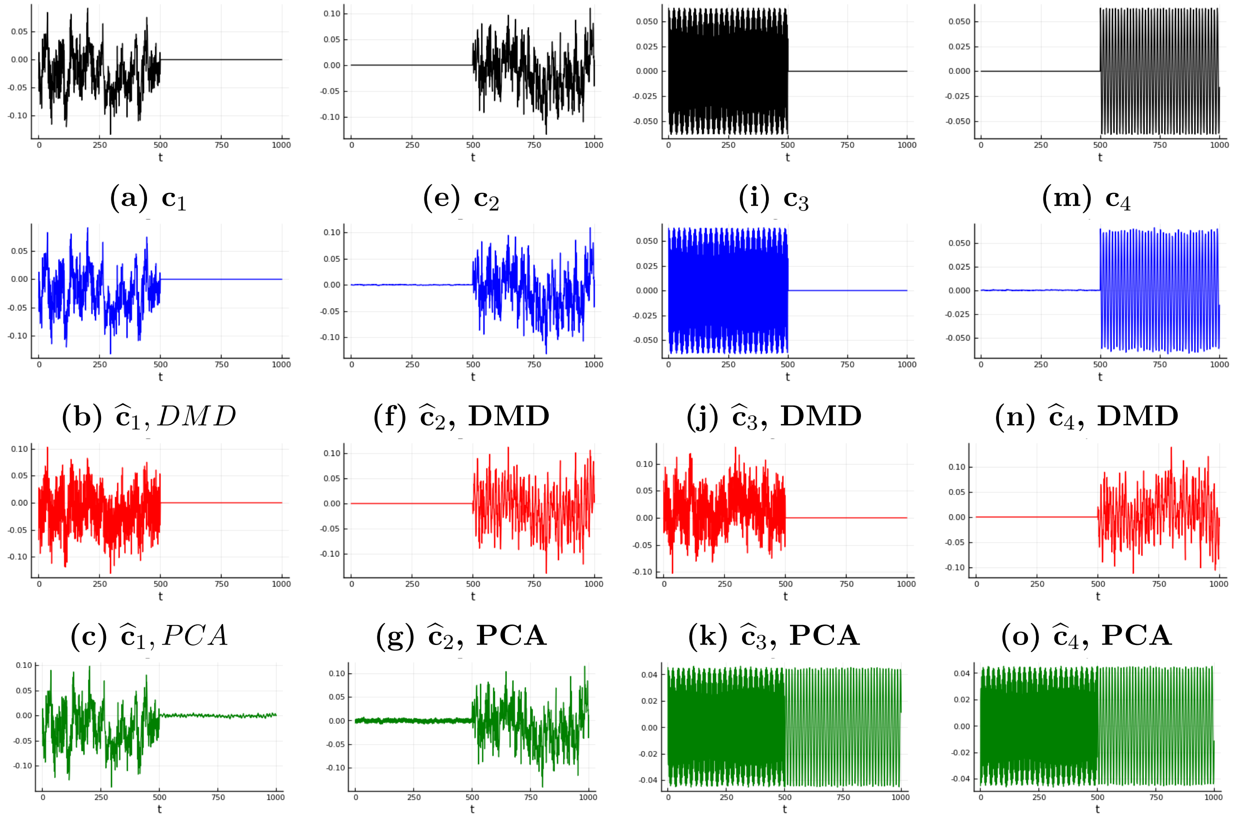
```
Input: Data $X = \begin{bmatrix}\mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_n\end{bmatrix}$, Integer lag $0 < \tau < n$.
Goal: $X = \widehat{Q}\widehat{C}^T$.
  Compute $\widehat{\boldsymbol{\mu}}$ and $\overline{X} = \begin{bmatrix}\bar{\mathbf{x}}_1 & \bar{\mathbf{x}}_2 & \dots & \bar{\mathbf{x}}_n\end{bmatrix}$ as in (2.52).
  Form $\overline{X}^{\tau}_{(0)} = \begin{bmatrix}\bar{\mathbf{x}}_1 & \bar{\mathbf{x}}_2 & \dots & \bar{\mathbf{x}}_{n-\tau}\end{bmatrix}$ and $\overline{X}^{\tau}_{(1)} = \begin{bmatrix}\bar{\mathbf{x}}_{1+\tau} & \bar{\mathbf{x}}_{2+\tau} & \dots & \bar{\mathbf{x}}_n\end{bmatrix}$.
  Compute $\widehat{A}_{\tau} = \overline{X}^{\tau}_{(1)} \left[\overline{X}^{\tau}_{(0)}\right]^{+}$.
  Compute $\widehat{A}_{\tau} = \widehat{Q}\widehat{\Lambda}\widehat{Q}^{-1}$ with eigenvalues sorted by decreasing order of magnitude.
  Compute $\widetilde{C}^T = \widehat{Q}^{-1}\overline{X}$.
  Compute $\widehat{C}^T = \widehat{Q}^{-1}\widehat{\boldsymbol{\mu}}\,\mathbf{1}_n^T + \widetilde{C}^T$.
Return: $\widehat{Q}$, $\widehat{C}$.
```

when a changepoint occurs. Indeed, we find that the squared error for both Q is approximately 0.069 and that for S is 0.035, and that the estimated signals are correctly identified. Moreover, the changepoints are clearly visible. Note that PCA fails to pick out the individual signals, while preserving the changepoints; this is expected behavior, due to the non-orthogonality of the mixing. Kurtosis-based ICA also fails, as the two AR processes have Gaussian marginals.

2.9 Conclusions

Our analysis has revealed that DMD unmixes deterministic signals and stationary, ergodic time series that are uncorrelated at a lag of 1 time-step. We have analyzed the unmixing performance of DMD in the finite sample setting with and without randomly missing data, and have introduced and analyzed a natural higher-lag extension of DMD. We have provided numerical simulations to verify our theoretical results. We have shown (empirically) how the higher lag DMD can outperform conventional (lag-1) DMD for time series for which there is a higher autocorrelation at higher lags than at lag 1: this is a natural extension of DMD that practitioners should adopt and experiment with. Moreover, we showed how DMD (like ICA-family methods) can successfully solve the cocktail party problem. Our results reveal why DMD will succeed in unmixing Gaussian time series while kurtois-based ICA fails, and also why applying DMD to a multivariate mixture of Fourier series type data, like in the eigen-walker model, can better reveal non-orthogonal mixing matrices in a way that PCA fundamentally cannot.

There many directions for extending this research. Analyzing and improving the perfor-



(a) $\mathbf{c}_1$

(b) $\widehat{\mathbf{c}}_1, DMD$

(c) $\widehat{\mathbf{c}}_1, PCA$

(d) $\widehat{\mathbf{c}}_1, ICA$

(e) $\mathbf{c}_2$

(f) $\widehat{\mathbf{c}}_2$**, DMD**

(g) $\widehat{\mathbf{c}}_2$**, PCA**

(h) $\widehat{\mathbf{c}}_2$**, ICA**

(i) $\mathbf{c}_3$

(j) $\widehat{\mathbf{c}}_3$**, DMD**

(k) $\widehat{\mathbf{c}}_3$**, PCA**

(l) $\widehat{\mathbf{c}}_3$**, ICA**

(m) $\mathbf{c}_4$

(n) $\widehat{\mathbf{c}}_4$**, DMD**

(o) $\widehat{\mathbf{c}}_4$**, PCA**

(p) $\widehat{\mathbf{c}}_4$**, ICA**

Figure 2.10: We generate $k = 4$ signals of length $n = 1000$, and mix them. Each signal has a changepoint, in that it switches from all zeros to a definite, non-zero signal. We find that the DMF algorithm perfectly captures the underlying signals, in addition to estimating Q and S (squared errors of 0.0098 and 0.0096, respectively) very well. We plot the estimated $\mathbf{c}_i$ beside the true signals, and observe perfect overlap. As a comparison, we plot the results from using PCA and ICA below those from DMD. We observe that PCA fails dramatically, due to the non-orthogonality of the mixing, and that ICA does as well, due to the Gaussianity of the marginal distributions of the AR(2) processes.

mance of DMD and the tSVD-DMD algorithm and comparing it to that of SOBI in the noisy, finite sample setting is a natural next step. We have taken some preliminary steps in this direction in [100], where we have given performance bounds for the tSVD-DMD algorithm. Additionally, selecting a lag at which to perform DMD is an open problem. Note that the performance of SOBI is known to be sensitive to the choice of the lag parameter [116], and that in Figure 2.5, we presented an example of a mixed time series for which τ-DMD with $\tau = 2$ outperforms conventional ($\tau = 1$) DMD. One might recast the lag selection problem into a problem of optimal weight selection for a weighted multi-lag DMD setup where we consider the eigenvectors of the matrix $\widehat{A}_{\text{agg}} = \sum_{i=1}^{l} w_i \widehat{A}_{\tau_i}$, where $\widehat{A}_{\tau_i}$ is the matrix in (2.27) and we optimize for the weights w_i which yield the best estimate for the mixing matrix Q in (2.16). There are intriguing connections between this formulation and spectral density estimation in time series analysis [93] and multi-taper spectral estimation [7, 49, 5] that suggest ways of improving the performance of DMD, and also SOBI (as the work in [120] does), in the presence of finite, noisy data in a manner that makes it robust to the lag selection misspecification.

Finally, non-linear extensions of this work, particularly in the design and analysis of provably convergent DMD-based unmixing on non-linearly mixed ergodic time series are of interest and would complement related works on non-linear ICA [3, 39, 57, 78, 55, 22, 58, 46, 4, 133] and non-linear DMD [130, 124].

Acknowledgements

We thank Amit Surana and J. Nathan Kutz for introducing us to and intriguing us with their research applying the DMD algorithm during their respective seminars at the University of Michigan. We thank MIDAS and the MICDE seminar organizers for inviting them and especially thank J. Nathan Kutz for his thought-provoking statement that “DMD is Fourier meets the Eigenvalue Decomposition”. This thought-provoking comment seeded our inquiry and led to the formulation in Section 2.4.3, from which the rest of our results flowered. We thank Hao Wu and Asad Lodhia for their detailed comments and suggestions on earlier versions of this manuscript, and particularly Florica Constantine for her suggestions and numerous edits. We thank Harish Ganesh for his perspectives and thoughts on the use of DMD in its original domain of experimental fluid mechanics. We thank Alfred Hero and Jeff Fessler for feedback and suggestions, and Shai Revzen for his thought-provoking comments and insights about when DMD does and does not work in real-world settings–they have provided us with fodder for many more questions than this work answers. The Julia package `https://github.com/aprasadan/DMF.jl` contains code to reproduce all the simulations herein.

2.A Proof of Theorem 2.1 for $\tau = 1$

Recall the definitions of $X_{(0)}$ and $X_{(1)}$ from (2.21). Noting that $X = QDS^T$, we may define $S_{(0)}$ and $S_{(1)}$, where

$$S_{(0)} = \begin{bmatrix} s_{1,1} & s_{2,1} & \cdots & s_{k,1} \\ s_{1,2} & s_{2,2} & \cdots & s_{k,2} \\ \vdots & \vdots & \cdots & \vdots \\ s_{1,n-1} & s_{2,n-1} & \cdots & s_{k,n-1} \end{bmatrix} \text{ and } S_{(1)} = \begin{bmatrix} s_{1,2} & s_{2,2} & \cdots & s_{k,2} \\ s_{1,3} & s_{2,3} & \cdots & s_{k,3} \\ \vdots & \vdots & \cdots & \vdots \\ s_{1,n} & s_{2,n} & \cdots & s_{k,n} \end{bmatrix}. \tag{2.53}$$

Then, we have that

$$X_{(0)} = QDS_{(0)}^T \text{ and } X_{(1)} = QDS_{(1)}^T. \tag{2.54}$$

We make the key observation that

$$S_{(1)}^T = \begin{bmatrix} \mathbf{s}_{1,2} & \mathbf{s}_{1,3} & \cdots & \mathbf{s}_{1,n-1} & \mathbf{s}_{1,1} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \mathbf{s}_{k,2} & \mathbf{s}_{k,3} & \cdots & \mathbf{s}_{k,n-1} & \mathbf{s}_{k,1} \end{bmatrix} + \begin{bmatrix} 0 & \cdots & 0 & \mathbf{s}_{1,n} - \mathbf{s}_{1,1} \\ \vdots & \cdots & \vdots & \vdots \\ 0 & \cdots & 0 & \mathbf{s}_{k,n} - \mathbf{s}_{k,1} \end{bmatrix}. \tag{2.55}$$

Let P be the $(n-1) \times (n-1)$ lag-1 circular shift matrix as described in the construction of the lag-1 inner-product matrix $L = L_1$ in (2.29). A comparison of the first term on the right-hand side in the decomposition of $S_{(1)}^T$ in (2.55) with the column partition decomposition of $S_{(0)}^T$ in (2.53) reveals that this first term is a lag-1 circular shift of the matrix $S_{(0)}^T$. Consequently, we may express $S_{(1)}^T$ as

$$S_{(1)}^T = S_{(0)}^T P + \Delta_1, \tag{2.56}$$

where $S_{(0)}^T P$ is the lag-1 circular shift of $S_{(0)}^T$ and Δ_1 is the rank 1 error matrix given by the second term in the right-hand side of (2.55). Thus, from (2.54) we have that

$$X_{(1)} = QD(S_{(0)}^T P + \Delta_1) = QDS_{(0)}^T P + \Delta_X, \tag{2.57}$$

where $\Delta_X = QD\Delta_1$. Consequently, by substituting the expression of $X_{(1)}$ from (2.57) and $X_{(0)}$ from (2.54), we can express $\widehat{A}$ as

$$\widehat{A} = X_{(1)} X_{(0)}^+ = QL_D Q^+ + \widehat{\Delta}_X \tag{2.58}$$

where

$$\widehat{\Delta}_X = \Delta_X \left(S_{(0)}^T\right)^+ D^+ Q^+ \text{ and } L_D = D S_{(0)}^T P \left(S_{(0)}^T\right)^+ D^+. \tag{2.59}$$

Let diag$(\cdot)$ denote the diagonal matrix determined by the main diagonal of its argument. Then, the matrix L_D can be decomposed as

$$L_D = \underbrace{\text{diag}(L_D)}_{=:\Lambda} + \Delta_L. \tag{2.60}$$

Substituting the expression of L_D in (2.60) into the first term on the right hand side of (2.58) gives us the expression

$$\widehat{A} = Q\Lambda Q^+ + \widehat{\Delta}_A, \text{ where } \widehat{\Delta}_A = Q\Delta_L Q^+ \widehat{\Delta}_X. \tag{2.61}$$

The essence of our proof lies in bounding the size of $\widehat{\Delta}_A$. To this end, we first unpack $\widehat{\Delta}_A$. A key observation, to be substantiated in what follows, is that we may write $S_0^+ = S_0^T + \Delta_{Sp}$, where $\|\Delta_{Sp}\|_2$ is small (to be quantified in what follows). When we substitute this quantity into the definition of $\widehat{\Delta}_X$ in (2.59) and expand the terms in $\widehat{\Delta}_A$, we obtain:

$$\widehat{\Delta}_A = QD\Delta_L D^{-1} Q^+ + QDS_0^T P_1 \Delta_{Sp}^T D^{-1} Q^+ + QD\Delta_1 S_0 D^{-1} Q^+ + QD\Delta_1 \Delta_{Sp}^T D^{-1} Q^+. \tag{2.62}$$

It is now relatively straightforward to bound the size of $\widehat{\Delta}_A$: we bound each term individually by bounding the factors therein. The most involved part of this argument comes from bounding the size of Δ_{Sp}, as we will do next. Then, we will state a bound on the size of $\widehat{\Delta}_A$. Given the bound on $\widehat{\Delta}_A$, we will appeal to results from perturbation theory to bound the deviation of the eigenvectors $\widehat{\mathbf{q}}_i$ of $\widehat{A}$ from $\mathbf{q}_i$.

2.A.1 Bounding Δ_{Sp}

We now bound the size of Δ_{Sp}. We proceed in three steps, separated into lemmas. Through our lemmas, we characterize the singular vectors and values of S_0, so that we may understand the pseudoinverse S_0^+.

Lemma 2.1 (The right singular vectors of S_0)**.** *The right singular vectors of S_0 are, up to a bounded perturbation, the columns of the $k \times k$ identity matrix, I_k, with the j^{th} column denoted by $\mathbf{e}_{j,k}$.*

Proof. $S_0^T S_0$ is a $k \times k$ matrix with diagonal entries between $1 - O(n^{-\alpha})$ and 1, and off-diagonal entries bounded in size by $O(f(n))$ (recall (2.30d)). I.e., $S_0^T S_0 = \mathrm{I}_k + \Delta_V$, $\|\Delta_V\|_F^2 = O\left(k^2 f(n)^2 + kn^{-2\alpha}\right)$. Then, the eigenvectors of $S_0^T S_0$ are the columns of the identity matrix, up to a perturbation Δ_V: $\mathrm{I}_k + \Delta_V$.

To see that $\mathbf{e}_{j,k}$ is almost an eigenvector of $S_0^H S_0$: $\left\|\mathbf{e}_{j,k} - S_0^T S_0 e_{j,k}\right\|_2^2 = O\left(kf(n)^2 + n^{-2\alpha}\right)$. Hence, $\|\Delta_V\|_F^2 = O\left(k^2 f(n)^2 + kn^{-2\alpha}\right)$. □

Before considering the left singular vectors and singular values, we need the following fact.

Lemma 2.2. *For $a > 0$ and $a \neq 1$, there exists a constant $b(a)$ such that $\frac{1}{1-a} \leq 1 + b(a) \times a$. Choosing $b(a) \geq \frac{1}{1-a}$ is sufficient.*

Lemma 2.3 (The left singular vectors and the singular values of S_0)**.** *The left singular vectors of S_0 are approximately the columns of S_0, and the non-zero singular values are approximately* 1*.*

Proof. The left singular vectors of S_0 are found by normalizing the columns of S_0 times the right singular vectors. I.e., $S_0\left[\mathrm{I} + \Delta_V\right]$, but normalized. The size of $S_0 \Delta_V$ can be bounded by $\|S_0 \Delta_V\|_F^2 = O\left(k^3 f(n)^2 + k^2 n^{-2\alpha}\right)$, since $\|S_0\|_F^2 \leq \|S\|_F^2 = k$. Moreover, the norms of individual columns are bounded above by 1 and below by

$$\sqrt{1 - O(kf(n)^2 + n^{-2\alpha})} \geq 1 - O\left(k^{1/2} f(n) + n^{-\alpha}\right).$$

Using Lemma (2.2) and assuming that $O(k^{1/2} f(n) + n^{-\alpha})$ is bounded away from 1, e.g., by 9/10, a normalized column of $S_0 + S_0 \Delta_V$ has norm $1 + O(k^{1/2} f(n) + n^{-\alpha})$. Then, writing the normalization as multiplication by a diagonal matrix, we have $(S_0 + S_0 \Delta_V)(\mathrm{I} + \Delta_N) = S_0 + S_0 \Delta_V + S_0 \Delta_V \Delta_N$. The norm of Δ_N is bounded by $\|\Delta_N\|_F^2 = O(k^2 f(n)^2 + kn^{-2\alpha})$. Then, the norm of S_0 minus the error terms is:

$$\|S_0 - S_0 \Delta_V - S_0 \Delta_V \Delta_N\|_F^2 = O\left(k^3 f(n)^2 + k^2 n^{-2\alpha}\right).$$

□

Now, we may combine the previous results to bound Δ_{Sp}.

Lemma 2.4 (The Pseudoinverse of S_0)**.** *The pseudoinverse of S_0 is $S_0^+ = S_0^T + \Delta_{Sp}$, where $\|\Delta_{Sp}\|_F$ is small.*

Proof. Writing the SVD of S_0 as $(S_0 + \Delta_U)(\mathrm{I} + \Delta_N)(\mathrm{I} + \Delta_V)^T$, applying Lemma 2.2 to the individual elements of $\mathrm{I} + \Delta_N$ and noting that $\|\Delta'_N\|_F = \Theta(\|\Delta_N\|_F)$ yields that the pseudoinverse is $(\mathrm{I} + \Delta_V)(\mathrm{I} + \Delta'_N)(S_0 + \Delta_U)^T$. Once again assuming that $f(n) \to 0$ and noting that $f(n) \leq 1$,

$$\|\Delta_{Sp}\|_F^2 = O\left(k^3 f(n)^2 + k^2 n^{-2\alpha}\right). \tag{2.63}$$

□

2.A.2 Bounding the size of $\widehat{\Delta}_A$

Now that we have computed the pseudoinverse of S_0, we may return to the main computation. Recall that we wrote

$$\widehat{\Delta}_A = QD\Delta_L D^{-1}Q^+ + QDS_0^T P_1 \Delta_{Sp}^T D^{-1}Q^+ + QD\Delta_1 S_0 D^{-1}Q^+ + QD\Delta_1\Delta_{Sp}^T D^{-1}Q^+. \tag{2.64}$$

First, note that each factor of Q and $Q^\dagger$ adds a factor of k to the squared Frobenius norm. The pre- and post-multiplication by D and D^{-1} respectively adds a factor of $(d_1/d_k)^2$. By assumption, $L = \left[S_0^T P_1 S_0\right]$ is a $k \times k$ matrix with diagonal entries that are $\Theta(1)$ and off-diagonal entries that are bounded as $O(f(n))$, so that $\|\Delta_L\|_F^2 \leq O(kf^2(n))$. Once again by assumption,

$$\|\Delta_1\|_F^2 \leq O(kn^{-2\alpha}), \tag{2.65}$$

and S_0 and $S_0^H P_1$ each contribute factors of k to the squared Frobenius norm. Then, we have

$$\|\widehat{\Delta}_A\|_F^2 = O\left((d_1/d_k)^2 k^6 \times [f(n)^2 + n^{-2\alpha}]\right). \tag{2.66}$$

2.A.3 Eigenvectors and Eigenvalues

We have written $\widehat{A}$ as $Q\Lambda Q^\dagger + \widehat{\Delta}_A$, and we know the size of $\widehat{\Delta}_A$. The next step is to compute the eigenvectors of $\widehat{A}$. Ideally, these are the columns of Q, notated by $\mathbf{q}_j$ and estimated by $\widehat{\mathbf{q}}_j$, which are stacked into $\widehat{Q}$.

There are two basic propositions from the perturbation theory of eigenvalues and eigenvectors that we need to complete our analysis. First, we have the following proposition bounding the error in the eigenvalues as a consequence of [33, Theorem 4.4]:

Proposition 2.1. *Let λ_i be a simple eigenvalue of $A = Q\Lambda Q^+$, where the columns of Q,* *denoted by* $\mathbf{q}_i$*, are unit-norm, fixed, and linearly independent. Then, there is a eigenvalue* $\widehat{\lambda}_i$ *of the perturbed matrix* $\widehat{A} = A + \widehat{\Delta}_A$ *such that* $\left|\lambda_i - \widehat{\lambda}_j\right|^2 \leq O\left(\left\|\widehat{\Delta}_A\right\|_2^2\right)$.

Proof. From [33, Theorem 4.4], we have that

$$\widehat{\lambda}_i = \lambda_i + \frac{\mathbf{y}_i^H \widehat{\Delta}_A \mathbf{q}_i}{\mathbf{y}_i^H \mathbf{q}_i} + O\left(\left\|\widehat{\Delta}_A\right\|_2^2\right),$$

where $\mathbf{q}_i$ is the corresponding unit-norm right eigenvector to λ_i, and $\mathbf{y}_i$ is the corresponding unit-norm left eigenvector. Hence,

$$\left|\widehat{\lambda}_i - \lambda_i\right| = O\left(\frac{\mathbf{y}_i^H \widehat{\Delta}_A \mathbf{q}_i}{\mathbf{y}_i^H \mathbf{q}_i}\right).$$

Noting that λ_i is simple and that the $\mathbf{q}_i$ are linearly independent, we have that $\mathbf{y}_i^H \mathbf{q}_i$ is fixed and non-zero (see [129, Chapter 2] for a discussion of this quantity), and we obtain the desired result. □

Then, we have the following proposition as a consequence of [79, Theorem 2]:

Proposition 2.2. *Let* λ_i *be a simple eigenvalue of* $A = Q\Lambda Q^+$ *where the columns of* Q*, denoted by* $\mathbf{q}_i$*, are unit-norm, fixed, and linearly independent. Let* $\mathbf{q}_i$ *be the corresponding unit-norm right eigenvector* $\mathbf{q}_i$ *to* λ_i*, and* $\widehat{\mathbf{q}}_i$ *is the estimated eigenvector from* $\widehat{A} = A + \widehat{\Delta}_A$*. Then, we have that*

$$\|\mathbf{q}_i - p_i \widehat{\mathbf{q}}_i\|_2^2 \leq O\left(\frac{\left\|\widehat{\Delta}_A\right\|_2^2}{\delta_L^2}\right),$$

where $p_i = sign\left(\widehat{\mathbf{q}}_i^T \mathbf{q}_i\right)$ *and* $\delta_L = \min_{j \neq l} |\lambda_l - \lambda_j|$.

Proof. As a consequence of [79, Theorem 2], we may write

$$\widehat{\mathbf{q}}_i = \mathbf{q}_i + \frac{(\lambda_i \, \mathrm{I}_p - A)^D \widehat{\Delta}_A \mathbf{q}_i}{\mathbf{y}_i^H \mathbf{q}_i} + O\left(\left\|\widehat{\Delta}_A\right\|_2^2\right),$$

where $\mathbf{y}_i$ is the corresponding unit-norm left eigenvector for λ_i, and A^D denotes the Drazin Inverse (also called the Group Inverse) of $A = Q\Lambda Q^+$. The discussion in the proof of [79, Corollary 4] indicates that we may bound $(\lambda_i \, \mathrm{I}_p - A)^D$ in Proposition 2.2 by

$\left\|(\lambda_i \mathrm{I}_p - A)^D\right\|_2 \leq 1/\delta_L$. Noting that λ_i is simple and that the $\mathbf{q}_i$ are linearly independent, we have that $\mathbf{y}_i^H \mathbf{q}_i$ is fixed and non-zero; see [129, Chapter 2] for a discussion of this quantity. Hence, we may bound

$$\left\| \frac{(\lambda_i \mathrm{I}_p - A)^D \widehat{\Delta}_A \mathbf{q}_i}{\mathbf{y}_i^H \mathbf{q}_i} + O\left(\left\|\widehat{\Delta}_A\right\|_2^2\right) \right\|_2^2 \leq O\left(\frac{\left\|\widehat{\Delta}_A\right\|_2^2}{\delta_L^2}\right). \tag{2.67}$$

□

Proposition 2.2 provides a bound on the individual eigenvector errors. Summing over the eigenvector errors, we have that

$$\sum_{i=1}^{k} \| \mathbf{q}_i - p_i \widehat{\mathbf{q}}_i \|_2^2 \leq O\left(k \frac{\left\|\widehat{\Delta}_A\right\|_2^2}{\delta_L^2}\right).$$

Noting that $\left\|\widehat{\Delta}_A\right\|_2^2 \leq \left\|\widehat{\Delta}_A\right\|_F^2$, we may substitute our bound from (2.66) to complete the proof.

2.B Bridging Corollary 2.1 and Theorem 2.1 with $\tau = 1$

When C is a matrix of cosines, we may bridge the gap as follows. To apply Theorem 2.1 to a matrix C with columns $\mathbf{c}_i$ of the form

$$c_{it} = \cos\left(\omega_i t + \phi_i\right), \tag{2.68}$$

we need to show that L_{ii} does not tend to zero, that L_{ij} does tend to zero for $i \neq j$, and that size of the elements of S is bounded. Moreover, we need bounds on the convergence of the L_{ij} and the elements of S. Recall that L was defined in (2.29), and is the matrix of circular inner products of the $\mathbf{s}_i$, where the $\mathbf{s}_i$, defined in (2.15), are the normalized $\mathbf{c}_i$ and form the columns of the matrix S.

To tackle these three tasks, we require the following two identities governing sums of products of cosines:

$$\begin{aligned}\sum_{t=1}^{n} &\cos\left(\omega_1 t + \phi_1\right) \times \cos\left(\omega_2 t + \phi_2\right) \\ &= \frac{1}{2\left(\cos\omega_1 - \cos\omega_2\right)}\Bigg(\cos\left(\omega_1[n+1] + \phi_1\right)\cos\left(\omega_2 n + \phi_2\right) \\ &- \cos\left(\omega_2[n+1] + \phi_2\right)\cos\left(\omega_1 n + \phi_1\right) \\ &- \cos\phi_2 \cos\left(\omega_1 + \phi_1\right) + \cos\phi_1 \cos\left(\omega_2 + \phi_2\right)\Bigg),\end{aligned} \tag{2.69}$$

when $\omega_1 \neq \omega_2$, and

$$\sum_{t=1}^{n} \cos^2\left(\omega_1 t + \phi_1\right) = \frac{n}{2} + \frac{1}{2}\frac{\sin\left(\omega_1 n\right)}{\sin\omega_1}\cos\left(\omega_1[n+1] + 2\phi_1\right). \tag{2.70}$$

We first consider the simplest of the three tasks: the bound on the size of S_{ij}. Since the $\mathbf{c}_i$ have entries of the form (2.68), applying (2.70), we have that

$$\|\mathbf{c}_i\|_2^2 = \frac{n}{2} + \frac{1}{2}\frac{\sin\left(\omega_i n\right)}{\sin\omega_i}\cos\left(\omega_i[n+1] + 2\phi_i\right). \tag{2.71}$$

Note that if ω_i is not 0 or π, (2.71) behaves like $\Theta(n)$. If ω_i is 0 or π, (2.71) is equal to $n\cos^2\phi_1$, which is also $\Theta(n)$: if $\cos^2\phi_i = 0$ and $\omega_i = 0$ or π, $\mathbf{c}_i$ is identically zero, and not part of a linearly independent set of vectors. Hence, the square of the norm of each $\mathbf{c}_i$ is $\Theta(n)$, and the elements of $\mathbf{c}_i$ are bounded in size by 1. It follows that the elements of S cannot be larger than $O(1/\sqrt{n})$, or that $\alpha = 1/2$.

Next, we consider the bound for L_{ij} for $i \neq j$. Assuming that $\omega_i \neq \omega_j$, we may bound the right-hand size of (2.69) by

$$\frac{2}{\left|\cos\omega_i - \cos\omega_j\right|}. \tag{2.72}$$

But (2.69) is exactly the inner product of $\mathbf{c}_i$ and $\mathbf{c}_j$, for $i \neq j$. Since the elements of L_{ij} are the inner products of the $\mathbf{s}_i$ with $\mathbf{s}_j$, dividing (2.72) by the norm of each $\mathbf{c}_i$ yields a bound on the size of L_{ij}. Since the norm of each $\mathbf{c}_i$ is $\Theta(\sqrt{n})$, the size of L_{ij} is bounded by

$$|L_{ij}| = O\left(\frac{1}{\sqrt{n}} \cdot \frac{1}{\left|\cos\omega_i - \cos\omega_j\right|}\right).$$

Taking the maximum over i and j yields that $|L_{ij}| \leq O\left(\frac{1}{\sqrt{n}} \cdot \frac{1}{\delta_L}\right)$, where

$$\delta_L = \min_{i \neq j} |\cos \omega_i - \cos \omega_j| \,.$$

Hence, we have that $f(n) = \frac{1}{\sqrt{n}} \frac{1}{\delta_L}$. Note that $f(n)$ in the corollary contains a factor of δ_L: this is the origin of the δ_L^4 dependence, relative to Theorem 2.1, which has a δ_L^2 dependence.

Finally, we characterize the elements L_{ii}. The third and final identity we need is a version of (2.69) with $\omega_1 = \omega_2$ and $\phi_2 = \phi_1 + \omega_1$:

$$\sum_{t=1}^{n} \cos(\omega_1 t + \phi_1) \times \cos(\omega_1 [t+1] + \phi_1) = \frac{n}{2} \cos \omega_1 + \frac{1}{2} \frac{\sin(\omega_1 n)}{\sin \omega_1} \cos(\omega_1 [n+1] + 2\phi_1) \,. \tag{2.73}$$

Unless ω_1 is $\pi/2$, L_{ii} will not have limit 0. For $\omega_1 \neq \pi/2$, (2.73) is $\Theta(n)$. Dividing by (2.70) yields that L_{ii} is the ratio of two $\Theta(n)$ quantities: for large n, the mixed sine-cosine terms in both equations are negligible, so that L_{ii} has limit $\cos \omega_i$.

Combining these steps, we obtain the result of Corollary (2.1) from Theorem (2.1).

Note that more generally, we may write a version of (2.73) for larger lags τ. That is, let $\omega_1 = \omega_2$, and $\phi_2 = \phi_1 + \tau \omega_1$, so that

$$\begin{aligned} &\sum_{t=1}^{n} \cos(\omega_1 t + \phi_1) \times \cos(\omega_1 [t+\tau] + \phi_1) \\ &\quad = \frac{n}{2} \cos(\tau \omega_1) + \frac{\sin(\omega_1 n)}{2 \sin \omega_1} \cos(\omega_1 [n + \tau + 1] + 2\phi_1) \,. \end{aligned} \tag{2.74}$$

That is, looking ahead to Theorem 2.1, unless $\omega_1 \tau$ is an odd multiple of $\pi/2$, $L_{ii}(\tau)$ will not have limit 0. Moreover, in the large n limit, we would have $L_{ii}(\tau) = \cos(\tau \omega_1)$.

2.C The proof of Theorem 2.1 for $\tau > 1$

We may define

$$S^{\tau}_{(0)} = \begin{bmatrix} s_{1,1} & s_{2,1} & \cdots & s_{k,1} \\ s_{1,2} & s_{2,2} & \cdots & s_{k,2} \\ \vdots & \vdots & \cdots & \vdots \\ s_{1,n-\tau} & s_{2,n-\tau} & \cdots & s_{k,n-\tau} \end{bmatrix} \text{ and } S^{\tau}_{(1)} = \begin{bmatrix} s_{1,1+\tau} & s_{2,1+\tau} & \cdots & s_{k,1+\tau} \\ s_{1,2+\tau} & s_{2,2+\tau} & \cdots & s_{k,2+\tau} \\ \vdots & \vdots & \cdots & \vdots \\ s_{1,n} & s_{2,n} & \cdots & s_{k,n} \end{bmatrix}. \quad (2.75a)$$

Then, we have that

$$X^{\tau}_{(0)} = QW\left(S^{\tau}_{(0)}\right)^T \text{ and } X_{(1)} = QW\left(S^{\tau}_{(1)}\right)^T. \quad (2.76)$$

We make the key observation that

$$\left(S^{\tau}_{(1)}\right)^T = \begin{bmatrix} s_{1,1+\tau} & \cdots & s_{1,n-\tau} & s_{1,1} & \cdots & s_{1,\tau} \\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ s_{k,1+\tau} & \cdots & s_{k,n-\tau} & s_{k,1} & \cdots & s_{k,\tau} \end{bmatrix} + \begin{bmatrix} 0 & \cdots & 0 & s_{1,n-\tau+1} - s_{1,1} & \cdots & s_{1,n} - s_{1,\tau} \\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ 0 & \cdots & 0 & s_{k,n-\tau+1} - s_{k,1} & \cdots & s_{k,n} - s_{k,\tau} \end{bmatrix}, \quad (2.77)$$

so that $\left(S^{\tau}_{(1)}\right)^T$ can be written as a τ-times shift of $\left(S^{\tau}_{(0)}\right)^T$, plus an error term, Δ_τ, where Δ_τ is the second term in (2.77). Mimicking the proof of Theorem 2.1 for the $\tau = 1$ case and assuming that τ is sufficiently small reveals that the only change is that Δ_1 is replaced with Δ_τ in (2.64) and (2.65). Hence, we replace $n^{-2\alpha}$ with $\tau n^{-2\alpha}$ in the final result.

2.D The Proof of Theorem 2.2

In this section, we provide the details behind the results of Theorem 2.2. Relative to the deterministic Theorems 2.1, Theorem 2.2 differs only in that the quantities $L(\tau)$ and d_i are random variables, where these quantities are defined in (2.29) and (2.18) respectively. Hence, it is sufficient to demonstrate that L_τ and the d_i are close to their expected values with high probability. In what follows, we suppress the τ dependence of L and other related quantities.

2.D.1 Conditions for the convergence of L to $\mathbb{E}\, L$

We first consider the convergence of L. For convergence of L to its expectation, we need a series of technical assumptions on the $\mathbf{c}_i$. In stating these, we mimic the notation and state the conditions for Theorem 2 (equations (1) through (4)) in [52]. Essentially, at each time t, we have p values: we have a p-dimensional time series. We will denote this series as $\widetilde{\mathbf{c}}_t$, with $\widetilde{\mathbf{c}}_t = \begin{bmatrix} \mathbf{c}_{1,t} & \mathbf{c}_{2,t} & \dots & \mathbf{c}_{p,t} \end{bmatrix}^T$. We require that each coordinate of $\widetilde{\mathbf{c}}_t$ is individually an ergodic, wide-sense (covariance) stationary process with zero mean and finite variance. Formally, if $\boldsymbol{\epsilon}_t \in \mathbb{R}^p$ is the sequence of linear innovations, we are able to write $\widetilde{\mathbf{c}}_t = \sum_{j=0}^{\infty} \kappa_j \, \boldsymbol{\epsilon}_{t-j}$, where the κ_j are $p \times p$ matrices. We require $\sum_{j=0}^{\infty} \|\kappa_j\|_F^2 < \infty$ and $(\kappa_0)_{il} = 1$. Moreover, if we define $K(z) = \sum_{j=0}^{\infty} \kappa_j z^j$, for $|z| < 1$, we require that the determinant of $K(z)$ is non-zero. We further require that if $\mathcal{F}_{t-1}$ is the σ-algebra generated by $\boldsymbol{\epsilon}_s$ for $s \leq t$,

$$\mathbb{E}\left[\boldsymbol{\epsilon}_t \mid \mathcal{F}_{t-1}\right] = \mathbf{0}_p, \mathbb{E}\left[\boldsymbol{\epsilon}_t \, \boldsymbol{\epsilon}_t^T \mid \mathcal{F}_{t-1}\right] = \Sigma_\epsilon, \text{ and } \mathbb{E}\left[|(\boldsymbol{\epsilon}_t)_i|^r \mid \mathcal{F}_{t-1}\right] \leq \infty, \tag{2.78a}$$

for $r \geq 4$. Moreover, Σ_ϵ is a fixed, deterministic $p \times p$ matrix.

2.D.2 The convergence of L to $\mathbb{E}\, L$

Given these many conditions, what can we say? We first consider all of the entries of L, diagonal and off-diagonal. Recall that the elements of L are (up to a scaling of $1/n$ and some neglected terms from the circularity) the auto- and cross-correlations of the $\mathbf{c}_i$ at the lag τ. Let $\mathbb{E}\, L_{ij}$ be the expected value of L_{ij}, for all i and j. Applying Theorem 2 of [52] (a strengthening of Theorems 1 and 2 from [48]), we have that

$$\max_{i,j} \max_{0 \leq \tau \leq n^{\frac{r}{2(r-2)}}} |L_{ij} - \mathbb{E}\, L_{ij}| = o\left((\tau \log n)^{2/r} \, (\log \log n)^{(1+\delta)2/r} \, n^{-1/2} \right), \tag{2.79}$$

almost surely, for some $r \geq 4$ and $\delta > 0$. I.e., for any reasonably small lag, as n grows (and p is fixed), we expect the auto- and cross-correlations to converge to their expected values, with strongly bounded deviations. Indeed, for a threshold

$$\psi = (\tau \log n)^{2/r} \, (\log \log n)^{(1+\delta)2/r} \, n^{-1/2},$$

we have that

$$\mathbb{P}\left[\max_{i,j} \max_{0\le\tau\le n^{\frac{r}{2(r-2)}}} |L_{ij} - \mathbb{E}\, L_{ij}| \ge \psi\right] \le O\left(\left[\log n\,(\log\log n)^{1+\delta}\right]^{-1}\right). \tag{2.80}$$

Hence, as n increases, the L matrix is close to its expected value with high probability.

There are two more quantities of interest. First, the separation δ_L: from the discussion above, it follows that the empirical value of $\min_{i\neq j}|L_{ii} - L_{jj}|$ is close to $\delta_L = \min_{i\neq j}|\mathbb{E}\, L_{ii} - \mathbb{E}\, L_{jj}|$ with high probability. Moreover, the lag-0 auto-covariance provides values of $\mathbb{E}\, d_1^2$ and $\mathbb{E}\, d_k^2$. It follows that the d_i are within $f(n)[1 + o(1)]$ of the $\mathbb{E}\, d_i$.

2.D.3 The desired properties of $\mathbb{E}\, L$

We have established that L and the other quantities has the desired convergence properties. Next, we discuss what properties we want $\mathbb{E}\, L$ to have. Assume that we are operating at a reasonable lag τ (per the conditions above). Then, we consider the lag τ autocorrelations and cross-correlations of the $\mathbf{c}_i$. We want the cross-correlations to be 0 in expectation, and the autocorrelations to be non-zero. Note that we do not demand that the $\mathbf{c}_i$ be independent or uncorrelated at every lag: just at the desired lag τ. In this setup, the right-hand side of (2.79) provides the bounding function $f(n)$ for the Theorem, as $\mathbb{E}\, L_{ij} = 0$ for the off-diagonal elements.

2.D.4 Special Case: ARMA

From Theorem 3 in [52], in the special case of a stationary ARMA process, we may strengthen these bounds. That is, if the $\mathbf{c}_i$ are drawn as contiguous realizations of an ARMA process, we may replace the right-hand side of (2.79) with $o\left((\log\log n/n)^{1/2}\right)$, for lags τ such that $0 \le \tau \le O\left([\log n]^a\right)$ for some $a > 0$, and with no further work reuse the same probability bound as in (2.80), with $\delta = 0$.

2.D.5 Obtaining the Theorem Statements

We have computed $f(n)$ and shown that with high probability L is close to $\mathbb{E}\, L$. We have further discussed the desired properties of $\mathbb{E}\, L$, and shown that the d_i are close to $\mathbb{E}\, d_i$ and that $\min_{i\neq j}|L_{ii} - L_{jj}|$ is close to $\min_{i\neq j}|\mathbb{E}\, L_{ii} - \mathbb{E}\, L_{jj}|$. Essentially, we have computed all of the quantities that appear in Theorem 2.1 with relevant probabilities. In Theorem 2.1, we replace these quantities with their expectations, and obtain the desired result.

2.E Proof of Theorem 2.3

Proof. Recall that the proof of Theorem 2.1 begins by bounding the perturbation of $\widehat{A}$ from $Q\Lambda Q^+$, as in written in (2.61). Hence, we may note that $\widehat{A}^T = (Q^+)^T \Lambda Q^T + \widehat{\Delta}_A^T$, and note that $\widehat{\Delta}_A^T$ has the same norm as $\widehat{\Delta}_A$. Following the rest of the proof to its conclusion reveals that we may estimate the left eigenvectors of $\widehat{A}$ with the same error bound as for the right.

Assume that our estimate of the left eigenvectors $\left(\widehat{Q^+}\right)^T$ has normalized columns. Then, writing $(Q^+)^T = \left(\widehat{Q^+}\right)^T + \Delta_{Q^+}^T$, we may write $\left(\widehat{Q^+}X\right)^T = SD + X^T\Delta_{Q^+}^T$. Let $\boldsymbol{\epsilon}_i$ denote the i^{th} column of $X^T\Delta_{Q^+}^T$, so that $\widehat{\mathbf{s}}_i = \frac{d_i\,\mathbf{s}_i + \boldsymbol{\epsilon}_i}{\|d_i\,\mathbf{s}_i + \boldsymbol{\epsilon}_i\|_2}$. We may write

$$\|\mathbf{s}_i - \widehat{\mathbf{s}}_i\|_2 = \left\| \mathbf{s}_i \left(1 - \frac{d_i}{\|d_i\,\mathbf{s}_i + \boldsymbol{\epsilon}_i\|_2}\right) + \boldsymbol{\epsilon}_i \frac{1}{\|d_i\,\mathbf{s}_i + \boldsymbol{\epsilon}_i\|_2} \right\|,$$

where we have implicitly assumed (without loss of generality) that $\mathbf{s}_i^T\,\widehat{\mathbf{s}}_i$ is positive. By the triangle inequality, we may write $d_i - \|\,\boldsymbol{\epsilon}_i\,\|_2 \le \|d_i\,\mathbf{s}_i + \boldsymbol{\epsilon}_i\|_2 \le d_i + \|\,\boldsymbol{\epsilon}_i\,\|_2$. Then, we have that

$$\|\mathbf{s}_i - \widehat{\mathbf{s}}_i\|_2 \le \max_{\pm} \left\{ \left| 1 - \frac{d_i}{d_i \pm \|\,\boldsymbol{\epsilon}_i\,\|_2} \right| + \frac{\|\,\boldsymbol{\epsilon}_i\,\|_2}{|d_i \pm \|\,\boldsymbol{\epsilon}_i\,\|_2|} \right\}, \tag{2.81}$$

where the maximum is taken over combinations of the ± signs in both terms.

Before proceeding, we need the following lemma:

Lemma 2.5. *Let $0 < y < x$, and assume that there is a constant $c > 0$ such that $x > 1/c$. Then,*

$$\left|1 - \frac{x}{x \pm y}\right| < cy \text{ and } \left|\frac{y}{x \pm y}\right| < cy.$$

Continuing, if $\|\,\boldsymbol{\epsilon}_i\,\|_2 < d_i$ for all $i = 1, 2, \ldots, k$, then by applying the lemma to each term in the right-hand side of (2.81) with $c = 2/d_k$, we have that $\|\mathbf{s}_i - \widehat{\mathbf{s}}_i\|_2 \le (4/d_k)\|\,\boldsymbol{\epsilon}_i\,\|_2$. Hence, summing over all i yields that

$$\sum_{i=1}^{k} \|\mathbf{s}_i - \widehat{\mathbf{s}}_i\|_2^2 \le \frac{4}{d_k^2} \sum_{i=1}^{k} \|\,\boldsymbol{\epsilon}_i\,\|_2^2 = \frac{16}{d_k^2} \|X^T\Delta_{Q^+}^T\|_F^2.$$

Recall that we have bounded $\|\Delta_{Q^+}^T\|_F^2$ by $\epsilon_{d,v}^2$, and $\|X^T\|_F^2$ by kd_1^2. It follows that

$$\sum_{i=1}^{k} \|\mathbf{s}_i - \widehat{\mathbf{s}}_i\|_2^2 \leq \frac{16d_1^2}{d_k^2} k\epsilon_{d,v}^2. \tag{2.82}$$

We have assumed that $\| \boldsymbol{\epsilon}_i \|_2 < d_i$ for all $i = 1, 2, \ldots, k$; a sufficient condition is that

$$\|X^T \Delta_{Q^+}^T\|_2^2 \leq \|X^T \Delta_{Q^+}^T\|_F^2 < d_k^2,$$

or that $kd_1^2\epsilon_{d,v}^2 < d_k^2$. □

2.F Proof of Theorem 2.4

Before proceeding, we remind the reader that the relevant notation and setup were presented in Section 2.6, and that (2.47) and (2.48) contain the required definitions and assumptions for the proof of the theorem.

Following the approach taken in the proof of Theorem 2.4 in [88], we write

$$\widetilde{X} = \mathbb{E}_M \widetilde{X} + \left(\widetilde{X} - \mathbb{E}\,\widetilde{X}\right) = qX + \left(\widetilde{X} - \mathbb{E}\,\widetilde{X}\right) = qX + \Delta_S, \tag{2.83}$$

where we define $\Delta_S = \left(\widetilde{X} - \mathbb{E}\,\widetilde{X}\right)$. We will first control the size of $\mathbb{E}\,\|\Delta_S\|_2$. Then, noting that the tSVD-DMD algorithm performs DMD on a truncated SVD $\widehat{X}_k$ of $\widetilde{X}$, we will bound the error in estimating X and X^+ from the low rank approximation of $\widetilde{X}$. That is, we will bound the deviation of the estimated singular vectors $\widehat{\mathbf{u}}_i$ and $\widehat{\mathbf{v}}_i$ and values $\widehat{\sigma}_i$ from the true values $\mathbf{u}_i$, $\mathbf{v}_i$, and σ_i, respectively, using the results from [92]. We will then compute the estimation error in $\left(\widehat{X}_{(0)}^{\tau}\right)^+$ and $\widehat{X}_{(1)}^{\tau}$, and hence write $\widetilde{A} = \widehat{X}_{(1)}^{\tau}\left[\widehat{X}_{(0)}^{\tau}\right]^+ = \widehat{A} + \Delta_A$. We will bound the size of Δ_A, and then bound the error in the eigenvectors of $\widetilde{A}$ from those of $\widehat{A}$. The final result will follow by an application of the triangle inequality.

2.F.1 Bounding $\mathbb{E}\,\|\Delta_S\|_2$

The first tool is a result of Latała [69]:

$$\mathbb{E}\,\sigma_1(\Delta_S) \leq C\left[\max_i \sqrt{\sum_j \mathbb{E}(\Delta_S)_{i,j}^2} + \max_j \sqrt{\sum_i \mathbb{E}(\Delta_S)_{i,j}^2} + \sqrt[4]{\sum_{i,j} \mathbb{E}(\Delta_S)_{i,j}^4}\right], \tag{2.84}$$

for some constant $C > 0$. We find that $\mathbb{E}\,\sigma_1(\Delta_S) \leq g(n,p,k,q)$, where

$$g(n,p,k,q) = O\bigg(\sqrt[4]{q(1-q)}d_1 k \times \max\left\{n^{1/4-\alpha}p^{1/4-\beta}, n^{-\alpha}, p^{-\beta}\right\}\bigg), \tag{2.85}$$

Next, we need a bound on the probability that $\mathbb{E}\left\|\Delta_S\right\|_2$ is close to $\left\|\Delta_S\right\|_2$. Noting that the first singular value is a 1-Lipschitz, convex function, and that $|(\Delta_S)_{i,j}| \leq O\left(d_1 n^{-\alpha} p^{-\beta} k\right)$, we may apply Talagrand's concentration inequality [117, Theorem 2.1.13, pp. 73]:

$$\mathbb{P}\left[|\sigma_1(\Delta_S) - \mathbb{E}\,\sigma_1(\Delta_S)| > t\right] \leq 2\exp\left(-ct^2\frac{n^{2\alpha}p^{2\beta}}{d_1^2 k^2}\right) = 2\exp\left(-c\gamma t^2\right), \tag{2.86}$$

for some constant $c > 0$.

2.F.2 The Low Rank Approximation

We apply the results from [92] to characterize the finite-sample performance of the low-rank approximation. Given the low-rank approximation that fills in the missing entries, we have an estimate $\widehat{X}$ of qX. Then, we have $\widehat{X}_0^+$ and $\widehat{X}_1$ that are passed into the DMD algorithm. Given $\widetilde{X}$, we will characterize how far $\widehat{X}$ is from qX. Then, (by assumptions on the density of qX) these bounds are close to those for $X_{(1)}$ and $X_{(0)}$, and we can apply them to write $\widehat{X}_{(0)}^+$ as $\frac{1}{q}X_{(0)}^+ + \Delta_{S_0}$ and $\widehat{X}_{(1)}$ as $qX_{(1)} + \Delta_{S_1}$. Furthermore, we assume that we have oracular knowledge of the rank k.

Before proceeding, note that we have controlled the size of the entries of Δ_S, shown that its norm concentrates and is bounded, and bounded the expectation of the norm. Moreover, Δ_S is trivially zero mean and and random (from the randomness in masking the entries of X). Hence, we are able to apply the results from [92].

The Singular Vectors of $\widetilde{X}$

We have previously found that

$$\mathbb{P}\left(|\sigma_1(\Delta_S)| > \widetilde{t}\right) \leq 2\exp\left(-c_0\gamma(\widetilde{t} - g(n,p,k,q))^2\right).$$

Let $t = \widetilde{t} - g(n,p,k,q)$ for some $\widetilde{t}$.

Recall that for two unit norm vectors $\mathbf{x}$ and $\mathbf{y}$, $\sin^2\angle(\mathbf{x},\mathbf{y}) = 1 - (\mathbf{x}^T\mathbf{y})^2 \leq \epsilon^2$ means that if $\mathbf{x}^T \mathbf{y} \geq 0$,

$$\| \mathbf{x} - \mathbf{y} \|_2^2 = 2 \left(1 - \mathbf{x}^T \mathbf{y}\right) \leq 2 \left(1 - \sqrt{1 - \epsilon^2}\right) \leq 2\epsilon^2.$$

Applying Corollary 20 from [92] and noting that $\|\Delta_S\|_2 \leq t$ with high probability, we have that

$$\sin \angle(\mathbf{v}_i, \widehat{\mathbf{v}}_i) \leq 8\sqrt{2}\frac{\sqrt{k}}{\delta_{\sigma,q}} \left[t(\sqrt{k} + 1) + t^2\right], \tag{2.87}$$

with probability at least

$$\left[1 - 24 \cdot 9^k \exp\left(-\gamma \frac{\delta_{\sigma,q}^2}{64}\right) - 8 \cdot 81^k \exp\left(-\gamma k \frac{t^2}{16}\right)\right] \cdot \left[1 - 2\exp\left(-c_0 \gamma t^2\right)\right]. \tag{2.88}$$

Then, if V contains the first k right singular vectors of X, and assuming that $t \to 0$ and that $\delta_{\sigma,q} \nrightarrow 0$, we have that

$$\left\| V - \widehat{V} \right\|_F \leq O\left(\frac{k^2 t}{\delta_{\sigma,q}}\right), \tag{2.89}$$

with probability at least

$$1 - O\left(9^k \exp\left(-\gamma \frac{\delta_{\sigma,q}^2}{64}\right)\right) - O\left(81^k \exp\left(-\gamma k \frac{t^2}{16}\right)\right) - O\left(\exp\left(-c_0 \gamma t^2\right)\right). \tag{2.90}$$

We have an identical result for U and $\widehat{U}$.

The Singular Values of $\widetilde{X}$

Applying Theorem 23 from [92], we next have that $\widehat{\sigma}_j(\widetilde{X}) \geq \sigma_j(qX) - t$ with probability at least

$$1 - 4 \cdot 9^j \exp\left(-c_0 \gamma \frac{t^2}{16}\right), \tag{2.91}$$

and that

$$\widehat{\sigma}_j(\widetilde{X}) \leq \sigma_j(qX) + \sqrt{k}t + 2\sqrt{j}\frac{t^2}{\sigma_j(qX)} + j\frac{t^3}{\left(\sigma_j(qX)\right)^2}, \tag{2.92}$$

with probability at least

$$1 - 4 \cdot 81^k \exp\left(-c_0 \gamma \frac{t}{16}\right) - 2\exp\left(-c_0 \gamma t^2\right). \tag{2.93}$$

It follows that

$$|\widehat{\sigma}_i - \sigma_i| \leq t(\sqrt{k}+1) + 2\sqrt{j}\frac{t^2}{\sigma_j(qX)} + j\frac{t^3}{\left(\sigma_j(qX)\right)^2} \tag{2.94}$$

with probability at least

$$1 - 4 \cdot 81^k \exp\left(-c_0\gamma\frac{t}{16}\right) - 2\exp\left(-c_0\gamma t^2\right) - 4 \cdot 9^j \exp\left(-c_0\gamma\frac{t^2}{16}\right). \tag{2.95}$$

Lemma 2.6. *For positive scalars a, x, and y, $\frac{1}{x-y} \leq \frac{1}{x} + ay$ if $y > x$ or if $x \geq \sqrt{\frac{1}{a}}$ and $y \leq x - \frac{1}{ax}$. Moreover, $\frac{1}{x+y} \geq \frac{1}{x} - ay$ if $x \geq \sqrt{\frac{1}{a}}$, or if $0 < x \leq \sqrt{\frac{1}{a}}$ and $y > \frac{1}{ax} - x$.*

Applying the lemma, we find that if $t \leq \frac{3}{4}\sigma_k(qX)$ (true for sufficiently large n and p, by assumption), we may write

$$\left|\frac{1}{\widehat{\sigma}_j} - \frac{1}{\sigma_j(qX)}\right| \leq \frac{4}{\sigma_k^2}\left[(\sqrt{k}+1)t + 2\sqrt{j}\frac{t^2}{\sigma_j} + j\frac{t^3}{\sigma_j^2}\right]$$

with probability at least (2.95).

Then, it follows that

$$\left\|\Sigma - \widehat{\Sigma}\right\|_F \leq O\left(kt\right) \text{ and } \left\|\Sigma^+ - \widehat{\Sigma^+}\right\|_F \leq O\left(\frac{kt}{\sigma_k^2}\right), \tag{2.96a}$$

with probability at least

$$1 - O\left(81^k \cdot k \cdot \exp\left(-c\frac{\gamma t^2}{16}\right)\right). \tag{2.96b}$$

We have assumed that $\sigma_k \nrightarrow 0$ and that $t^2\gamma \nrightarrow 0$.

The Error in $\widehat{X}$

Finally, we may combine all of the above results and write the following where if $qX = U\Sigma V^T$ is the (thin) SVD of qX, $\widehat{X} = \left(U + \Delta_U\right)\left(\Sigma + \Delta_{\Sigma,q}\right)\left(V + \Delta_V\right)^T$. We may then write $\widehat{X} = qX + \Delta_X$, where

$$\Delta_X = U\Sigma\Delta_V^T + U\Delta_{\Sigma,q}V^T + U\Delta_{\Sigma,q}\Delta_V^T + \Delta_U\Sigma V^T + \Delta_U\Sigma\Delta_V^T + \Delta_U\Delta_{\Sigma,q}V^T + \Delta_U\Delta_{\Sigma,q}\Delta_V^T. \tag{2.97}$$

Then we may write $\widehat{X} = qX + \Delta_X$, where Δ_X is defined as all but the first term in (2.97). We now plug in our bounds for the sizes of the Δ terms, note that each U and V add factors of $\sqrt{k}$ to the Frobenius norm, and note that Σ adds a factor bounded by $\sqrt{k}\sigma_1(qX)$. Then, when g is sufficiently small, we have that

$$\|\Delta_X\|_F \leq O\left(k^3 t \frac{\sigma_1(qX)}{\delta_{\sigma,q}}\right), \tag{2.98}$$

with probability at least

$$1 - O\left(k \cdot 81^k \exp\left(-c_0\gamma k t^2/16\right)\right) - O\left(k \cdot 9^k \exp\left(-c_0\gamma k \delta_{\sigma,q}/64\right)\right). \tag{2.99}$$

The result for $\widehat{X}^+$ is similar: we may expand $\widehat{X}^+$ as we did for $\widehat{X}$ in (2.97), and obtain that with the same probability, we have $\widehat{X}^+ = X^+ + \Delta_{X^+}$, where

$$\|\Delta_{X^+}\|_F \leq O\left(k^3 t \frac{1}{\delta_{\sigma,q}}\right). \tag{2.100}$$

2.F.3 Using $\widehat{X}_k$ to estimate $\widehat{A}$

Next, we consider the estimation of $\widehat{A}$ with $\widetilde{A} = \widehat{X}^\tau_{(1)}\left[\widehat{X}^\tau_{(0)}\right]^+$. That is, we estimate $\widehat{X}$, and take the sub-matrices $\widehat{X}^\tau_{(1)}$ and $\widehat{X}^\tau_{(0)}$ as inputs to DMD. Our previous bounds may be applied with $g(n,p,k,q)$ replaced with $\sqrt{\tau}g(n,p,k,q)$: note that the sum of squares of the norms of τ columns of X is bounded by $kd_1^2\tau n^{-2\alpha}$, and all of these factors except τ appear in $g(n,p,k,q)^2$. Writing $\widehat{X}^\tau_{(1)} = X_{(1)} + \Delta_{X_1}$ and $\left(\widehat{X}^\tau_{(0)}\right)^+ = X^+_{(0)} + \Delta_{X_0^+}$, we may write $\widetilde{A} = \widehat{A} + \Delta_A$, where Δ_A is the sum of all but the first term in

$$\widetilde{A} = X_{(1)}X^+_{(0)} + \Delta_{X_1}X^+_{(0)} + X_{(1)}\Delta_{X_0^+} + \Delta_{X_1}\Delta_{X_0^+}. \tag{2.101}$$

Note that we have dropped the τ dependence for ease of reading. Each factor of $X_{(1)}$ adds $\sqrt{k} \times \sigma_1(qX_{(1)})$ to the Frobenius norm, and each factor of $X^+_{(0)}$ adds $\sqrt{k}/\sigma_k(qX_{(0)})$. Hence, we may write

$$\|\Delta_A\|_F \leq O\left(\frac{\sqrt{k}}{\sigma_k(qX_0)}\|\Delta_{X_1}\|_F + \sqrt{k}\sigma_1(qX_1)\left\|\Delta_{X_0^+}\right\|_F\right). \tag{2.102}$$

Ideally, we would have (2.102) in terms of X. First, note that by the Cauchy Interlacing Theorem [40], $\sigma_1(qX_{(1)}) \leq \sigma_1(qX)$. It follows that we may replace $X_{(1)}$ with X without any further work.

Since $X_{(0)}$ has the same singular values as a version of X with the last τ columns set to 0, we may replace $X_{(0)}$ with a perturbation of X, denoted by $\widetilde{X}_{(0)}$: $\widetilde{X}_{(0)} = X + \widetilde{\Delta}_{X_0}$, where

$$\left\|\widetilde{\Delta}_{X_0}\right\|_F \leq \sqrt{k\tau} d_1 n^{-\alpha} \leq \sqrt{\frac{\tau}{\sqrt{q(1-q)}}} \times g(n,p,k,q).$$

An application of the Weyl Inequality [59, Theorem 4.3.1] yields that

$$\frac{1}{\sigma_k(qX_{(0)})} = \frac{1}{\sigma_k(q\widetilde{X}_{(0)})} \leq \frac{1}{\sigma_k(qX) - q\widetilde{\Delta}_{X_0}}.$$

By assumption, $\sigma_k(X)$ does not have limit 0. Moreover, by assumption, the norm of $\widetilde{\Delta}_{X_0}$ does have limit zero. Hence, for sufficiently large n, we may write

$$\frac{1}{\sigma_k(qX_{(0)})} \leq \frac{1}{\sigma_k(qX)} + \frac{1}{q} O\left(\left\|\widetilde{\Delta}_{X_0}\right\|_F\right) \leq \frac{1}{\sigma_k(qX)} + \frac{\sqrt{\tau}}{q} O(g(n,p,k,q)).$$

Now, let $t = ag(n,p,k,q)$ for some $a > 1$. Putting the previous work together, we find that

$$\|\Delta_A\|_F \leq O\left(k^{7/2} a \sqrt{\tau} g(n,p,k,q) \frac{\sigma_1(qX)}{\delta_{\sigma,q}}\right). \tag{2.103}$$

This bound holds with probability at least

$$1 - O\left(k \cdot 81^k \exp\left(-\left(1-\frac{1}{a}\right)^2 c_0 \gamma \frac{(\sqrt{\tau} g(n,p,k,q))^2}{16}\right)\right) - O\left(k \cdot 9^k \exp\left(-c_0 \gamma \frac{\delta_{\sigma,q}}{64}\right)\right). \tag{2.104}$$

2.F.4 The DMD Eigenvectors

Finally, we have previously bounded the deviation of $\widehat{A} = X_{(1)} X_{(0)}^+$ from $Q\Lambda Q^+$. We have just bounded the deviation of $\widetilde{A}$ from $\widehat{A}$ due to missing data. We may combine the effects of missing data and the deterministic noiseless deviation bound via the triangle inequality. Then, we apply the the union bound over the k eigenvectors. Let $\epsilon_{d,v}^2$ be the deterministic deviation of the $\mathbf{q}_k$, i.e., the right-hand side of (2.33a). Then, with probability at least

$$1 - O\left(k^2 \cdot 81^k \exp\left(-\left(1-\frac{1}{a}\right)^2 c_0\gamma \frac{\tau\left(g(n,p,k,q)\right)^2}{16}\right)\right) - O\left(k^2 \cdot 9^k \exp\left(-c_0\gamma\frac{\delta_{\sigma,q}}{64}\right)\right), \tag{2.105}$$

$$\sum_{i=1}^{k} \|\widehat{\mathbf{q}}_i - p_i\,\mathbf{q}_i\|_2^2 \leq O\left(\frac{\tau}{q^2}a^2\left(g(n,p,k,q)\right)^2 \frac{\sigma_1^2(X)}{\delta_\sigma^2}\frac{k^8}{\delta_L^2} + \epsilon_{d,v}^2\right), \tag{2.106}$$

where we have adapted the final step in the proof of Theorem 2.1.

Finally, let $\epsilon_{d,e}^2$ be the deterministic deviation of the L_{ii}, i.e., the right-hand side of (2.33c). Once again adapting the final step in the proof of Theorem 2.1, we have that for each L_{ii}, there is an eigenvalue of $\widetilde{A}$ such that

$$|L_{ii} - \lambda_i|^2 \leq O\left(\frac{\tau}{q^2}a^2\left(g(n,p,k,q)\right)^2 \frac{\sigma_1^2(X)}{\delta_\sigma^2}k^7 + \epsilon_{d,e}^2\right), \tag{2.107}$$

with probability at least (2.105).

Chapter 3

The Performance of the DMD Algorithm with a Denoising Step

We have previously analyzed the DMD algorithm in the noiseless data setting; now, we analyze the tSVD-DMD algorithm in the presence of noisy and missing data. In the noisy data or missing data settings, it is advantageous to 'clean' or denoise the data before applying DMD. We use a truncated SVD (tSVD) as a denoising step and find that the tSVD-DMD algorithm inherits the phase transition from the trucnated SVD. Moreover, we derive an shrinkage-like estimator in the same spirit as the OptShrink algorithm [88]. We also provide some preliminary characterizations of and conjectures about DMD performed directly on noisy data. [2]

3.1 Introduction

This chapter is a direct continuation of Chapter 2. In Chapter 2, we studied the Dynamic Mode Decomposition (DMD) algorithm in the noise-free setting and derived performance bounds for DMD applied to the Blind Source Separation (BSS) problem [27]. We also derived results for the performance of DMD on missing data pre-processed with a truncated SVD. We move on in this chapter to DMD applied to data that is corrupted by both noise and missing values: this setting is the most physically realistic and the most interesting to practitioners. In the field of fluid mechanics, DMD is generally applied to real measure-

[2] This chapter describes joint work with Asad Lodhia and Raj Rao Nadakuditi. Preliminary work on this topic has appeared in [100].

ments of fluid flowing. Of course, no set of sensors or cameras is perfect, and there will necessarily be some noise in the collected data. Additionally, it is very possible that some measurements may be so corrupted that they should be dropped (treated like they were missing to begin with), or that some measurements are lost due to sensor failure or the like. Moving beyond fluid mechanics, e.g., to blind source separation of real audio signals, it is extremely easy to imagine noise or missingness in the collected audio streams.

As we will see, DMD applied to noisy data leads to poor results. Nonetheless, there is hope: if the latent signal is low rank, i.e., there are only a few latent signals to estimate relative to the dimensionality of the problem, the low-rank structure of the signal means that a truncated SVD (tSVD) is a natural denoising choice [88]. In this chapter, we analyze the performance of DMD with a tSVD denoising step. Additionally, we take some preliminary steps toward analyzing DMD applied directly to noisy data, as well as DMD applied to data that is entirely composed of noise.

3.2 Model

Consider a collection of latent signals $\{\mathbf{s}_i\}_{i=1}^r$ where $\mathbf{s}_i \in \mathbb{C}^n$ and $r < n$. Define

$$\theta_i = \|\mathbf{s}_i\|_2 \text{ and } \mathbf{v}_i = \frac{1}{\theta_i}\mathbf{s}_i\,. \tag{3.1}$$

Assume that the θ_i and $\mathbf{v}_i$ are ordered so that $\theta_1 > \theta_2 > \ldots > \theta_r > 0$, and that the signal strengths θ_i and rank r are not growing with n. Moreover, assume that the $\mathbf{v}_i$ are mutually orthogonal (or orthogonal in expectation, if stochastic), so that $\mathbf{v}_i^H\,\mathbf{v}_j$ is zero (or is zero in expectation) for $i \neq j$. Moreover, assume that the $\mathbf{v}_i$ have a non-trivial autocorrelation structure, i.e., if P is the circular left-shift matrix, $\mathbf{v}_i^H\,P\,\mathbf{v}_i = \gamma_i$, and $\mathbf{v}_i^H\,P\,\mathbf{v}_j = 0$ for $i \neq j$ (these statements once again hold in expectation for a stochastic $\mathbf{v}_i$). Finally, let $V \in \mathbb{C}^{n\times r}$ be the matrix with the $\mathbf{v}_i$ as columns and $\Theta \in \mathbb{R}^{r\times r}$ be a diagonal matrix with the θ_i on the diagonal.

Assume that we observe a mixture of the signals: let $U \in \mathbb{C}^{p\times r}$ be a mixing matrix with orthonormal columns $\mathbf{u}_i$ ($p = p(n) \geq r$) and consider a signal matrix

$$Y = \sum_{i=1}^{r} \theta_i\,\mathbf{u}_i\,\mathbf{v}_i^H = U\Theta V^H. \tag{3.2}$$

Here, Y is latent and we instead observe the noisy matrix

$$X = Y + G, \tag{3.3}$$

where the entries of G are *i.i.d.* $\mathcal{N}(0, 1/n)$ (Gaussian) random variables.

Remark 3.1. *At first glance the orthogonality of the signals $\mathbf{v}_i$ may appear restrictive; however, the results in [102, Thm. 3.2] require the asymptotic decay of the inner products $\mathbf{v}_i^H P \mathbf{v}_j$ and $\mathbf{v}_i^H \mathbf{v}_j$. Moreover, to prevent recovery of linear combinations of the signals given the orthogonality of the $\mathbf{u}_i$, we require the θ_i to be distinct. Hence, the only constraining part of this setup is the orthogonality of the $\mathbf{u}_i$.*

3.2.1 Denoising X

To denoise X and estimate Y, we perform the truncated SVD. We assume that we have oracle knowledge of the rank r, and hence obtain

$$\widehat{X} = \widehat{U}\widehat{\Theta}\widehat{V}^H \tag{3.4}$$

where $\widehat{U} \in \mathbb{C}^{p\times r}$ is an estimator of U, and so on.

3.3 Circular tSVD-DMD

The DMD algorithm applied to X, which has columns $\{\mathbf{x}_i\}_{i=1}^n$, would proceed by forming

$$X_{(1)} = \begin{bmatrix} \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_n \end{bmatrix} \text{ and } X_{(0)} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{n-1} \end{bmatrix}, \tag{3.5a}$$

and then taking an eigendecomposition of the matrix

$$\widehat{A} = X_{(1)} X_{(0)}^{+}, \tag{3.5b}$$

where the + denotes the Moore-Penrose pseudoinverse. The eigenvectors of $\widehat{A}$, denoted by $\widehat{\mathbf{q}}_i$, would be estimators of the $\mathbf{u}_i$.

However, we may observe that $X_{(1)} \approx X_{(0)} P$ up to a small rank-1 perturbation. Hence, we will consider the object

$$\widehat{A} = \widehat{X} P \widehat{X}^{+} = \widehat{U} \widehat{\Theta} \left[\widehat{V}^H P \widehat{V} \right] \widehat{\Theta}^{+} \widehat{U}^H. \tag{3.6}$$

3.3.1 Performance

Note that in the noise-free setting, we would have

$$\widehat{A} = Y P Y^{+} = U \Theta \left[V^H P V \right] \Theta^{+} U^H, \tag{3.7}$$

and by assumption $\left[V^H P V\right]$ is a diagonal matrix. Thus, our goal will be to show that $\left[\widehat{V}^H P \widehat{V}\right]$ is asymptotically diagonal. Then, it is immediate that $\widehat{A}$ has eigenvectors $\widehat{\mathbf{u}}_i$ with eigenvalues $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_i$. Moreover, we also have estimates of the latent signals $\mathbf{v}_i$. Theorem 3.1, taken from [12, Sec. 3.1], quantifies the performance of the estimators $\widehat{\mathbf{u}}_i$ and $\widehat{\mathbf{v}}_i$.

Theorem 3.1 (Performance of the tSVD)**.** *Let the data X be formed according to the model described in Section 5.2 and let $\widehat{\mathbf{u}}_i$, $\widehat{\theta}_i$, and $\widehat{\mathbf{v}}_i$ be as defined in (3.4). Let $c = p/n$. Then, we have that almost surely, [12, Sec. 3.1]*

1. $\left|\mathbf{u}_i^H \widehat{\mathbf{u}}_i\right|^2 \to \alpha_{u_i}^2 = \begin{cases} 1 - \frac{c\left(1+\theta_i^2\right)}{\theta_i^2\left(\theta_i^2+c\right)} & \textit{if } \theta_i \geq c^{1/4}, \\ 0 & \textit{otherwise.} \end{cases}$

2. $\left|\mathbf{v}_i^H \widehat{\mathbf{v}}_i\right|^2 \to \alpha_{v_i}^2 = \begin{cases} 1 - \frac{\left(c+\theta_i^2\right)}{\theta_i^2\left(\theta_i^2+1\right)} & \textit{if } \theta_i \geq c^{1/4}, \\ 0 & \textit{otherwise.} \end{cases}$

Theorem 3.2 and Corollary 3.1 describe the performance of the tSVD-DMD algorithm. We see that regardless of signal strength, $\widehat{V}^H P \widehat{V}$ is asymptotically diagonal. Moreover, we may write $\widehat{A}$ as a symmetric matrix plus a noise term whose magnitude converges to 0.

Theorem 3.2. *Let $\widehat{A}$ be formed according to (3.6), the data X and Y formed according to the model described in Section 5.2, and the results of Theorem 3.1 hold. Then, we have that*

1. *$\widehat{V}^H P \widehat{V}$ is asymptotically almost surely diagonal.*

2. *The diagonal entries of $\widehat{V}^H P \widehat{V}$, $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_i$, have an almost sure limit of $\alpha_{v_i}^2 \gamma_i$.*

3. *In the almost sure limit, the eigenvalues of $\widehat{A}$ are $\alpha_{v_i}^2\gamma_i$ with corresponding eigenvectors $\widehat{\mathbf{u}}_i$.*

Corollary 3.1. *We may write $\widehat{A}$, defined in (3.6), as*

$$\widehat{A} = \widehat{U}\widehat{\Lambda}\widehat{U}^H + \Delta_A, \tag{3.8}$$

where $\widehat{\Lambda}$ is diagonal with entries $\alpha_{v_i}^2\gamma_i$ and

$$\|\Delta_A\|_F \to 0$$

almost surely.

3.4 Some Intuitions behind Theorem 3.2

Before formally proving Theorem 3.2, we will walk through a rough heuristic derivation that motivates why we expect it to be true.

Note that we may write

$$\widehat{\mathbf{v}}_i = \alpha_i \, \mathbf{v}_i + \sqrt{1-\alpha_i^2}\, \mathbf{v}_{i,\perp}, \tag{3.9}$$

where (abusing notation), $\mathbf{v}_{i,\perp}$ is some unit vector that is orthogonal to $\mathbf{v}_i$ and

$$\alpha_i = \widehat{\mathbf{v}}_i^H \, \mathbf{v}_i \, .$$

Note that $\mathbf{v}_{i,\perp}$ is random.

The entries of $\widehat{V}^H P \widehat{V}$ are $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j$. Using (3.9), we may write

$$\begin{aligned}\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j = {} & \overline{\alpha_i}\alpha_j \, \mathbf{v}_i^H \, P \, \mathbf{v}_j + \overline{\alpha_i}\sqrt{1-\alpha_j^2}\, \mathbf{v}_i^H \, P \, \mathbf{v}_{j,\perp} \\ & + \overline{\sqrt{1-\alpha_i^2}}\alpha_j \, \mathbf{v}_{i,\perp}^H \, P \, \mathbf{v}_j + \overline{\sqrt{1-\alpha_i^2}}\sqrt{1-\alpha_j^2}\, \mathbf{v}_{i,\perp}^H \, P \, \mathbf{v}_{j,\perp} \, .\end{aligned} \tag{3.10}$$

We now use the randomness of $\mathbf{v}_{i,\perp}$: all inner products involving these vectors have expected limit zero. Then, we note that since the θ_i are distinct, we expect the inner product $\left|\widehat{\mathbf{v}}_i^H \, \mathbf{v}_i\right|^2$ to have limit α_i^2 (given by Theorem 3.1) and $\widehat{\mathbf{v}}_i^H \, \mathbf{v}_0$ to have limit zero. Finally, we recall that $\mathbf{v}_i^H \, P \, \mathbf{v}_j$ has limit $\gamma_i \neq 0$ and that $\mathbf{v}_i^H \, P \, \mathbf{v}_j$ has limit zero, so that

$$\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j \to \begin{cases} \alpha_i^2 \gamma_i & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

It follows that

$$\widehat{A} = \widehat{X} P \widehat{X}^+ = \widehat{U} \widehat{\Theta} \left[\widehat{V}^H P \widehat{V} \right] \widehat{\Theta}^+ \widehat{U}^H$$

has limit

$$\widehat{A} \to \widehat{U} \mathrm{diag} \left(\alpha_i^2 \gamma_i \right) \widehat{U}^H,$$

which is a Hermitian matrix. Hence, the eigenvectors of $\widehat{A}$, denoted by $\widehat{\mathbf{q}}_i$, will be exactly the $\widehat{\mathbf{u}}_i$. Then, if $\widehat{Q} = \begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_r \end{bmatrix}$, estimating the latent signals by $\left(\widehat{Q}^+ \widehat{X} \right)^H$ yields vectors that are proportional to the $\widehat{\mathbf{v}}_i$. Hence, the truncated SVD denoising procedure has the side effect of symmetrizing or Hermitianizing the DMD eigenvalue problem.

3.5 Missing Data

We modify our data model to include missing data. That is, instead of observing the matrix X, we observe

$$X \odot M,$$

where the entries of M are *i.i.d.* random variables with

$$M_{ij} = \begin{cases} 1 & \text{with probability } q, \\ 0 & \text{with probability } 1 - q, \end{cases} \tag{3.11}$$

where $\odot$ denotes the element-wise or Hadamard product. Additionally, we impose the following *incoherence* or *density* conditions on $\mathbf{u}_i$ and $\mathbf{v}_i$:

$$\max_{1 \leq i \leq k} \|\mathbf{u}_i\|_\infty \leq C_u \frac{(\log p)^{\eta_u}}{\sqrt{p}} \text{ and } \max_{1 \leq i \leq k} \|\mathbf{v}_i\|_\infty \leq C_v \frac{(\log n)^{\eta_v}}{\sqrt{p}}, \tag{3.12}$$

for some positive constants C_u, C_v, η_u, and η_v that do not depend on n and p. Intuitively, if we have missing data, we cannot expect to recover sparse vectors, as we would be unable to distinguish between a missing entry and a zero. We then have the following result from [114, Thm. 2]:

Theorem 3.3 (Performance of the tSVD with Noisy, Missing Data)**.** *Let the data X be formed according to the model described in Section 5.2 and (3.12), and then masked/observed with probability q, as in (3.11). Let $\widehat{\mathbf{u}}_i$, $\widehat{\theta}_i$, and $\widehat{\mathbf{v}}_i$ be as defined in (3.4). Define $c = p/n$ and $\widetilde{\theta}_i = \sqrt{q}\theta_i$. Then, we have that almost surely, [114, Thm. 2]*

1. $\left|\mathbf{u}_i^H \widehat{\mathbf{u}}_i\right|^2 \to \widetilde{\alpha}_{u_i}^2 = \begin{cases} 1 - \frac{c\left(1+\widetilde{\theta}_i^2\right)}{\widetilde{\theta}_i^2\left(\widetilde{\theta}_i^2+c\right)} & \textit{if } \widetilde{\theta}_i \geq c^{1/4}, \\ 0 & \textit{otherwise.} \end{cases}$

2. $\left|\mathbf{v}_i^H \widehat{\mathbf{v}}_i\right|^2 \to \widetilde{\alpha}_{v_i}^2 = \begin{cases} 1 - \frac{\left(c+\widetilde{\theta}_i^2\right)}{\widetilde{\theta}_i^2\left(\widetilde{\theta}_i^2+1\right)} & \textit{if } \widetilde{\theta}_i \geq c^{1/4}, \\ 0 & \textit{otherwise.} \end{cases}$

Theorem 3.3 indicates that the effect of missing data is to lower the SNR by a factor of $\sqrt{q}$, where q is the probability of observing an individual entry. Relative to Theorem 3.1, the rescaling of θ_i is the only change.

Given Theorem 3.3, we immediately are able to state Theorem 3.4 and Corollary 3.2, analogous to Theorem 3.2 and Corollary 3.1, respectively. Once again, the form of these new theorems is exactly analogous to their non-missing counterparts.

Theorem 3.4. *Let $\widehat{A}$ be formed according to (3.6) from the missing data model (3.11), and let the conditions and results of Theorem 3.3 hold. Then, we have that*

1. *$\widehat{V}^H P \widehat{V}$ is asymptotically almost surely diagonal.*

2. *The diagonal entries of $\widehat{V}^H P \widehat{V}$, $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_i$, have an almost sure limit of $\widetilde{\alpha}_{v_i}^2 \gamma_i$.*

3. *In the almost sure limit, the eigenvalues of $\widehat{A}$ are $\widetilde{\alpha}_{v_i}^2 \gamma_i$ with corresponding eigenvectors $\widehat{\mathbf{u}}_i$.*

Corollary 3.2. *We may write $\widehat{A}$, defined in (3.6), as*

$$\widehat{A} = \widehat{U}\widehat{\Lambda}\widehat{U}^H + \Delta_A, \tag{3.13}$$

where $\widehat{\Lambda}$ is diagonal with entries $\widetilde{\alpha}_{v_i}^2 \gamma_i$ and

$$\|\Delta_A\|_F \to 0$$

almost surely.

3.6 Simulations

In this section, we provide some numerical verifications of our results. We use a rank-2 signal

$$Y = \theta\, \mathbf{u}_1\, \mathbf{v}_1^H + \frac{\theta}{2}\, \mathbf{u}_2\, \mathbf{v}_2^H,$$

where $\theta_2 = \theta_1/2$, the entries of $\mathbf{v}_1$ are proportional to a realization of an $AR(2)$ process with coefficients $(1/6, 2/3)$ (so that $\gamma_1 = 1/2$), and the entries of $\mathbf{v}_2$ are proportional to $\cos \omega t$ where $\cos \omega = 1/4$ (so that $\gamma_2 = 1/4$). The $\mathbf{u}_i$ are randomly generated and orthogonal to each other. We vary the ratio $c = p/n$, the signal strength θ, and the observation probability q. Using oracle knowledge of the rank r, we compute the truncated SVD and compute $\widehat{V}^H P \widehat{V}$. We compute $\left|\widehat{\mathbf{q}}_i^H\, \mathbf{u}_i\right|^2$ where $\widehat{\mathbf{q}}_i$ are the DMD eigenvectors generated from both X and the denoised $\widehat{X}$. Then, given the estimated $\widehat{Q}$, we compute $\widehat{S}$ by normalizing the columns of $\left(\widehat{Q}^+ \widehat{X}\right)^H$ or $\left(\widehat{Q}^+ X\right)^H$. Note that $\widehat{Q}$ has columns $\mathbf{q}_i$ and $\widehat{S}$ has columns $\mathbf{s}_i$.

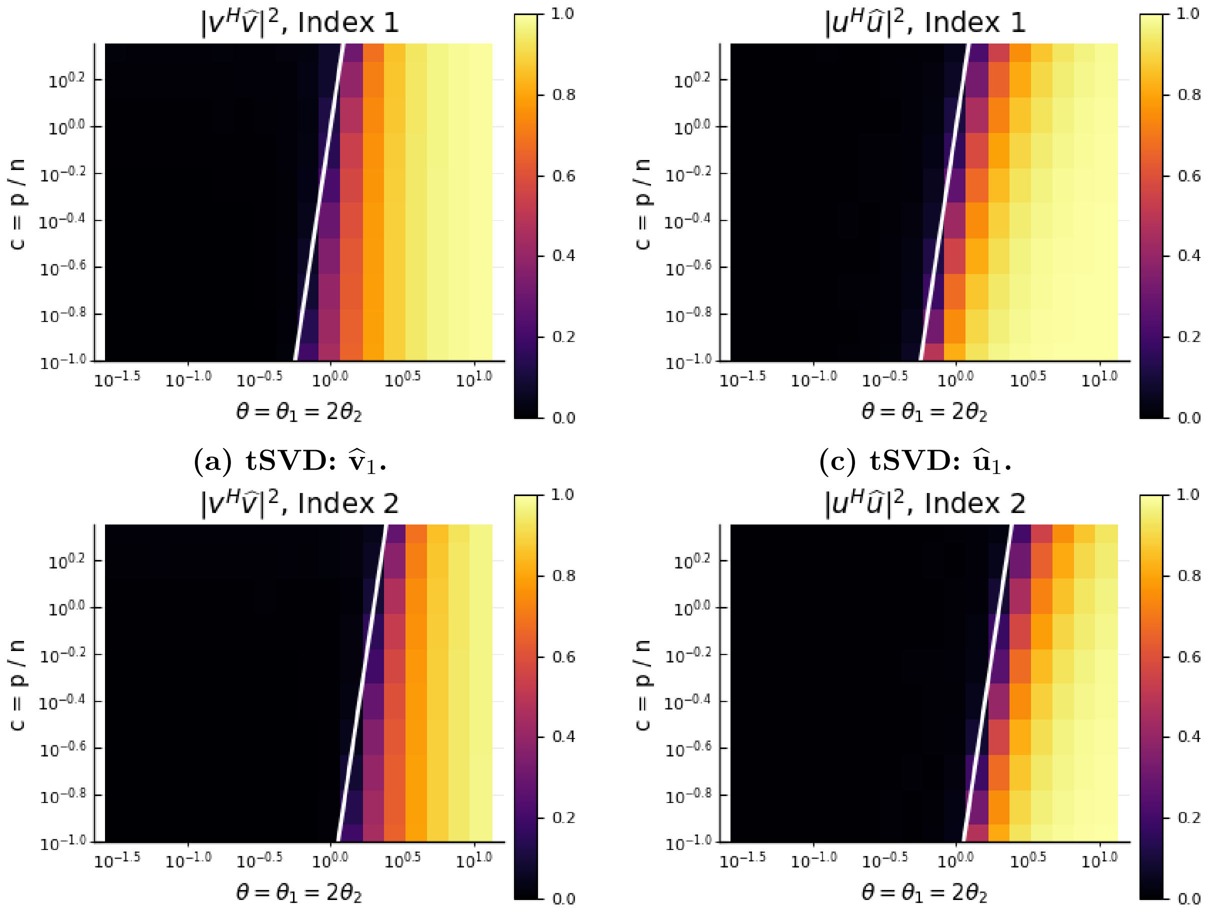
We first verify the noisy data cases, i.e., Theorem 3.2 and Corollary 3.1. Before proceeding, we present a verification of Theorem 3.1 in Figure 3.1. We then verify the performance of the tSVD-DMD algorithm in Figure 3.2, and compare it with DMD in Figure 3.3. We see that the phase transition is correctly predicted, and that DMD performs significantly worse that the tSVD-DMD algorithm. In Figure 3.4, we see that $\widehat{V}^H P \widehat{V}$ is asymptotically diagonal, as expected.

We next verify the results for missing data. We begin with a verification of Theorem 3.3 in Figures 3.5 and 3.6, and see that the truncated SVD behaves as predicted. We then verify the performance of the tSVD-DMD algorithm in Figure 3.7 and 3.8, and compare it with DMD in Figure 3.9 and 3.10. We see that the phase transition is correctly predicted, and that DMD performs significantly worse that the tSVD-DMD algorithm. In Figure 3.11, we see that $\widehat{V}^H P \widehat{V}$ is asymptotically diagonal, as expected.

3.7 Optimally Weighted DMD

Recall that for the orthogonal model, we consider

$$\widehat{A} = \widehat{X} P \widehat{X}^+ = \widehat{U} \widehat{\Theta} \left[\widehat{V}^H P \widehat{V}\right] \widehat{\Theta}^+ \widehat{U}^H, \tag{3.14}$$



(a) tSVD: $\widehat{\mathbf{v}}_1$.

(c) tSVD: $\widehat{\mathbf{u}}_1$.

(b) tSVD: $\widehat{\mathbf{v}}_2$.

(d) tSVD: $\widehat{\mathbf{u}}_2$.

Figure 3.1: We verify Theorem 3.1. The white line in each heatmap indicates the phase transition, as predicted in the theorem.

and the equivalent noise-free version is

$$A = XPX^{+} = U\Theta\left[V^{H}PV\right]\Theta^{+}U^{H} = U\mathrm{diag}\left(\gamma_i\right)U^{H}. \tag{3.15}$$

Moreover, we have shown that using the truncated SVD is equivalent (in the almost sure limit) to writing

$$\widehat{A} = \widehat{U}\mathrm{diag}\left(\alpha_{v_i}^2\gamma_i\right)\widehat{U}^{H}.$$

In the noisy setting, we are given $\widehat{U}$ instead of U: we may then ask the question, is there an alternative weighting of the $\widehat{\mathbf{u}}_i$ that yields a lower error? I.e., what is

$$\underset{W}{\operatorname{argmin}} \left\| \widehat{U} W \widehat{U}^H - U \operatorname{diag}(\gamma_i) U^H \right\|_F . \tag{3.16}$$

Manipulating this equation, we see that the optimal W is given by

$$\widehat{W} = \widehat{U}^H U \operatorname{diag}(\gamma_i) U^H \widehat{U}. \tag{3.17}$$

We will refer to this as the *optimal* value of W. Noting that the 'true' W is diagonal, we may go one step further and write down an *optimal approximation*

$$\widehat{W} = \operatorname{diag}\left(\widehat{U}^H U \operatorname{diag}(\gamma_i) U^H \widehat{U}\right). \tag{3.18}$$

Finally, using random matrix theory, we know that in the almost sure limit, we may write down the following predicted result for W

$$\widehat{W} = \operatorname{diag}\left(\alpha_{u_i}^2 \gamma_i\right), \tag{3.19}$$

and call it the *estimator* for the optimal W. Note that $\alpha_{u_i}^2$ and $\alpha_{v_i}^2$ are defined in Theorem 3.3. Of course, we do not have access to γ_i directly: however, note that the diagonals of $\widehat{V}^H P \widehat{V}$ are approximately $\alpha_{v_i}^2 \gamma_i$. Moreover, we do not have access to the $\alpha_{v_i}^2$ directly either, as these depend on knowledge of the θ_i. However, we may estimate the θ_i and $\alpha_{v_i}^2$ using [12, Thm. 2.8, 2.9]. It follows that we may find the optimal weights by rescaling these values, and that the estimator is completely data-driven.

We verify the power of this estimator, and plot the numerical error in estimating A and the DMD eigenvectors Q, and the weights used in our formulation. We fix p and rotate between varying one of n, θ, and the estimated rank $\widehat{r}$. We fix the rank $r = 2$ and use the orthogonal cosines model once again. In Figures 3.12, 3.13, 3.14, we see that our estimator outperforms the tSVD-DMD procedure. Moreover, we see that the behavior of our procedure is either that of a shrinkage estimator or the opposite (an expansion) depending on the relative values of p and n.

Note that in the limit, we have shown that $\widehat{V}^H P \widehat{V}$ is diagonal; in the finite sample regime, there will necessarily be some error or non-zero values in the off-diagonals. We repeat our simulations and compare the values of our estimators where we discard the off-diagonal entries of $\widehat{V}^H P\widehat{V}$, referred to as the oracle approximation. Additionally, we go further and replace the values of $\left|\widehat{\mathbf{u}}_i^H\,\mathbf{u}_i\right|^2$ (and similarly for the $\mathbf{v}_i$) with their theoretical limits (the $\alpha_{u_i}^2$), and refer to this as the 'RMT' version of the oracle estimators. In Figures 3.15, 3.16, 3.17, we see that replacing the oracle estimators with their approximation and RMT approximations does not lead to a notable change: all of the empirical values of the oracle estimator are very close to the approximations. The important takeaway here is that we are able to predict the behavior of the oracle estimator.

3.7.1 Extension to the non-orthogonal model

Here, we extend our previous result for the optimal W to the setting in which the latent signals and the mixing matrix are not orthogonal. I.e., $X = QC^H$ is a rank r matrix, with an SVD $X = U\Theta V^H$, but the columns of Q and C are not necessarily orthogonal. We still impose that the columns of Q are unit norm and that the columns of C are mutually 1-lag uncorrelated, with distinct, non-zero lag-1 autocorrelations γ_i. Recall that we consider

$$\widehat{A} = \widehat{X}P\widehat{X}^{+} = \widehat{U}\widehat{\Theta}\left[\widehat{V}^H P\widehat{V}\right]\widehat{\Theta}^{+}\widehat{U}^H. \tag{3.20}$$

Effectively, we are working with $\widehat{U}\widehat{K}\widehat{U}^H$, where

$$\widehat{K} = \widehat{\Theta}\left[\widehat{V}^H P\widehat{V}\right]\widehat{\Theta}^{+},$$

and in the noise-free setting we would have

$$K = \Theta\left[V^H P V\right]\Theta^{+}.$$

We may ask, is there a better matrix $\widehat{W}$ that would yield a lower error, i.e., what is

$$\underset{W}{\operatorname{argmin}}\left\|\widehat{U}W\widehat{U}^H - UKU^H\right\|_F. \tag{3.21}$$

We see that the optimal W is given by

$$\widehat{W} = \widehat{U}^H UKU^H\widehat{U}, \tag{3.22}$$

and refer to this as the *optimal* value of W. The design of an estimator of W is still an open question, and will be the subject of future work.

We repeat our simulations with all of the parameters the same, except we do not use an orthogonal model. In Figures 3.18, 3.19, 3.20, we see that our estimator outperforms the tSVD-DMD procedure. Moreover, we see that the behavior of our procedure is either that of a shrinkage estimator or the opposite (an expansion) depending on the relative values of p and n.

3.8 Conjectures about DMD in the Noise-only Setting

We state a conjecture about DMD in the noise-only setting.

Conjecture 3.1. *Let G be a $p \times n$ matrix with i.i.d. $\mathcal{N}\left(0, \frac{1}{n}\right)$ entries and let $c = \frac{p}{n-1}$. If we define $\widehat{A}$ as in (3.5) with G in place of X, let $\lambda_1, \lambda_2, \ldots, \lambda_p$ be the eigenvalues of $\widehat{A}$. If we write $\lambda_i = r_i \exp\left(i\omega_i\right)$, where $\omega_i \in [-\pi, \pi)$ and $r_i \geq 0$, we have that:*

1. *The limiting distribution of the ω_i is uniform on $[-\pi, \pi)$.*
2. *If $c \geq 1$, the limiting distribution of r_i has a density*

$$f(r) = \left(1 - \frac{1}{c}\right)\delta_1(r) + \frac{1}{c}\delta_p(r).$$

3. *If $c < 1$, the limiting distribution of r_i has a density*

$$f(r) = 2\left(\frac{1}{c} - 1\right)\frac{r}{\left[1 - r^2\right]^2}\mathbb{1}_{[0,\sqrt{c}]}(r).$$

Here, $\mathbb{1}_{[0,\sqrt{c}]}(r)$ denotes the indicator function of the interval $[0, \sqrt{c}]$ and $\delta_x(r)$ denotes the Dirac delta function centered at the value x.

Note that the density in the $c < 1$ case is exactly that of a product of truncated unitary matrices, as described in [1, Sec. 3.1.1, (3.12)].

3.8.1 Simulations

We present a few simulations to verify Conjecture 3.1. We fix $p = 500$ and vary n. We plot the empirical densities of the eigenvalues and overlay the theoretical predictions in Figure

3.21, and observe that our conjecture is numerically substantiated.

3.9 Some Expressions for DMD in the Rank-1 Setting

While the general analysis of DMD in the low-rank signal-plus-noise setting is still open, we are able to give a partial characterization in the rank-1 setting.

Here, our model is

$$X = \theta\, \mathbf{u}\, \mathbf{v}^H + G \in \mathbb{C}^{p \times n}, \tag{3.23}$$

where we assume that $\mathbf{v}^H\, P\, \mathbf{v} = \gamma \neq 0$, $\theta > 0$, and that $\mathbf{u}$ and $\mathbf{v}$ have unit ℓ_2 norm. Moreover, the entries of G are once again *i.i.d.* $\mathcal{N}(0, 1/n)$ random variables. We will denote the columns of G by $\mathbf{g}_i$ and the entries of $\mathbf{v}$ by v_i. P once again denotes the circular left-shift matrix, $\mathbf{v}_{(0)}$ will denote the vector composed of the first $n-1$ elements of $\mathbf{v}$, and $\mathbf{e}_i$ will denote the vector with a 1 in the i^{th} position and 0 elsewhere.

We have the following result for DMD in the rank-1 plus noise setting:

Theorem 3.5. *Given X as defined in (3.23), let $\widehat{A}$ be formed as in (3.5). Then, $\widehat{A}$ is a rank-4 perturbation of $\widetilde{G} = G_{(0)} P G_{(0)}^{+}$, where $G_{(0)}$ is defined analogously to $X_{(0)}$ in (3.5).*

Note that we conjecture that the eigenvalue distribution of $\widetilde{G}$ is as described in Conjecture 3.1, but our results do not depend on the conjecture. In the rest of this section, we prove this result. We also state the following conjecture about the behavior of higher rank signals plus noise:

Conjecture 3.2. *Let X be a rank r plus white noise signal matrix, analogous to (3.23), and let $\widehat{A}$ be formed as in (3.5). Then, $\widehat{A}$ is a rank-$\widetilde{r}$ perturbation of $\widetilde{G} = G_{(0)} P G_{(0)}^{+}$, where $G_{(0)}$ is defined analogously to $X_{(0)}$ in (3.5) and $\widetilde{r} = \min\{p, n-1, 3r+1\}$.*

3.9.1 Perturbation: $p \leq n - 1$

We begin with the case where $p \leq n - 1$.

Definitions

Assume that $p \leq n - 1$. Following [80, Theorem 3], let

$$\widetilde{\mathbf{u}} = G_{(0)}^{+}\, \mathbf{u} \text{ and } \widetilde{\mathbf{v}} = \left(I - G_{(0)}^{+} G_{(0)} \right) \mathbf{v}_{(0)},$$

so that we may define

$$\bar{\beta} = 1 + \theta \overline{\left[\mathbf{v}_{(0)}^H \, \widetilde{\mathbf{u}}\right]},$$

where the over-line denotes a complex conjugate; and

$$\widetilde{\sigma} = \theta^2 \|\widetilde{\mathbf{u}}\|_2^2 \|\widetilde{\mathbf{v}}\|_2^2 + |\beta|^2.$$

Note that with probability 1, $\mathbf{u}$ is in the range of $G_{(0)}$ and $\bar{\beta}$ is non-zero.

Pseudoinverse Perturbation

If we write

$$X_{(0)}^+ = G_{(0)}^+ + \Delta_G,$$

from applying the result of [80, Theorem 3], we have that

$$\Delta_G = \left\{ \frac{\theta}{\bar{\beta}} \left(1 - \frac{\theta^2 \|\widetilde{\mathbf{u}}\|_2^2 \|\widetilde{\mathbf{v}}\|_2^2}{\widetilde{\sigma}} \right) \widetilde{\mathbf{v}} \widetilde{\mathbf{u}}^H - \frac{\theta^2 \|\widetilde{\mathbf{u}}\|_2^2}{\widetilde{\sigma}} \widetilde{\mathbf{v}} \, \mathbf{v}_{(0)}^H - \frac{\theta^2 \|\widetilde{\mathbf{v}}\|_2^2}{\widetilde{\sigma}} \widetilde{\mathbf{u}} \widetilde{\mathbf{u}}^H - \frac{\theta \bar{\beta}}{\widetilde{\sigma}} \widetilde{\mathbf{u}} \, \mathbf{v}_{(0)}^H \right\} \times G_{(0)}^+.$$

For future use, we will define c_1 through c_4 and write

$$\Delta_G = \left\{ c_1 \widetilde{\mathbf{v}} \widetilde{\mathbf{u}}^H + c_2 \widetilde{\mathbf{v}} \, \mathbf{v}_{(0)}^H + c_3 \widetilde{\mathbf{u}} \widetilde{\mathbf{u}}^H + c_4 \widetilde{\mathbf{u}} \, \mathbf{v}_{(0)}^H \right\} \times G_{(0)}^+.$$

Perturbation of $G_{(0)} P G_{(0)}^+$

Note that

$$X_{(1)} = X_{(0)} P + \Delta,$$

where

$$\Delta = \left([\mathbf{g}_n - \mathbf{g}_1] + \theta \, [v_n - v_1] \, \mathbf{u} \right) \mathbf{e}_{n-1}^H = \boldsymbol{\delta} \, \mathbf{e}_{n-1}^H \, .$$

Then,

$$X_{(1)} X_{(0)}^+ = \left(\left[\theta \, \mathbf{u} \, \mathbf{v}_{(0)}^H + G_{(0)} \right] P + \Delta \right) \left(G_{(0)}^+ + \Delta_G \right),$$

which can be written as

$$X_{(1)} X_{(0)}^+ = G_{(0)} P G_{(0)}^+ + \theta \, \mathbf{u} \, \mathbf{v}_{(0)}^H \, P G_{(0)}^+ + \Delta G_{(0)}^+ + \left[\theta \, \mathbf{u} \, \mathbf{v}_{(0)}^H + G_{(0)} \right] P \Delta_G + \Delta \Delta_G.$$

We will (temporarily) ignore the first term, and the common $G_{(0)}^{+}$ term. Let c_1 through c_4 denote the coefficients of the outer products in Δ_G. We have:

$$\begin{aligned}\theta\,\mathbf{u}\,\mathbf{v}_{(0)}^{H}\,P + \boldsymbol{\delta}\,\mathbf{e}_{n-1}^{H} &+ X_{(0)}P\left[c_1\widetilde{\mathbf{v}} + c_3\widetilde{\mathbf{u}}\right]\widetilde{\mathbf{u}}^{H} + X_{(0)}P\left[c_2\widetilde{\mathbf{v}} + c_4\widetilde{\mathbf{u}}\right]\mathbf{v}_{(0)}^{H} \\ &+ \boldsymbol{\delta}\left[c_1\widetilde{\mathbf{v}}_{n-1} + c_3\widetilde{\mathbf{u}}_{n-1}\right]\widetilde{\mathbf{u}}^{H} + \boldsymbol{\delta}\left[c_2\widetilde{\mathbf{v}}_{n-1} + c_4\widetilde{\mathbf{u}}_{n-1}\right]\mathbf{v}_{(0)}^{H}\,.\end{aligned} \tag{3.24}$$

A Rank-4 Perturbation

Let

$$W = \begin{bmatrix} X_{(0)}P\widetilde{\mathbf{v}} & X_{(0)}P\widetilde{\mathbf{u}} & \theta\,\mathbf{u} & \boldsymbol{\delta} \end{bmatrix},$$

$$Y = \begin{bmatrix} c_1 & c_3 & 0 & c_1\widetilde{v}_{n-1} + c_3\widetilde{u}_{n-1} \\ c_2 & c_4 & 0 & c_2\widetilde{v}_{n-1} + c_4\widetilde{u}_{n-1} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{H},$$

and

$$Z = \begin{bmatrix} \widetilde{\mathbf{u}} & \mathbf{v}_{(0)} & P^{H}\,\mathbf{v}_{(0)} & \mathbf{e}_{n-1} \end{bmatrix}.$$

Then, we have that

$$X_{(1)}X_{(0)}^{+} = G_{(0)}PG_{(0)}^{+} + WYZ^{H}G_{(0)}^{+}.$$

That is, $X_{(1)}X_{(0)}^{+}$ is a rank-4 perturbation of $G_{(0)}PG_{(0)}^{+}$.

Note that if $\theta = 0$, the perturbation is rank-1, and is equal to

$$\left[\mathbf{g}_n - \mathbf{g}_1\right]\mathbf{e}_{n-1}^{H}\,G_{(0)}^{+} = \boldsymbol{\delta}\,\mathbf{e}_{n-1}^{H}\,G_{(0)}^{+}.$$

3.9.2 Perturbation: $p \geq n-1$

We now finish with the case where $p \geq n-1$.

Definitions

Assume that $p \geq n-1$. Following [80, Theorem 5], if we define

$$\widetilde{\mathbf{u}} = \left[I - G_{(0)}G_{(0)}^{+}\right]\mathbf{u} \text{ and } \widetilde{\mathbf{v}} = \left[v_{(0)}^{H}G_{(0)}^{+}\right]^{H},$$

we have that

$$\widetilde{\sigma} = \theta^2 \|\widetilde{\mathbf{u}}\|_2^2 \|\widetilde{\mathbf{v}}\|_2^2 + |\bar{\beta}|^2,$$

and β exactly as before, with

$$\bar{\beta} = 1 + \theta \overline{\left[\mathbf{v}_{(0)}^H \, G_{(0)}^+ \, \mathbf{u}\right]}$$

in the current notation.

Note that with probability 1, $\mathbf{v}_{(0)}$ is in the range of $G_{(0)}^H$ and $\bar{\beta}$ is non-zero.

Pseudoinverse Perturbation

Similar to what we wrote previously, applying the result in [80, Theorem 5] yields that

$$\Delta_G = G_{(0)}^+ \times \left\{ \frac{\theta}{\bar{\beta}} \left[1 - \frac{\theta^2}{\widetilde{\sigma}} \|\widetilde{\mathbf{u}}\|_2^2 \|\widetilde{\mathbf{v}}\|_2^2\right] \widetilde{\mathbf{v}} \widetilde{\mathbf{u}}^H + -\frac{\theta^2}{\widetilde{\sigma}} \|\widetilde{\mathbf{u}}\|_2^2 \widetilde{\mathbf{v}} \widetilde{\mathbf{v}}^H + -\frac{\theta^2}{\widetilde{\sigma}} \|\widetilde{\mathbf{v}}\|_2^2 \, \mathbf{u} \, \widetilde{\mathbf{u}}^H + -\frac{\theta \bar{\beta}}{\widetilde{\sigma}} \, \mathbf{u} \, \widetilde{\mathbf{v}}^H \right\}.$$

Similarly, we may define c_1 through c_4 to be the coefficients of the outer products.

A Rank-4 Perturbation

We define

$$W = \left[\left(X_{(0)} P + \boldsymbol{\delta} \, \mathbf{e}_{n-1}^H\right) G_{(0)}^+ \widetilde{\mathbf{v}} \quad \left(X_{(0)} P + \boldsymbol{\delta} \, \mathbf{e}_{n-1}^H\right) G_{(0)}^+ \, \mathbf{u} \quad \theta \, \mathbf{u} \quad \boldsymbol{\delta} \right],$$

$$Y = \begin{bmatrix} c_1 & c_3 & 0 & 0 \\ c_2 & c_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^H ,$$

and

$$Z = \left[\widetilde{\mathbf{u}} \quad \widetilde{\mathbf{v}} \quad \left[\mathbf{v}_{(0)}^H \, P G_{(0)}^+\right]^H \quad \left[\mathbf{e}_{n-1}^H \, G_{(0)}^+\right]^H \right],$$

so that

$$X_{(1)} X_{(0)}^+ = G_{(0)} P G_{(0)}^+ + W Y Z^H.$$

Once again, $X_{(1)} X_{(0)}^+$ is a rank-4 perturbation of $G_{(0)} P G_{(0)}^+$.

Note that we require $n \geq 2$; before, we required $p \geq 1$ and $n \geq 2$. When $\min\{p, n-1\} < 4$, the rank of $W Y Z^H$ is obviously smaller than 4.

Once again, we have that if $\theta = 0$, the perturbation is rank-1, and is equal to

$$[\mathbf{g}_n - \mathbf{g}_1]\,\mathbf{e}_{n-1}^H\,G_{(0)}^{+} = \boldsymbol{\delta}\,\mathbf{e}_{n-1}^H\,G_{(0)}^{+}.$$

3.10 Proof of Theorem 3.2 (and Theorem 3.4)

Note that proving that $\widehat{V}^H P \widehat{V}$ is diagonal yields that the diagonal entries of $\widehat{V}^H P \widehat{V}$ are the eigenvalues for the eigenvectors $\widehat{\mathbf{u}}_i$. I.e., item 3 is a simple consequence of items 1 and 2 in Theorem 3.2. Our proof will proceed by first characterizing $\widehat{\mathbf{v}}_i$ and then writing out the full form of $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j$. We will then note that the expression for $\widehat{\mathbf{v}}_i$ involves a resolvent, so that $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j$ involves a product of resolvents: we will analyze this product. Then, we will combine all of our results, make simplifications, and show that the matrix $\widehat{V}^H P \widehat{V}$ is asymptotically diagonal.

3.10.1 The form of $\widehat{\mathbf{v}}_i$

Let

$$R(z) = \left(G^H G - z\,\mathrm{I}_n\right)^{-1}. \tag{3.25}$$

From [12, Lem. 5.1, Proof], we have that

$$\widehat{\mathbf{v}}_i \propto R\left(\widehat{\theta}_i^2\right)\left[\widehat{\theta}_i Y^H \widehat{\mathbf{u}}_i + G^H Y \widehat{\mathbf{v}}_i\right]. \tag{3.26}$$

Hence, we may write

$$\widehat{\mathbf{v}}_i = \frac{R\left(\widehat{\theta}_i^2\right)\left[\widehat{\theta}_i Y^H \widehat{\mathbf{u}}_i + G^H Y \widehat{\mathbf{v}}_i\right]}{\left[\widehat{\theta}_i^2 \widehat{\mathbf{u}}_i^H Y R\left(\widehat{\theta}_i^2\right)^2 Y^H \widehat{\mathbf{u}}_i + \widehat{\mathbf{v}}_i^H Y^H G R\left(\widehat{\theta}_i^2\right)^2 G^H Y \widehat{\mathbf{v}}_i + 2\widehat{\theta}_i \mathrm{Re}\left\{\widehat{\mathbf{u}}_i^H Y R\left(\widehat{\theta}_i^2\right)^2 G^H Y \widehat{\mathbf{v}}_i\right\}\right]^{1/2}}. \tag{3.27}$$

3.10.2 The form of $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j$

We consider the general form of $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j$. We first consider the numerator of the product:

$$\begin{aligned} a_n &= \widehat{\theta}_i \widehat{\theta}_j \widehat{\mathbf{u}}_i^H Y \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] Y^H \widehat{\mathbf{u}}_j \\ &= \widehat{\theta}_i \widehat{\theta}_j \sum_{l,k} \theta_l \theta_k \left(\mathbf{u}_l^H \, \widehat{\mathbf{u}}_i\right) \left(\widehat{\mathbf{u}}_j^H \, \mathbf{u}_k\right) \mathbf{v}_k^H \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] \mathbf{v}_l, \end{aligned} \tag{3.28a}$$

$$\begin{aligned} b_n &= \widehat{\theta}_i \widehat{\mathbf{u}}_i^H Y \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] G^H Y \widehat{\mathbf{v}}_j \\ &= \widehat{\theta}_i \sum_{l,k} \theta_l \theta_k \left(\widehat{\mathbf{u}}_i^H \, \mathbf{u}_k\right) \left(\mathbf{v}_l^H \, \widehat{\mathbf{v}}_j\right) \mathbf{v}_k^H \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] G^H \, \mathbf{u}_l, \end{aligned} \tag{3.28b}$$

$$\begin{aligned} c_n &= \widehat{\theta}_j \widehat{\mathbf{v}}_i^H Y^H G \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] Y^H \widehat{\mathbf{u}}_j \\ &= \widehat{\theta}_j \sum_{l,k} \theta_l \theta_k \left(\widehat{\mathbf{v}}_i^H \, \mathbf{v}_k\right) \left(\mathbf{u}_l^H \, \widehat{\mathbf{u}}_j\right) \mathbf{u}_k^H \, G \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] \mathbf{v}_l, \end{aligned} \tag{3.28c}$$

$$\begin{aligned} d_n &= \widehat{\mathbf{v}}_i^H Y^H G \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] G^H Y \widehat{\mathbf{v}}_j \\ &= \sum_{l,k} \theta_l \theta_k \left(\mathbf{v}_k^H \, \widehat{\mathbf{v}}_i\right) \left(\widehat{\mathbf{v}}_j^H \, \mathbf{v}_l\right) \mathbf{u}_k^H \, G \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] G^H \, \mathbf{u}_l \, . \end{aligned} \tag{3.28d}$$

Next, we consider the denominator, noting that we have two sets of variables (for $\mathbf{v}_i$ and $\mathbf{v}_j$):

$$\begin{aligned} a_{d,i} &= \widehat{\theta}_i^2 \widehat{\mathbf{u}}_i^H Y R\left(\widehat{\theta}_i^2\right)^2 Y^H \widehat{\mathbf{u}}_i \\ &= \widehat{\theta}_i^2 \sum_{k,l} \theta_k \theta_l \left(\mathbf{u}_l^H \, \widehat{\mathbf{u}}_i\right) \left(\widehat{\mathbf{u}}_i^H \, \mathbf{u}_k\right) \mathbf{v}_k^H \, R\left(\widehat{\theta}_i^2\right)^2 \mathbf{v}_l, \end{aligned} \tag{3.29a}$$

$$\begin{aligned} b_{d,i} &= \widehat{\mathbf{v}}_i^H Y^H G R\left(\widehat{\theta}_i^2\right)^2 G^H Y \widehat{\mathbf{v}}_i \\ &= \sum_{k,l} \theta_k \theta_l \left(\mathbf{v}_l^H \, \widehat{\mathbf{v}}_i\right) \left(\widehat{\mathbf{v}}_i^H \, \mathbf{v}_k\right) \mathbf{u}_k^H \, G R\left(\widehat{\theta}_i^2\right)^2 G^H \, \mathbf{u}_l, \end{aligned} \tag{3.29b}$$

$$
\begin{aligned}
c_{d,i} &= 2\widehat{\theta}_i \mathrm{Re}\left\{ \widehat{\mathbf{u}}_i^H Y R\left(\widehat{\theta}_i^2\right)^2 G^H Y \widehat{\mathbf{v}}_i \right\} \\
&= 2\widehat{\theta}_i \mathrm{Re}\left\{ \sum_{k,l} \theta_k \theta_l \left(\widehat{\mathbf{u}}_i^H \mathbf{u}_k\right) \left(\mathbf{v}_l^H \widehat{\mathbf{v}}_i\right) \mathbf{v}_k^H R\left(\widehat{\theta}_i^2\right)^2 G^H \mathbf{u}_l \right\}.
\end{aligned} \tag{3.29c}
$$

Putting everything together, we find that

$$
\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j = \frac{a_n + b_n + c_n + d_n}{\sqrt{a_{d,i} + b_{d,i} + c_{d,i}}\sqrt{a_{d,j} + b_{d,j} + c_{d,j}}}. \tag{3.30}
$$

3.10.3 The entries of $R(z)PR(w)$

Note that for any unit vectors $\mathbf{a}$ and $\mathbf{b} \in \mathbb{C}^n$, if $m_\phi(z)$ is the Stieltjes transform of the Marcenko-Pastur law and the imaginary part of z is denoted by η, we have that

$$
\left|\mathbf{a}^H R(z)\, \mathbf{b} - m_\phi(z)\, \mathbf{a}^H \mathbf{b}\right| \prec \Psi(z) = \sqrt{\frac{\operatorname{Im} m_\phi(z)}{n\eta}} + \frac{1}{n\eta}, \tag{3.31}
$$

where the symbol $\prec$ denotes stochastic domination [18, Thm. 2.4]. We want to bound an expression of the form

$$
\left|\mathbf{a}^H R(z) P R(w)\, \mathbf{b} - m_\phi(z) m_\phi(w)\, \mathbf{a}^H P\, \mathbf{b}\right|,
$$

where z and w may be different. Assume that $\mathbf{a}$ has entries a_i and $\mathbf{b}$ has entries b_i, and that all subscripts are taken modulo the number of coordinates n (e.g., a_0 is a_n). For notational clarity, we will define $Z = R(z)$ and $W = R(w)$. We may then write:

$$
\mathbf{a}^H R(z) P R(w)\, \mathbf{b} = \sum_{l,k} a_l b_k \sum_q Z_{lq} W_{q-1,k}. \tag{3.32}
$$

Expanding this summation, we may write

$$
\begin{aligned}
\mathbf{a}^H R(z) P R(w)\, \mathbf{b} &= \sum_l a_l b_{l-1} Z_{ll} W_{l-1,l-1} + \sum_l a_l b_{l-1} Z_{l,l-1} W_{l-1,l-1} \\
&\quad + \sum_l \sum_{q: q \neq l, l-1} a_l b_{l-1} Z_{lq} W_{q-1,l-1} + \sum_{l,k \neq l-1} a_l b_k Z_{ll} W_{l-1,k} \\
&\quad + \sum_{l,k \neq l-1} a_l b_k Z_{l,k+1} W_{kk} + \sum_{l,k \neq l-1} \sum_{q: q \neq l, k+1} a_l b_k Z_{lq} W_{q-1,k}.
\end{aligned} \tag{3.33}
$$

Notice that only the first summation has no off-diagonal entries of Z and W: this is the term that we care about, and our goal is to show that the remaining terms are bounded.

To bound the remaining terms, we will point to the derivation in [18, Sec. 5]. In particular, we note that expanding the off-diagonal entries of Z and W using the various resolvent identities (see [18, Lem. 3.8]) enables us to find products of the $(i.i.d)$ entries of G, G_{ij} inside each term of the summations. We are in the same setting as [18, Sec. 5], and it is hence an immediate consequence that the size of the remaining five terms is stochastically dominated by $\Psi(z)\Psi(w)$.

To bound the diagonal terms, we note that individually we have

$$|Z_{ll} - m_\phi(z)| \prec \Psi(z) \text{ and } |W_{ll} - m_\phi(z)| \prec \Psi(w)$$

uniformly in z, w, and the index l. Using [18, Lem. 3.1, 3.2], it follows that

$$\left| \sum_l a_l b_{l-1} Z_{ll} W_{l-1,l-1} - \sum_l a_l b_{l-1} m_\phi(z) m_\phi(w) \right| \prec \Psi(z)\Psi(w).$$

We have proven that

$$\left| \mathbf{a}^H R(z) P R(w) \, \mathbf{b} - m_\phi(z) m_\phi(w) \, \mathbf{a}^H P \, \mathbf{b} \right| \prec \Psi(z)\Psi(w). \tag{3.34}$$

3.10.4 Putting things together

Following [12, Lemma 4.1, Proof], we immediately conclude that b_n, c_n, and $c_{d,i}$ and $c_{d,j} \to 0$ almost surely. Moreover, [12, Eqn. (17)] indicates that we may reduce the summations in (3.28) to only terms such that $\theta_l = \theta_i$ and $\theta_k = \theta_j$, and to $\theta_k = \theta_l = \theta_i$ in (3.29). However, by assumption, the θ_i are distinct, so that the summations in the surviving terms collapse. Hence, we may write

$$\begin{aligned} a_n &= \widehat{\theta}_i \widehat{\theta}_j \widehat{\mathbf{u}}_i^H Y \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] Y^H \widehat{\mathbf{u}}_j \\ &= \widehat{\theta}_i \widehat{\theta}_j \theta_i \theta_j \left(\mathbf{u}_i^H \, \widehat{\mathbf{u}}_i \right) \left(\widehat{\mathbf{u}}_j^H \, \mathbf{u}_j \right) \mathbf{v}_i^H \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right) \right] \mathbf{v}_j + o(1), \end{aligned} \tag{3.35a}$$

$$\begin{aligned} d_n &= \widehat{\mathbf{v}}_i^H Y^H G \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right)\right] G^H Y \widehat{\mathbf{v}}_j \\ &= \theta_i \theta_j \left(\mathbf{v}_i^H \, \widehat{\mathbf{v}}_i\right) \left(\widehat{\mathbf{v}}_j^H \, \mathbf{v}_j\right) \mathbf{u}_i^H \, G \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right)\right] G^H \, \mathbf{u}_j + o(1) \end{aligned} \tag{3.35b}$$

$$\begin{aligned} a_{d,i} &= \widehat{\theta}_i^2 \widehat{\mathbf{u}}_i^H Y R\left(\widehat{\theta}_i^2\right)^2 Y^H \widehat{\mathbf{u}}_i \\ &= \widehat{\theta}_i^2 \theta_i^2 \left|\mathbf{u}_i^H \, \widehat{\mathbf{u}}_i\right|^2 \mathbf{v}_i^H \, R\left(\widehat{\theta}_i^2\right)^2 \mathbf{v}_i + o(1), \end{aligned} \tag{3.35c}$$

$$\begin{aligned} b_{d,i} &= \widehat{\mathbf{v}}_i^H Y^H G R\left(\widehat{\theta}_i^2\right)^2 G^H Y \widehat{\mathbf{v}}_i \\ &= \theta_i^2 \left|\mathbf{v}_i^H \, \widehat{\mathbf{v}}_i\right|^2 \mathbf{u}_i^H \, G R\left(\widehat{\theta}_i^2\right)^2 G^H \, \mathbf{u}_i + o(1). \end{aligned} \tag{3.35d}$$

We have characterized the behavior of $\mathbf{v}_i^H \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right)\right] \mathbf{v}_j$ and shown that it concentrates around

$$m_\phi\left(\widehat{\theta}_i^2\right) m_\phi\left(\widehat{\theta}_j^2\right) \mathbf{v}_i^H \, P \, \mathbf{v}_j \, .$$

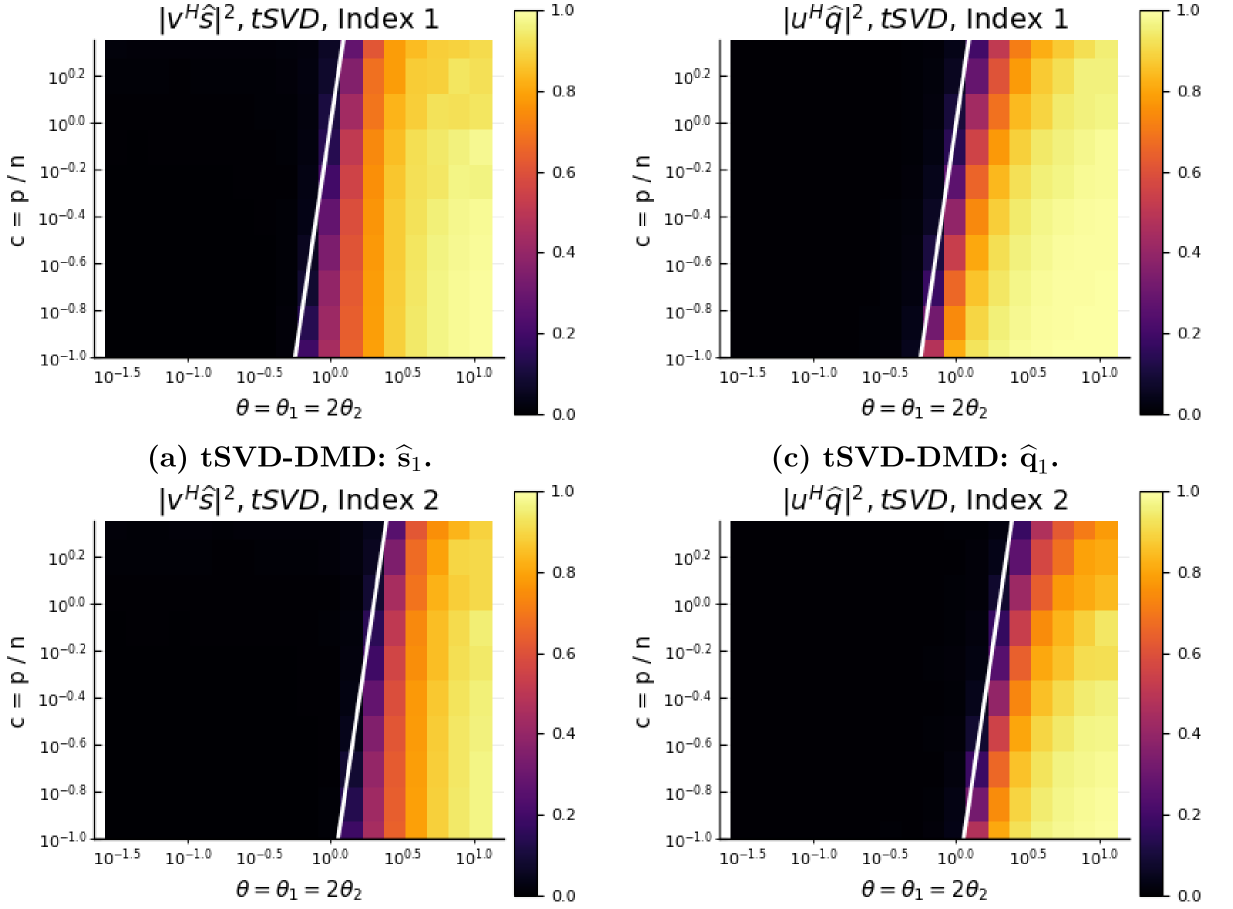
However, by assumption, $\mathbf{v}_i^H \, P \, \mathbf{v}_j$ has limit zero, so that unless $i = j$, a_n vanishes. The randomness of the $\mathbf{u}_i$ means that $\mathbf{u}_i^H \, G \left[R\left(\widehat{\theta}_i^2\right)^H P R\left(\widehat{\theta}_j^2\right)\right] G^H \, \mathbf{u}_j$ vanishes for all i and j, so that d_n vanishes. Hence, we have that $\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_j \to 0$ for $i \neq j$, and

$$\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_i = \frac{\widehat{\theta}_i^2 \theta_i^2 \left|\mathbf{u}_i^H \, \widehat{\mathbf{u}}_i\right|^2 m_\phi\left(\widehat{\theta}_i^2\right)^2 \gamma_i}{\widehat{\theta}_i^2 \theta_i^2 \left|\mathbf{u}_i^H \, \widehat{\mathbf{u}}_i\right|^2 \mathbf{v}_i^H \, R\left(\widehat{\theta}_i^2\right)^2 \mathbf{v}_i + \theta_i^2 \left|\mathbf{v}_i^H \, \widehat{\mathbf{v}}_i\right|^2 \mathbf{u}_i^H \, G R\left(\widehat{\theta}_i^2\right)^2 G^H \, \mathbf{u}_i} + o(1).$$

However, this is exactly equal to the expression in [12, Thm. 2.9] times γ_i, and we conclude that

$$\widehat{\mathbf{v}}_i^H P \widehat{\mathbf{v}}_i \to \alpha_{v_i}^2 \gamma_i$$

almost surely, as desired.



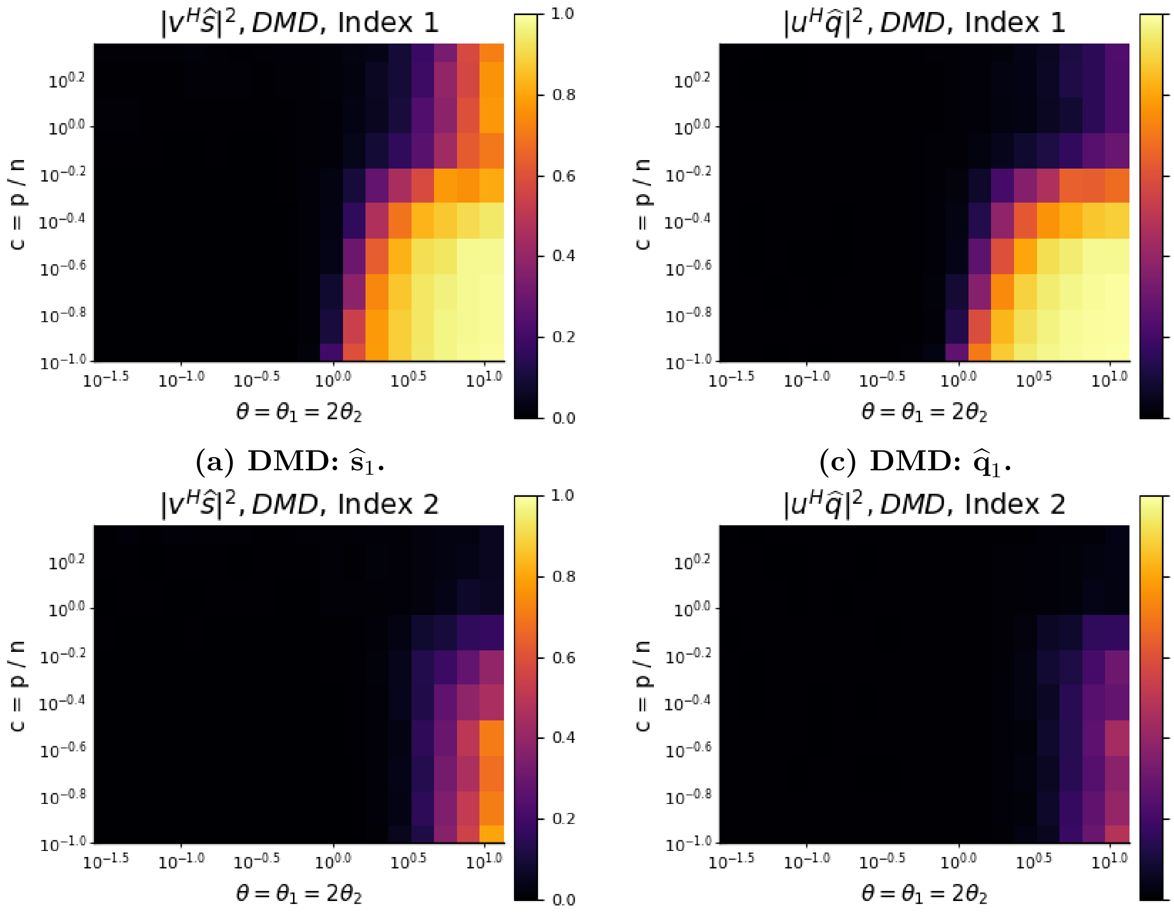
(a) tSVD-DMD: $\hat{\mathrm{s}}_1$.

(c) tSVD-DMD: $\hat{\mathrm{q}}_1$.

(b) tSVD-DMD: $\hat{\mathrm{s}}_2$.

(d) tSVD-DMD: $\hat{\mathrm{q}}_2$.

Figure 3.2: Here, we verify Theorem 3.2 by presenting the performance of the tSVD-DMD algorithm. We see that the estimates from the algorithm inherit the tSVD phase transition of Theorem 3.1, and that above the phase transition they are correct.



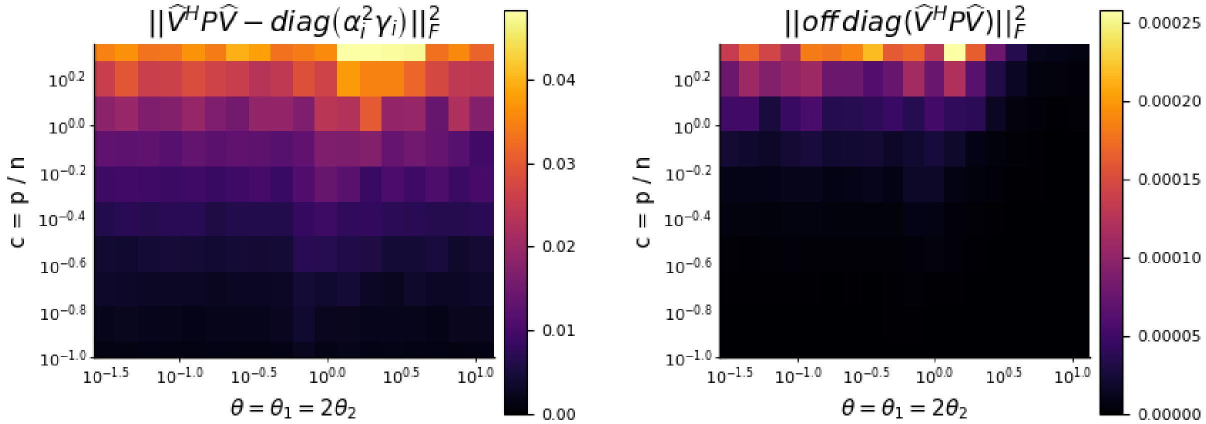
(a) DMD: $\widehat{s}_1$.

(c) DMD: $\widehat{q}_1$.

(b) DMD: $\widehat{s}_2$.

(d) DMD: $\widehat{q}_2$.

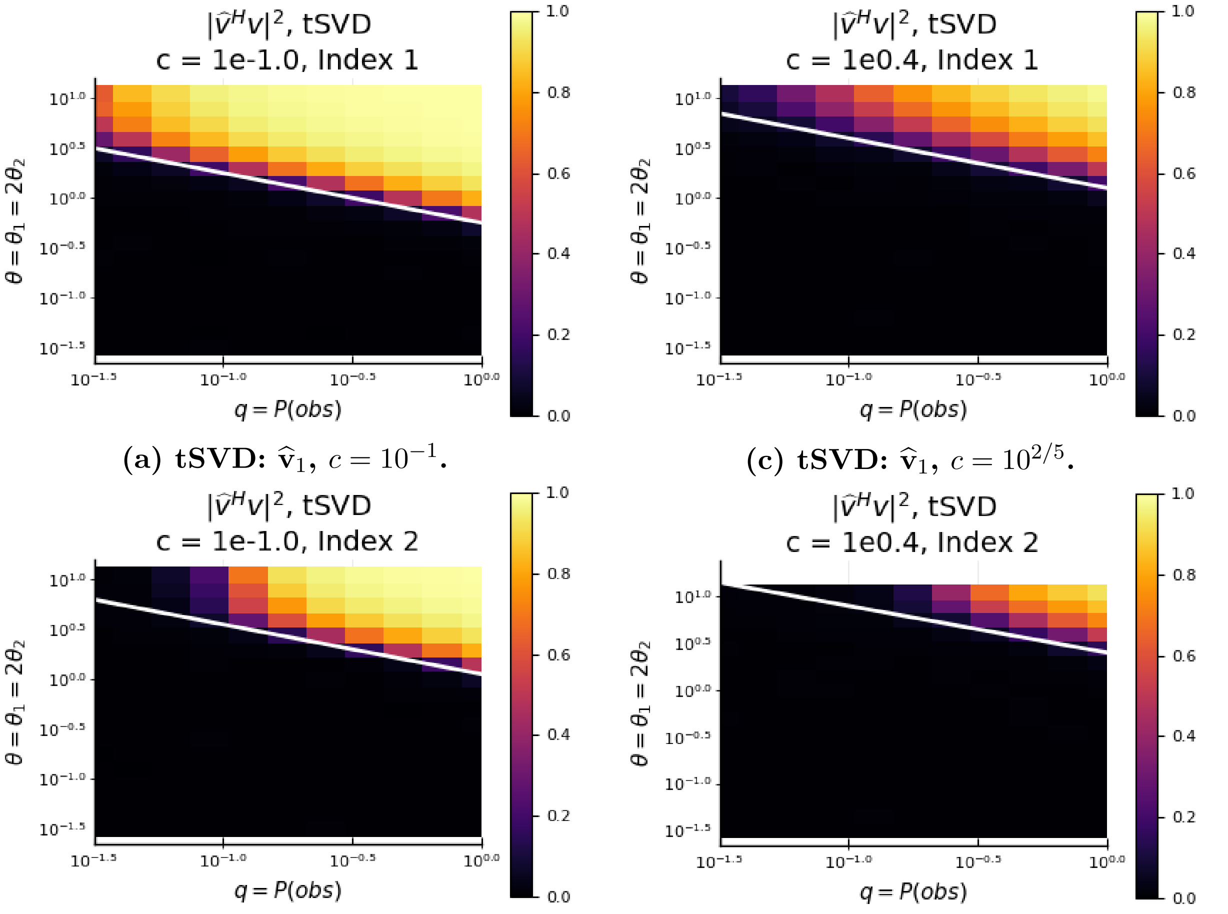
Figure 3.3: Here, we present performance results for the DMD algorithm on noisy data. We see that the performance is much worse than that of the tSVD-DMD algorithm (Figure 3.2).



(a) tSVD: Error in $\widehat{V}^H P \widehat{V}$.

(b) tSVD: Error in the off-diagonal entries of $\widehat{V}^H P \widehat{V}$.

Figure 3.4: We verify the results of Theorem 3.2 by plotting the error between the limiting diagonal matrix and $\widehat{V}^H P \widehat{V}$. We separate the total error and the error in the off-diagonals, and note that both are small.



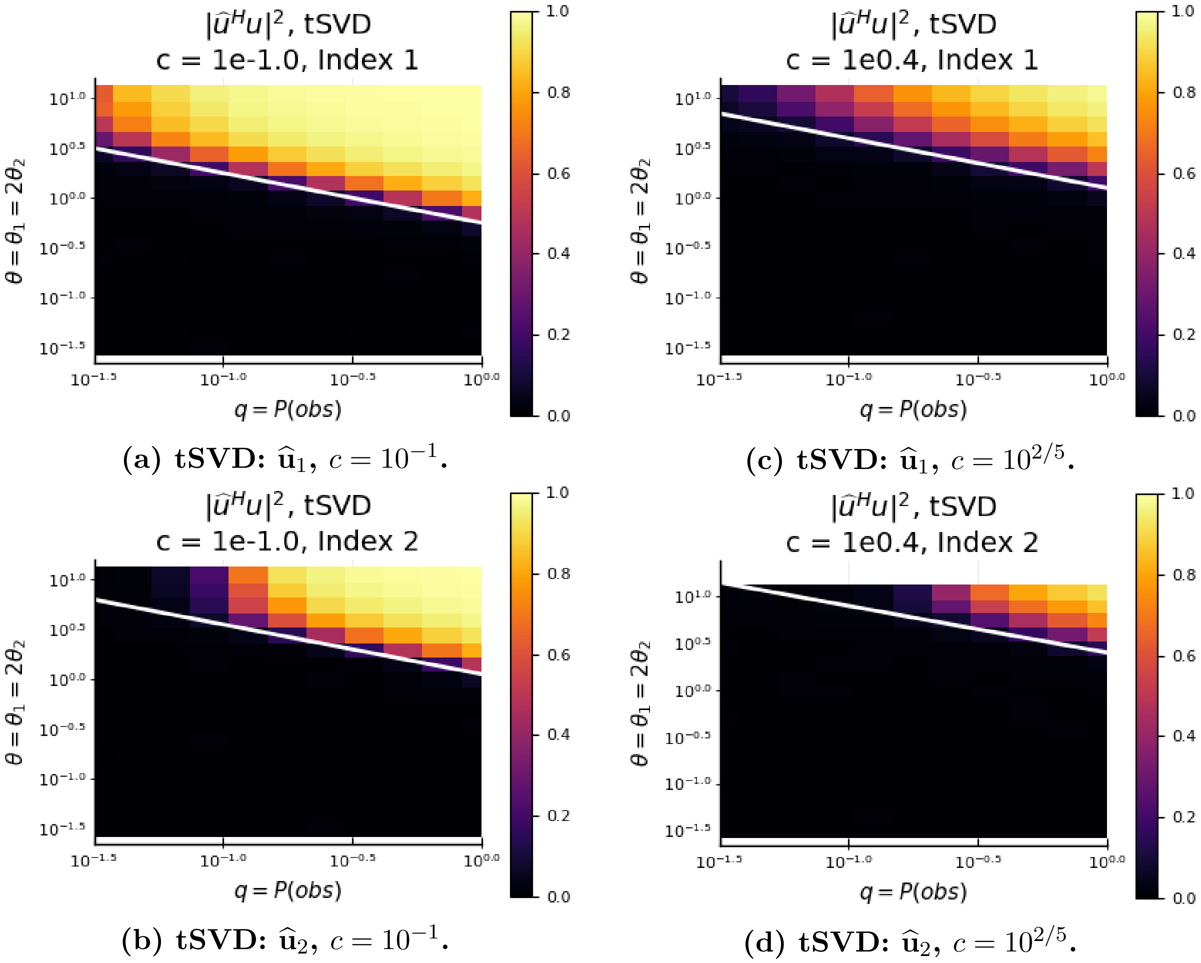
(a) **tSVD: $\widehat{\mathbf{v}}_1$, $c = 10^{-1}$.**

(c) **tSVD: $\widehat{\mathbf{v}}_1$, $c = 10^{2/5}$.**

(b) **tSVD: $\widehat{\mathbf{v}}_2$, $c = 10^{-1}$.**

(d) **tSVD: $\widehat{\mathbf{v}}_2$, $c = 10^{2/5}$.**

Figure 3.5: We verify Theorem 3.3. The white line in each heatmap indicates the phase transition, as predicted in the theorem.



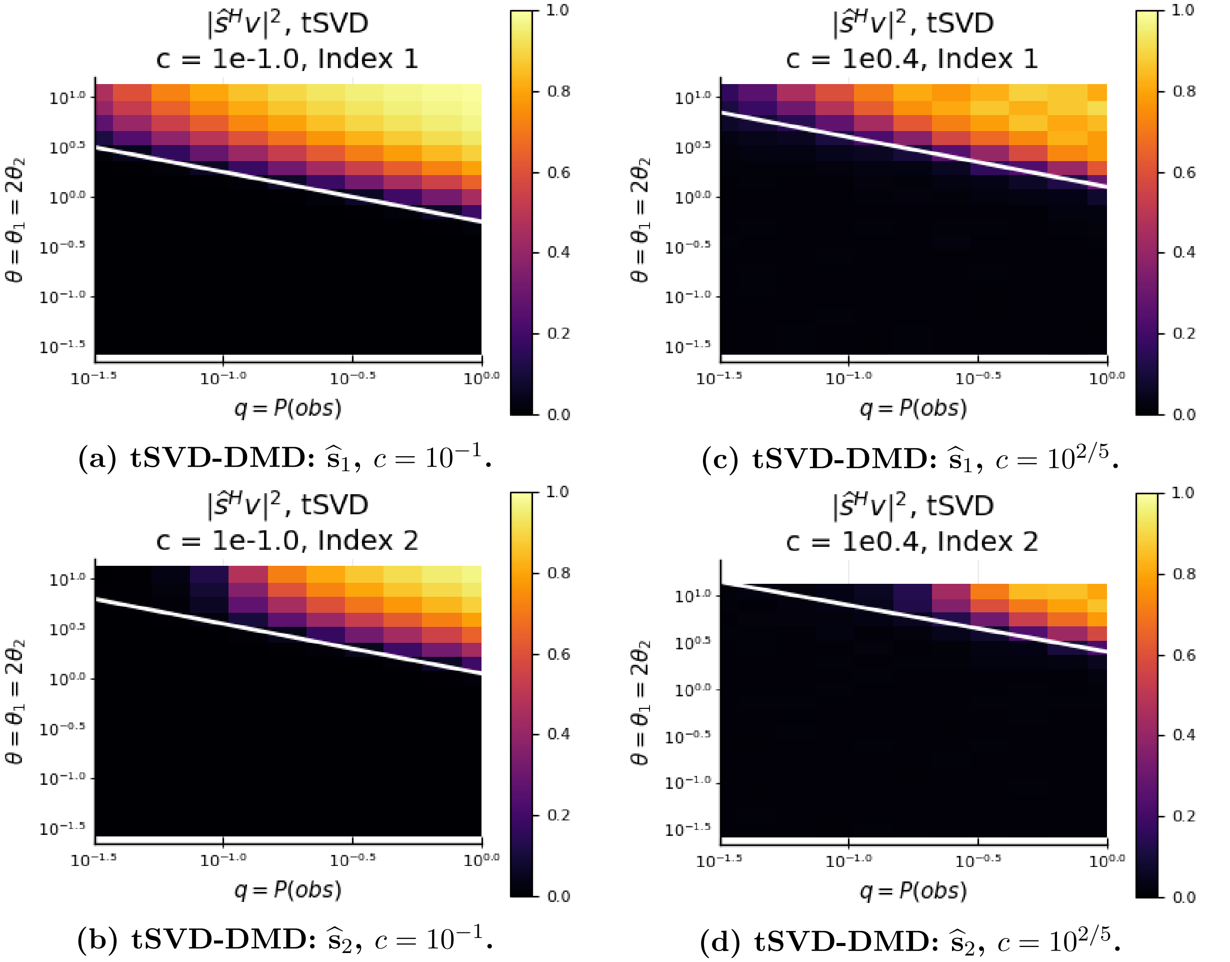
(a) **tSVD: $\hat{\mathbf{u}}_1$, $c = 10^{-1}$.**

(c) **tSVD: $\hat{\mathbf{u}}_1$, $c = 10^{2/5}$.**

(b) **tSVD: $\hat{\mathbf{u}}_2$, $c = 10^{-1}$.**

(d) **tSVD: $\hat{\mathbf{u}}_2$, $c = 10^{2/5}$.**

Figure 3.6: We verify Theorem 3.3. The white line in each heatmap indicates the phase transition, as predicted in the theorem.



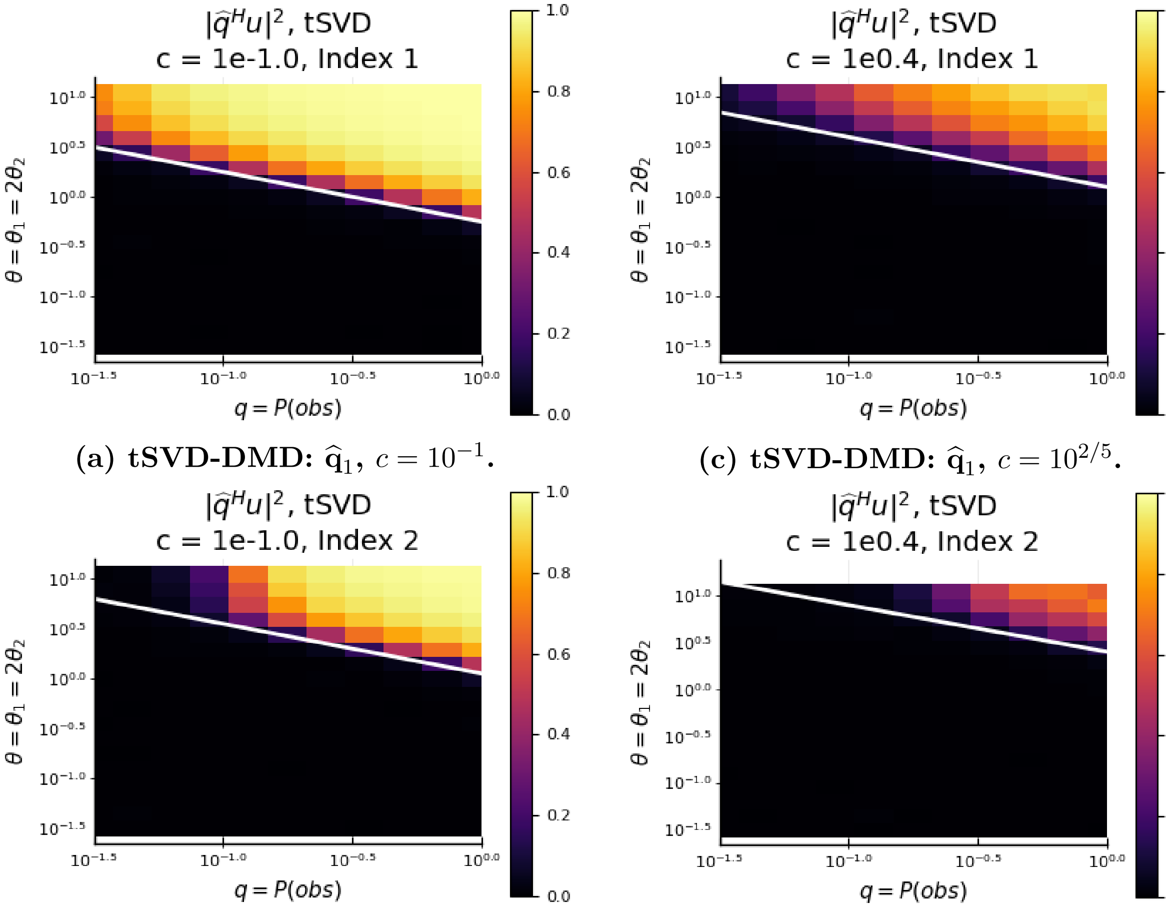
(a) **tSVD-DMD: $\widehat{\mathrm{s}}_1$, $c = 10^{-1}$.**

(c) **tSVD-DMD: $\widehat{\mathrm{s}}_1$, $c = 10^{2/5}$.**

(b) **tSVD-DMD: $\widehat{\mathrm{s}}_2$, $c = 10^{-1}$.**

(d) **tSVD-DMD: $\widehat{\mathrm{s}}_2$, $c = 10^{2/5}$.**

Figure 3.7: Here, we verify Theorem 3.4 by presenting the performance of the tSVD-DMD algorithm. We see that the estimates from the algorithm inherit the tSVD phase transition of Theorem 3.3, and that above the phase transition they are correct.



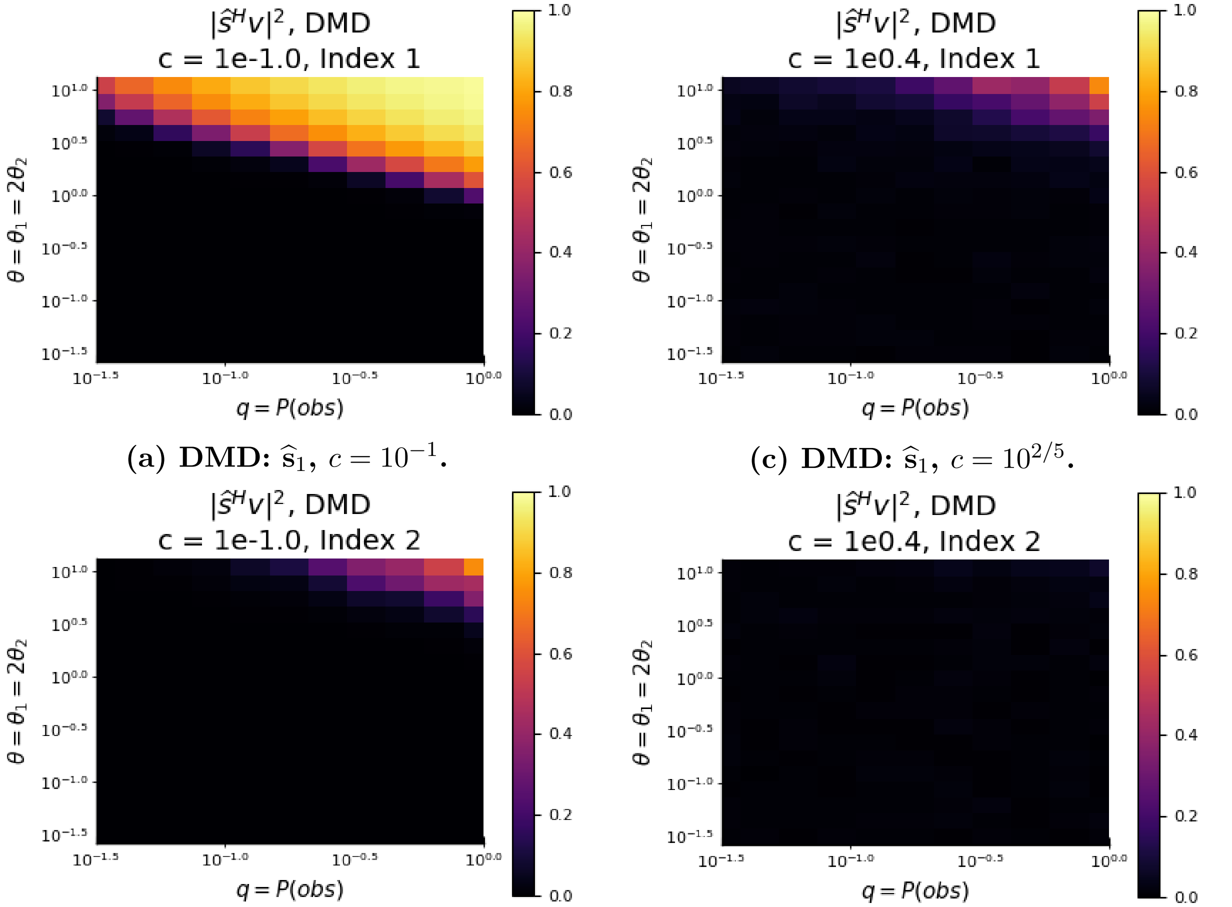
(a) **tSVD-DMD: $\widehat{\mathrm{q}}_1$, $c = 10^{-1}$.**

(c) **tSVD-DMD: $\widehat{\mathrm{q}}_1$, $c = 10^{2/5}$.**

(b) **tSVD-DMD: $\widehat{\mathrm{q}}_2$, $c = 10^{-1}$.**

(d) **tSVD-DMD: $\widehat{\mathrm{q}}_2$, $c = 10^{2/5}$.**

Figure 3.8: Here, we verify Theorem 3.4 by presenting the performance of the tSVD-DMD algorithm. We see that the estimates from the algorithm inherit the tSVD phase transition of Theorem 3.3, and that above the phase transition they are correct.



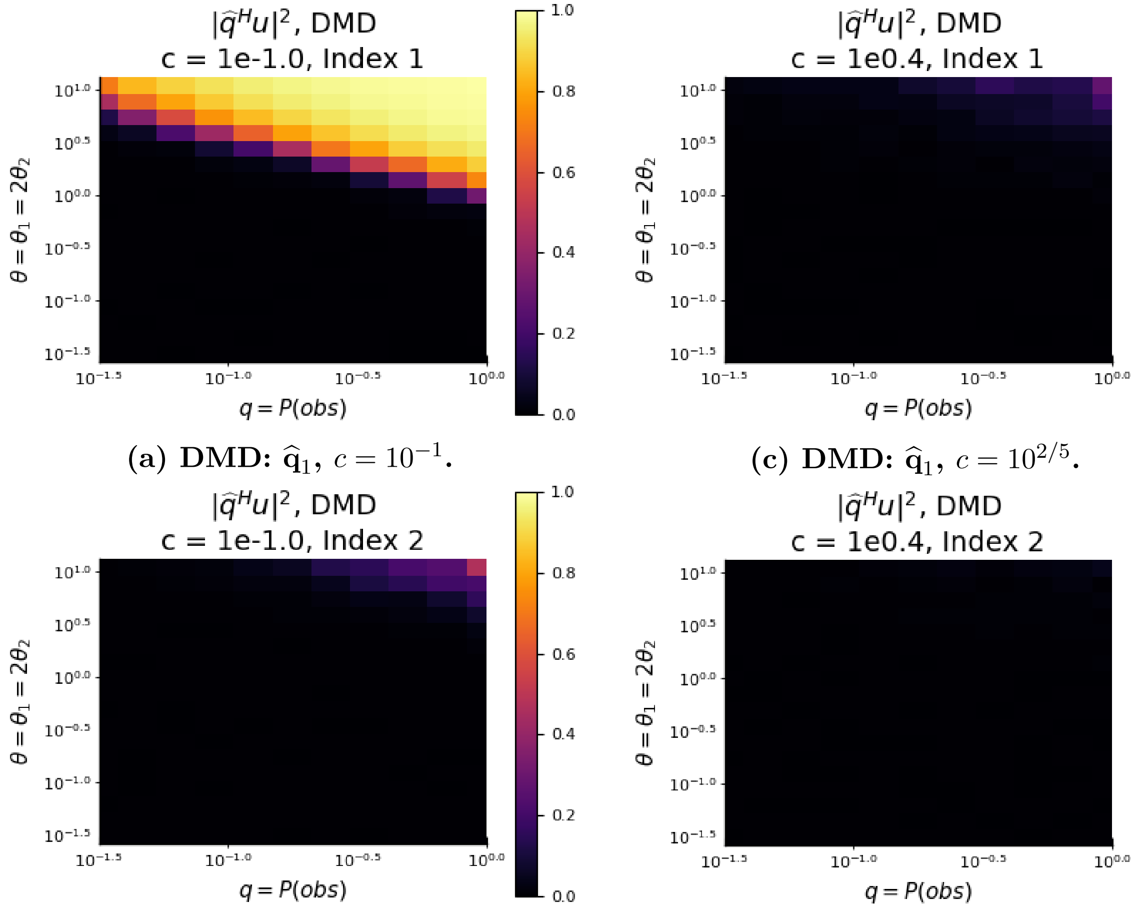
(a) **DMD:** $\hat{s}_1$, $c = 10^{-1}$.

(c) **DMD:** $\hat{s}_1$, $c = 10^{2/5}$.

(b) **DMD:** $\hat{s}_2$, $c = 10^{-1}$.

(d) **DMD:** $\hat{s}_2$, $c = 10^{2/5}$.

Figure 3.9: Here, we present performance results for the DMD algorithm on noisy data. We see that the performance is much worse than that of the tSVD-DMD algorithm (Figure 3.7).



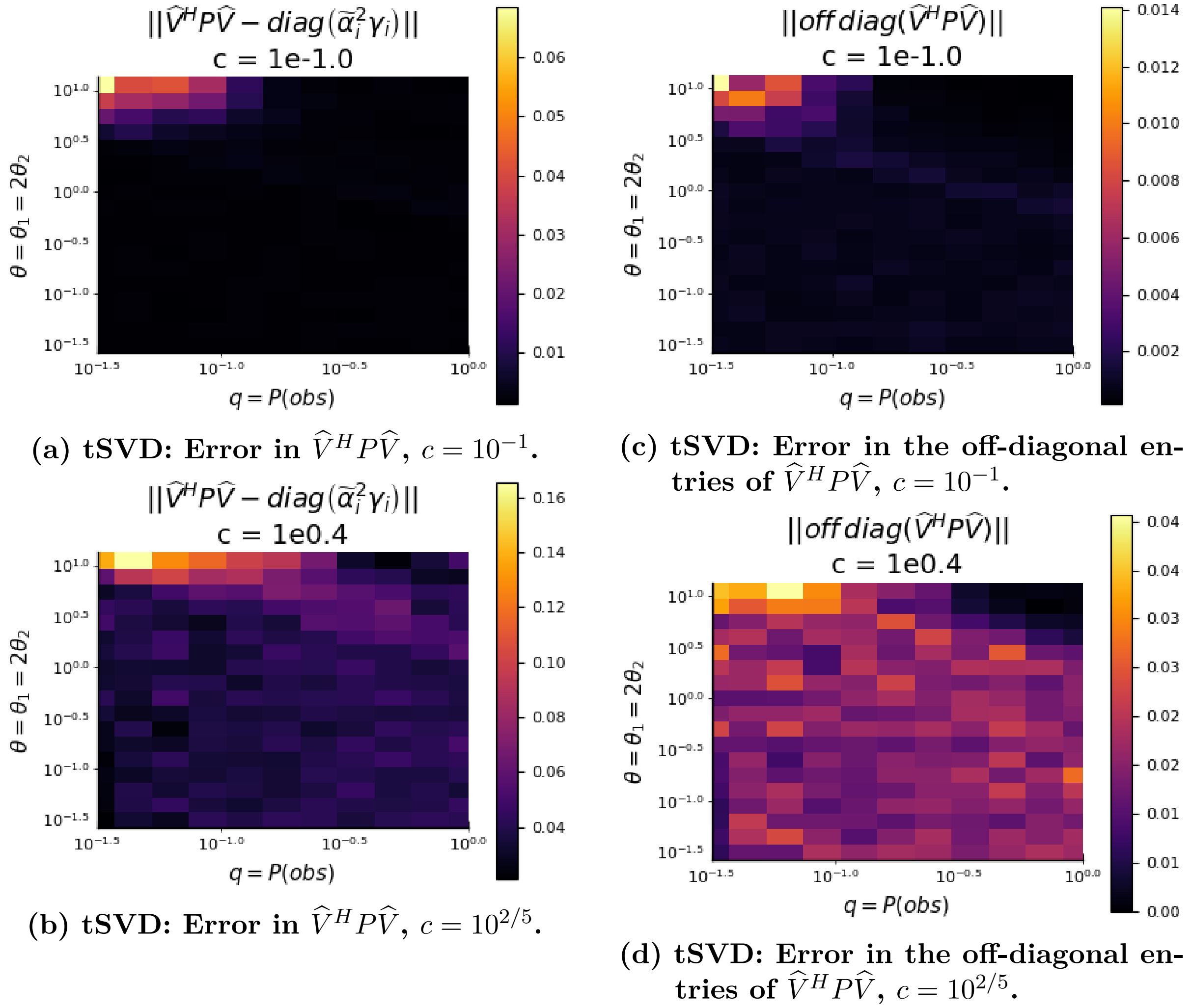
(a) **DMD: $\widehat{\mathrm{q}}_1$, $c = 10^{-1}$.**

(c) **DMD: $\widehat{\mathrm{q}}_1$, $c = 10^{2/5}$.**

(b) **DMD: $\widehat{\mathrm{q}}_2$, $c = 10^{-1}$.**

(d) **DMD: $\widehat{\mathrm{q}}_2$, $c = 10^{2/5}$.**

Figure 3.10: Here, we present performance results for the DMD algorithm on noisy data. We see that the performance is much worse than that of the tSVD-DMD algorithm (Figure 3.8).



(a) **tSVD: Error in $\widehat{V}^H P\widehat{V}$, $c = 10^{-1}$.**

(b) **tSVD: Error in $\widehat{V}^H P\widehat{V}$, $c = 10^{2/5}$.**

(c) **tSVD: Error in the off-diagonal entries of $\widehat{V}^H P\widehat{V}$, $c = 10^{-1}$.**

(d) **tSVD: Error in the off-diagonal entries of $\widehat{V}^H P\widehat{V}$, $c = 10^{2/5}$.**

Figure 3.11: We verify the results of Theorem 3.4 by plotting the error between the limiting diagonal matrix and $\widehat{V}^H P\widehat{V}$. We separate the total error and the error in the off-diagonals, and note that both are small.

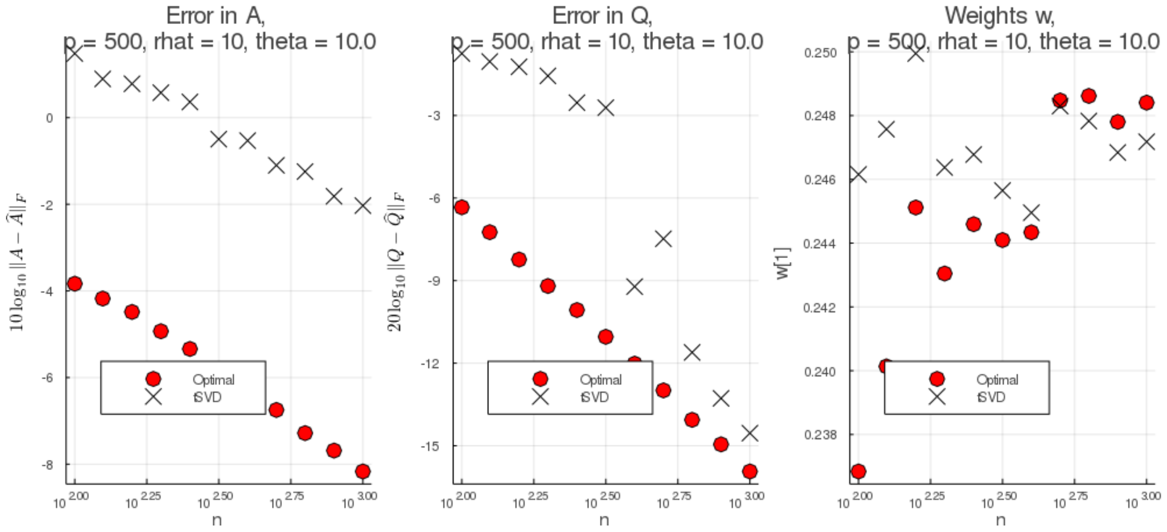


Figure 3.12: We study our Optimally Weighted DMD procedure, where we fix all parameters except the sample size n. We see that the estimation error of A and Q is lower with the reweighted DMD procedure than with the tSVD. Moreover, we see a transition in the behavior of the weights from shrinkage to growth as n changes.

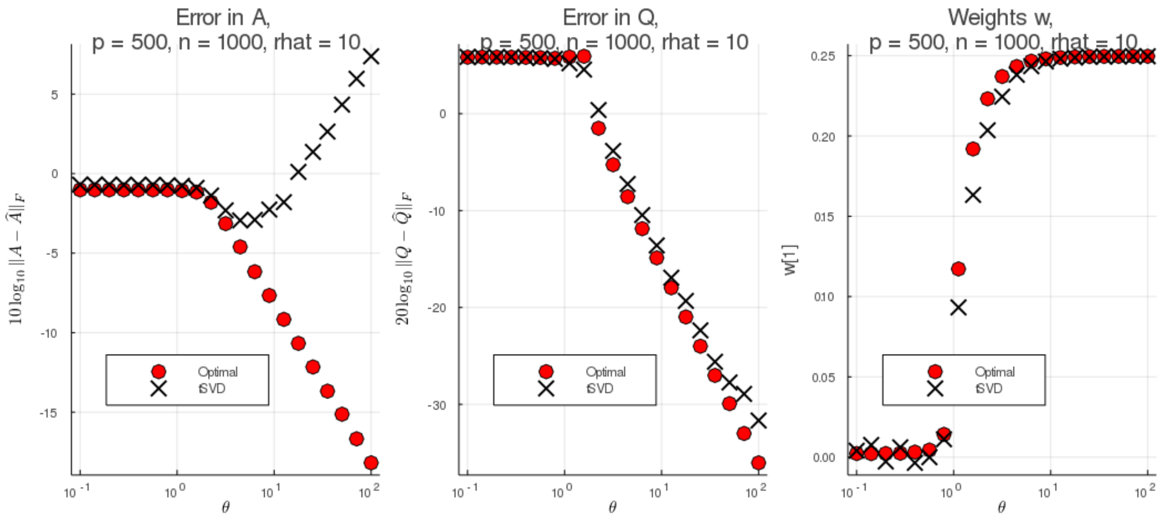


Figure 3.13: We study our Optimally Weighted DMD procedure, where we fix all parameters except the SNR θ. We see that the estimation error of A and Q is lower with the reweighted DMD procedure than with the tSVD.

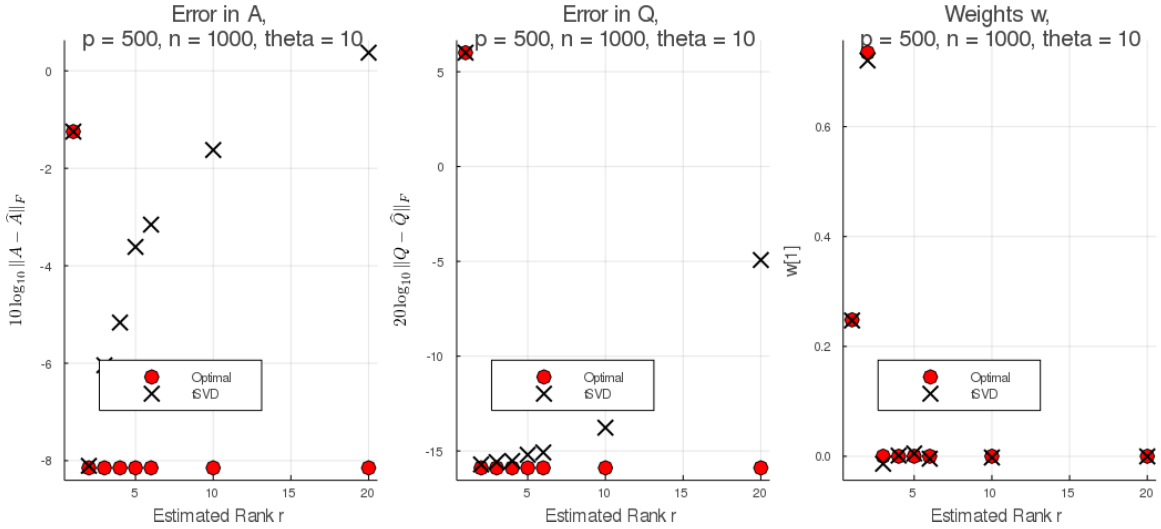


Figure 3.14: We study our Optimally Weighted DMD procedure, where we fix all parameters except the estimated rank $\widehat{r}$. We see that the estimation error of A and Q is lower with the reweighted DMD procedure than with the tSVD.

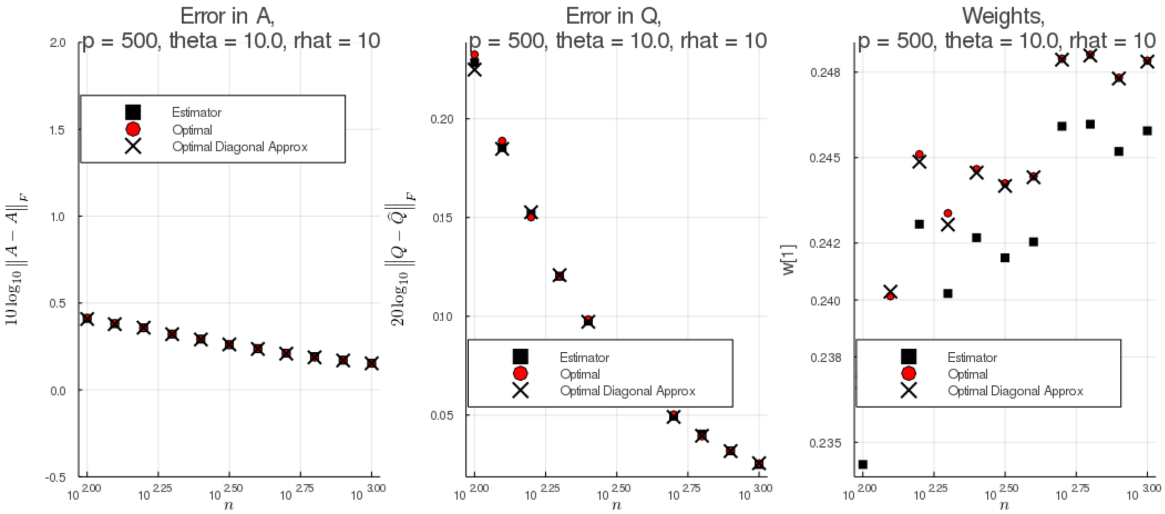


Figure 3.15: We study our Optimally Weighted DMD procedure, where we fix all parameters except the sample size n. We see that the diagonal approximation and the data driven estimator are very close to the optimal.

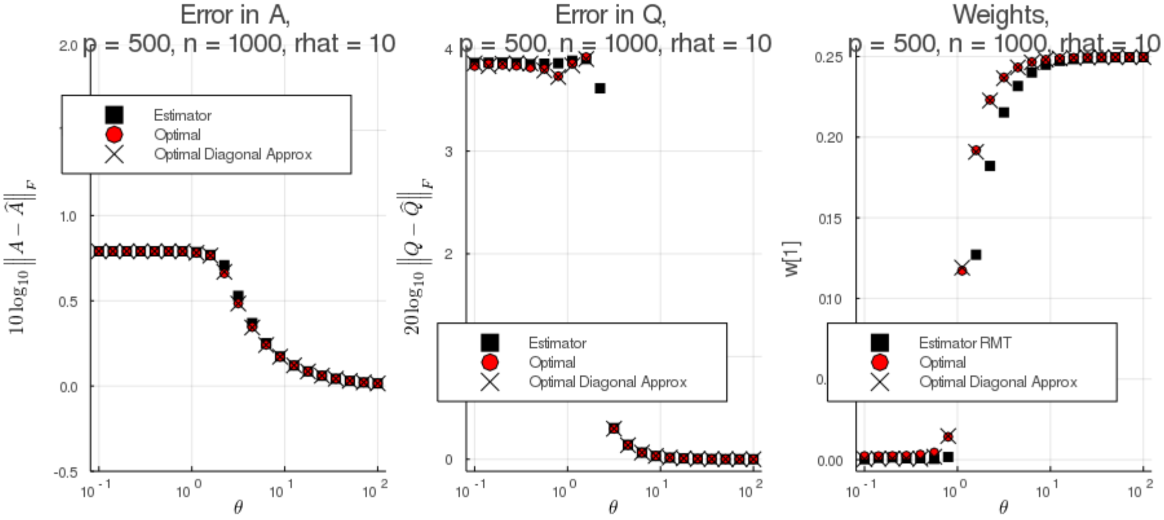


Figure 3.16: We study our Optimally Weighted DMD procedure, where we fix all parameters except the SNR θ. We see that the diagonal approximation and the data driven estimator are very close to the optimal.

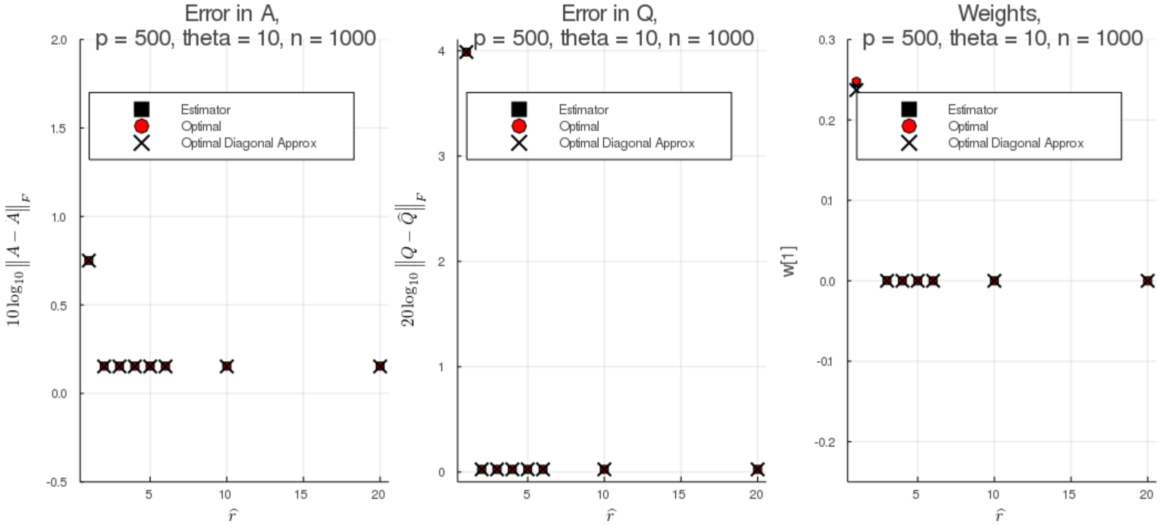


Figure 3.17: We study our Optimally Weighted DMD procedure, where we fix all parameters except the estimated rank $\widehat{r}$. We see that the diagonal approximation and the data driven estimator are very close to the optimal.

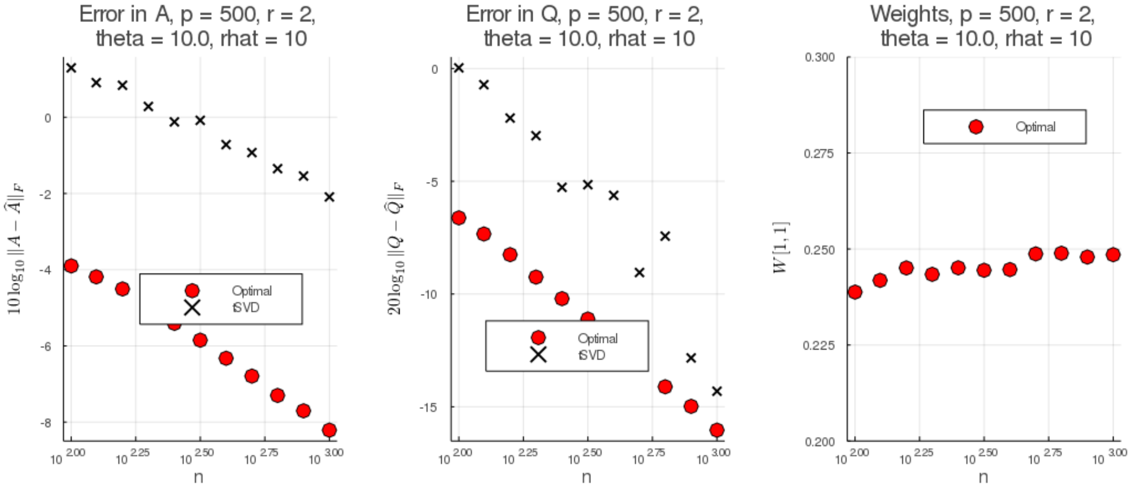


Figure 3.18: We study our Optimally Weighted DMD procedure for the non-orthogonal model, where we fix all parameters except the sample size n. We see that the estimation error of A and Q is lower with the reweighted DMD procedure than with the tSVD. Moreover, we see a transition in the behavior of the weights from shrinkage to growth as n changes.

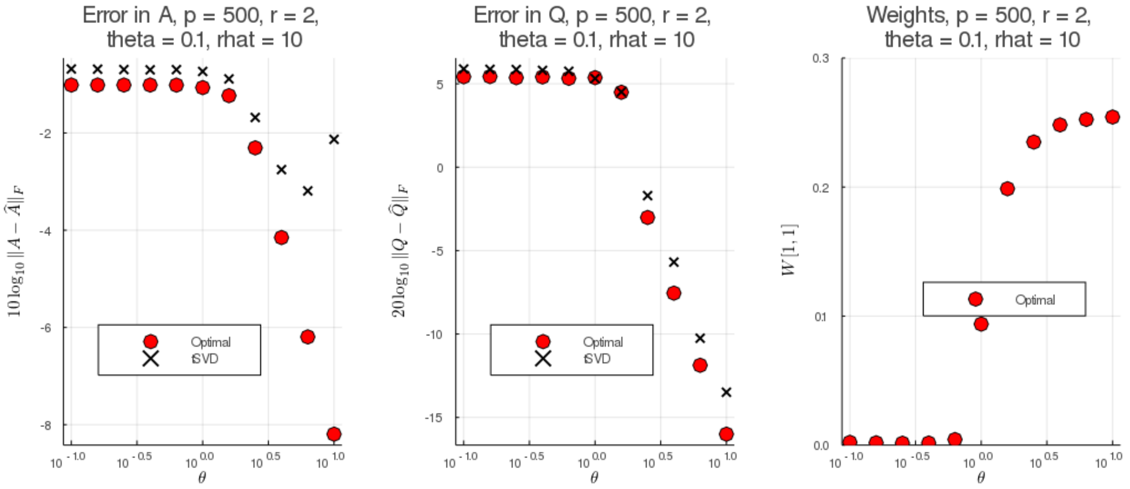


Figure 3.19: We study our Optimally Weighted DMD procedure for the non-orthogonal model, where we fix all parameters except the SNR θ. We see that the estimation error of A and Q is lower with the reweighted DMD procedure than with the tSVD.

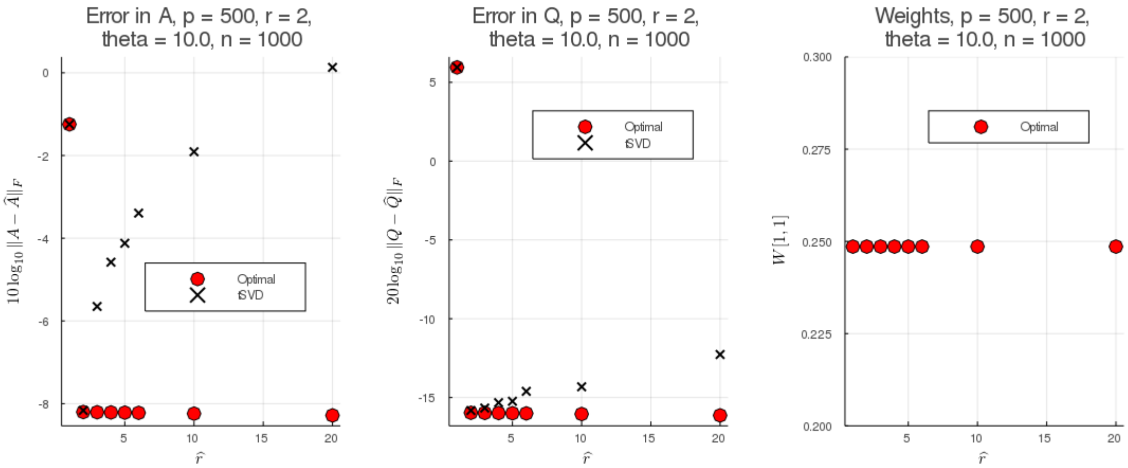
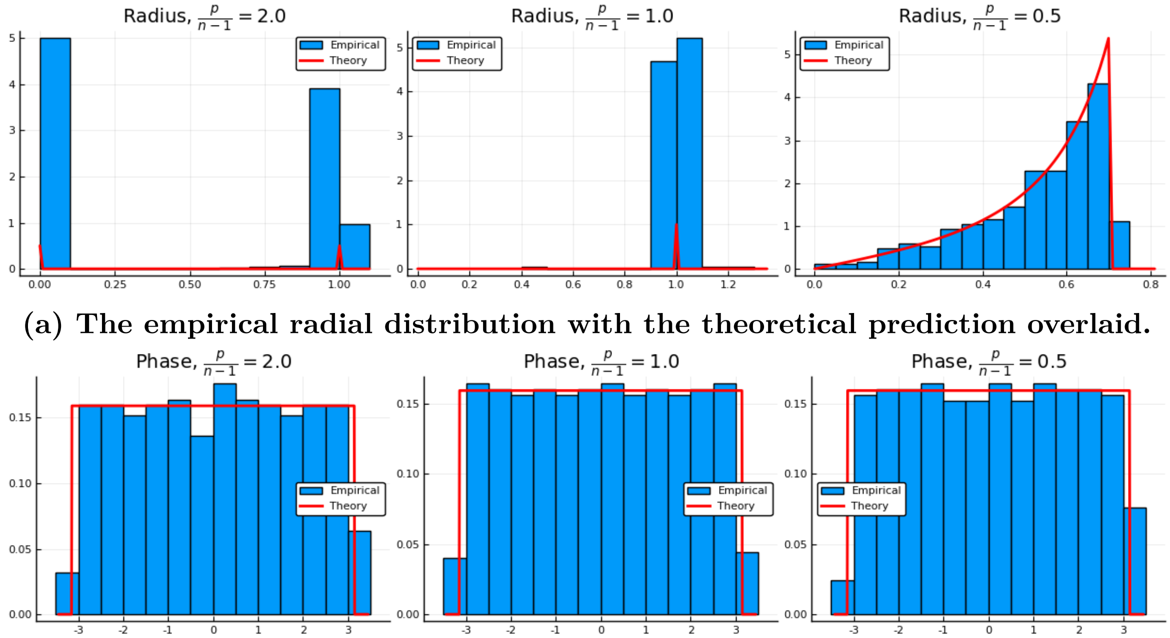


Figure 3.20: We study our Optimally Weighted DMD procedure for the non-orthogonal model, where we fix all parameters except the estimated rank $\widehat{r}$. We see that the estimation error of A and Q is lower with the reweighted DMD procedure than with the tSVD.



(a) The empirical radial distribution with the theoretical prediction overlaid.

(b) The empirical phase distribution with the theoretical prediction (uniform) overlaid.

Figure 3.21: We numerically verify Conjecture 3.1. We see that the distribution of phases is indeed uniform, and that for for $p \geq n-1$ the radial distribution concentrates around 1 and 0. The non-trivial case for $p \leq n-1$ is also observed to closely align with the theoretical prediction.

Chapter 4

Extensions and Applications of the DMD Algorithm

In Chapter 2, we introduced the DMD (Dynamical Mode Decomposition) algorithm and provided a performance analysis in the noise-free and missing data settings. In Chapter 3, we extended our analysis to the noisy data setting. In this chapter, we provide some alternate perspectives on and applications of the DMD algorithm.

First, recall that the DMD algorithm solves an eigenvalue problem, and that our results in Chapter 2 indicate that the solutions / outputs from the algorithm have certain qualities. We present an optimization-based framework that mimics the eigenvalue problem solved by the algorithm. We provide an algorithm with some convergence guarantees, as well as some numerical results. Additionally, we show that our formulation can be used to impose additional structure on the DMD outputs, e.g., sparsity.

Next, we discuss the utility of the Hilbert transform for DMD. Originating from signal processing, the Hilbert transform is the integral transform that takes in a real-valued signal and produces a complex signal that is analytic on the complex plane. We show that using the Hilbert transform on certain datasets prior to applying DMD may be advantageous. We follow the Hilbert transform discussion with a two-dimensional, spatial DMD extension. The DMD algorithm operates on time series, but we might imagine a spatial variation instead of temporal variation (e.g., images instead of time series).

We then apply the DMD algorithm to some real data. We apply DMD to two datasets: an fMRI dataset and a real, still-camera video. We next revisit audio unmixing, and present a comparison of DMD and kurtosis-based ICA. We derive an example of real audio signals that DMD can unmix, but fail to be unmixed by ICA. Finally, we briefly compare DMD and Singular Spectrum Analysis (SSA), another time series decomposition technique.

4.1 An Optimization-Based Formulation of DMD

Once again, assume that we are given a $p \times n$ data matrix X with columns $\mathbf{x}_t$ and that we seek to write X as a linear combination of some latent signals $\{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_k\}$. That is, for a given matrix X, we seek matrices

$$B = \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \cdots \mathbf{b}_k \end{bmatrix} \in \mathbf{s}^{p \times k} \text{ and } S = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \cdots & \mathbf{s}_k \end{bmatrix} \in \mathbf{s}^{n \times k},$$

such that $X = BS^H$. We will assume that we have oracle knowledge of the rank k. Once again, we use P to denote the circular left-shift matrix, where we will leave the dimension unspecified and clear from context.

4.1.1 The Objective Function

The conclusion of Theorem 2.1 was that DMD unmixes or produces signals that are uncorrelated at a lag of one-time step but also have non-vanishing lag-1 autocorrelations. Additionally, given estimates $\widehat{B}$ and $\widehat{S}$, we want $X = \widehat{B}\widehat{S}^H$. These two properties suggest minimizing the following objective function:

$$L\left(X, B, S, \boldsymbol{\lambda}\right) = \frac{1}{2}\left\|X - BS^H\right\|_F^2 + \frac{\lambda_1^2}{2}\sum_{i \neq j}\left(\mathbf{s}_i^H\, P\, \mathbf{s}_j\right)^2 - \frac{\lambda_2^2}{2}\sum_{i}\left(\mathbf{s}_i^H\, P\, \mathbf{s}_i\right)^2,$$

while constraining $\left\|\mathbf{s}_i\right\|_2 = 1$. However, this function is non-convex, is not even block convex, and has a subtraction of terms. Moreover, the unit norm constraint for the $\mathbf{s}_i$ is difficult to work with. A relaxation of the following form may be easier to work with:

$$L\left(X, B, S, \boldsymbol{\lambda}\right) = \frac{1}{2}\left\|X - BS^H\right\|_F^2 + \frac{\lambda_1^2}{2}\left\|\mathrm{I}_k - S^H P S\right\|_F^2.$$

We have relaxed the unit norm constraint by asking that the diagonal terms of $S^H PS$ (equal to $\mathbf{s}_i^H\, P\, \mathbf{s}_i$) are close to 1. This formulation eliminates the subtraction of terms, but is still not block-convex due to the $S^H PS$ term. A natural way to create block-convexity would be to use variable splitting [44]. We then may write

$$L\left(X, B, S, Z, \boldsymbol{\lambda}\right) = \frac{1}{2}\left\|X - BS^H\right\|_F^2 + \frac{\lambda_1^2}{2}\left\|\mathrm{I}_k - Z^H PS\right\|_F^2 + \frac{\lambda_2^2}{2}\left\|Z - S\right\|_F^2. \tag{4.1}$$

Essentially, splitting S into Z and S and then penalizing the difference between Z and S creates a block-convex function.

Gradients and Updates

Given (4.1), we may use an alternating minimization approach to seek the minima [132]. That is, we take gradient descent steps in each variable until convergence.

We see that the gradient in S is given by

$$\nabla_S L = \left[BS^H - X\right]^H B + \lambda_1^2\left(Z^H P\right)^H\left[Z^H PS - \mathrm{I}_k\right] + \lambda_2^2\left[S - Z\right], \tag{4.2a}$$

and that in Z is given by

$$\nabla_Z L = \lambda_1^2\left(PS\right)\left[\left(PS\right)^H Z - \mathrm{I}_k\right] + \lambda_2^2\left[Z - S\right]. \tag{4.2b}$$

Moreover, as in [105, Proof of Prop. 1], we find that B has a closed form update of the form

$$\mathbf{b}_i \mapsto \frac{1}{\mathbf{s}_i^H\,\mathbf{s}_i}\left(X - BS^H\right)\mathbf{s}_i + \mathbf{b}_i. \tag{4.2c}$$

Sparsity

At this point, we may extend the original DMD problem to include sparsity: if we believe that the modes $\mathbf{b}_i$ and hence the mixing matrix B are sparse, can we find sparse solutions? A natural choice would be to include a term of the form

$$\sum_i \left\|\mathbf{b}_i\right\|_0$$

in the optimization:

$$L\left(X, B, S, Z, \boldsymbol{\lambda}\right) = \frac{1}{2}\left\|X - BS^H\right\|_F^2 + \frac{\lambda_1^2}{2}\left\|\mathrm{I}_k - Z^H PS\right\|_F^2 + \frac{\lambda_2^2}{2}\left\|Z - S\right\|_F^2 + \lambda_3^2\sum_{i=1}^{k}\left\|\mathbf{b}_i\right\|_0. \tag{4.3}$$

The only change is to the update for $\mathbf{b}_i$ in (4.2) [105, Proof of Prop. 1]:

$$\mathbf{b}_i \mapsto \mathcal{H}_{\lambda_3}\left(\frac{1}{\mathbf{s}_i^H\,\mathbf{s}_i}\left(X - BS^H\right)\mathbf{s}_i + \mathbf{b}_i\right), \tag{4.4}$$

where $\mathcal{H}_t$ denotes the hard-thresholding operator:

$$\mathcal{H}_t(x) = \begin{cases} 0 & \text{if } |x| \leq t, \\ \left[|x| - t\right]\exp\left(i\angle x\right) & \text{otherwise.} \end{cases} \tag{4.5}$$

4.1.2 Algorithm

Formally, we are solving the following problem:

$$\widehat{B}, \widehat{S}, \widehat{Z} = \arg\min_{B\in\mathbb{C}^{p\times k};S,Z\in\mathbb{C}^{n\times k}} L\left(B, S, Z, \boldsymbol{\lambda}\right), \tag{4.6a}$$

where

$$L\left(X, B, S, Z, \boldsymbol{\lambda}\right) = \frac{1}{2}\left\|X - BS^H\right\|_F^2 + \frac{\lambda_1^2}{2}\left\|\mathrm{I}_k - Z^H Ps\right\|_F^2 + \frac{\lambda_2^2}{2}\left\|Z - S\right\|_F^2 + \lambda_3^2\sum_{i=1}^{k}\left\|\mathbf{b}_i\right\|_0. \tag{4.6b}$$

Note that

$$\boldsymbol{\lambda} = \begin{bmatrix}\lambda_1 & \lambda_2 & \lambda_3\end{bmatrix},$$

and that $\lambda_i \geq 0$. Then, our algorithm is an alternating minimization procedure, where we iterate updates of the form

1. for $i \in \{1, 2, \ldots, k\}$,

$$\mathbf{b}_i \mapsto \mathcal{H}_{\lambda_3}\left(\frac{1}{\mathbf{s}_i^H\,\mathbf{s}_i}\left(X - BS^H\right)\mathbf{s}_i + \mathbf{b}_i\right),$$

2.

$$S \mapsto S - \alpha_S \nabla_S L,$$

3.

$$Z \mapsto Z - \alpha_Z \nabla_Z L,$$

where Z and $\nabla_S L$ and $\nabla_Z L$ are defined in (4.2), and α_S and α_Z are positive step-sizes.

4.1.3 Convergence Analysis

As written, we are unable to state any guarantees or convergence results for the problem (4.6). However, if we were to modify the objective function and problem slightly, we are able say something [132].

First, we need to impose boundedness on the variables $\mathbf{b}_i$, $\mathbf{s}_i$, and $\mathbf{z}_i$, e.g., by saying that

$$\|\mathbf{b}_i\|_\infty \leq \lambda_4,$$

and similarly for the $\mathbf{s}_i$ and $\mathbf{z}_i$ with λ_5 and λ_6. Once again, the only change to our updates that is made is to threshold from above [105, Proof of Prop. 1]: that is, given the current variable value, we perform a capping operation $\mathcal{C}_t(x)$ at λ_4, where

$$\mathcal{C}_t(x) = \begin{cases} x & \text{if } |x| \leq t, \\ t \exp\left(i\angle x\right) & \text{otherwise.} \end{cases} \tag{4.7}$$

Note that we need $\lambda_3 \leq \lambda_4$.

Second, we need to ensure that each 'block' and subproblem is strongly convex. Adding

$$\frac{\lambda_7^2}{2}\left[\|B\|_F^2 + \|S\|_F^2 + \|Z\|_F^2\right]$$

to the objective function L, where λ_7 is a small, positive constant, achieves this. Note that λ_7 must be smaller than λ_1 and λ_2. Here, the updates are now different, but if λ_7 is sufficiently small, heuristically, we expect that the solution/limit points will be close to what they would be with $\lambda_7 = 0$.

Given these two small modifications to our problem, we are able to state the following convergence result for the problem (4.6) as a consequence of [132, Cor. 2.4]:

Theorem 4.1. *When algorithm in Section 4.1.2 is applied to the modified version of (4.6), every limit point is a Nash equilibrium point.*

4.1.4 Simulations

We numerically solve the problem (4.6) using the algorithm specified in Section 4.1.2 and compare the performance with ordinary DMD.

We fix $p = 200$ and vary the number of samples n between 10^2 and $10^{3.5}$. We generate two different rank-2 datasets, both composed of cosines. In the first, the columns of S are proportional to $\cos\frac{t}{4}$ and $\cos\frac{t}{2}$, and in the second, they are proportional to $\cos\frac{t}{2}$ and $\cos 2t$. We use a sparse B with 10 non-zero entries. We add *i.i.d.* Gaussian noise (mean 0 and variance σ^2/n to the data, and vary σ.

We perform a coarse grid-search over $\boldsymbol{\lambda}$ (4^3 values), and use 10^3 iterations of the alternating descent updates. We search over 5 starting points and perform 5 trials of different noise instances. The total run time for this procedure is $\sim$ 27 hours.

We present results for the estimation error of B and S (after normalization/rescaling to unit ℓ_2 norm), as defined in (2.33a). In Figures 4.1 and 4.2, we see that the optimization formulation is slightly more robust to noise than the original DMD algorithm. Of course, this robustness comes at a high computational cost, and it is unclear whether this cost is worth it relative to denoising with a truncated SVD followed by the original algorithm. Of course, the main cost here comes from the parameter grid search: one immediate improvement to the tractability would be to replace the parameter grid search with some more intelligent, like a Bayesian Hyperparameter Search [112] or even a simple random search [15].

4.2 Complex Exponentials and Hilbert DMD

Assume that we have a data matrix $X \in \mathbb{C}^{p\times n}$ that can be written as

$$X = BC^H, \tag{4.8}$$

where $B \in \mathbb{C}^{p\times k}$ has linearly independent columns and the j^{th} column of C, denoted by $\mathbf{c}_j$ has entries $(\mathbf{c}_j)_t = \exp(i\omega_j t)$. I.e., each sample $\mathbf{x}_j$ (column j) in X is a linear combination of complex exponentials. Note that if any of the $\mathbf{c}_i$ had a phase shift, we would simply absorb the constant into the corresponding column of B, $\mathbf{b}_i$.

The DMD algorithm proceeds by forming

$$X_{(1)} = \begin{bmatrix}\mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_n\end{bmatrix} \text{ and } X_{(0)} = \begin{bmatrix}\mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{n-1}\end{bmatrix},$$

and takes the eigendecomposition of

$$\widehat{A} = X_{(1)}X_{(0)}^+.$$

Our goal is to characterize the behavior of DMD in this setting (complex exponentials).

First, we observe that if we define $C_{(0)}$ and $C_{(1)}$ analogously to $X_{(1)}$ and $X_{(0)}$ (we subset to the first/last $n-1$ rows), we may write

$$C_{(1)}^H = W C_{(0)}^T,$$

where $W \in \mathbb{C}^{k \times k}$ is diagonal with entries $\exp(i\omega_j)$. I.e., multiplication by W advances every entry of $C_{(0)}^H$ by one timestep, thus forming $C_{(1)}^H$. We may then write

$$\widehat{A} = X_{(1)} X_{(0)}^+ = B C_{(1)}^H \left(B C_{(0)}^H \right)^+ = B W C_{(0)}^H \left(B C_{(0)}^H \right)^+ . \tag{4.9}$$

By construction, $C_{(0)}^H$ has linearly independent rows and B has linearly independent columns. It follows that

$$\widehat{A} = X_{(1)} X_{(0)}^+ = B W C_{(0)}^H \left(C_{(0)}^H \right)^+ B^+. \tag{4.10}$$

Moreover, since $C_{(0)}^H$ has linearly independent rows, we may write $C_{(0)}^H \left(C_{(0)}^H \right)^+ = \mathrm{I}_k$, so that

$$\widehat{A} = X_{(1)} X_{(0)}^+ = B W B^+. \tag{4.11}$$

That is, DMD yields a matrix $\widehat{A}$ whose non-zero eigenvalues are $\exp(i\omega_j)$ with corresponding eigenvectors proportional to $\mathbf{b}_j$.

4.2.1 Hilbert DMD

A large portion of the simulations and examples in this thesis are of linearly mixed cosines, where the error from performing DMD is small but non-zero. Having observed that complex exponentials are perfectly recovered with DMD, one might wonder whether it is possible to *transform* the cosines into complex exponentials, and hence have perfect recovery.

The Hilbert Transform of a real-valued signal $f(t)$ is a linear transformation that returns the real-valued signal $g(t)$ such that $f(z) + ig(z)$ is an analytic function in the complex plane [91]. Colloquially, this analytic function is referred to as the Hilbert transform of the input signal. In the case of the signal $a \cos(\omega t + \phi)$, the corresponding transformed output would be the complex exponential $a \exp(i\omega t + i\phi)$.

Hence, if we observe a data matrix X whose rows are linear combinations of cosines of various frequencies and phases, applying the Hilbert transform to each row yields linear combinations of complex exponentials. Then, applying DMD to the transformed data yields perfect recovery of the mixing matrix and complex exponentials; taking the real part would yield the original latent signals.

4.2.2 Simulations

We test the power of Hilbert-DMD on simulated data. Our data model is

$$X = \theta \, \mathbf{q}_1 \, \mathbf{s}_1^T + \frac{1}{2}\theta \, \mathbf{q}_2 \, \mathbf{s}_2^T \, .$$

We fix $p = 250$ and vary the number of samples n. We use a rank-2 dataset, where the $\mathbf{s}_i$ are cosines of frequencies $1/4$ and 2. We vary the signal strength θ and add *i.i.d.* Gaussian noise of variance $1/n$ and mean zero. We compare the estimation error of DMD, the tSVD-DMD, and the Hilbert-DMD algorithms for Q and S.

We begin with the noise-free setting in Figure 4.3. We observe that the Hilbert-DMD algorithm does not perform quite as well as does vanilla DMD, but that its error is still low. In Figure 4.4, we present the noisy data setting. We see that there is an intermediate regime wherein Hilbert-DMD performs slightly better than DMD, when the signal strength is not too large (but is sufficiently large to not be lost) and the sample size is sufficiently large. I.e., it is possible that Hilbert-DMD can offer a slight edge over vanilla DMD.

4.3 A Two-Dimensional Version of DMD

In Chapter 2, we discovered that DMD is able to solve the Blind Source Separation problem. I.e., it is able to unmix linear combinations of latent signals, *if* the signals are uncorrelated at lags of zero and one timesteps. However, the DMD algorithm is inherently one-dimensional, and relies on uncorrelatedness in a temporal sense. In this section, we develop a spatial version of DMD, that is, a version of DMD that can unmix linear combinations of images.

4.3.1 Model

Assume that we have images

$$Z_1, Z_2, \ldots, Z_r \in \mathbb{R}^{m \times n}.$$

Assume that we observe linear mixtures of the images,

$$Y_1, Y_2, \ldots, Y_p \in \mathbb{R}^{m \times n},$$

where $r \leq p$. Moreover, assume that the mixtures are generated by

$$Y_i = \sum_{l=1}^{r} B[i,l] Z_l,$$

for $B \in \mathbb{R}^{p \times r}$. Without loss of generality, assume that the images Z_r have zero mean and unit norm.

The 1-dimensional DMD algorithm is able to unmix linearly mixed signals that are uncorrelated at a lag of zero and one more, non-zero lag (traditionally, a lag of 1). Here, we continue the generalization and impose uncorrelatedness at a lag (τ_x, τ_y). That is, the inner product

$$\text{Tr}\left[Z_i[(\tau_x + 1) : m, (\tau_y + 1) : n] Z_j[1 : (m - \tau_x), 1 : (n - \tau_y)]^T\right]$$

vanishes for $i \neq j$ and does not vanish for $i = j$.

Let $\widehat{A}$ be the $p \times p$ matrix with entries

$$\text{Tr}\left[Y_i[(\tau_x + 1) : m, (\tau_y + 1) : n] Y_j[1 : (m - \tau_x), 1 : (n - \tau_y)]^T\right]. \tag{4.12}$$

If the inner product of the Z_i, Z_j vanishes for pairs $i \neq j$, we have that

$$\widehat{A} \approx B\text{diag}\left(\text{Tr}\left[Z_i[(\tau_x + 1) : m, (\tau_y + 1) : n] Z_i[1 : (m - \tau_x), 1 : (n - \tau_y)]^T\right]\right) B^T. \tag{4.13}$$

Notice that if we define a matrix Q where the i^{th} column of Q is the corresponding column of B scaled to have unit norm, we may write

$$\widehat{A} \approx Q\text{diag}\left(\|B[:,i]\|_2^2 \, Tr\left[Z_i[(\tau_x + 1) : m, (\tau_y + 1) : n] Z_i[1 : (m - \tau_x), 1 : (n - \tau_y)]^T\right]\right) Q^T. \tag{4.14}$$

If the columns of B are orthogonal, so that the columns of Q are orthonormal, the columns of Q are the eigenvectors of $\widehat{A}$. Then, up to a scaling, we may recover the latent images by

$$\widehat{Z}_i = \sum_{l=1}^{p} Q[l,i] Y_l. \tag{4.15}$$

We will refer to the computation of $\widehat{A}$ as in (4.12), the subsequent eigendecomposition $\widehat{A} = Q\Lambda Q^T$, and the unmixing in (4.15) as the orthogonal 2D-DMD algorithm.

Remark 4.1. *If the images are not orthogonally mixed, we may first whiten the mixtures Y_i, as in [131]. That is, we compute*

$$\widetilde{Y}_i = \sum_{l=1}^{p} C[i,l] Y_l,$$

where $C = \widehat{A}^{-1/2}$. We may then apply the orthogonal 2D-DMD algorithm to the $\widetilde{Y}_i$.

4.3.2 Application

In this section, we demonstrate the ability of the 2D-DMD algorithm to unmix mixtures of images. In Figures 4.5 and 4.6, we present results for two orthogonally and non-orthogonally mixed images, respectively. We see that our method works to unmix the images.

4.4 Data Analysis

In this section, we present applications of DMD to real data.

4.4.1 Functional fMRI Data

Courtesy of J. Fessler and M. Karker, we were given access to a $3D$ functional MRI dataset. The data is a cleaned, reconstructed image, with dimensions $(x, y, z) = (72, 48, 12)$ and 235 samples in time. We reshaped each sample into a vector and formed a $[72 \cdot 48 \cdot 12] \times 235$ matrix. This data is reconstructed to be low-rank and sparse (in the Fourier domain) [65, 72]. We compare the results of DMD on the de-meaned logged/not-logged data with the SVD.

Our analysis ‘pipeline’ is as follows:

1. Reshape the data into a matrix.
2. Take the base-10 logarithm of each element in the matrix (for the logged data; otherwise do nothing).
3. Subtract the mean from each row of the data.
4. Perform DMD (at a lag of 1) on both the logged/not logged matrices.
5. Take the SVD of both matrices to compare with DMD.

Our motivation for taking the logarithm is twofold. First, we observed that after performing DMD, some of the signals in $\widehat{C}$ looked like they were modulated cosines, i.e., a product of two different signals; and second, the scale of values is quite large, and the logarithm is a monotonic transformation reduces the absolute variation in the data.

We first present the mean images in Figures 4.7 and 4.8. Next, we present the leading estimated components for DMD, logged-DMD, and the SVD in Figures 4.9, 4.10, 4.11, and 4.12. We see that none of these are particularly convincing relative to the mean image, but that the edges of the image are captured well. Moreover, all of the images are qualitatively similar.

When we consider the temporal variations C and the right singular vectors V, given in Figures 4.13, 4.14, 4.15, and 4.16, we see a stark difference. Recall that this image is reconstructed from a sparse representation in a Fourier basis, i.e., as a linear combination of complex exponentials. The results from DMD are exactly sinusoidal: they are the latent Fourier series that reconstructed this image. However, the SVD does not capture any of this structure, and it is unclear what the right singular vectors convey.

4.4.2 Video Data

We recorded a video containing two laptop screens on two chairs, where the two screens flashed at different frequencies; in Figure 4.17, we display the background of the scene. The first screen flashed at a frequency $f_1 = 1.0Hz$ and the second at a frequency $f_2 = 2/3Hz$. There are 233 frames of size 160×120, so that we have a video data cube of size $160 \times 120 \times 233$. We reshaped the cube into a matrix (each frame is a column of length 160×120) before proceeding with our analysis.

There are three components to the video: the two flashing screens and the background, and we hope that any decomposition would capture this behavior. We first apply DMD to the video, as-is. In Figures 4.18 and 4.19, we present these results. We see that the second screen and accompanying frequency is lost, and that the first is duplicated (as a complex conjugate). The background is estimated well, but the decomposition is somehow incomplete. Noting that there are three components in the video, we might expect that truncating the video to a rank-3 matrix might yield better results. Indeed, it does. In Figures 4.20 and 4.21, we see that we recover all three frequencies and components, and there is no visual degradation.

We next repeat our experiment, but with the video corrupted by additive white noise with standard deviation equal to 0.01% of the Frobenius norm of the video matrix. Our results are shown in Figures 4.22 and 4.23. The estimated modes are noisy (although all of the components are visible), and the temporal variations are inaccurate. Once again, however, denoising with a truncated SVD before applying DMD is a good solution, as seen in Figures 4.24 and 4.25.

In general, there is the question of whether the latent rank is correctly estimated: in our video, we know that there are three components, and hence use a rank-3 approximation. We study the effect of a rank overestimation on the output: we now repeat our previous simulations for a non-oracle, estimated rank of 4. In Figures 4.26 and 4.27, we see that the rank-4 truncation loses all the components except the background; however, in Figures 4.28 and 4.29, we see that using the OptShrink method recovers all three components [88]. The OptShrink method is a modification of the tSVD low-rank matrix estimator that applies a shrinkage to the empirical, noisy singular values, so that the estimated low rank matrix has a lower error relative to the truth. With the tSVD, there is no shrinkage, and the fourth singular value (pure noise) is strong enough to negatively affect the DMD results. However, with OptShrink the shrinkage applied to the fourth singular value (noise) is sufficient to mitigate the rank overestimation. This story is repeated for the noisy video setting, with the tSVD results shown in Figures 4.30 and 4.31 and OptShrink in Figures 4.32 and 4.33. Once again, we see that the OptShrink shrinkage is sufficient to mitigate the rank overestimation, whereas the tSVD is not robust.

4.5 Audio Unmixing Revisited

In Chapter 2, we demonstrated that DMD is able to unmix linearly mixed, real audio signals. We also saw that kurtosis-based ICA (referred to as ICA in this work) was able to unmix the signals. In this section, we revisit this example and modify it so that we have an example where DMD works to unmix the signals but ICA does not.

Here, we have three speech signals and one music signal, all sampled at $48000 Hz$ and with 141540 samples (approximately 2.9 seconds). The three speech signals are from different speakers, all adult. The first signal has an adult male saying ‘The DMD algorithm is very interesting’, the second has an adult female saying ‘DMD is a very boring algorithm’, and the third has an adult female saying ‘I don’t know what DMD is’. The music is an instrumental segment from the Reggaeton song ‘Rakata’, by Wisin y Yandel [87].

We begin by normalizing the signals to have identical ranges and zero mean, and then we mix the signals with a 4×4, randomly generated mixing matrix. We apply both DMD and ICA to the mixture, and see that we are able to recover the signals with squared errors (defined in (2.39)) of 2×10^{-3} and 1×10^{-3}, respectively. Our results are shown in Figure 4.34.

4.5.1 Breaking Kurtosis-Based ICA

Recall that kurtosis-based ICA fails to unmix signals with a Gaussian marginal [56, Ch. 7]. Using this fact, we will construct an example of ‘real’ signals that DMD can unmix but cannot be unmixed with ICA. Note that for a random variable x with distribution function $F(t) = \mathbb{P}\left(x \leq t\right)$, we may transform x into a normal random variable by

$$\widetilde{x} = F_g^{-1}\left(F\left(x\right)\right), \tag{4.16}$$

where F_g^{-1} is the functional inverse of the standard normal distribution function. We have used the fact that $F\left(x\right)$ has a uniform distribution on $[0, 1]$.

We return to our four real signals, and transform each of them according to (4.16). Once again, we mix the signals with a 4×4, randomly generated mixing matrix. We apply both DMD and ICA to the mixture, and see that ICA completely fails to recover the signals, but that DMD comes reasonably close with a squared error of 2×10^{-1}. Our results are shown in Figure 4.35. We note that since the success of DMD is tied to the lag-1 cross correlations and autocorrelations of the signals, the continued success of DMD is due to the fact that the transformation does not change the correlations significantly. In particular, computing the lag-1 correlation matrix L from the signal and $\widetilde{L}$ from the transformed signals, defined in (2.29), we find that

$$\left\| L - \widetilde{L} \right\|_F / \left\| L \right\|_F \approx 0.03.$$

Qualitatively, the transformed signals are akin to ‘bad radio transmissions’, and are still recognizable and intelligible. I.e., this transformation is not too extreme or even terribly unrealistic.

4.6 DMD and SSA

In Chapter 2, we have shown that DMD can be used as a multivariate time series decomposition method: given a p-variate time series, we can recover up p latent time series. In this section, we take a brief look at another time series decomposition method and provide a comparison of the two methods.

Singular spectrum analysis (SSA) is a decomposition method for univariate time series [45]. The method relies on a Singular Value Decomposition (SVD) of an embedding matrix. I.e., given a times series $\mathbf{x}_t$ with entries $x_1, x_2, \ldots, x_n$ and a window parameter l, the $l \times (n - l + 1)$ matrix is fomed:

$$X = \begin{bmatrix} x_1 & x_2 & \cdots & x_{n-l+1} \\ x_2 & x_3 & \cdots & x_{n-l} \\ \vdots & \vdots & \cdots & \vdots \\ x_l & x_{l+1} & \cdots & x_n \end{bmatrix}.$$

The SVD of X is then grouped and reshaped into several component time series vectors. A multivariate extension exists, wherein each individual time series is embedded and the individual embedding matrices are stacked up.

There are well established separability (recovery) results for SSA [45, Sec. 1.5]. Assume that a univariate time series $\mathbf{x}_t$ is a sum of components $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_k$. First, for a fixed window l, we say that the $\mathbf{c}_i$ are separable (recoverable) from $\mathbf{x}_t$ in a weak sense if every contiguous subsequence of $\mathbf{c}_i$ is orthogonal to every contiguous subsequence of $\mathbf{c}_j$ for $i \neq j$ and subsequences of identical length. Strong separability is guaranteed by the singular values of the embedding matrix being distinct. Moreover, there are approximate and asymptotic notions of strong and weak separability (that the conditions hold approximately or in the limit $n, l \to \infty$). Concretely, cosines of different frequencies are asymptotically weakly separable, and if their magnitudes are distinct, they will be asymptotically strongly separable [50].

The first point of comparison is the number of components that can be distinguished with the algorithms. Given a p-variate time series of length n, DMD is able to extract $\min\{p, n-1\}$ series. It follows that DMD is inherently multivariate: given a single, univariate time series, DMD would not be able to extract anything meaningful. Contrastingly, SSA is by definition a univariate time series decomposition. The number of components is necessarily a function of the window size, and will be at most $\min\{l, n-l+1\}$, for a maximum value of $n/2$ components when $l = n/2$. With a multivariate SSA decomposition of a p-variate series, the number of components is scaled by p.

The second point of comparison is the computational properties of the algorithms. Given a p-variate time series of length n, DMD involves the pseudoinverse of a $p \times n-1$ matrix, the product of a $p \times n-1$ matrix with an $n-1 \times p$ matrix, and the eigendecomposition of a $p \times p$ matrix. To recover the latent series, if there are k components, the pseudoinverse of a $p \times k$ matrix and the product of this pseudoinverse with a $p \times n$ matrix is required. SSA is dominated by the reshaping and SVD steps: for each variable/series, there is an $l \times n-l+1$ matrix formed. Then, a multivariate SSA procedure would require the SVD of a $pl \times n-l+1$ matrix. If l and n are both large, the scaling of the algorithm with p may quickly become unfeasible.

The third point of comparison is the choice of parameters. With DMD, unless we are using the extension to lags other than 1, the only parameter is the number of components. With SSA, both the number of components and the window size are parameters that must be identified. In particular, the window size plays a critical role in the identifiability of signals, and is directly related to the temporal resolution of the recovered signals [45, Ch. 1]. In practice, a larger window size is balanced against computational considerations, as well as any known periodicities in the data.

We next compare DMD and SSA numerically. We generate a random 10×2 mixing matrix Q and two cosines that are to be mixed: the first with frequency $\omega_1 = 3/2$ and the second with frequency $\omega_2 = 2$. We fix the length $n = 1000$ and use a window size of $l = n/2$. We compare SSA, multivariate SSA (MSSA) and DMD in Figures 4.36, 4.37, and 4.38. We see that SSA outperforms MSSA, but both are worse than DMD. Nonetheless, both SSA and MSSA return outputs that are recognizably close to the true signals. We repeat our experiment with $\omega_1 = 1/2$ and $\omega_2 = 2$, seen in Figures 4.39, 4.40, and 4.41. Here, we see that MSSA suffers from additional artefacts at the edges of the signals, and that SSA misses one of the cosines in every case. DMD once again performs well.

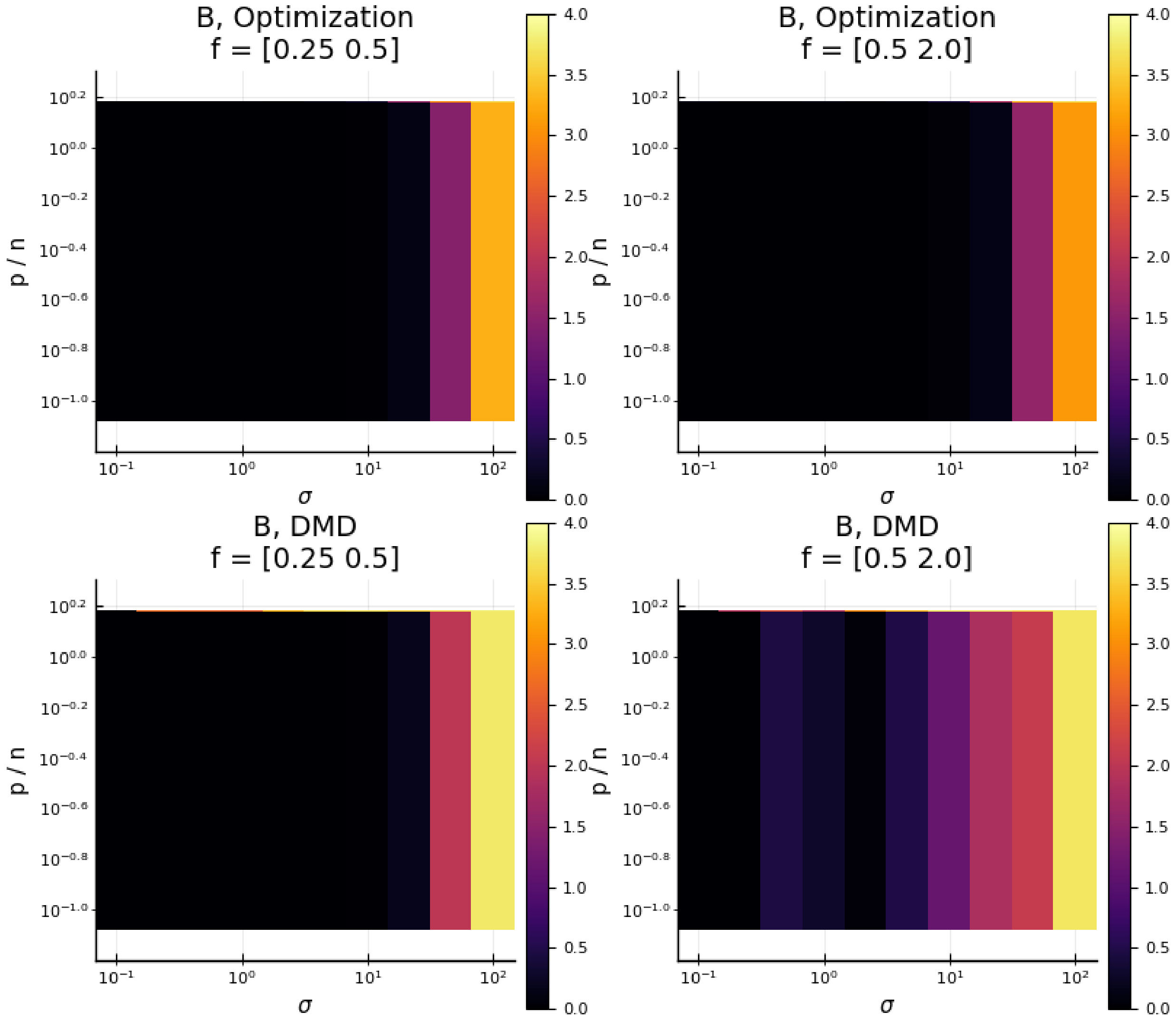


Figure 4.1: We present results for the estimation error of B (after normalization/rescaling to unit ℓ_2 norm), as defined in (2.33a). We fix $p = 200$ and vary n, the frequencies, and the noise level. We see that the optimization formulation is more robust to noise than DMD.

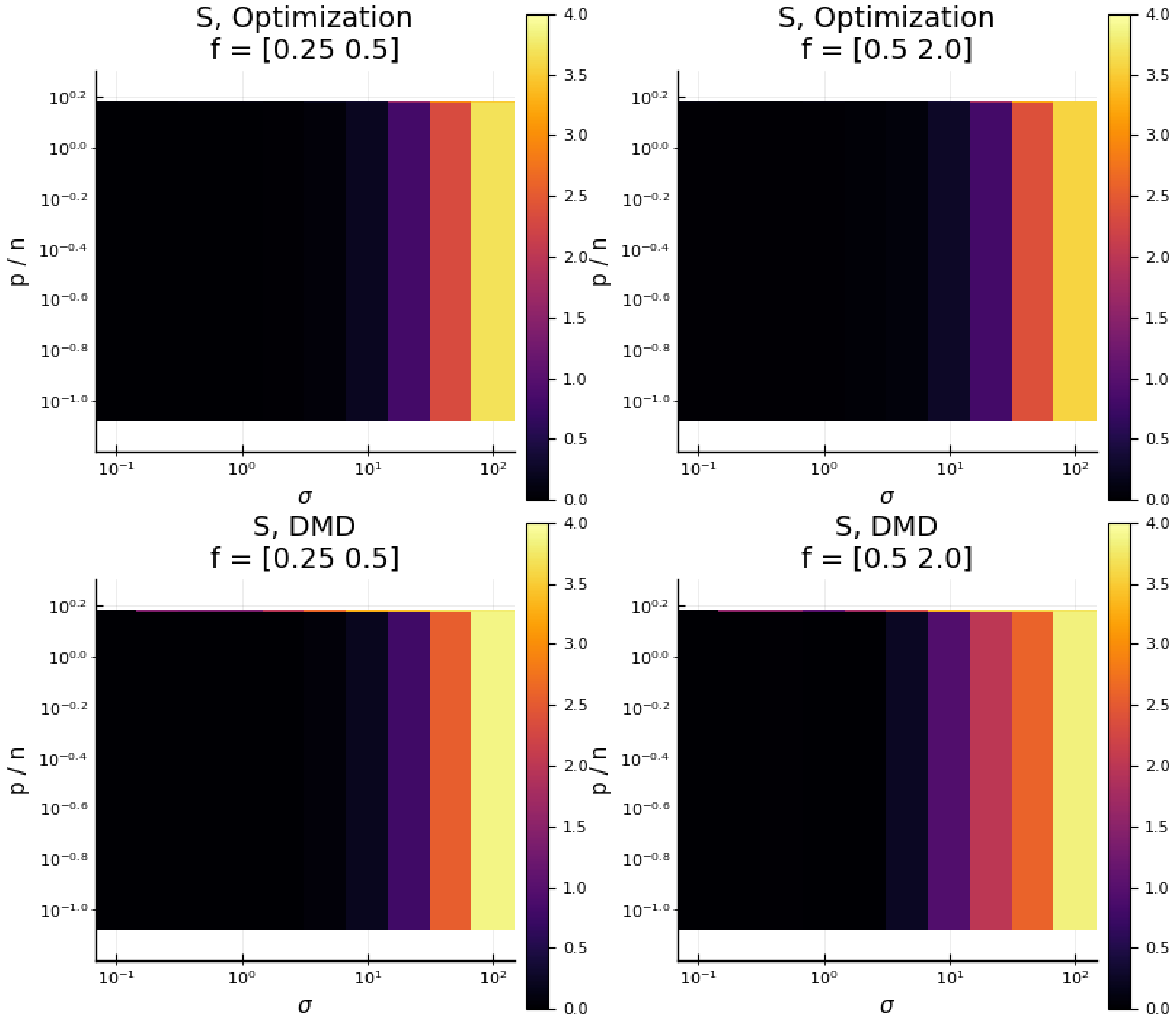


Figure 4.2: **We present results for the estimation error of S (after normalization/rescaling to unit ℓ_2 norm), as defined in (2.33a). We fix $p = 200$ and vary n, the frequencies, and the noise level. We see that the optimization formulation is more robust to noise than DMD.**

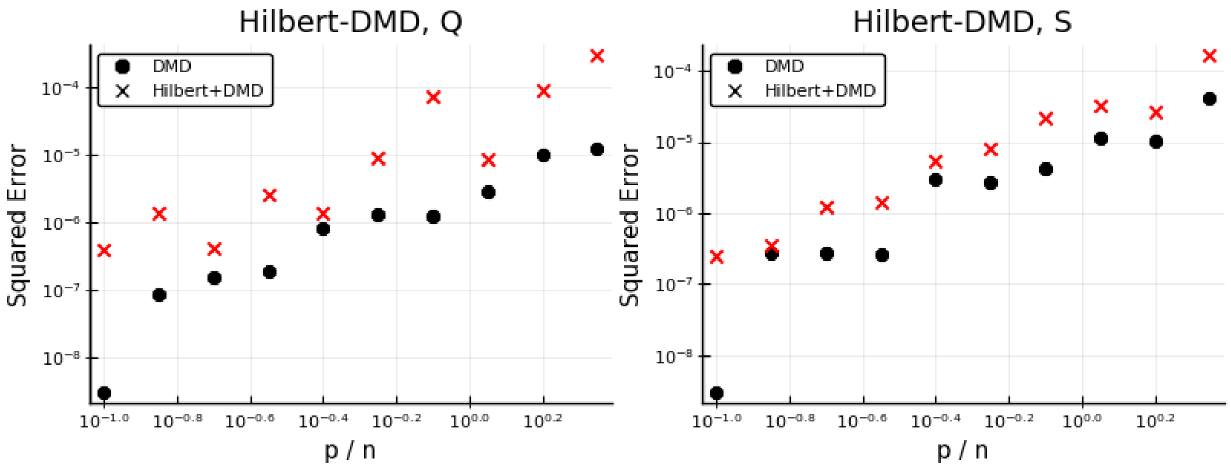


Figure 4.3: We present results for the estimation error of Q and S as defined in (2.33a) for the noise-free setting. We fix $p = 250$ and vary n. We see that Hilbert-DMD performs well, but not as well as vanilla DMD here.

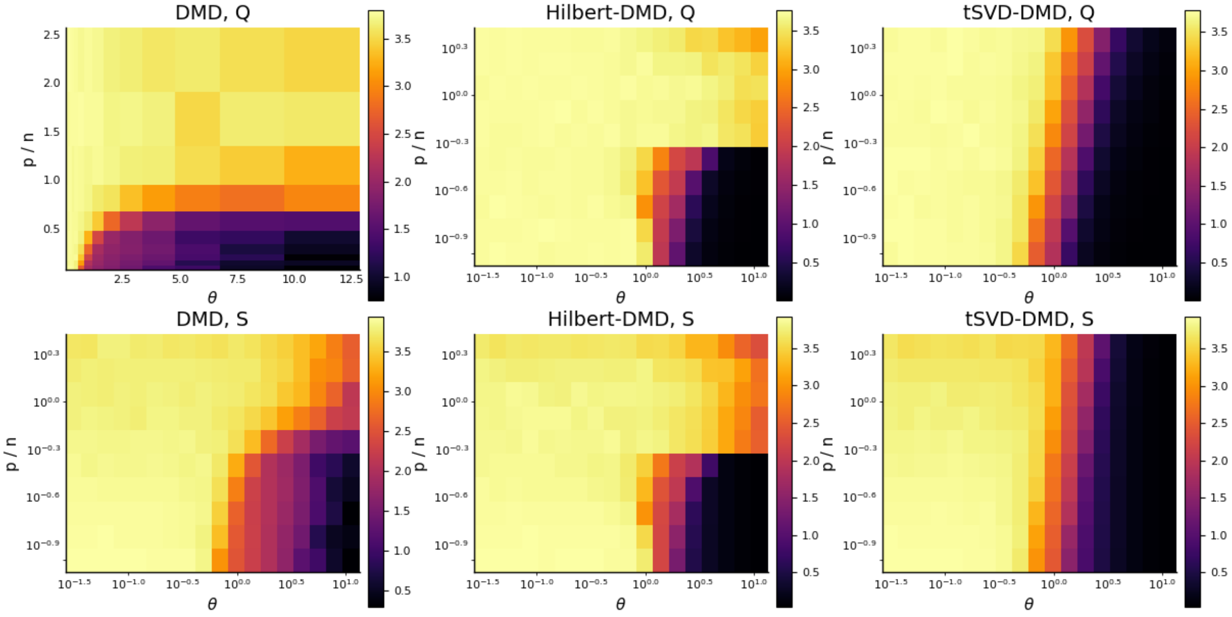


Figure 4.4: We present results for the estimation error of Q and S as defined in (2.33a) in the noisy setting. We fix $p = 250$ and vary n. We see that Hilbert-DMD performs better than DMD in the intermediate regime where the sample size n is sufficiently large and the signal strength θ is moderately large.

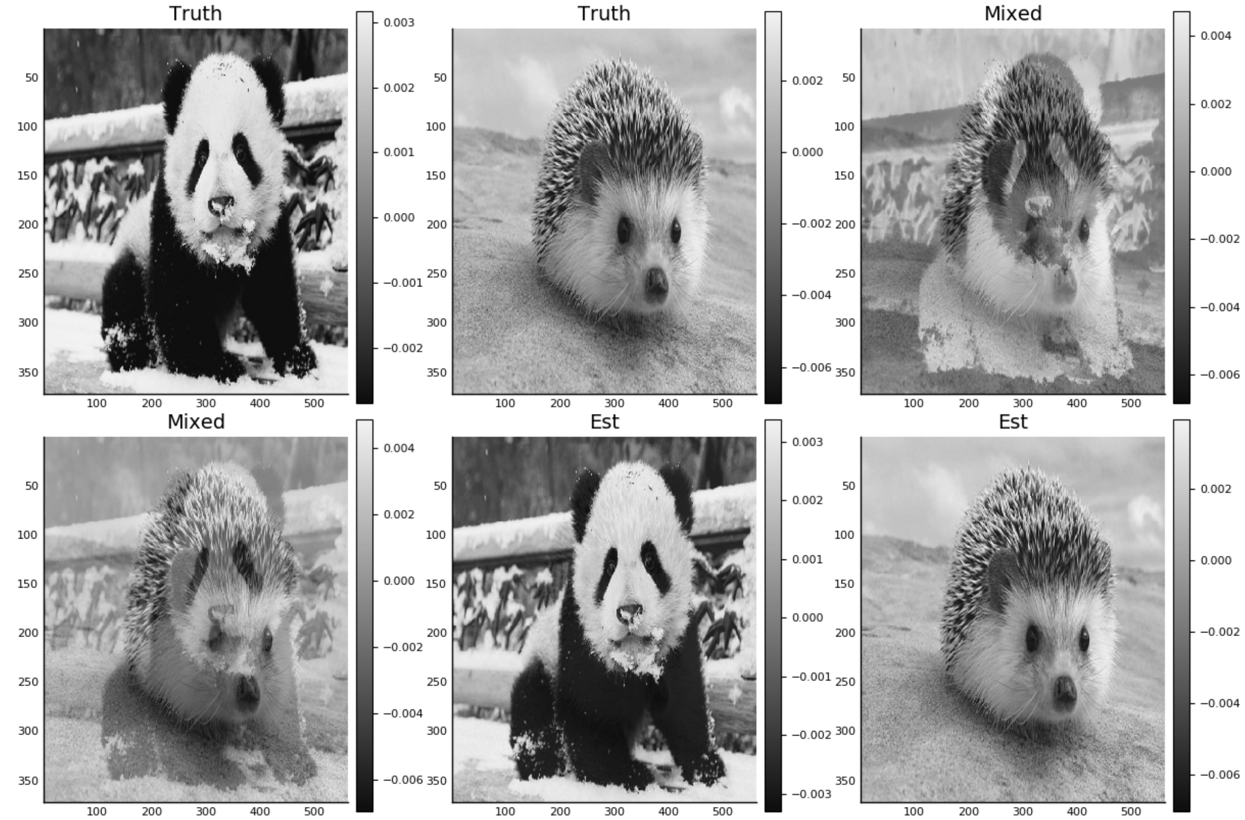


Figure 4.5: We present results for the orthogonal 2D-DMD algorithm for an orthogonal mixture of two images. We see that the algorithm successfully unmixes the two linearly mixed images, as expected.

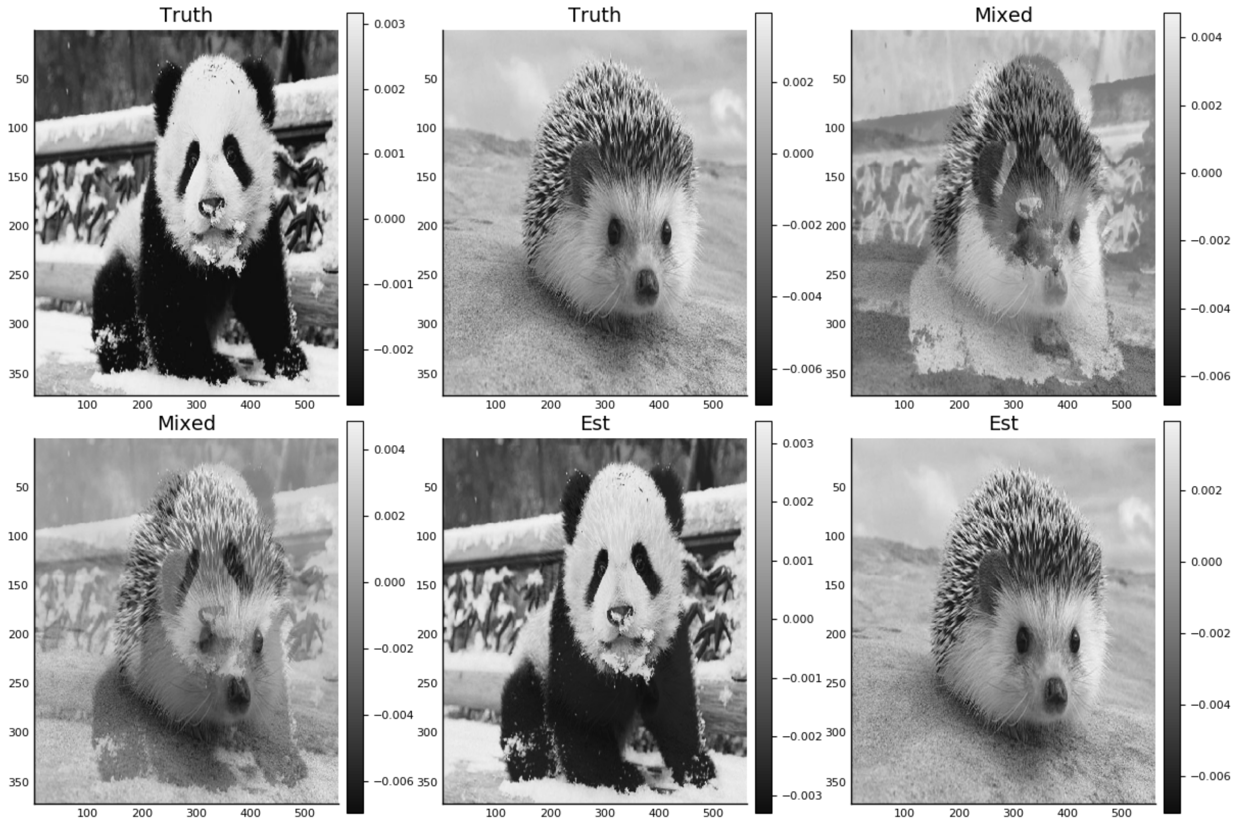


Figure 4.6: We present results for the orthogonal 2D-DMD algorithm for a non-orthogonal mixture of two images. We whiten the mixtures before applying the algorithm, and we see that the algorithm successfully unmixes the two linearly mixed images.

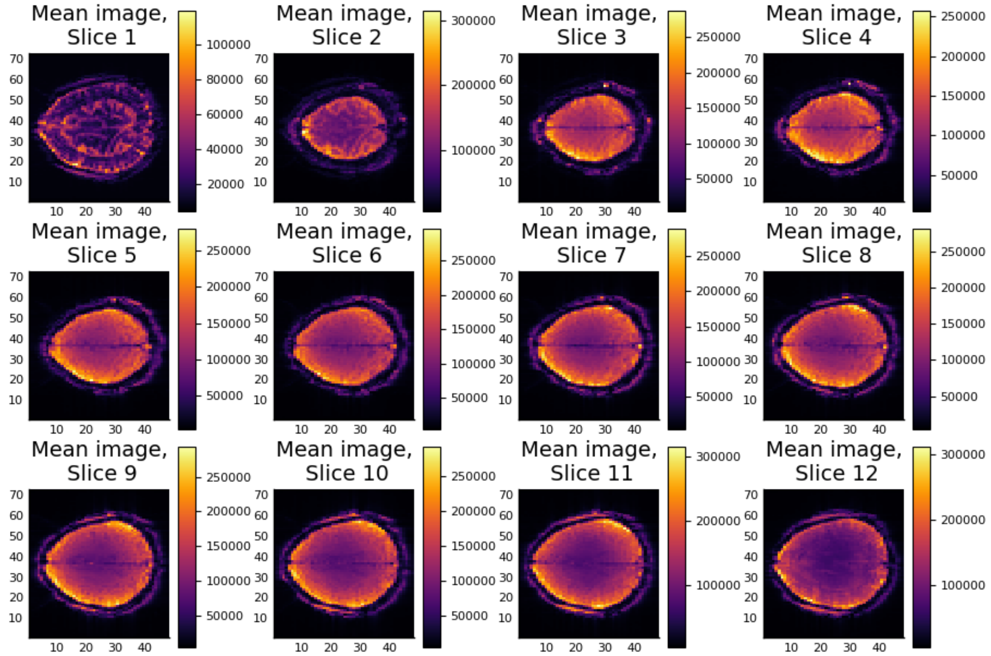


Figure 4.7: The mean image for the fMRI dataset.

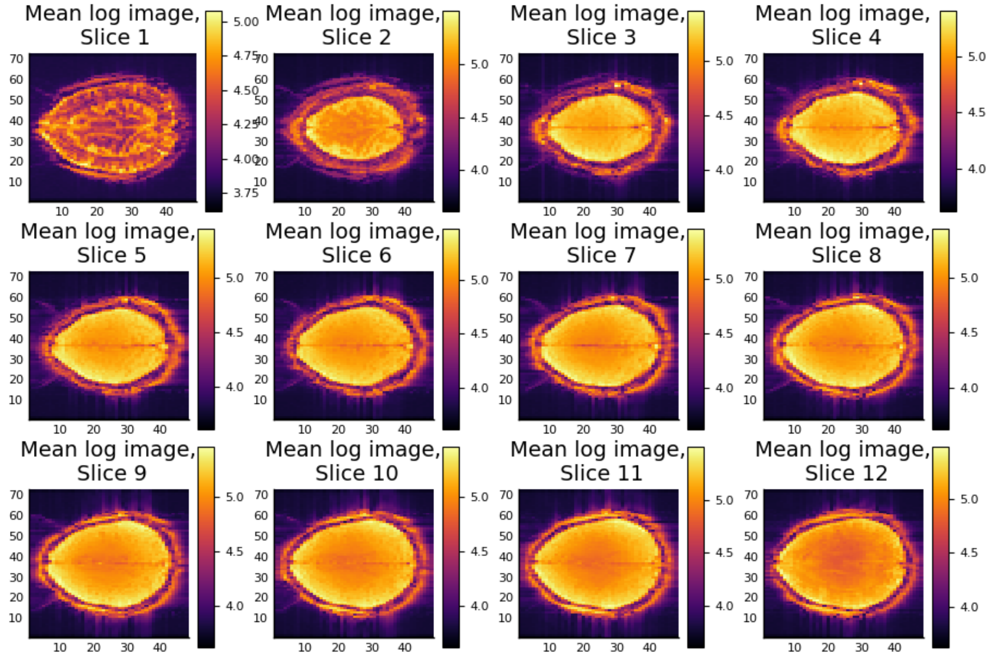


Figure 4.8: The mean image for the logged fMRI dataset.

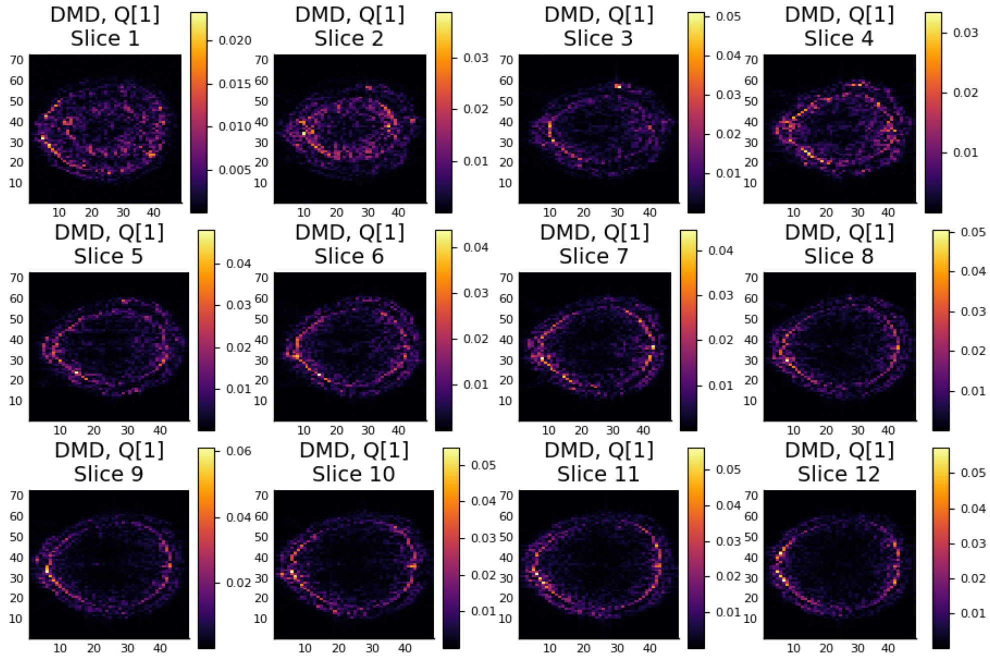


Figure 4.9: The leading DMD eigenvector for the fMRI dataset.

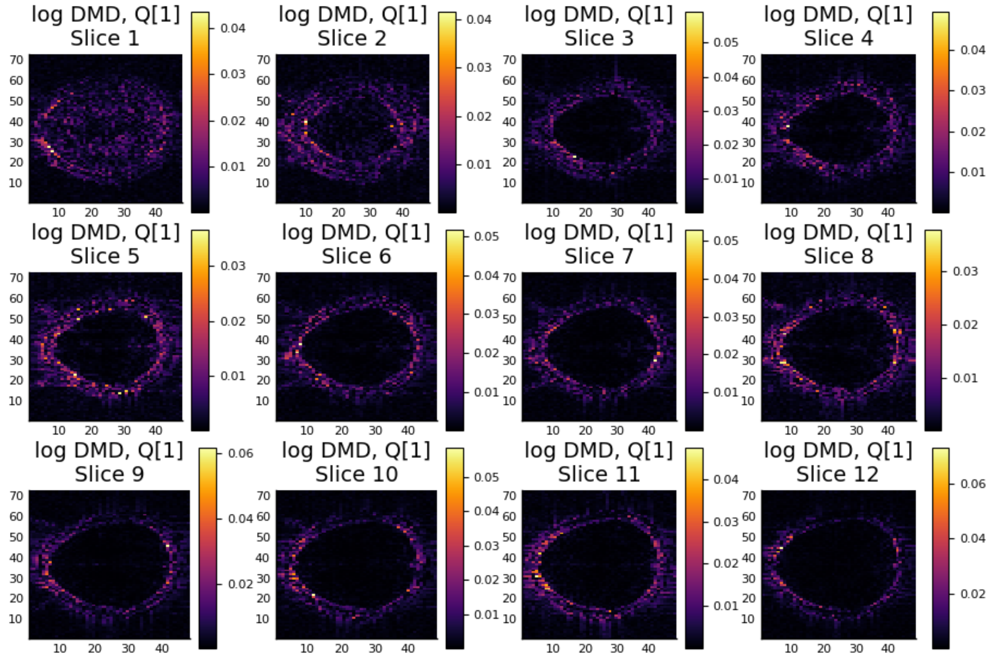


Figure 4.10: The leading DMD eigenvector for the logged fMRI dataset.

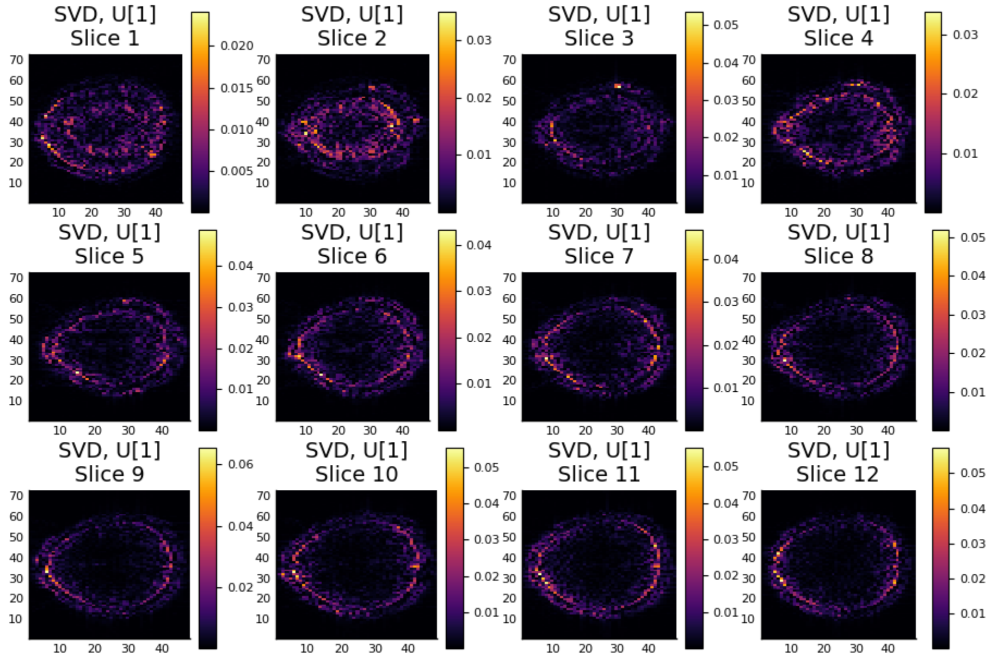


Figure 4.11: The leading left singular vector for the fMRI dataset.

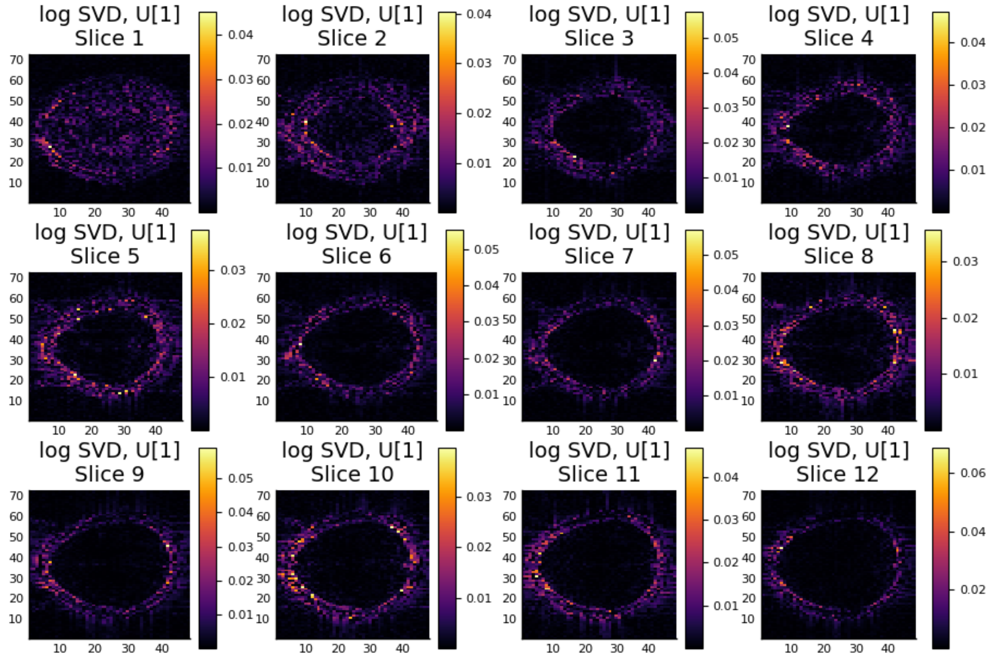


Figure 4.12: The leading left singular vector for the logged fMRI dataset.

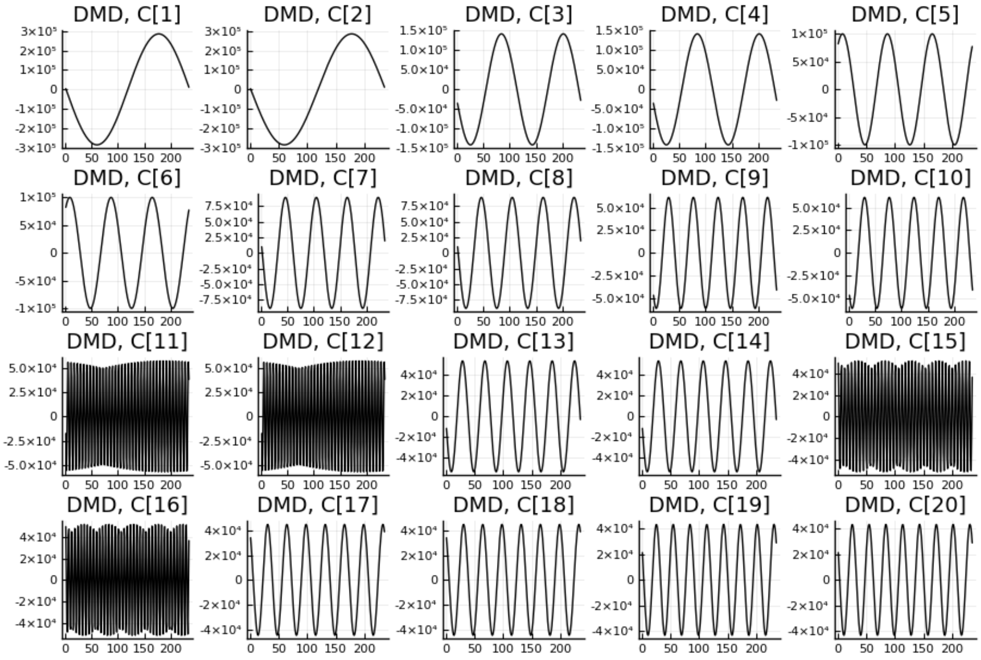


Figure 4.13: The leading DMD temporal variations for the fMRI dataset.

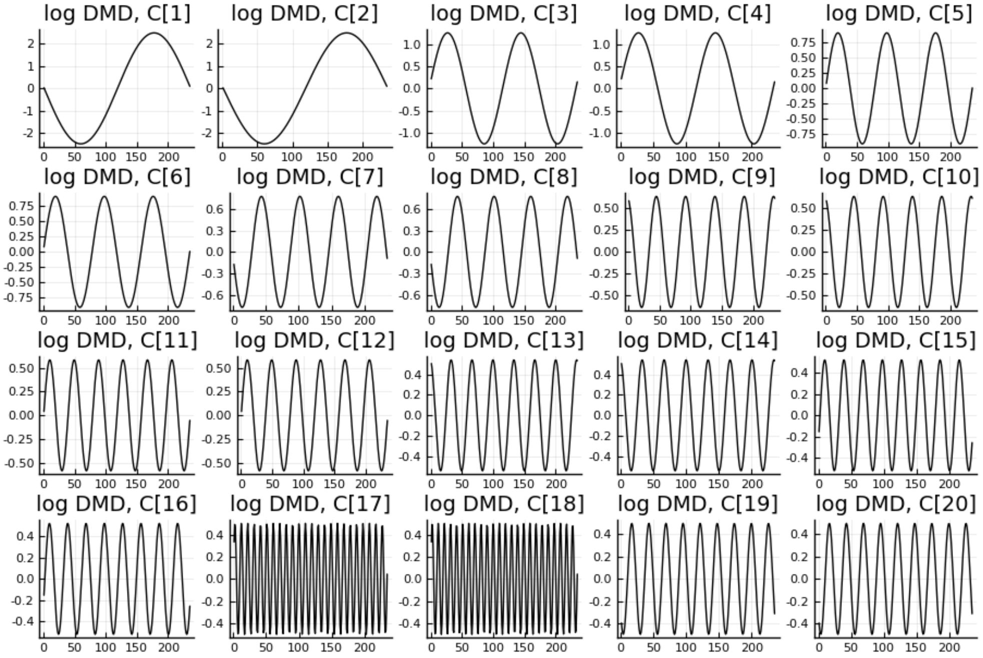


Figure 4.14: The leading DMD temporal variations for the logged fMRI dataset.

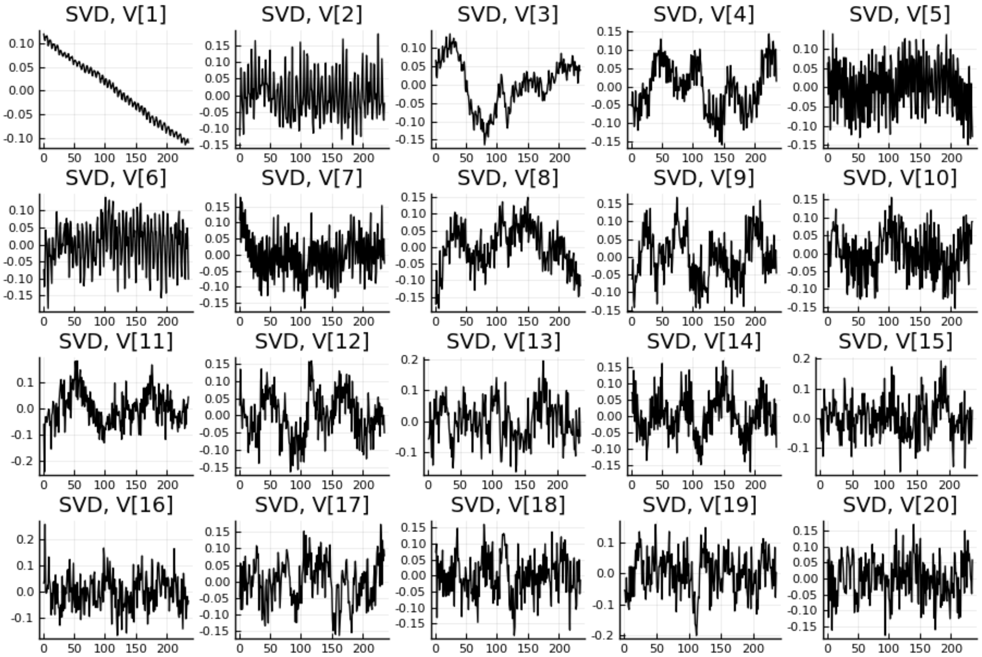


Figure 4.15: The leading right singular vectors for the fMRI dataset.

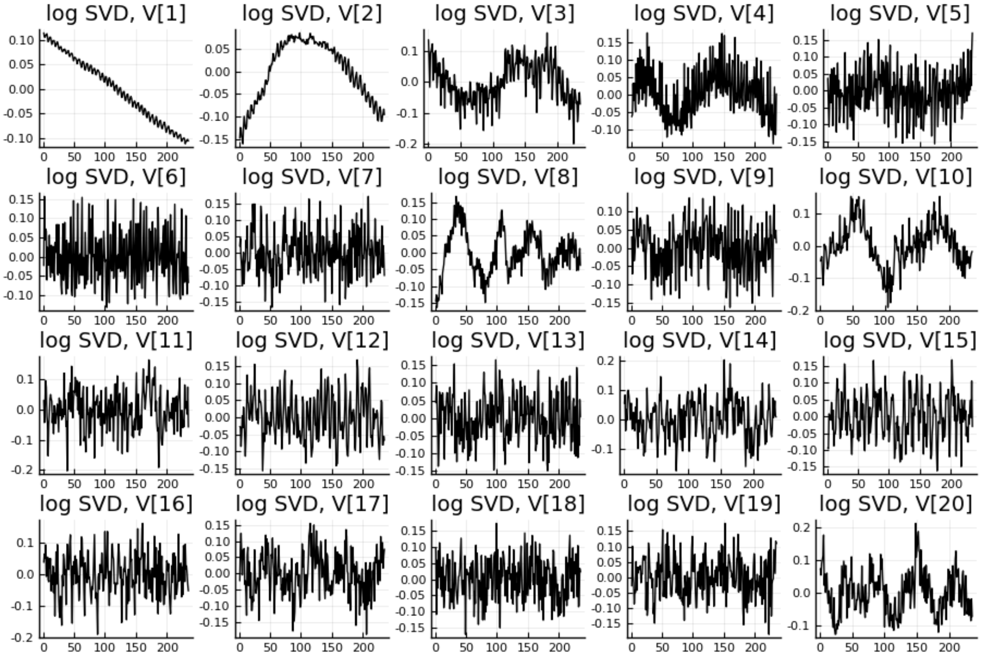


Figure 4.16: The leading right singular vectors for the logged fMRI dataset.

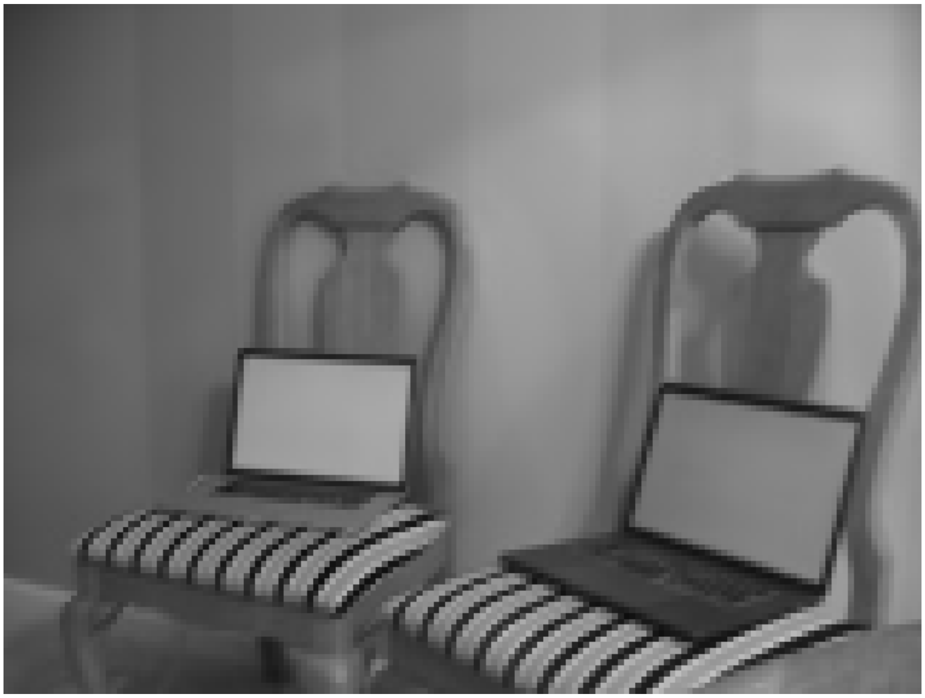


Figure 4.17: The mean frame (background) of the video.

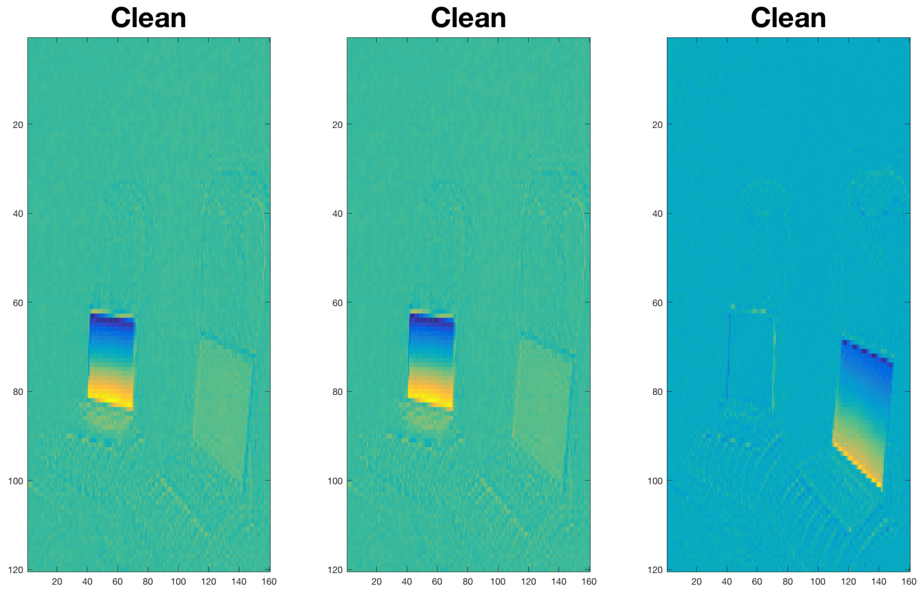


Figure 4.18: The modes from applying DMD to the video as-is. We see that the second screen is lost.

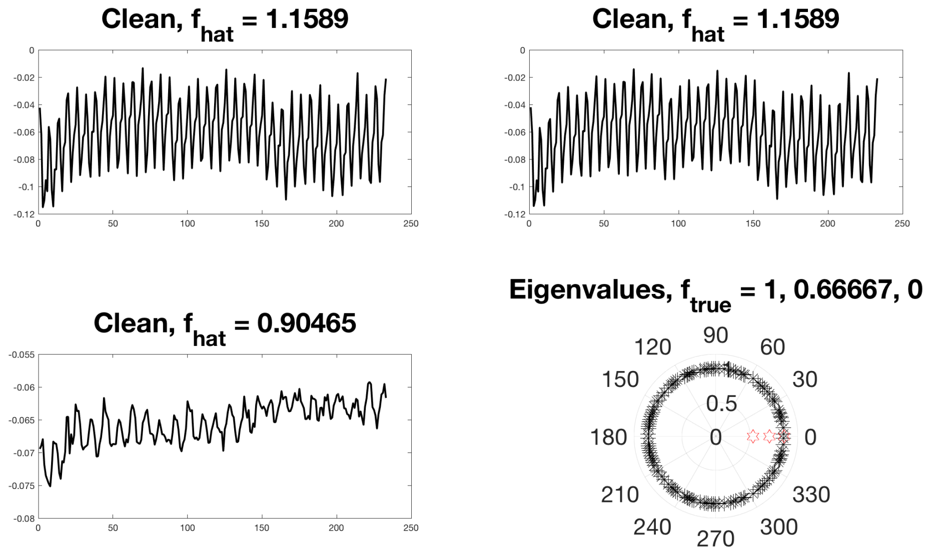


Figure 4.19: The temporal variations from applying DMD to the video as-is. We see that the second frequency is lost. In the eigenvalue plot, the black 'x' indicates the empirical DMD eigenvalues, and the red markers indicate the true values.

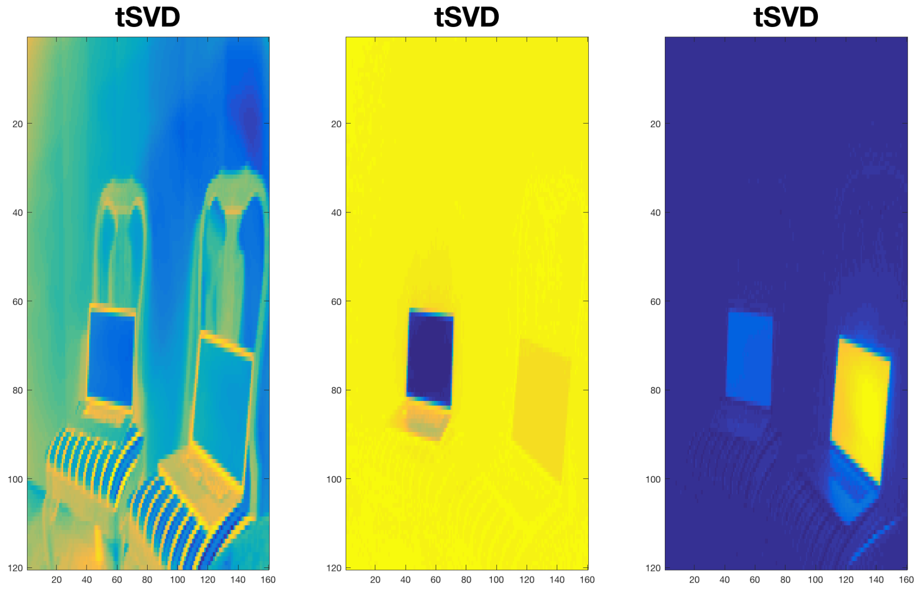


Figure 4.20: The modes from applying DMD to the truncated, rank-3 video. We see that all three components are present.

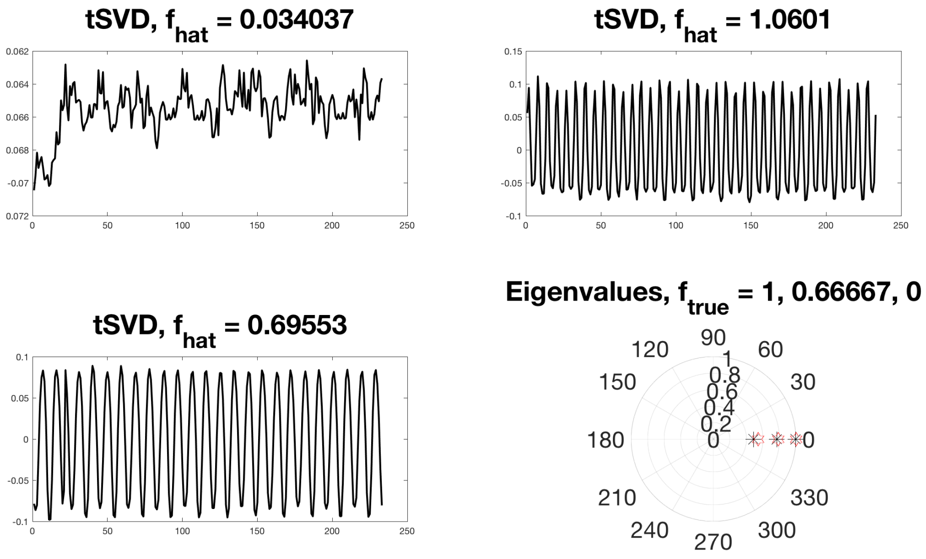


Figure 4.21: The temporal variations from applying DMD to the truncated, rank-3 video. We see that all three components are present. In the eigenvalue plot, the black 'x' indicates the empirical DMD eigenvalues, and the red markers indicate the true values.

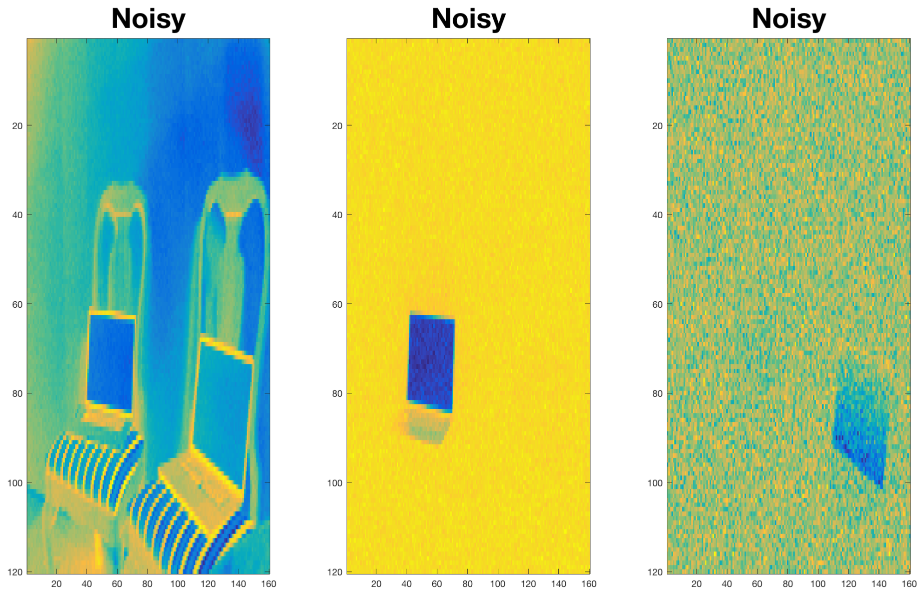


Figure 4.22: The modes from applying DMD to the noisy video as-is. We see that the second screen is present, but the quality of the estimated modes is poor.

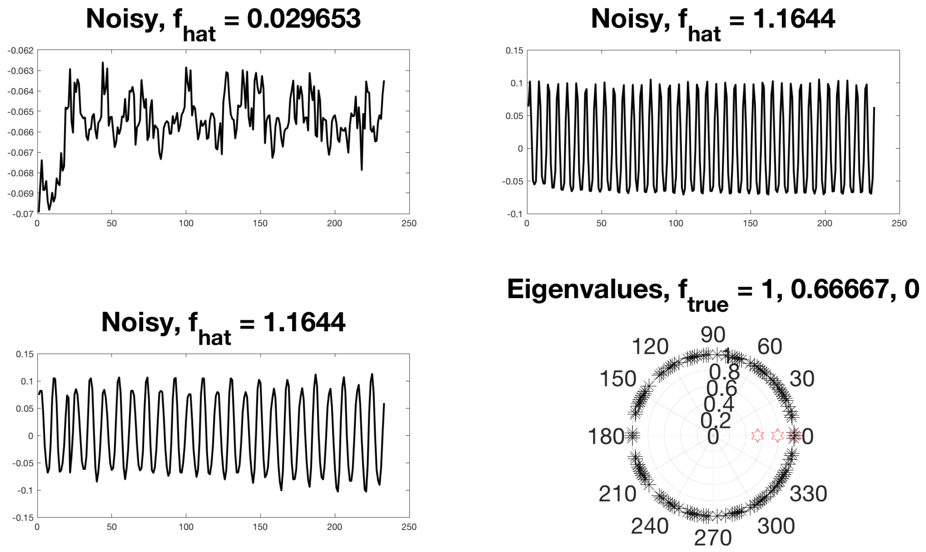


Figure 4.23: The temporal variations from applying DMD to the noisy video as-is. We see that the estimated frequencies are not accurate. In the eigenvalue plot, the black 'x' indicates the empirical DMD eigenvalues, and the red markers indicate the true values.

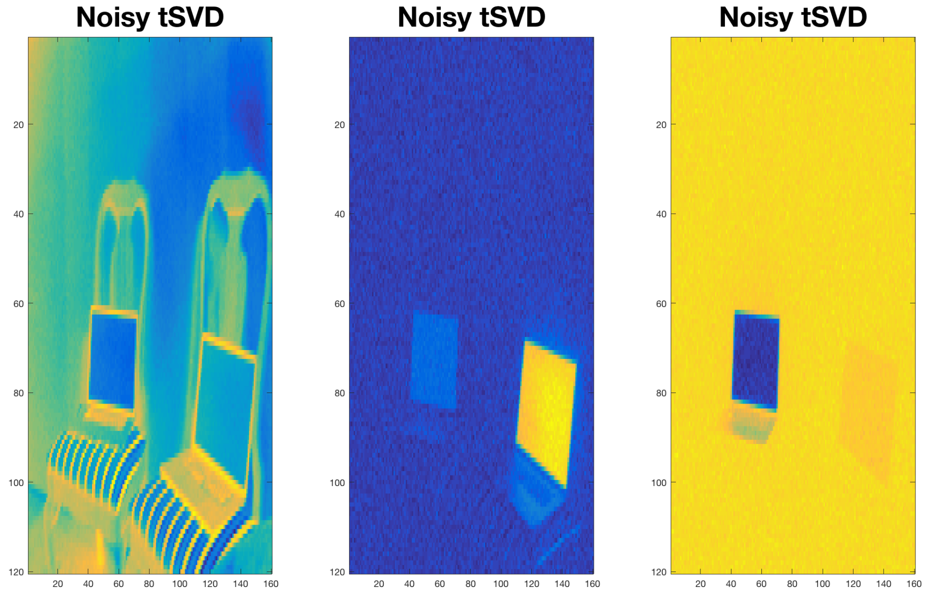


Figure 4.24: The modes from applying DMD to the noisy video after a rank-3 truncation. We see that all components are estimated well.

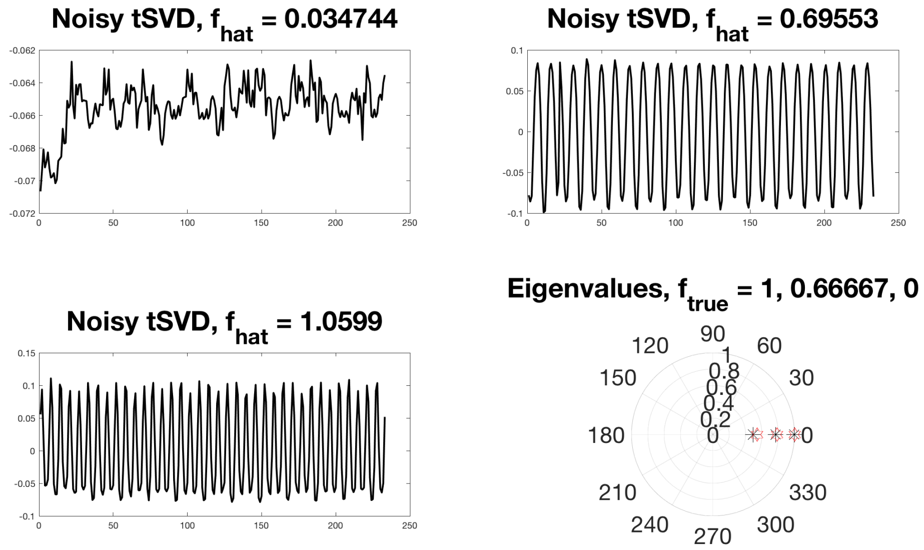


Figure 4.25: The temporal variations from applying DMD to the noisy video after a rank-3 truncation. We see that all frequencies are estimated well. In the eigenvalue plot, the black 'x' indicates the empirical DMD eigenvalues, and the red markers indicate the true values.

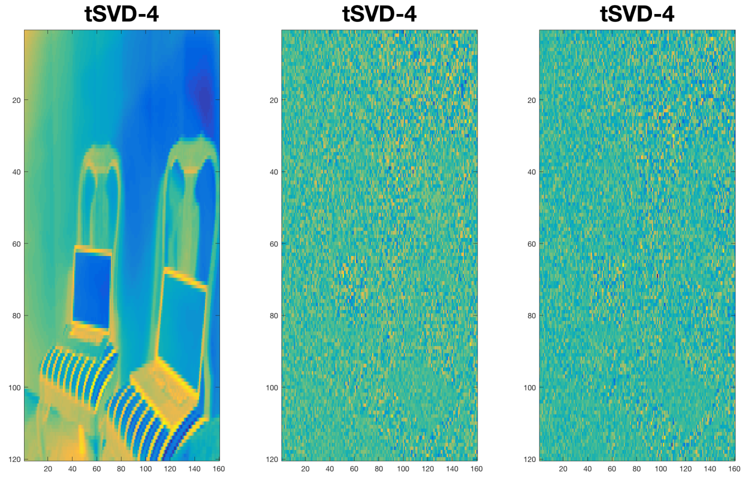


Figure 4.26: The modes from applying DMD to the truncated, rank-4 video. We see that of the three components, only the background is present.

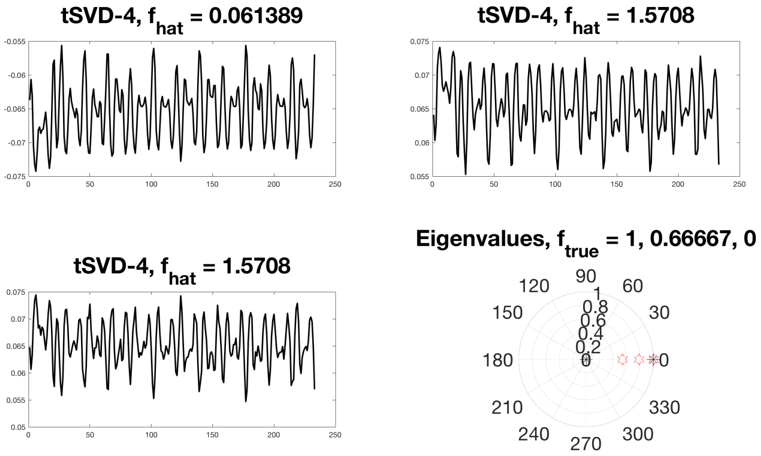


Figure 4.27: The temporal variations from applying DMD to the truncated, rank-4 video. We see that only the background is present. In the eigenvalue plot, the black 'x' indicates the empirical DMD eigenvalues, and the red markers indicate the true values.

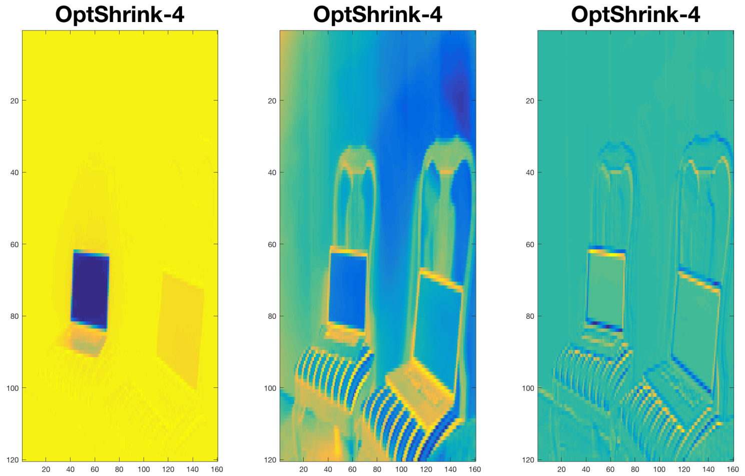


Figure 4.28: The modes from applying DMD to the OptShrink-truncated, rank-4 video. We see that of the three components, only the background is present.

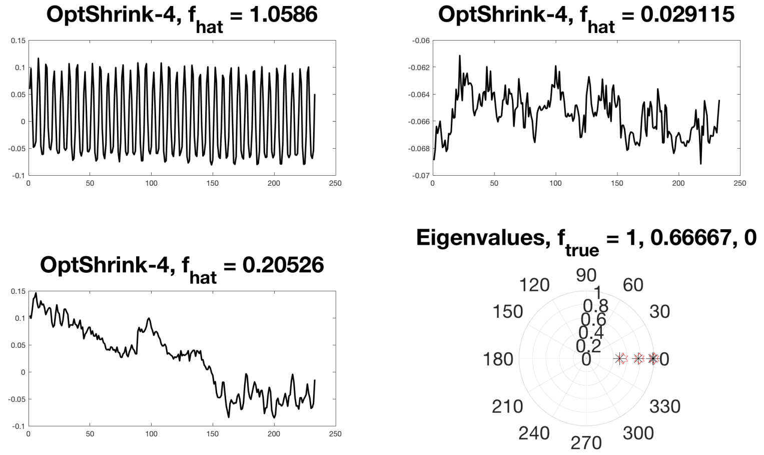


Figure 4.29: The temporal variations from applying DMD to the OptShrink-truncated, rank-4 video. We see that only the background is present. In the eigenvalue plot, the black 'x' indicates the empirical DMD eigenvalues, and the red markers indicate the true values.

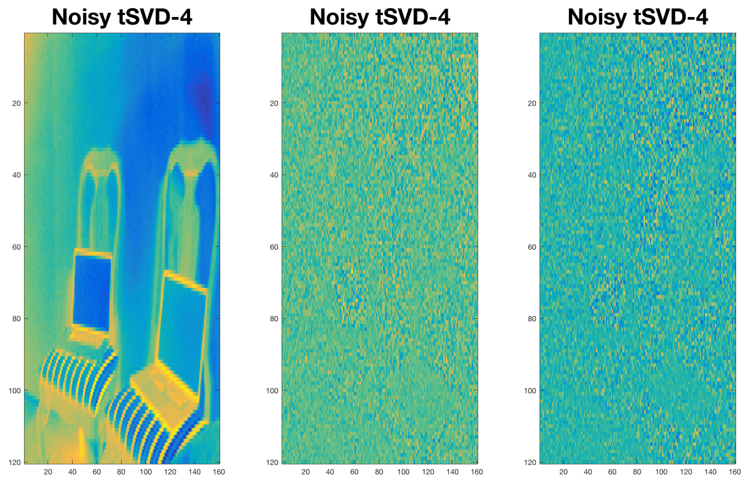


Figure 4.30: The modes from applying DMD to the noisy video after a rank-4 truncation. We see that only the background is estimated well.

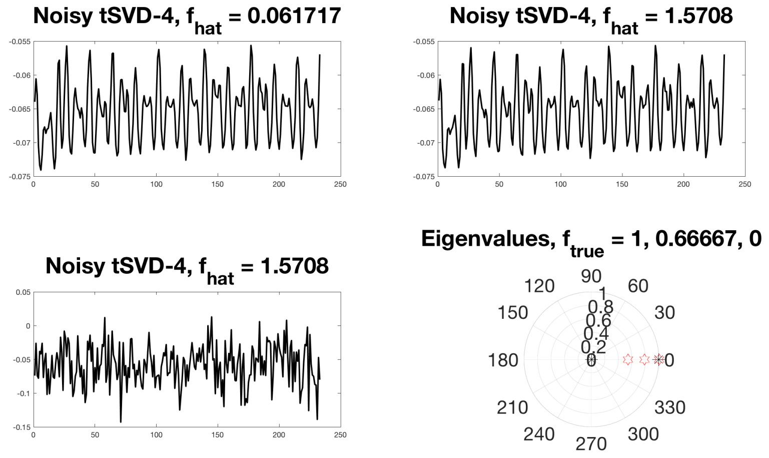


Figure 4.31: The temporal variations from applying DMD to the noisy video after a rank-4 truncation. We see that only the background is estimated well. In the eigenvalue plot, the black 'x' indicates the empirical DMD eigenvalues, and the red markers indicate the true values.

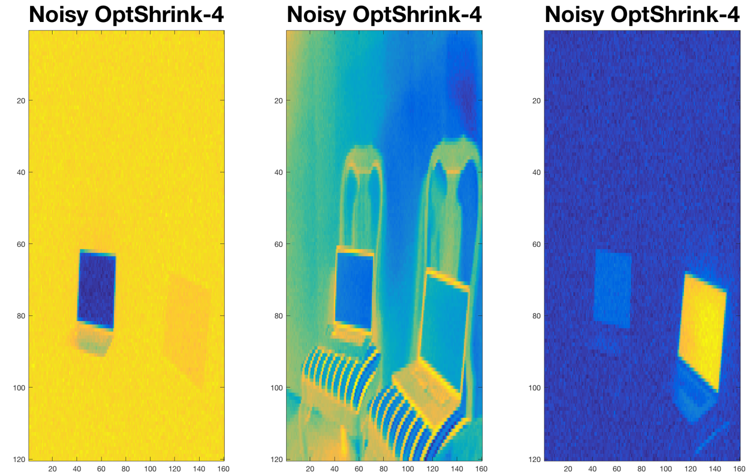


Figure 4.32: The modes from applying DMD to the noisy video after a rank-4 OptShrink truncation. We see that all three components are estimated well.

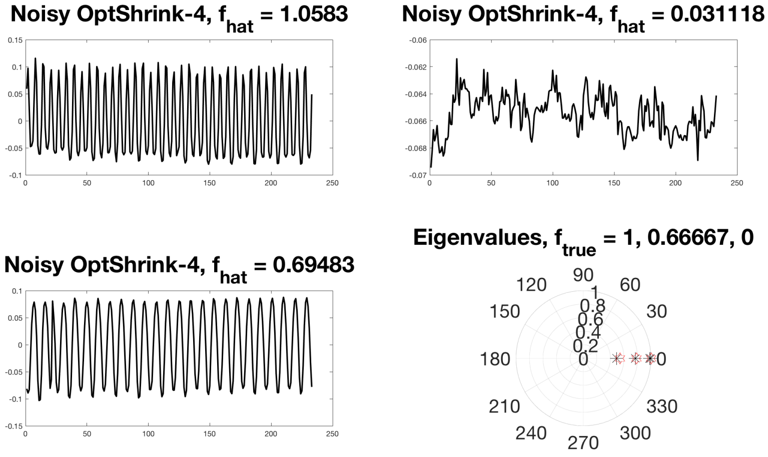
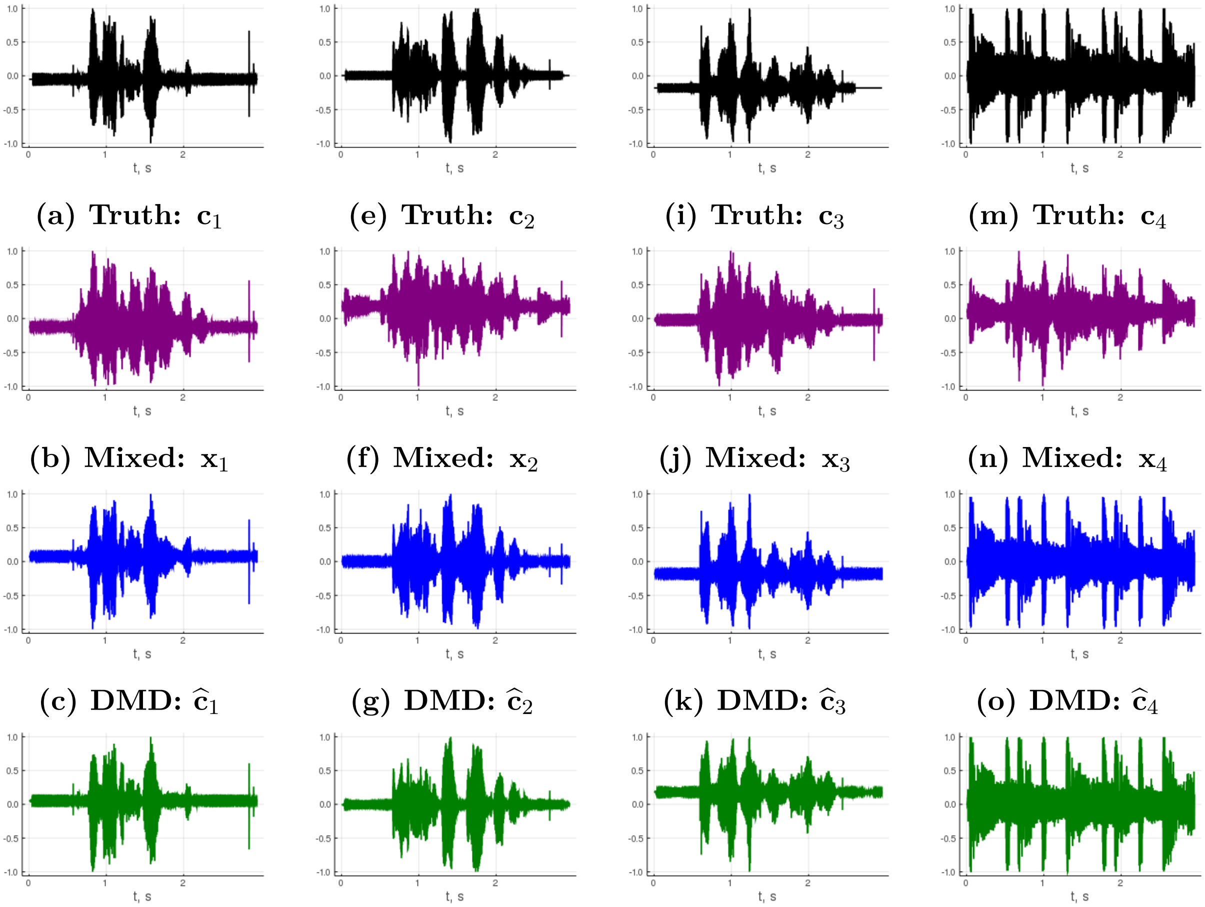


Figure 4.33: The temporal variations from applying DMD to the noisy video after a rank-4 OptShrink truncation. We see that all three components are estimated well. In the eigenvalue plot, the black 'x' indicates the empirical DMD eigenvalues, and the red markers indicate the true values.



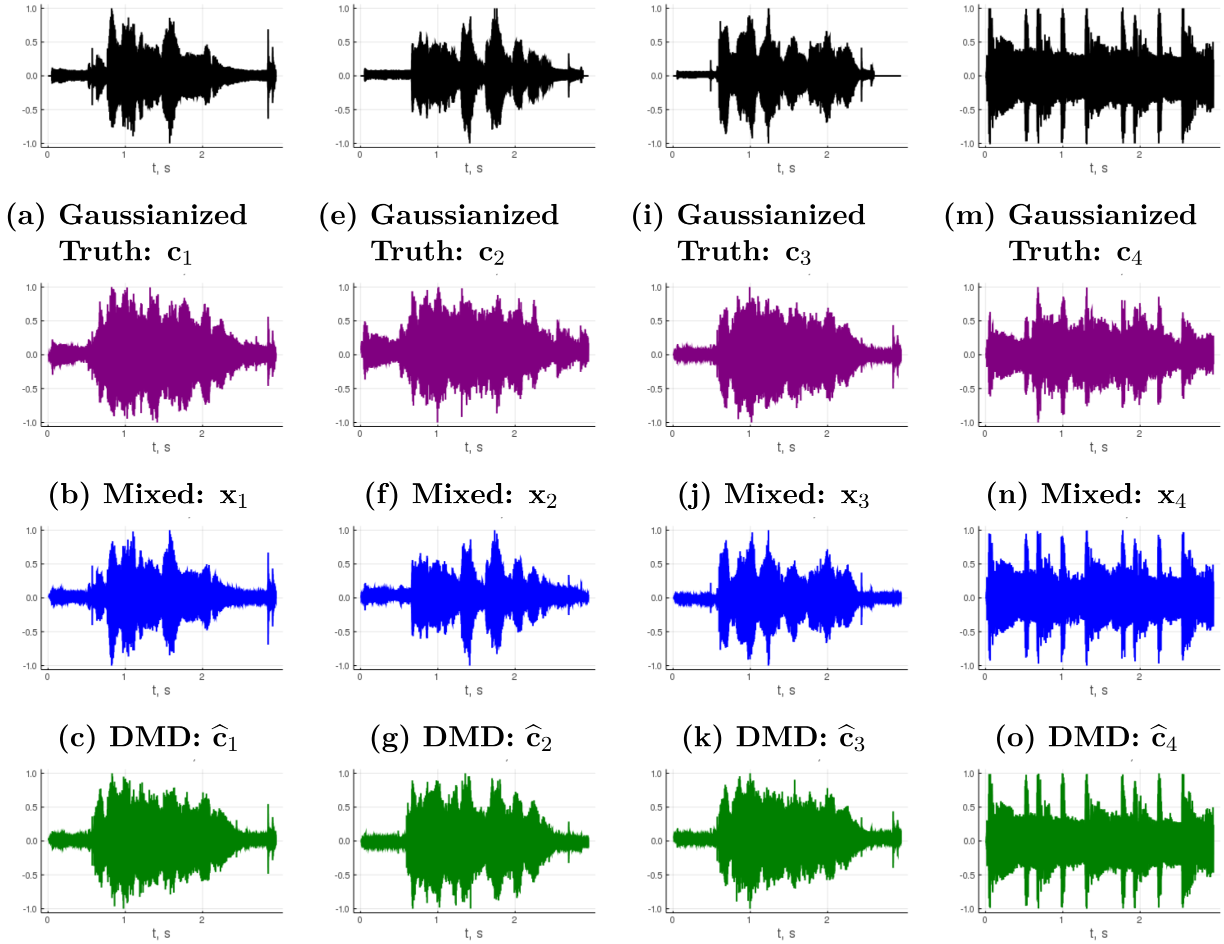
(a) Truth: $\mathbf{c}_1$ (e) Truth: $\mathbf{c}_2$ (i) Truth: $\mathbf{c}_3$ (m) Truth: $\mathbf{c}_4$

(b) Mixed: $\mathbf{x}_1$ (f) Mixed: $\mathbf{x}_2$ (j) Mixed: $\mathbf{x}_3$ (n) Mixed: $\mathbf{x}_4$

(c) DMD: $\widehat{\mathbf{c}}_1$ (g) DMD: $\widehat{\mathbf{c}}_2$ (k) DMD: $\widehat{\mathbf{c}}_3$ (o) DMD: $\widehat{\mathbf{c}}_4$

(d) ICA: $\widehat{\mathbf{c}}_1$ (h) ICA: $\widehat{\mathbf{c}}_2$ (l) ICA: $\widehat{\mathbf{c}}_3$ (p) ICA: $\widehat{\mathbf{c}}_4$

Figure 4.34: We present the waveforms for the true audio signals (three speakers and one music signal), the mixed signals, and the unmixed signals. We see that both DMD and ICA unmix the signals, with squared errors of 2×10^{-3} and 1×10^{-3}, respectively (defined in (2.39)).



(a) Gaussianized Truth: $\mathbf{c}_1$

(e) Gaussianized Truth: $\mathbf{c}_2$

(i) Gaussianized Truth: $\mathbf{c}_3$

(m) Gaussianized Truth: $\mathbf{c}_4$

(b) Mixed: $\mathbf{x}_1$

(f) Mixed: $\mathbf{x}_2$

(j) Mixed: $\mathbf{x}_3$

(n) Mixed: $\mathbf{x}_4$

(c) DMD: $\widehat{\mathbf{c}}_1$

(g) DMD: $\widehat{\mathbf{c}}_2$

(k) DMD: $\widehat{\mathbf{c}}_3$

(o) DMD: $\widehat{\mathbf{c}}_4$

(d) ICA: $\widehat{\mathbf{c}}_1$

(h) ICA: $\widehat{\mathbf{c}}_2$

(l) ICA: $\widehat{\mathbf{c}}_3$

(p) ICA: $\widehat{\mathbf{c}}_4$

Figure 4.35: We present the waveforms for the Gaussianized audio signals (three speakers and one music signal), the mixed signals, and the unmixed signals. We see that DMD unmixes the signals with a squared error of 2×10^{-1} (defined in (2.39)), whereas ICA fails completely.

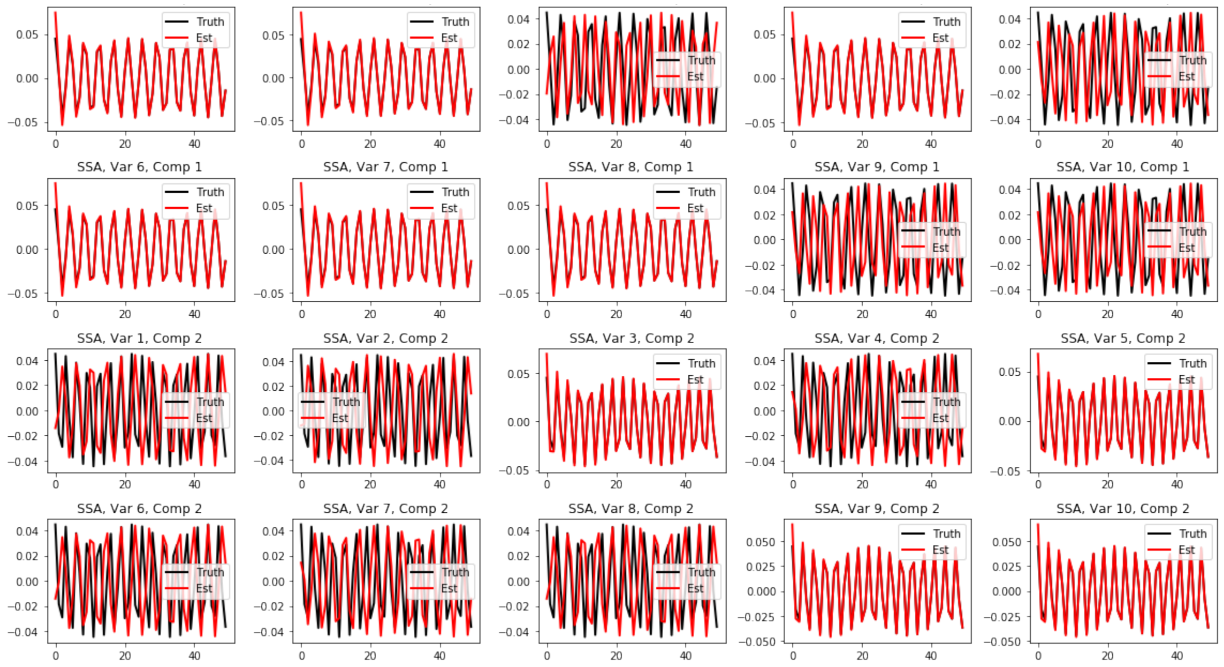


Figure 4.36: We present the waveforms for SSA applied to $p = 10$ mixtures of two cosines with $\omega_1 = 3/2$ and $\omega_2 = 2$. We see that SSA applied to the individual channels performs well.

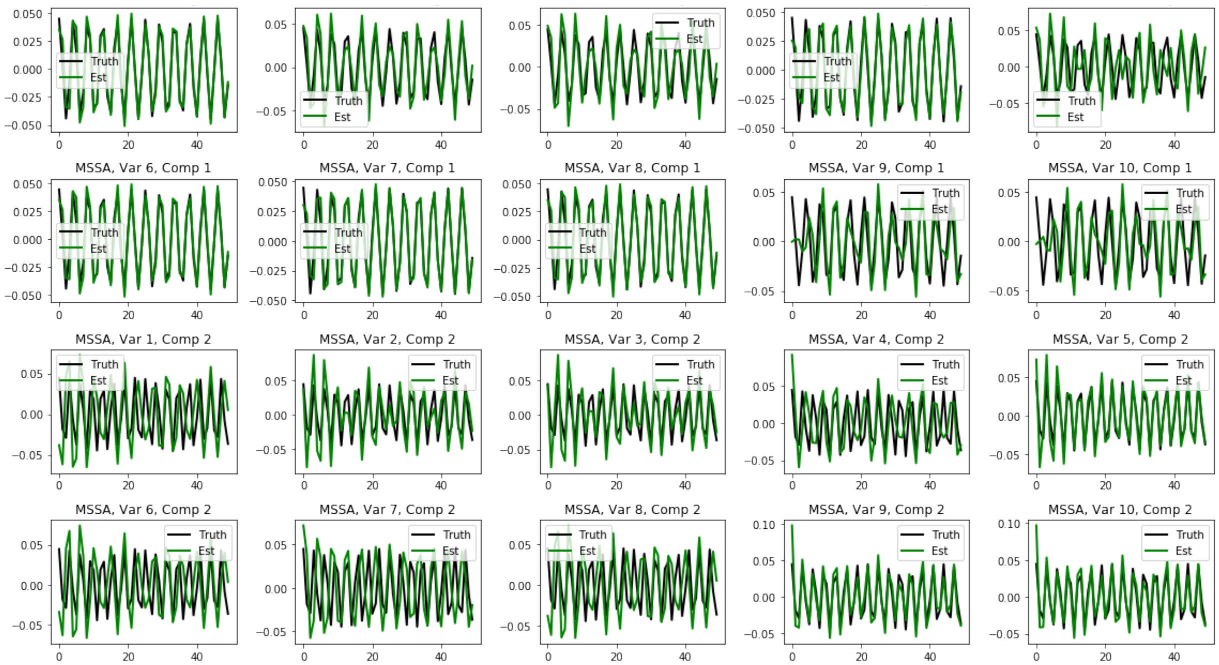


Figure 4.37: We present the waveforms for multivariate SSA (MSSA) applied to $p = 10$ mixtures of two cosines with $\omega_1 = 3/2$ and $\omega_2 = 2$. We see that MSSA performs well, but not as well as SSA (Figure 4.36).

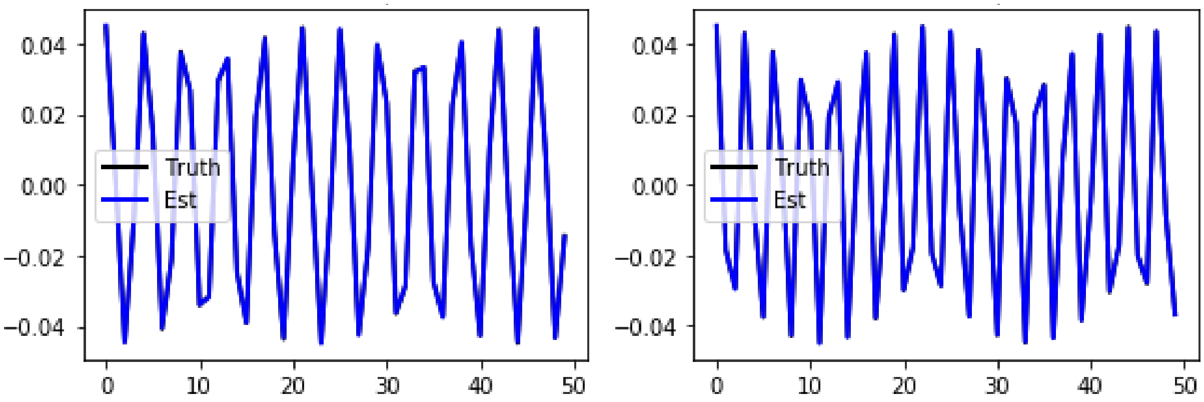


Figure 4.38: We present the waveforms for DMD applied to $p = 10$ mixtures of two cosines with $\omega_1 = 3/2$ and $\omega_2 = 2$. We see that DMD performs better than SSA (Figure 4.36) and MSSA (Figure 4.37).

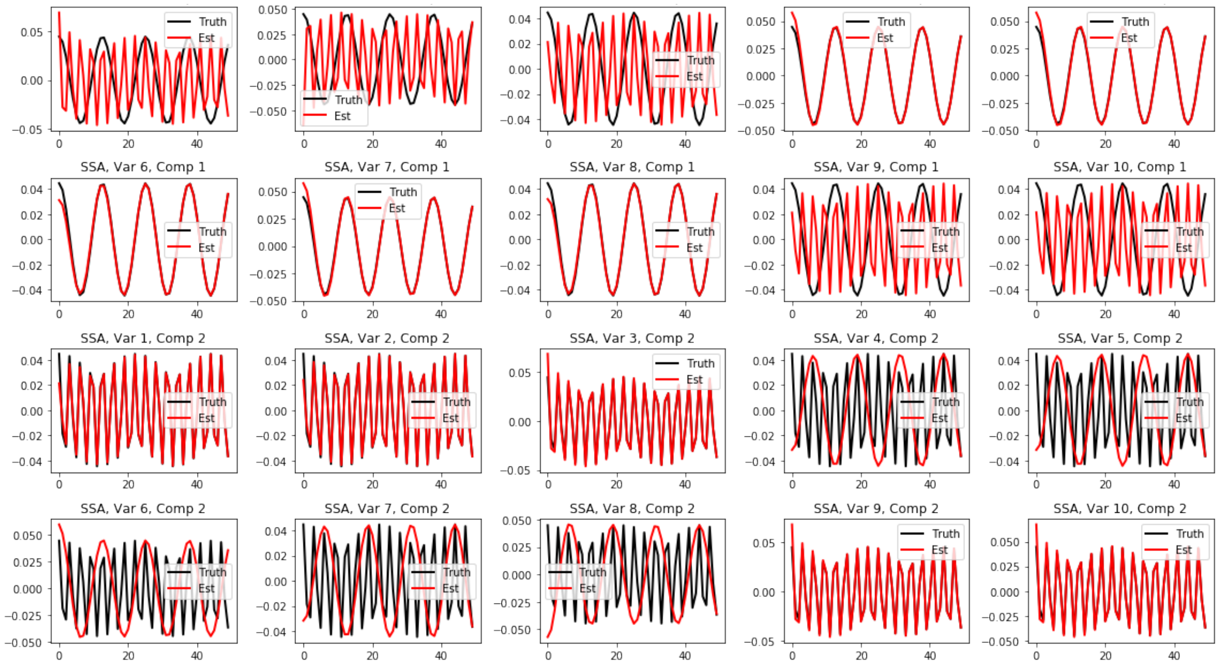


Figure 4.39: We present the waveforms for SSA applied to $p = 10$ mixtures of two cosines with $\omega_1 = 1/2$ and $\omega_2 = 2$. We see that SSA applied to the individual channels tends to lose one of the components.

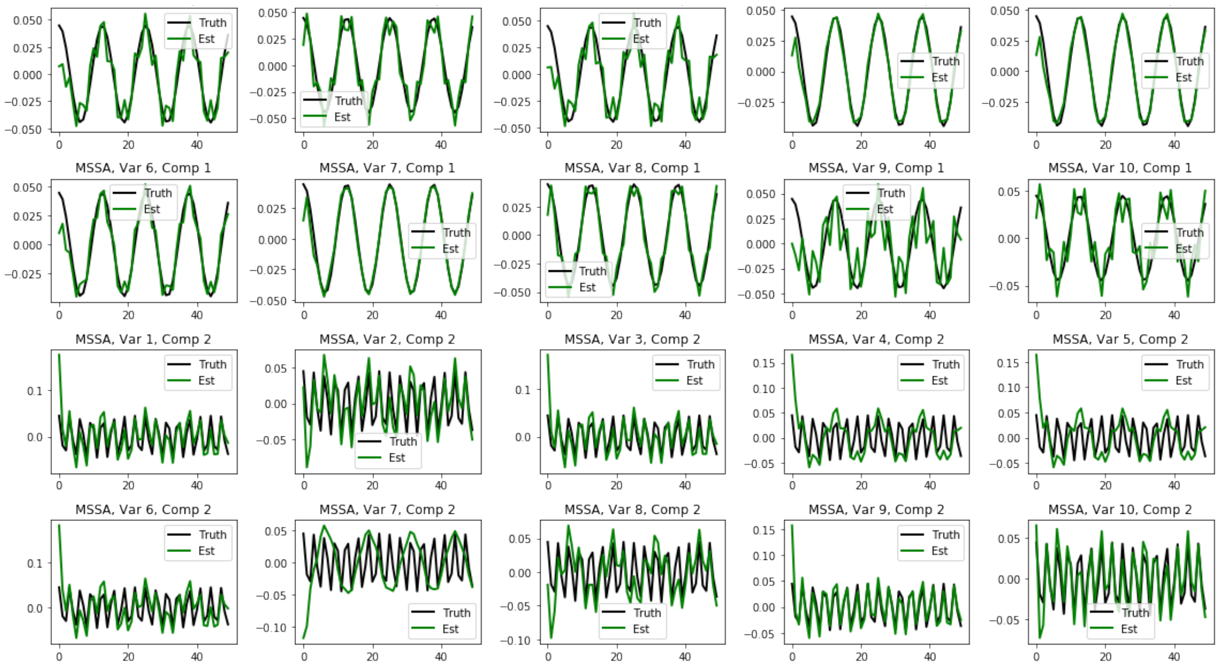


Figure 4.40: We present the waveforms for multivariate SSA (MSSA) applied to $p = 10$ mixtures of two cosines with $\omega_1 = 1/2$ and $\omega_2 = 2$. We see that MSSA performs better than SSA (Figure 4.36).

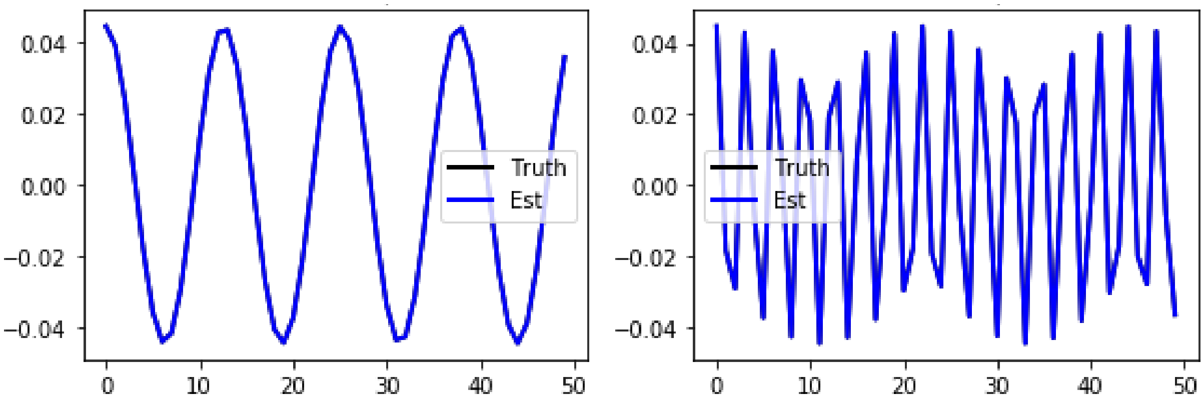


Figure 4.41: We present the waveforms for DMD applied to $p = 10$ mixtures of two cosines with $\omega_1 = 1/2$ and $\omega_2 = 2$. We see that DMD performs better than SSA (Figure 4.39) and MSSA (Figure 4.40).

Chapter 5

Sparse Equisigned PCA: Algorithms and Performance Bounds in the Noisy Rank-1 Setting

Singular value decomposition (SVD) based principal component analysis (PCA) breaks down in the high-dimensional and limited sample size regime below a certain critical eigen-SNR that depends on the dimensionality of the system and the number of samples. Below this critical eigen-SNR, the estimates returned by the SVD are asymptotically uncorrelated with the latent principal components. We consider a setting where the left singular vector of the underlying rank one signal matrix is assumed to be sparse and the right singular vector is assumed to be equisigned, that is, having either only nonnegative or only nonpositive entries. We consider six different algorithms for estimating the sparse principal component based on different statistical criteria and prove that by exploiting sparsity, we recover consistent estimates in the low eigen-SNR regime where the SVD fails. Our analysis reveals conditions under which a coordinate selection scheme based on a *sum-type decision statistic* outperforms schemes that utilize the ℓ_1 and ℓ_2 norm-based statistics. We derive lower bounds on the size of detectable coordinates of the principal left singular vector and utilize these lower bounds to derive lower bounds on the worst-case risk. Finally, we verify our findings with numerical simulations and a illustrate the performance with

a video data where the interest is in identifying objects.[3]

5.1 Introduction

It is well-understood that singular value decomposition (SVD) based principal component analysis (PCA) breaks down in the high-dimensional and limited sample size regime below a certain critical eigen-SNR (eigenvalue signal-to-noise ratio) that depends on the dimensionality of the system and the number of samples [61, 17]. Several sparse PCA algorithms have been proposed in the literature (see [61, 17, 31, 75, 134, 16]) and have been shown to successfully estimate the principal components in the low eigen-SNR regime where the SVD fails.

Prior work in this area primarily considers the Gaussian signal-plus-noise model with random effects, where the signal matrix is assumed to have sparse left singular vectors, normally distributed right singular vectors, and the noise matrix is assumed to have normally distributed *i.i.d.* entries. Here, we consider the setting where the left singular vector of the rank one signal matrix is sparse *and* the right singular vector is assumed to be equisigned. We say that a vector is equisigned if its entries are all non-negative or all non-positive. This is motivated by applications such as diffusion imaging in MRI where the right singular vector represents a physical quantity (e.g. intensity as the diffusion agent is absorbed by a tissue) that is non-negative, by imaging problems such as foreground-background separation in video data [99, 127] and object detection in astronomy [107], where the data are naturally non-negative, and by problems in bioinformatics where the data are (non-negative) counts of genes [118]. When analyzing data that are non-negative, it is logical to take advantage of this property, and investigate how we may use this knowledge to do better than the (generic) alternatives. Alternatively, a practitioner may seek to use techniques that constrain or impose non-negativity to preserve interpretability of the results, e.g., non-negative matrix factorization. Additionally, we motivate the rank-1 assumption by noting that for a video with a static background, the foreground is a perturbation of a rank-1 background [86, 42]. Finally, even though we do not pursue this angle here, our framework can be extended to deal with the scenario where the signal can be viewed of a rank 1 tensor with all but one of the representors in the Kroneker product representation of the tensor is an equisigned vector.

[3] This chapter describes joint work with Raj Rao Nadakuditi and Debashis Paul, and has appeared in [103].

There is precedent for and prior work on non-negative PCA, including the sparse biased PCA in [23], the sparse PCA with non-negativity priors in [98], and the work in [85]. These works differ from our work in that they impose non-negativity on the factors or the left singular vectors. In this work, we study sparse factors with non-negative loadings; i.e., we are solving a different problem in this work.

A natural question at this juncture is the following: *how does our problem differ from that solved by Non-Negative Matrix Factorization (NNMF)?* NNMF takes a given matrix $\mathbf{X}$ and looks for non-negative matrices $\mathbf{F}$ and $\mathbf{G}$ such that $\mathbf{X} = \mathbf{F}\,\mathbf{G}^T$ [54, 128]. Ordinary NNMF has no sparsity constraints. We might impose such constraints, as is done in [53] and [73], but except in special cases, these solutions have no known theoretical guarantee of statistical performance. This problem partly stems from the fact that solutions to the corresponding optimization problems may not be unique. In contrast, our problem only constrains the right singular vectors, while the left singular vectors are free to take any sign. The work in [34] extends the NNMF framework to one wherein only one of the factors is non-negative; nevertheless, the rest of the constraints we impose are not included. The work in [135] seeks factors (left singular vectors) with disjoint supports and non-negative loadings, but this definition of sparsity does not match that from the sparse PCA literature. Hence, NNMF is not an answer to the problem we consider herein.

The main contribution of this chapter is a rigorous sparsistency analysis of the various algorithms that brings into focus the various very-low eigen-SNR regimes where the new algorithms work and the SVD based methods provably fail. Additionally, a major novelty of this work is the integration of FDR-controlling (False Discovery Rate) hypothesis testing to the Sparse PCA problem.

Our analysis illustrates the situations where the sum based coordinate selection scheme dramatically outperforms the ℓ_1 and ℓ_2 [61, 17] based sparse PCA schemes. Additionally, our proposed algorithms are non-iterative, do not require the computation of the sample covariance matrix, and do not require knowledge of the sparsity level. We separate our algorithms into two groups: one where the Family-Wise Error Rate (FWER) is controlled, and another where the False Discovery Rate (FDR) is controlled. We utilize sharp tail probability bounds for relevent statistics to derive our FWER-controlling estimators [20]. For the FDR controlling estimators, we relate the problem at hand to that of the sparse normal means problem [36].

This chapter is organized as follows. In Section 5.3, we describe three algorithms for estimating the sparse principal component that utilize a coordinate selection scheme based on the sum, ℓ_1, and ℓ_2 norm-based statistics respectively. We call our family of algorithms *SEPCA*, an abbreviation for Sparse Equisigned PCA. Section 5.4 proposes three FDR-controlling refinements of the sum- and ℓ_2-based algorithms in Section 5.3 by relating coordinate detection to the sparse normal means estimation problem. In Section 5.5 we show how the estimation performance is governed by the size of the smallest detectable coordinate, which we analyze in Section 5.6 and validate using numerical simulations in Section 5.7. In Section 5.8, we provide some geometric intuitions about the relative performance of three of our algorithms. We show that the sum statistic is potentially the most powerful, while the ℓ_1 is the least powerful. We provide some concluding remarks in Section 5.9.

5.2 Problem Formulation

Let $\mathbf{X} \in \mathbb{R}^{p \times n}$ be a real-valued signal-plus-noise data matrix of the form

$$\mathbf{X} = \theta \, \mathbf{u} \, \mathbf{v}^T + \sigma \, \mathbf{G} \, . \tag{5.1}$$

The columns of the $p \times n$ data matrix $\mathbf{X}$ represent p-dimensional observations. In (5.1), $\mathbf{u}$ and $\mathbf{v}$ are the left and right singular vectors of the rank-one latent signal matrix, and have entries u_i and v_j, respectively. The entries of $\mathbf{G}$, the noise matrix, are assumed to be *i.i.d.* Gaussian random variables with mean 0 and variance $1/n$. We assume that $\mathbf{u} \in \mathbb{R}^p$ has unit norm and is sparse in the sense of small ℓ_0 norm, with $s \ll p$ non-zero entries, where $s/n \to 0$. That is, for a set $I = \{i_1, \cdots, i_s\} \subset \{1, \cdots, p\}$,

$$\begin{aligned} u_i &\neq 0 \quad \text{for } i \in I, \\ u_i &= 0 \quad \text{for } i \in I^C, \end{aligned} \tag{5.2}$$

where I^C denotes the complement of I. We further assume $\mathbf{v} \in \mathbb{R}^n$ to be of unit norm, deterministic, and equisigned. Given $\mathbf{X}$, our goal is to recover $\mathbf{u}$ and $\mathbf{v}$.

Note that the (i, k) entry of $\mathbf{X}$, X_{ik}, is a Gaussian random variable with mean $[\theta u_i] v_k$ and variance σ^2/n. Moreover, it follows that

$$\mathbb{E}(\mathbf{X} \, \mathbf{X}^T) = \theta^2 \, \mathbf{u} \, \mathbf{u}^T + \sigma^2 \, \mathrm{I}_p,$$

where I_p denotes the $p \times p$ identity matrix. The quantity $(\theta/\sigma)^2$ is, for this model, the eigen-SNR (signal-to-noise ratio).

5.2.1 Motivation: Breakdown of PCA / SVD

From [12], we have the following result: let $\widehat{\mathbf{u}}$ be the estimate of $\mathbf{u}$ given by the Singular Value Decomposition (SVD) of $\mathbf{X}$, and let $p(n)/n$ have limit $c \in [0, \infty]$ as n grows, with θ fixed and $\sigma = 1$. Then, with probability 1,

$$|\langle \widehat{\mathbf{u}}, \mathbf{u} \rangle|^2 \rightarrow \begin{cases} 1 - \frac{c\left(1+\theta^2\right)}{\theta^2(c+\theta^2)} & \text{if } \theta \geq c^{1/4}, \\ 0 & \text{otherwise.} \end{cases} \tag{5.3}$$

For general σ, we replace θ by θ/σ in (5.3). Hence, SVD based PCA leads to inconsistent estimates of $\mathbf{u}$ (and also for $\mathbf{v}$, which can be deduced from (5.3)) when the dimension p is comparable to or larger than the sample size n. Moreover, in the low eigen-SNR regime, the estimates break down completely. SVD does not exploit any assumed structure in $\mathbf{u}$ and $\mathbf{v}$. Consequently, (5.3) holds for arbitrary $\mathbf{u}$ and $\mathbf{v}$, including our setting where $\mathbf{u}$ is sparse and/or $\mathbf{v}$ is equisigned. Our goal, in what follows, is to derive consistent estimators for $\mathbf{u}$ and $\mathbf{v}$ that outperform the SVD by exploiting the sparsity of $\mathbf{u}$ and the equisigned nature of $\mathbf{v}$.

5.2.2 Problem Statement

Note that we have assumed that $\mathbf{u}$ is sparse and that the sparsity s is such that s/n has limit zero. Hence, if we had oracle knowledge of the sparsity pattern I (the indices of $\mathbf{u}$ that have non-zero coordinates), restricting the matrix X to those rows indexed by I and performing the SVD on the smaller matrix would yield a consistent estimator for the non-zero elements of $\mathbf{u}$ and the vector $\mathbf{v}$. This conclusion follows from (5.3), since the value c is replaced with s/n, which has limit zero. Thus, if we derived consistent estimators of the support of $\mathbf{u}$, we have a consistent two-stage estimation procedure of the vectors $\mathbf{u}$ and $\mathbf{v}$.

Formally, we are interested in finding a procedure that estimates I by $\widehat{I}$ such that in the limit $n \rightarrow \infty$,

$$\begin{cases} \mathbb{P}\left(i \in \widehat{I}\right) \rightarrow 1 & \text{if } i \in I, \\ \mathbb{P}\left(i \in \widehat{I}\right) \rightarrow 0 & \text{if } i \notin I. \end{cases} \tag{5.4}$$

Equivalently, noting that the Hamming distance of I and $\widehat{I}$, denoted by $d_H\left(I, \widehat{I}\right)$, is given by the cardinality of their symmetric set difference,

$$d_H\left(I, \widehat{I}\right) = \left|\left(I \cup \widehat{I}\right) \setminus \left(I \cap \widehat{I}\right)\right|,$$

we want the expected Hamming distance $\mathbb{E}\, d_H\left(I, \widehat{I}\right)$ to have limit 0, which is stronger than requiring consistency in recovering the support (or sparsity pattern) of $\mathbf{u}$. However, as the work in [24, 106] indicates, this limit will not in general be zero, and will depend on the noise level, signal strength, and sparsity.

5.3 Proposed Algorithms

We propose six different two-stage algorithms for estimating $\mathbf{u}$. The first three algorithms are designed to control the family-wise error rate (FWER), or, the probability of obtaining a false positive in the coordinate selection. The last three algorithms aim to control the false discovery rate (FDR), or, the proportion of false discoveries (coordinate detections) among all discoveries. We defer discussion of the FDR-based algorithms to Section 5.4.

All of the algorithms have the same basic form given in Algorithm 2. Given $\mathbf{X}$, we associate a test statistic T_i to each row of $\mathbf{X}$. The sparsity of $\mathbf{u}$ implies that the majority of the rows of $\mathbf{X}$ are purely noise, so that the majority of the T_i come from the null, noise-only distribution. Hence, based on the statistics $\{T_i\}$, we perform a form of multiple hypotheses testing procedure, and select the set $\widehat{I}$ of indices that are non-null. In this way, we can estimate the *support* of $\mathbf{u}$, thereby isolating the the rows of $\mathbf{X}$ that contain the signal. Then, taking the SVD of this submatrix (comprised of only the selected rows of $\mathbf{X}$) yields a better estimate of the non-zero coordinates in $\mathbf{u}$, as well as $\mathbf{v}$.

We begin by discussing the FWER-controlling algorithms. The work in [61] proposed a covariance thresholding method for Sparse PCA called DT-SPCA; this is equivalent to a coordinate selection scheme based on the ℓ_2 norm-based statistic. In our terminology and with our choice of thresholds, we label it as ℓ_2-*SEPCA*. We label the coordinate selection scheme based on the ℓ_1 norm-based statistic ℓ_1-*SEPCA*. Finally, the *sum-SEPCA* algorithm utilizes row sums of the data matrix.

Algorithm 2 Variable Selection and Estimation Algorithm

```
Input: Threshold $\tau_{n,p}$ and form of Test Statistic $T_i$ from Table 5.1
  Let $\widehat{I}$ be an empty list
  for all Rows $i$ of $\mathbf{X}$, $1 \leq i \leq p$ do
    Form test statistic $T_i$ from row $i$ of $\mathbf{X}$
    if $T_i \geq \tau_{n,p}$ then
      Add $i$ to $\widehat{I}$
    end if
  end for
  Let $[\widetilde{\mathbf{u}}, \widetilde{\theta}, \widehat{\mathbf{v}}] = \mathrm{SVD}(\mathbf{X}_{\widehat{I},:})$ be the rank-1 SVD of $\mathbf{X}$ restricted to rows in $\widehat{I} = [i_1, \cdots, i_{|\widehat{I}|}]$
  For $i_k \in \widehat{I}$, let $\widehat{u}_{i_k} = \widetilde{u}_k$; the other entries of $\widehat{\mathbf{u}}$ are set to 0.
```

5.3.1 Computational Complexity

Note that the variable selection part of our procedures has a computational complexity that is $O(pn)$: the formation of the test statistic is linear in the number of columns, and the formation is repeated once per row. Noting that for a $p \times n$ matrix, the complexity of the rank-1 SVD is $O(1 \times pn)$, we find that if $\left|\widehat{I}\right|$ coordinates are selected, we have an overall complexity of $O\left(pn + \left|\widehat{I}\right| n\right) = O(pn)$ [2].

Computation of the covariance matrix has a (naive) complexity of $O(p^2 n)$, and in practice is somewhere between $O(p^2)$ and $O(p^3)$ [41]. Immediately, our methods here are faster than those requiring explicit formation of the covariance matrix [17, 61, 75, 134, 16]. Additionally, there is no iteration or convergence of any optimization problems required. Note that a semi-definite programming-based formulation is at least polynomial in the problem size: $O(p^4)$ [31] or $O(p^3)$ [16]. The ITSPCA method applied to our rank-1 setting would have a cost of $O(ps)$ per iteration [75, Sec. 4]. TPower has a similar complexity of $O(sp + p)$ per iteration [134].

5.3.2 The DT-SPCA Algorithm and Two-Stage Procedures

The DT-SPCA algorithm was proposed in [61] and later used as the first stage of the ASPCA algorithm given in [17]. The algorithm thresholds the diagonals of the matrix XX^T to perform variable selection: note that in our setting, these values are

$$\theta^2 u_i^2 + \sigma^2 \sum_{j=1}^{n} G_{ij}^2,$$

with expectation $(\theta^2 u_i^2 + \sigma^2)$. The DT-SPCA algorithm thresholds these diagonal values at $\sigma^2\left(1+\gamma\sqrt{\frac{\log p}{n}}\right)$, where $\gamma > 0$, and then performs PCA on the reduced matrix formed from the selected variables. Noting that the diagonals of XX^T are the same as the row sum-of-squares of X, we see that ℓ_2-SEPCA is essentially the same (up to choice of threshold) as DT-SPCA.

However, the innovation of [61, 17] that we carry forward is the two-stage procedure. That is, we perform some sort of testing to estimate the support of the sparse singular vector $\mathbf{u}$, and then perform an SVD on the reduced matrix. As we will see in what follows, there is flexibility in the choice of testing or support estimation method.

5.3.3 Statement of Thresholds

We shall choose the thresholds $\tau_{n,p}$ for the coordinate selection scheme so that in the noise-only case,

$$\mathbb{P}\left(\max_{1\leq i\leq p} T_i \geq \tau_{n,p}\right) \leq \frac{1}{ep} \to 0, \tag{5.5}$$

where e is Euler's number, or the base of the natural logarithm. This choice ensures that the probability of a false positive tends to zero as $p \to \infty$. That is, the FWER is asymptotically zero and is bounded by $1/ep$ in the finite-dimensional case. Note that the constraint used to control the FWER is simply that the distribution of the noise is log-concave. In the Gaussian case, we obtain the specific expressions given summarized in Table 5.1; however, with knowledge of the moments $\mathbb{E}T_i$ and Var T_i, we can repeat our analysis and find thresholds for the ℓ_1 and ℓ_2-SEPCA algorithms with *any* log-concave noise distribution. The thresholds are summarized in Table 5.1.

Table 5.1: Test Statistics and Thresholds for Algorithm (2)

Algorithm	Statistic T_i	Threshold $\tau_{n,p}$
ℓ_1-SEPCA	$\frac{1}{\sqrt{n}}\sum_{k=1}^n \lvert X_{i,k}\rvert$	$\sigma\left(\sqrt{\frac{2}{\pi}} + C_1\frac{\log ep}{\sqrt{n}}\right)$
ℓ_2-SEPCA	$\sum_{k=1}^n X_{i,k}^2$	$\sigma^2\left(1 + C_2\frac{\log ep}{\sqrt{n}}\right)$
sum-SEPCA	$\frac{1}{\sqrt{n}}\lvert\sum_{k=1}^n X_{i,k}\rvert$	$\sigma C_U\sqrt{\frac{\log p}{n}}$

See (5.6) and (5.10) for definitions of the constants C_2, C_1, and C_U.

Remark 5.1. *Note that we impose strong control over the FWER and seek to reject* *individual null hypotheses, instead of weak control and considering the global null hypothesis as in [16].*

5.3.4 FWER Thresholds

ℓ_2- and ℓ_1-SEPCA

In the noise-only cases, the statistics for ℓ_2- and ℓ_1-SEPCA are distributed as scaled χ_n^2 and sums of half-normal, respectively. Both of these quantities are log-concave random variables, so we may apply the result in [68] to set the threshold $\tau_{n,p}$ in both cases.

Defining K to be some absolute constant (we may use $K = e$, as in [19]), we define the constants

$$C_2 = \sqrt{2}K \text{ and } C_1 = K\sqrt{(1 - 2/\pi)}. \tag{5.6}$$

sum-SEPCA

From Proposition 4.4 of [21], we obtain that the threshold for sum-SEPCA is given by

$$\tau_{n,p} = \frac{\sigma}{\sqrt{n}} \left(\sqrt{2 \log p} + \frac{1}{U(p)} \left(\frac{1}{3} \log ep + \sqrt{\log ep} \right) + \delta_p \right). \tag{5.7}$$

In (5.7), we have that

$$U(p) = \sqrt{2}\ \mathrm{Erf}^{-1} \left(1 - \frac{1}{p} \right) \text{ and } \delta_p \asymp \frac{\pi^2}{12} \left(\log p \right)^{-3/2}, \tag{5.8}$$

where Erf denotes the *error function*, or alternatively, the cumulative distribution function of a standard Gaussian random variable is given by

$$\Phi(x) = \frac{1}{2} \left(1 + \mathrm{Erf} \left(\frac{x}{\sqrt{2}} \right) \right). \tag{5.9}$$

Moreover, $\tau_{n,p} \leq \sigma C_U \sqrt{\frac{\log p}{n}}$ for some constant C_U. For a fixed value of p, choosing

$$\kappa_U \geq \frac{\sqrt{2}}{U(p)} \left(3 + \sqrt{\log p} \right) > 1 \text{ and } C_U = \sqrt{2} + \frac{\kappa_U}{3\sqrt{2}} \tag{5.10}$$

is sufficient. The choice of $1/ep$ is the largest bound justified by Proposition 4.4 of [21], so we have calibrated all of our algorithms to the same constant factor times $1/p$. The thresholds are summarized in Table 5.1.

5.3.5 Estimation of the Noise Variance, σ^2

In this work, we assume that the noise variance σ^2 is known; however, in general, estimation of σ^2 may not be straightforward [94]. Recently proposed procedures such as those proposed in [94, 96, 113] could be employed to estimate the noise variance, and we point the interested reader to these references for more theoretical background on the problem. We note that in most applications, including the video example we consider, one can obtain a relatively sparse representation of the object in a multiscale basis such as a wavelet basis [60, Sec. 7.5]. Under such circumstances, under the assumed additive, isotropic noise model, we can easily obtain a consistent estimate of σ^2 by utilizing the inherent sparsity of the signal, especially in finer scales. This can be done, for example, by computing the variance of the wavelet coefficients in the finest scale [60, Sec. 7.5]. One can obtain a more robust estimate by taking the median absolute deviation of the coefficients about their median and then by multiplying its square with a known scale factor (assuming normality) [96, 60].

5.4 Controlling the False Discovery Rate

So far, we have controlled the probability of a false alarms when detecting coordinates. However, there are two relevant observations to make. First, under the Gaussian noise, rank-1, and equisigned assumptions, the vector of test statistics $\{T_i\}$ in the sum-SEPCA algorithm looks like a sparse vector plus Gaussian noise (or a vector of χ_n^2-variates with varying non-centralities, in the ℓ_2-SEPCA algorithm). Secondly, controlling the false discovery rate, that is, the proportion of rejected nulls that are false positives, can lead to increased detection power relative to controlling the false positive rate. We hence look at FDR-controlling tests for the *Sparse Normal Means* problem.

That is, given a vector of test statistics (as before), we replace the thresholding and selection in Algorithm 2 with an FDR-controlling selection procedure. We summarize this change in Algorithm 3. There are three procedures we consider. The first two are known as Higher Criticism, and directly extend the sum- and ℓ_2-SEPCA algorithms [36, 37]. The third is a method for detection in the sparse normal means problem that comes out of complexity-penalized estimation theory for linear inverse problems [62].

Algorithm 3 FDR-Controlling Variable Selection and Estimation Algorithm

```
Input: Test Statistic $T_i$ from Table 5.1 and Selection Procedure
  Let $\widehat{I}$ be an empty list
  for all Rows $i$ of $\mathbf{X}$, $1 \leq i \leq p$ do
    Form test statistic $T_i$ from row $i$ of $\mathbf{X}$
  end for
  Perform an FDR-Controlling selection procedure, and add the selected indices to $\widehat{I}$
  Let $[\widetilde{\mathbf{u}}, \widetilde{\theta}, \widehat{\mathbf{v}}] = \mathrm{SVD}(\mathbf{X}_{\widehat{I},:})$ be the rank-1 SVD of $\mathbf{X}$ restricted to rows in $\widehat{I} = [i_1, \cdots, i_{|\widehat{I}|}]$
  For $i_k \in \widehat{I}$, let $\widehat{u}_{i_k} = \widetilde{u}_k$; the other entries of $\widehat{\mathbf{u}}$ are set to 0.
```

5.4.1 Higher Criticism

Formulation of Higher Criticism

Assume we have p independent tests of the form

$$\begin{aligned} H_{o,i}: &\quad W_i \sim \mathcal{N}(0,1), \\ H_{1,i}: &\quad W_i \sim \mathcal{N}(\mu_i,1), \end{aligned} \tag{5.11}$$

and assume that at most $p^{1-\beta}$ of the p hypotheses are truly non-null, for some $\beta \in (1/2, 1)$. Further assume that the non-null means have magnitude

$$\mu_i = \mu_p = \sqrt{2r \log p},$$

for $r \in (0,1)$. Here, the means will correspond to the coordinate size. Note that the expected maximum of p standard Gaussian random variables is upper bounded by $\sqrt{2 \log p}$, with the bound being asymptotically sharp.

If we let $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(p)}$ be the sorted p-values of the individual tests, we may define the Higher Criticism statistic:

$$HC_p = \max_{i:1/p \leq p_{(i)} \leq 1/2} \frac{\sqrt{p}\left(i/p - p_{(i)}\right)}{\sqrt{p_{(i)}(1-p_{(i)})}}. \tag{5.12}$$

Rejecting the global null hypothesis (that there are no non-null coordinates) when $HC_p > \sqrt{2 \log \log p}(1+o(1))$ leads to asymptotically full power when r is greater than some decision boundary ρ, and that under the global null,

$$\frac{HC_p}{\sqrt{2\log\log p}} \to 1 \tag{5.13}$$

in probability as $n, p \to \infty$. The function ρ depends on the sparsity index β, and as [37] indicate:

$$\rho(\beta) = \begin{cases} \beta - 1/2 & \text{when } \beta \in (1/2, 3/4), \\ \left(1 - \sqrt{1-\beta}\right)^2 & \text{when } \beta \in (3/4, 1). \end{cases} \tag{5.14}$$

If we replace the normal distribution with a χ_n^2 distribution, the same results hold for tests of the form

$$\begin{aligned} H_{o,i}:& \quad W_i \sim \chi_n^2, \\ H_{1,i}:& \quad W_i \sim \chi_n^2(\delta), \end{aligned} \tag{5.15}$$

where δ is a non-centrality parameter and we consider $r \in (0,1)$ such that $\delta = 2r\log p$.

Remark 5.2. *While Higher Criticism is typically formulated for the case of identical non-null means or parameters (all of the non-zero μ_i are identical), this constraint is not mandatory [6, 47]. Indeed, the results hold without modification for the Gaussian model with non-null means of size $\mu_i = \alpha_i\sqrt{2\log p}$, where α_i is a non-negative random variable with the property that $\mathbb{P}(\alpha_i \leq \sqrt{r}) = 1$ and $\mathbb{P}(\alpha_i > \sqrt{r} - \epsilon) > 0$ for all $\epsilon > 0$ [47]. The case of a χ_n^2 distribution is similar.*

As a point of interest, the test in (5.11) can be extended to (and potentially strengthened in) the case where the p tests are correlated, i.e., when the additive Gaussian noise has a non-identity covariance [47].

Application to our Problem

Recall that for the sum-SEPCA algorithm, we formed a vector of row-sums. That is, in the equisigned setting, taking sums across the rows of $\mathbf{X}$, we obtain a vector $\mathbf{y}$ where $y_i = \mu_i + \sigma z_i$, with $\mu_i = (\theta u_i)\|\,\mathbf{v}\,\|_1$: this situation is exactly that of a sparse mean vector embedded in Gaussian noise. Similarly, taking sums of squares across the rows of $\mathbf{X}$ (as in the ℓ_2-SEPCA algorithm) yields scaled χ_n^2 distributed random variables, of which only a few have non-zero non-centrality parameters.

With knowledge of the noise distribution, we may compute the p-values of each row statistic: these p-values are used to form the Higher Criticism statistic (5.12). As in [37], we may adapt the original global testing problem to a selection problem. For each p-value $p_{(i)}$, we have a value $HC_{p,i}$ of the higher criticism statistic (the value that is maximized in (5.12)). Rejecting each null hypothesis (that the coordinate of the corresponding row is zero) when $HC_{p,i}$ is larger than the threshold $\sqrt{2\log\log p}$ is a variable selection procedure. We refer to the procedure based on the sum statistic as HC-sum-SEPCA and that based on the sum of squares statistic as HC-ℓ_2-SEPCA. Importantly, we note that the form of the decision boundary ρ is identical to the global testing case, and that applying Higher Criticism to our row statistics is a viable global testing procedure [37].

5.4.2 FDR-SEPCA

In this section, we give an summary of the algorithm for uncorrelated noise and defer the general case and details to Appendix 5.D. We continue in the same vein as in the previous section on Higher Criticism.

We note that in the equisigned, rank-1 setting, coordinate selection is equivalent to the estimation of a sparse mean vector. Let $y_i = \mu_i + \sigma z_i$, where $i \in \{1, \cdots, p\}$ and the vector $\mathbf{z}$ of the z_i is normally distributed with mean 0 and covariance $\mathcal{I}_p$. The mean vector $\boldsymbol{\mu}$ of the μ_i is assumed to be sparse; the goal is to estimate $\boldsymbol{\mu}$. Taking sums across the rows of $\mathbf{X}$, we obtain a vector $\mathbf{y}$ where $y_i = \mu_i + \sigma z_i$, with $\mu_i = (\theta u_i)\| \mathbf{v} \|_1$. Hence, we are in the same setting as in the previous section.

The following penalized least squares formulation, taken from [62], yields an estimator for μ:

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \| \mathbf{y} - \boldsymbol{\mu} \|_2^2 + \sigma^2 \text{pen}\left(\| \boldsymbol{\mu} \|_0\right), \tag{5.16}$$

where pen(k) is defined as

$$\text{pen}(k) = \zeta k \left(1 + \sqrt{2\log(\nu p/k)}\right)^2, \tag{5.17}$$

with $\zeta > 1$; we may take $\zeta = 1 + o(1)$. The parameter ν is no smaller than e. We define $\| \boldsymbol{\mu} \|_0$ to be the number of non-zero coordinates of μ.

The solution to (5.16) is given by hard-thresholding. Let $|y|_{(i)}$ be the i^{th} order statistic of $|y_i|$, namely $|y|_{(1)} \geq \cdots \geq |y|_{(p)}$. Then if

$$\widehat{k} = \arg\min_{k\geq 0} \sum_{i>k} |y|_{(i)}^2 + \sigma^2 \text{pen}(k), \tag{5.18}$$

defining

$$t_k^2 = \mathrm{pen}(k) - \mathrm{pen}(k-1), \tag{5.19}$$

the solution is to hard threshold at $t_{\widehat{k}}$.

In this set-up, we have that

$$t_k \approx \sqrt{\zeta}(1 + \sqrt{2\log(\nu p/k)}).$$

We provide a precise quantification of t_k in Appendix 5.D.

Hence, by computing t_k and performing hard thresholding of the row sums, we can perform coordinate selection. Once again, this procedure replaces the test statistic/thresholding in Algorithm 2.

5.5 Estimation Error and Smallest Detectable Coordinate

As we will see, our theorems discuss the "detectability" of the coordinates u_i of $\mathbf{u}$. However, it is common in the sparse PCA literature to discuss lower bounds for the risk (estimation error) [61, 17, 75]. In what follows, we will show that these two notions are equivalent.

We define the L^2 estimation error for a principal component estimator as

$$L(\widehat{\mathbf{u}}, \mathbf{u}) = \|\mathbf{u} - \mathrm{sign}(\langle \mathbf{u}, \widehat{\mathbf{u}} \rangle)\widehat{\mathbf{u}}\|_2^2 . \tag{5.20}$$

The quantity in (5.20) is upper bounded by 2; this bound is attained when $\mathbf{u}$ and $\widehat{\mathbf{u}}$ are unit norm and mutually orthogonal. Following [17], we want to compute a lower bound for the maximum expected loss for the s-sparse vectors $\mathbf{u}$ (in the sense of ℓ_0 sparsity) defined as

$$\sup_{\mathbf{u} \in \mathbb{S}^{p-1} : \|\mathbf{u}\|_0 \leq s} \mathbb{E} L\left(\widehat{\mathbf{u}}, \mathbf{u}\right), \tag{5.21}$$

where $\mathbb{S}^{p-1}$ denotes the unit sphere in $\mathbb{R}^p$. Let $\widehat{I}$ be some index set of coordinates selected by an algorithm of the form given in Algorithm (2). We may take $\langle \mathbf{u}, \widehat{\mathbf{u}} \rangle$ to be non-negative, and decompose the loss as

$$\| \mathbf{u} - \widehat{\mathbf{u}} \|_2^2 = \underbrace{\| \mathbf{u}_{\widehat{I}} - \widehat{\mathbf{u}} \|_2^2}_{\substack{\text{Estimation Error from} \\ \text{detected coordinates}}} + \underbrace{\| \mathbf{u}_{\widehat{I}^c} \|_2^2}_{\substack{\text{Error from} \\ \text{missed coordinates}}} \geq \| \mathbf{u}_{\widehat{I}^c} \|_2^2. \tag{5.22}$$

Equation (5.22) shows that the loss is lower-bounded by the squared sum of the missed coordinates. Indeed, it is a natural consequence of the result in [12] that if the sparsity s grows slower than does n, and we have a consistent estimate of the support of $\mathbf{u}$, the estimation error will asymptotically be small. Essentially, we are estimating the singular vectors of an $s \times n$ matrix instead of a $p \times n$ matrix, so that if the ratio s/n has limit zero, our estimates will be consistent (see (5.3) and [12]). This suggests the following strategy for lower-bounding (5.21): we want to construct a non-trivial 'worst-case' sparse vector. That is, we want a vector $\mathbf{u}$ that has a non-trivial loss (less than 2), is sparse (fewer than s non-zero coordinates), and has maximal error from missed coordinates. To ensure a non-trivial loss, we set the first coordinate u_1 to be large, *i.e.*, $u_1 = \sqrt{1 - r^2}$, where $r = o(1)$. To ensure sparsity, we set $u_2, \cdots, u_{m+1}$ to be non-zero for some $m \leq s - 1$, with the subsequent coordinates of u set to 0. Then, the expected loss has the lower bound

$$\begin{aligned} \mathbb{E}L(\mathbf{u}, \widehat{\mathbf{u}}) &\geq \sum_{k=1}^{p} |u_k|^2 \mathbb{P}\left(\text{Not Selecting Coordinate k}\right) \\ &\geq \sum_{k=2}^{m+1} |u_k|^2 \mathbb{P}\left(\text{Not Selecting Coordinate k}\right), \end{aligned} \tag{5.23}$$

since u_1 is detected with probability approaching 1 and u_k is zero for $k > m + 1$. Now, let u_2 through u_{m+1} all have value $r/\sqrt{m}$, so that we may simplify the lower bound to

$$\mathbb{E}L(\mathbf{u}, \widehat{\mathbf{u}}) \geq r^2 \mathbb{P}\left(\text{Not Selecting Coordinate k}\right). \tag{5.24}$$

If coordinates of size $r/\sqrt{m}$ are not detected with a probability approaching 1, r^2 is a lower-bound on the risk. This construction shows that specifying the sizes of coordinates that are not detected with probability approaching 1 is equivalent to specifying a worst-case risk lower bound. Note that the value of r^2 depends on the specific algorithm and estimator, and that this is not a general or universal bound. Rather, the purpose of this construction is to show the equivalence between the two perspectives (a lower bound and detectable coordinate size).

Consequently, in what follows we focus on the smallest detectable and largest undetectable coordinates because they directly shed light on the attainable estimation error. The details of the risk calculations and extensions to approximate sparsity are deferred to Appendix 5.C, where we summarize our findings in Theorem 5.3.

5.6 Main Results

The following theorem characterizes consistent support recovery conditions. These results are the analogue of the 'sparsistency' guarantees found in the LASSO and ℓ_1-norm minimization literature [104]. Throughout, $\widehat{I}$ denotes the set of coordinates selected by the coordinate selection scheme.

Theorem 5.1. *For the model specified in (5.1) and (5.2) and the algorithms specified in Table 5.1, assume that $p(n), n \to \infty$, $s(n)/n \to 0$, and $\log p(n) = o(n)$. Let $\epsilon \in (0,1)$. We have that*

a. For $i \in I^c$,

$$\max_{i \in I^c} \mathbb{P}\left(i \in \widehat{I}\right) \to 0,$$

b. For $i \in I$,

$$\min_{i \in I \,:\, |\theta u_i| > \beta_{crit}(1+\epsilon)} \mathbb{P}\left(i \in \widehat{I}\right) \to 1,$$

$$\max_{i \in I \,:\, |\theta u_i| < \beta_{crit}(1-\epsilon)} \mathbb{P}\left(i \in \widehat{I}\right) \to 0.$$

Here

$$\beta_{crit} = \begin{cases} \sigma C_U \frac{\sqrt{\log p}}{\left|\sum_k v_k\right|} & \textit{for sum-SEPCA,} \\ \sigma \sqrt{C_2}\sqrt{\frac{\log ep}{\sqrt{n}}} & \textit{for } \ell_2\textit{-SEPCA,} \\ \sigma t_{\ell_1} & \textit{for } \ell_1\textit{-SEPCA,} \end{cases} \tag{5.25}$$

and t_{ℓ_1} satisfies the relation

$$\left(\sqrt{\frac{2}{\pi}} + C_1 \frac{\log ep}{\sqrt{n}}\right) = \frac{1}{n}\sqrt{\frac{2}{\pi}}[\sum_k \exp\left(-\left(\sqrt{n}\frac{(t_{\ell_1})v_k}{\sqrt{2}}\right)^2\right) + \sqrt{\pi}\sum_k \left(\sqrt{n}\frac{(t_{\ell_1})v_k}{\sqrt{2}}\right) \mathit{Erf}\left(\sqrt{n}\frac{(t_{\ell_1})v_k}{\sqrt{2}}\right)].$$

We defer the proof to Appendix 5.A.

Theorem 5.1 identifies a phase transition in the ability of the algorithms to accurately estimate the support of $\mathbf{u}$. Note that the analysis brings into sharp focus the dependence of β_{crit} on $\mathbf{v}$ for the ℓ_1- and sum-SEPCA algorithms, but not the ℓ_2-SEPCA algorithm. Consequently, we can expect the algorithms to perform differently depending on the structure of the underlying $\mathbf{v}$. It is important to note that the sparsity s of $\mathbf{u}$ is not a parameter in the thresholds and results.

It is also important to note that ℓ_2-SEPCA and ℓ_1-SEPCA do not rely on the equisigned character of $\mathbf{v}$. However, it is clear that the sum-SEPCA algorithm explicitly depends on the equisigned assumption.

5.6.1 Hamming Loss

It is also possible to state the above results in terms of the Hamming loss for the support of $\mathbf{u}$, and prove consistency of the coordinate selection scheme by assuming that all the nonzero coordinates of $\mathbf{u}$ lie above a critical threshold. A detailed decision-theoretic analysis of variable selection under a sequence model with *i.i.d.* noise and with respect to the Hamming loss has recently been carried out by [24]. Recall that the Hamming loss measures the number of elements in two sets that are different, so that here the loss between the true support I and the estimated support $\widehat{I}$ would be the size of the symmetric set difference of I and $\widehat{I}$. Let $d_H\left(I,\widehat{I}\right)$ denote the Hamming loss and assume that whatever algorithm we are using has a threshold β_{crit}. Then, for any $\epsilon \in (0,1)$, we may write

$$\begin{aligned}\mathbb{E}\, d_H\left(I,\widehat{I}\right) &= \sum_{i\in I:|\theta u_i|>\beta_{crit}(1+\epsilon)} \mathbb{P}\left(i\notin\widehat{I}\right) + \sum_{i\in I:|\theta u_i|<\beta_{crit}(1-\epsilon)} \mathbb{P}\left(i\notin\widehat{I}\right)\\ &+ \sum_{i\in I:(1-\epsilon)\le|\theta u_i|/\beta_{crit}\le(1+\epsilon)} \mathbb{P}\left(i\notin\widehat{I}\right) + \sum_{i\notin I}\mathbb{P}\left(i\in\widehat{I}\right). \qquad (5.26)\end{aligned}$$

We can then restate the results on coordinate selection in terms of the Hamming loss under a more restricted setting that assumes an exact form of sparsity of the vector $\mathbf{u}$.

Corollary 5.1. *For the model specified in (5.1) and (5.2), and an algorithm specified in Table 5.1, assume that the conditions of Theorem 5.1 hold. Let the support I of $\mathbf{u}$ be estimated by $\widehat{I}$. Moreover, assume that the algorithm has a threshold β_{crit} (given in (5.25))* *such that for a small, fixed $\epsilon_0 > 0$, the set*

$$I_0 := \{j : |\theta u_j| > \beta_{crit}(1 + \epsilon_0)\} \tag{5.27}$$

equals the set I. Then the expected Hamming loss satisfies

$$\mathbb{E}\, d_H \left(I, \widehat{I}\right) /|I| \to 0. \tag{5.28}$$

The proof of the corollary, given in Appendix 5.A.1, follows from applying Theorem 5.1, with a more detailed enumeration of the sets and the inclusion probabilities, to each term of (5.26).

5.6.2 FDR-Based Algorithms

We may summarize the coordinate selection properties of the FDR refinements as follows:

Theorem 5.2. *For the model specified in (5.1) and (5.2) and the three FDR-controlling algorithms summarized in Algorithm 3, assume that $p(n), n \to \infty$, $s(n)/n \to 0$, and $\log p(n) = o(n)$. Let $\epsilon \in (0, 1)$. We have that*

a. For all three algorithms and $i \in I^c$,

$$\max_{i \in I^c} \mathbb{P}\left(i \in \widehat{I}\right) \to 0,$$

b. For the Higher Criticism-based algorithms and $i \in I$,

$$\min_{i \in I \,:\, |\theta u_i| > \beta_{crit}(1+\epsilon)} \mathbb{P}\left(i \in \widehat{I}\right) \to 1,$$

$$\max_{i \in I \,:\, |\theta u_i| < \beta_{crit}(1-\epsilon)} \mathbb{P}\left(i \in \widehat{I}\right) \to 0.$$

c. For the FDR-SEPCA algorithm, uniformly over $i \in I$,
if $|\theta u_i| > \beta_{crit}(1 + \epsilon)$, coordinate i is selected;
if $|\theta u_i| < \beta_{crit}(1 - \epsilon)$, coordinate i is not selected
with probability tending to 1.

Here

$$\beta_{crit} = \begin{cases} \sigma\sqrt{\rho(\beta)}\frac{\sqrt{2\log p}}{\|\mathbf{v}\|_1} & \textit{for HC-sum-SEPCA,} \\ \sigma\rho(\beta)\frac{2\log p}{\sqrt{n}} & \textit{for HC-}\ell_2\textit{-SEPCA,} \\ \sigma\left(1-o(1)\right)\sqrt{\zeta}\frac{1+\sqrt{2\log(\nu p/\widehat{k})}}{\|\mathbf{v}\|_1} & \textit{for FDR-SEPCA,} \end{cases} \tag{5.29}$$

where $\zeta > 1$, $\nu > e$, and the FDR-SEPCA algorithm detects $\widehat{k}$ coordinates.

We defer the proof to Appendix 5.B.

Once again, we see that the structure of the underlying $\mathbf{v}$ plays a role in the performance of the sum-based algorithms, but not for the ℓ_2-based HC-ℓ_2-SEPCA algorithm. Unlike in the FWER-controlling cases, the sparsity of $\mathbf{u}$ plays a (small) role here, via the constant $\rho(\beta)$ for the Higher Criticism-based methods and via $\widehat{k}$ for FDR-SEPCA. Moreover, ℓ_2-HC-SEPCA, like ℓ_2-SEPCA, does not make use of the equisigned nature of $\mathbf{v}$.

5.6.3 Higher Ranks

In this work, we restrict our focus to the rank-1, equisigned setting. A natural question is are our results extensible to the higher rank setting?

The first point is concerned with the right singular vectors. To preserve orthogonality, we would need equisigned right-singular vectors $\mathbf{v}_i$ with disjoint supports. The second point is concerned with the left singular vectors. Our algorithms are based on thresholding row-statistics: it is possible that the union of supports of several sparse vectors is a relatively large set. The FWER-controlling algorithms (by design) are not sensitive to the increased supports, but the FDR-controlling algorithms are sensitive to this. Indeed, the decision boundaries for the FDR algorithms explicitly depend on the sparsity levels. Third point, once again, is concerned with the left singular vectors. It is possible that a sum-based statistic suffers from cancellations that decrease the size of the row-statistic. For example, in a rank-2 setting, if $\mathbf{u}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1 & 0 & \cdots & 0\end{bmatrix}^T$ and $\mathbf{u}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & -1 & 0 & \cdots & 0\end{bmatrix}^T$ and $\|\mathbf{v}_1\|_1$ and $\|\mathbf{v}_2\|_1$ have similar values and are both non-negative, the row-sum of the second row will be small. Note, however, that the ℓ_2-norm based methods do not suffer from this issue.

5.7 Simulations

To illustrate the relative powers of the six algorithms, we compute the theoretical limits on the sizes of detectable coordinates as a function of n. We use a unit-norm, equisigned $\mathbf{v}$ such that

$$v_k \propto \exp\left(-5\frac{k}{n}\right)\left|\sin\left(4\frac{k}{n}\right)\right| \quad \text{for } 1 \leq k \leq n. \tag{5.30}$$

This choice of $\mathbf{v}$ has a 'rise and fall' sort of behavior, and is motivated by physical signals, e.g., chemical reactions or nerve signals in the brain. The value of β_{crit} is shown in Figure 5.1; for this choice of $\mathbf{v}$, it is clear that the sum-SEPCA dramaticaly outperforms the other SEPCA variants in terms of size of the smallest detectable component. The FDR-SEPCA algorithm has similar performance to sum-SEPCA, and the HC-sum-SEPCA algorithm has the strongest performance.

In Figure 5.2, we plot the estimation error as a function of n and θ for all six algorithms. We also include results for the SVD and competing algorithms TPower [134] and ITSPCA [75]. In the simulations, we fix $p = 1000$ and vary n, since the dependence in p in the thresholds is logarithmic, whereas that in n is not. The left singular vector $\mathbf{u}$ is chosen to be the vector with 1 in the first coordinate and 0 elsewhere. We fix the noise variance σ^2 at 1, so that θ^2 is the eigen-SNR. The results should be interpreted as follows. For the particular $\mathbf{v}$ chosen here, we expect HC-sum-SEPCA to have the lowest detectable limit, and ℓ_1-SEPCA to have the largest. This behavior is confirmed. Moreover, the sum-based algorithms offer a slight strengthening of both ITSPCA and TPower. Importantly, note that the sum-based algorithms explicitly take advantage of the equisigned nature of $\mathbf{v}$: that is, algorithms that explicitly use the equisigned property outperform algorithms that do not (the ℓ_2 and ℓ_1 algorithms, as well as ITSPCA and TPower).

We repeat our simulations for a $\mathbf{u} \in \mathbb{R}^p$ with $\sqrt{p}$ non-zero coordinates (of equal size) and the same $\mathbf{v}$, as seen in Figure 5.3. We find similar conclusions as in Figure 5.2, where the sum-based algorithms offer a strengthening over ITSPCA and TPower; the ℓ_2-norm based algorithms do not perform well. Note that a sparsity of $\sqrt{p}$ is at the limit/valid edge for the higher criticism-based methods, but that these methods still perform well.

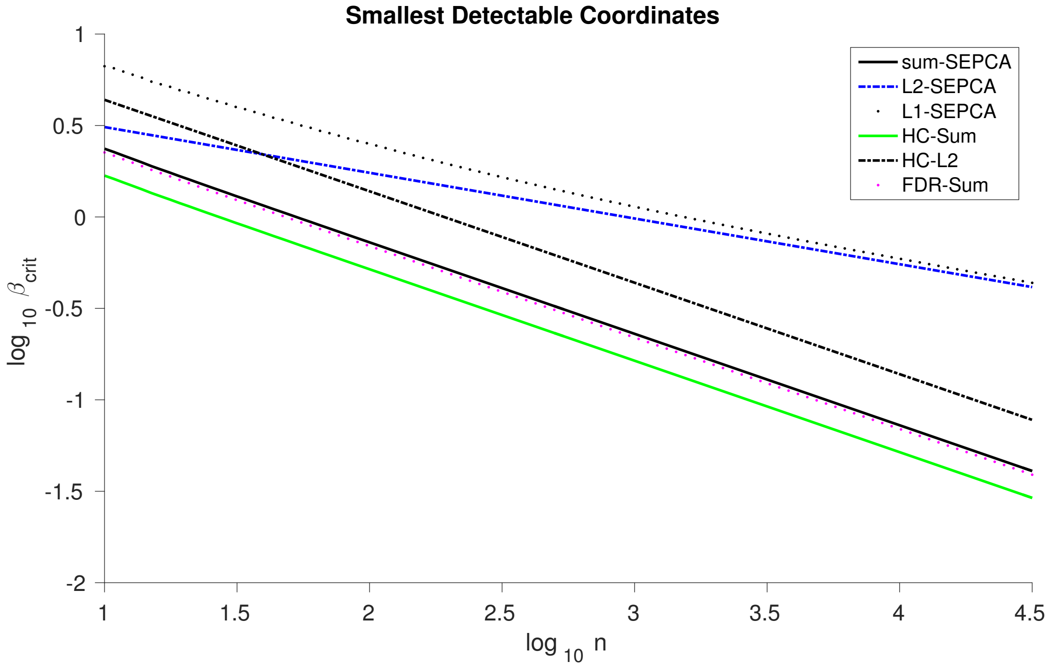


Figure 5.1: This plot shows β_{crit} for all six algorithms for the v described in (5.30).

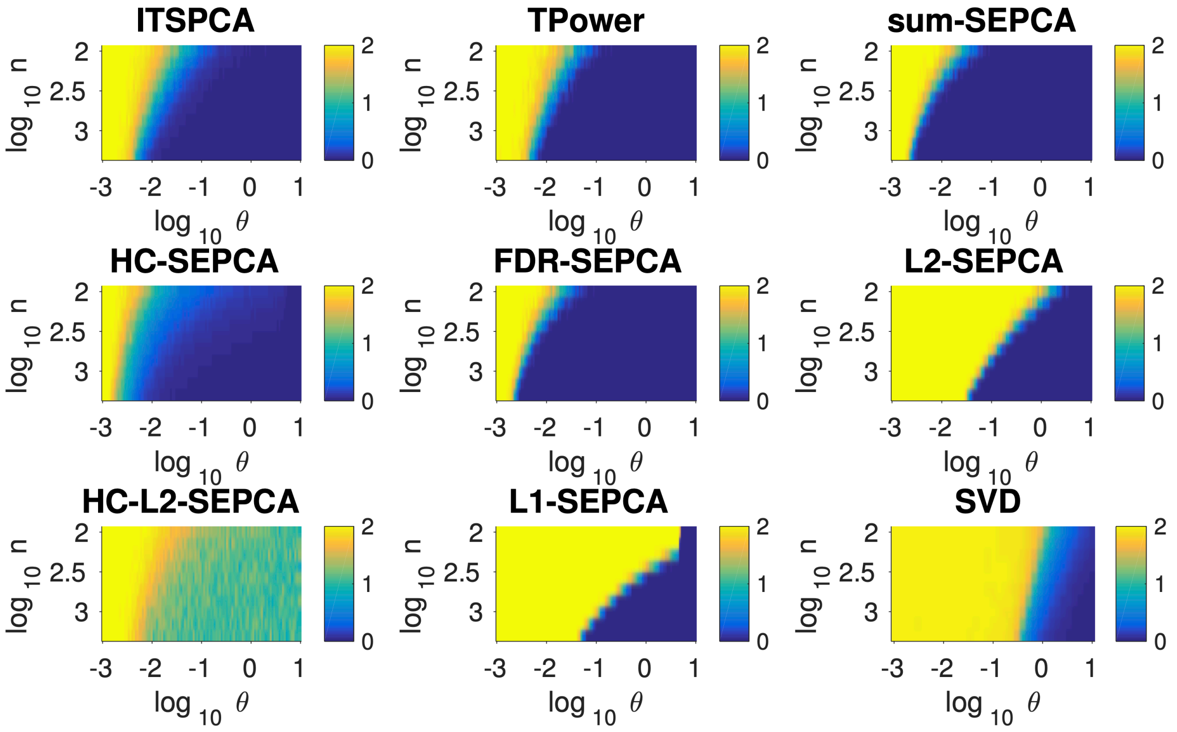


Figure 5.2: The plots show the empirical estimation error for all six algorithms for the u with one non-zero coordinate and the v described in (5.30). We include results from TPower, ITSPCA and the SVD for comparison.

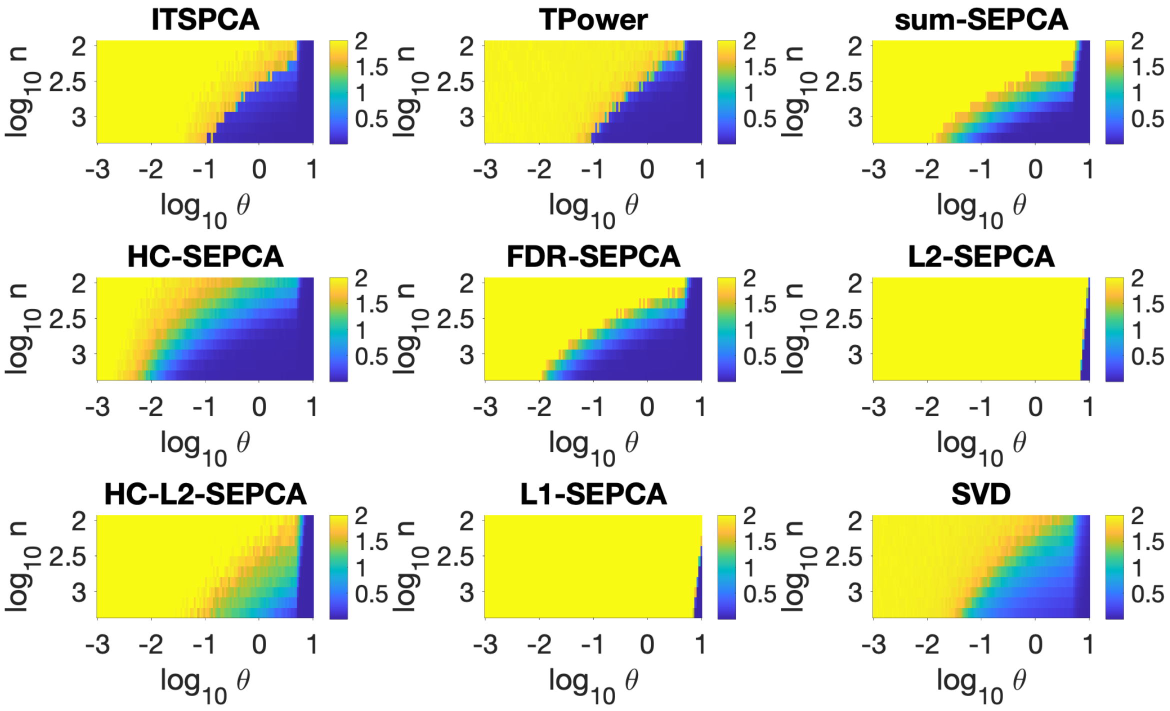


Figure 5.3: The plots show the empirical estimation error for all six algorithms for the u with $\sqrt{p}$ non-zero coordinates and the v described in (5.30). We include results from TPower, ITSPCA and the SVD for comparison.

5.7.1 Comments on the FDR-controlling procedures

The Higher Criticism for the χ^2_n-variates 'pushes back' the phase transition between detecting nothing and something to a lower value of θ relative to the ℓ_2-SEPCA algorithm, but is still less powerful than any of the sum-based algorithms. Moreover, even above the phase transition, the ℓ_2-SEPCA algorithm may be preferable, as the error is increased by unacceptably many false positives.

The Higher Criticism procedure for the sum statistic has the lowest phase transition point and hence the highest power. Its transition is more gradual than the penalized FDR thresholding procedure and sum-SEPCA, which have roughly the same performance in this simulation.

5.7.2 An example where ℓ_2-based algorithms outperform sum-based algorithms

Sum-SEPCA has a β_{crit} that depends on $\mathbf{v}$. Looking at the form in (5.25), if $\| \mathbf{v} \|_1$ is smaller than $n^{1/4}$, we would expect ℓ_2-SEPCA to detect a smaller coordinate size. Vectors with smaller coordinates have a smaller ℓ_1-norm, i.e., one that is closer to their ℓ_2-norm. Hence, if we choose

$$v_k \propto \frac{1}{k^2} \text{ for } 1 \leq k \leq n, \tag{5.31}$$

we expect sum-SEPCA to have worse performance relative to ℓ_2-SEPCA. Figures 5.4 and 5.5 confirm this expectation. The FDR refinements perform poorly. It should be noted, however, that TPower and ITSPCA retain their performance. This choice of $\mathbf{v}$ effectively corresponds to a very small value of n: the majority of coordinates are tiny in size and buried beneath noise regardless of the value of θ. If we 'corrected' the scenario and used a smaller n and a subset of $\mathbf{v}$, we would be in a situation closer to that given in (5.30).

5.7.3 A video data example

We conclude our sequence of examples with a real data study. This example is motivated by the problem of foreground-background separation in videos. Consider a grayscale video of stars twinkling against a black background [108]. Our goal is to estimate the locations of the stars: by reshaping the video, we may treat each frame as a vector and hence treat the video as a sparse matrix. Only a few locations have a star and are hence non-zero.

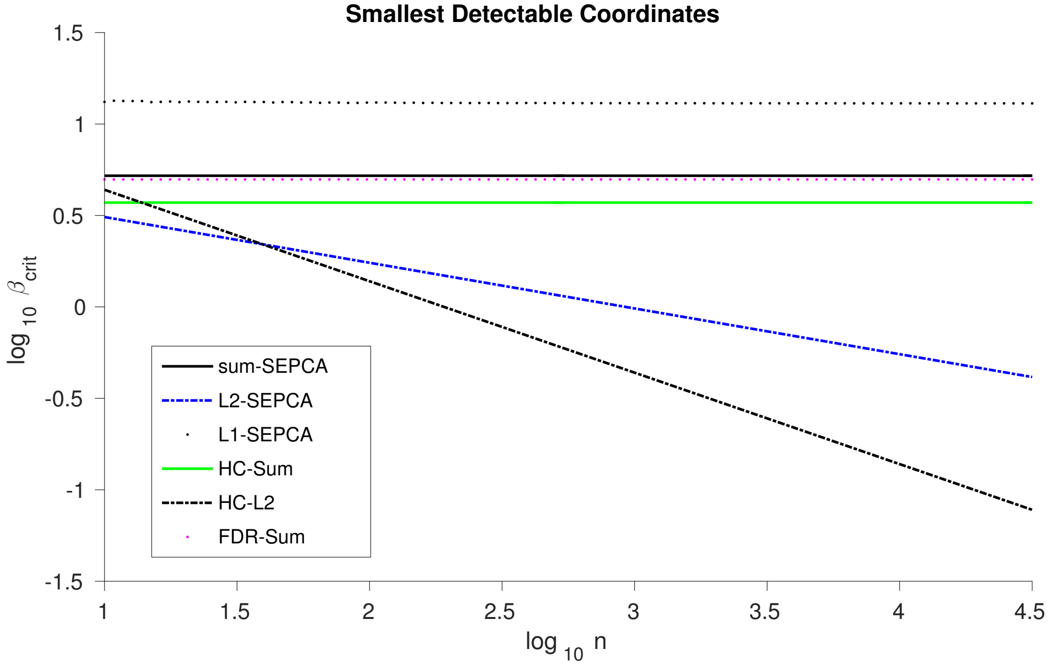


Figure 5.4: This plot shows β_{crit} for all six algorithms for the v described in (5.31).

The scale of the video pixels is between 0 and 255. We examine the top-left 72×64 pixels for 89 frames, as shown in Figure 5.2a. In Figure 5.2b, we plot the singular values of the video matrix. The first singular value stands out strongly against the rest, and at most two more singular values are well-separated from the bulk. This structure suggests that our rank-1 based approach is well suited to this problem.

We add Gaussian noise of variance σ^2 and study the True Positive Rates (TPR) and False Discovery Rates (FDR) across all algorithms and across different values of σ. In Figure (5.6), we show the results of our simulations. In terms of the TPR, everything other than the SVD has a similar performance, while the test-statistic SEPCA-based algorithms enjoy the best performance in terms of the FDR. In Figure 5.7 we zoom in on the top-right three stars and show how the algorithms perform as noise increases. Here, we see that the behavior alluded to in the TPR/FDR results actually occurs in the video.

5.8 A geometric view: which algorithm to use?

We have stated detectability results for each algorithm in Section 5.6 and provided a numerical verification and comparison in Section 5.7. In this section, we wish to analytically

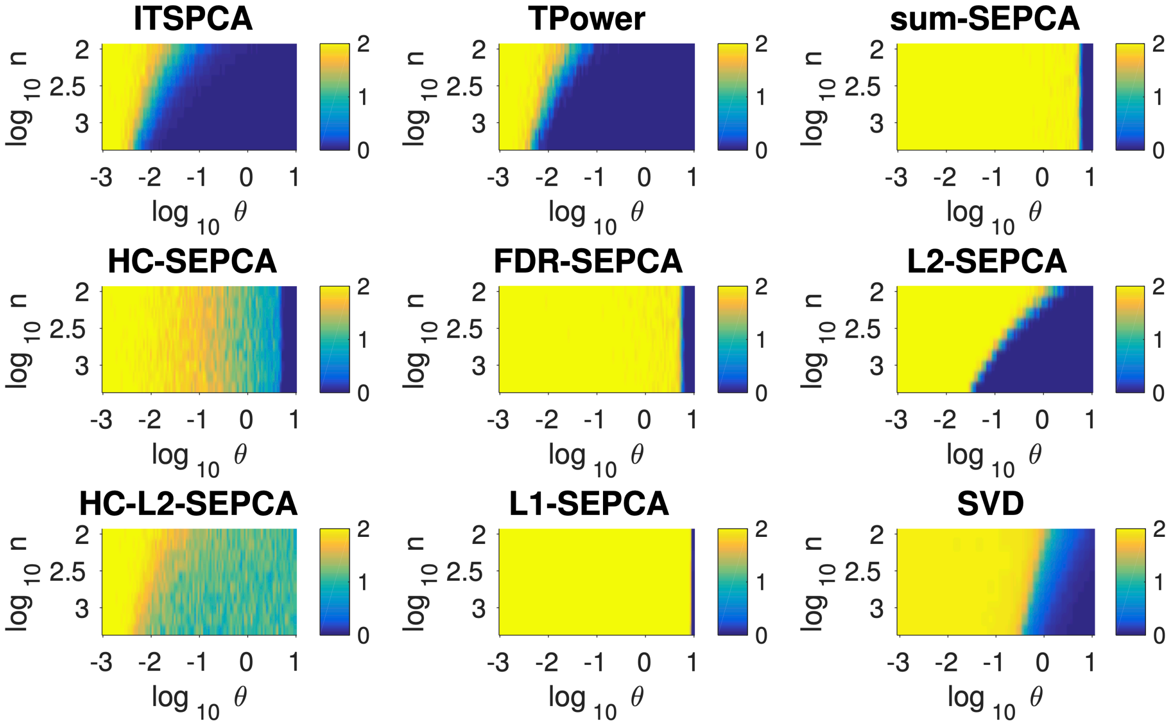


Figure 5.5: The plots show the empirical estimation error for all six algorithms for the u and v described in (5.31). We include results from TPower, ITSPCA and the SVD for comparison.

compare the algorithms. In particular, we have seen that the right singular vector $\mathbf{v}$ plays a critical role in the detectability and estimability of $\mathbf{u}$, and we will characterize this behavior carefully.

In this section, will use the following notational convenience: we absorb (θu_i) into $\mathbf{v} \in \mathbb{R}^n$, and write the detectability of coordinates in terms of $\mathbf{v}$. That is, if $\mathbf{v}^T$ is a row of $\mathbf{X}$, we specify when that row is selected. Moreover, we take $\sigma = 1$ for simplicity.

There are two 'classes' of detectability: in terms of $\| \mathbf{v} \|_1$ and in terms of $\| \mathbf{v} \|_2$. The sum-, HC-sum, and FDR-SEPCA algorithms select a coordinate if $|\sum_k v_k| = \| \mathbf{v} \|_1$ is large enough for a $\mathbf{v}$ in the orthant with all non-negative or all non-positive coordinates. Geometrically, the vector $\mathbf{v}$ is selected if it is 'outside' a hyperplane with a normal vector proportional to the vector of all 1s. The ℓ_1-SEPCA algorithm is similar, as it selects a coordinate when $\| \mathbf{v} \|_1$ is large enough, or if $\mathbf{v}$ lies outside an ℓ_1-ball of some radius. The connection between the previous three algorithms and ℓ_1-SEPCA comes from noting that the faces of an ℓ_1-ball are sections of hyperplanes with normal vectors proportional to a vector of ±1s. Finally, the ℓ_2- and HC-ℓ_2-SEPCA algorithms select a coordinate when

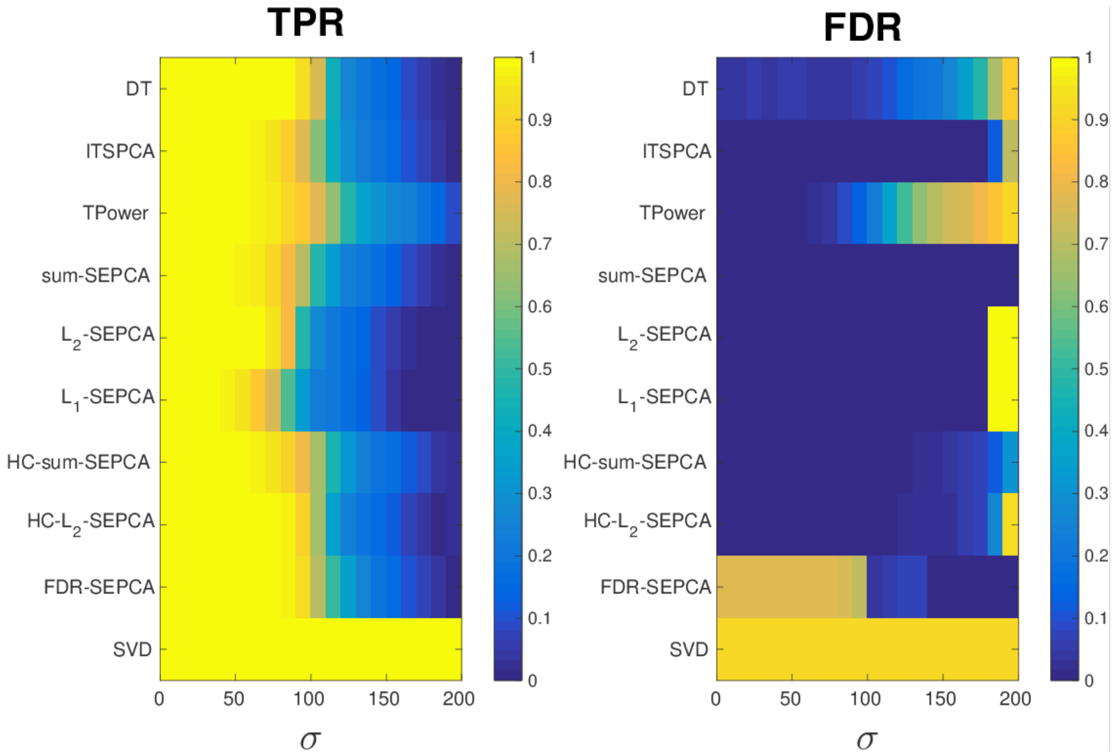
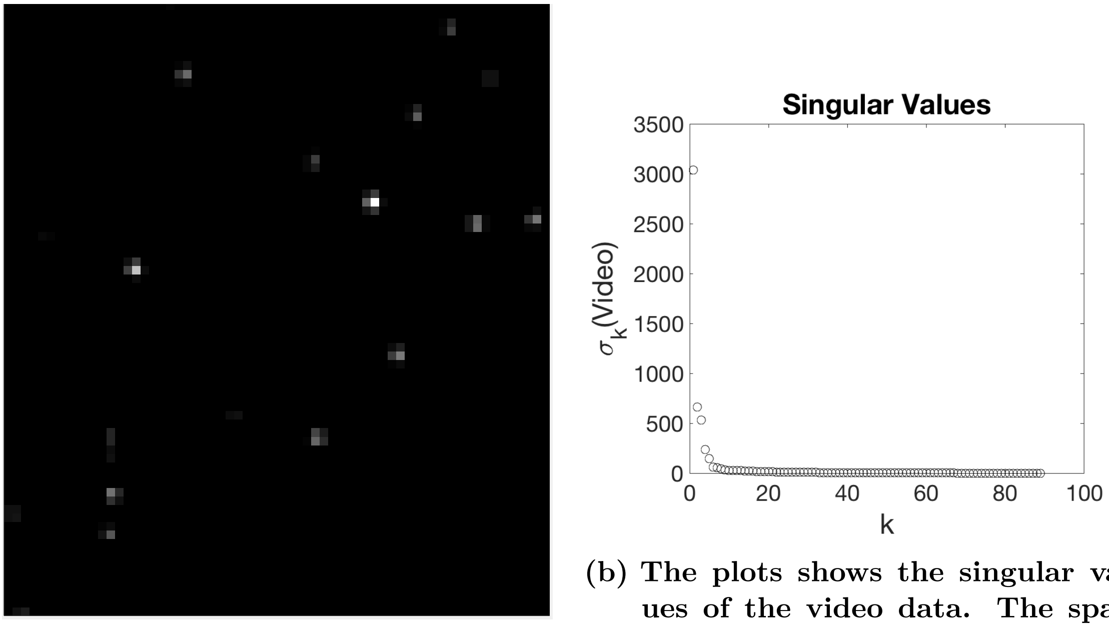


Figure 5.6: The left plot shows the True Positive Rate of the various algorithms as a function of the noise level σ. The right plot shows the False Discovery Rates.



(a) The image shows the mean intensity of pixels from the top-left 72×64 pixels for 89 frames. White indicates the presence of a star.

(b) The plots shows the singular values of the video data. The spacing suggest a low-rank-plus-noise structure.

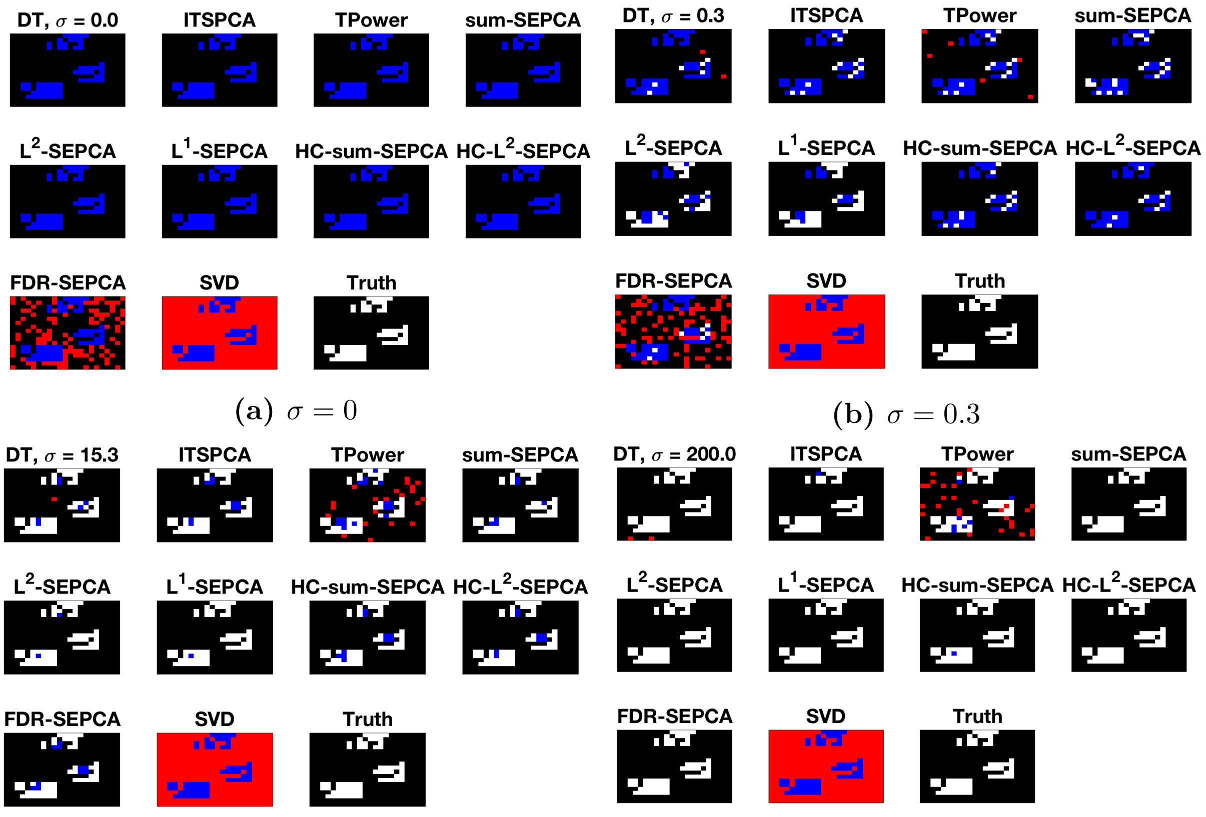
Table 5.2: Video Example Figures

$\| \mathbf{v} \|_2$ is large enough. I.e., when $\mathbf{v}$ lies outside some ℓ_2-ball.

Our goal in this section is to derive comparisons between the six algorithms. Specifically, for a given vector $\mathbf{v}$, which algorithm will have the greatest detection ability (we are, for the moment, only concerned with maximizing power)? Note that when $\mathbf{v}$ has a large norm, it does not matter which algorithm is used. Questions only arise when $\| \mathbf{v} \|_1$ or $\| \mathbf{v} \|_2$ are relatively small and are close to the thresholds.

5.8.1 Intersection of a hyperplane and a hypersphere

We may think of the ℓ_1 ball as a hyperplane when restricted to a single orthant. If a hypersphere of radius r intersects a hyperplane with a normal vector proportional to the vector of all ±1s and minimum distance to the origin of $r - h$, a hyperspherical cap of height h is formed: see Figure 5.8 for a simple illustration. Geometrically, a right triangle is formed, with hypotenuse r and leg $r - h$. Hence, the angle between the center of the



(a) $\sigma = 0$

(b) $\sigma = 0.3$

(c) $\sigma = 15$

(d) $\sigma = 200$

Figure 5.7: A zoomed-in view of the three top-right stars in the video example. White indicates a false negative (missed star), Red a false positive (a guessed pixel where there was nothing), and Blue a true positive (correctly identified pixel).

cap and the edge is:

$$\theta_{lim} = \cos^{-1}\frac{r-h}{r}. \tag{5.32}$$

It is sufficient to guarantee that

$$0 \leq \frac{r-h}{r} \leq 1$$

for the hyperspherical cap to exist. Moreover, a vector $\mathbf{v}$ has a direction contained inside the cap when the angle between $\mathbf{v}$ and the vector of ± 1 in the orthant containing $\mathbf{v}$ is smaller than θ_{lim}. In other words, defining the angle for a vector $\mathbf{v}$ as

$$\theta(\mathbf{v}) = \cos^{-1}\frac{\| \mathbf{v} \|_1}{\| \mathbf{v} \|_2\sqrt{n}}, \tag{5.33}$$

we need $\theta(\mathbf{v}) \leq \theta_{lim}$.

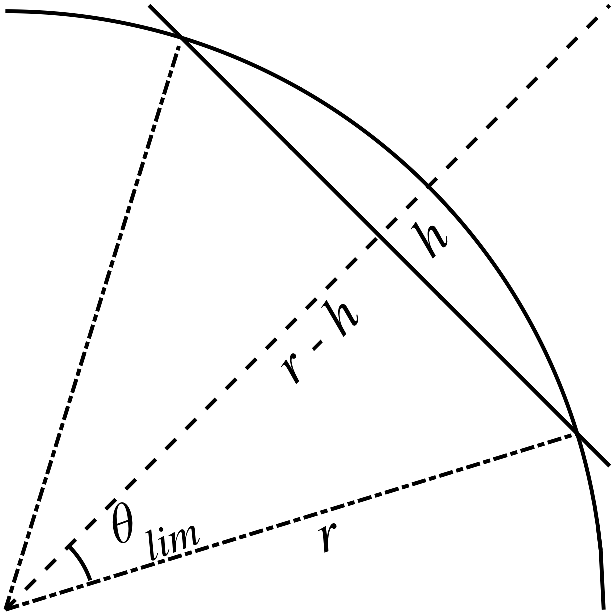


Figure 5.8: A spherical cap in $\mathbb{R}^2$

5.8.2 Comparison: ℓ_2-based versus sum-based statistics

We begin with a summary of the performance of each individual algorithm in Table 5.3. We first compare ℓ_2-SEPCA and then compare HC-ℓ_2-SEPCA with sum-, HC-sum, FDR-SEPCA in Tables 5.4 and 5.5. In our comparisons, we consider when the hyperspherical cap exists and give the angle of the cap. These are routine calculations, so we omit the details. We also omit ℓ_1-SEPCA from our comparisons, as we lack a closed-form expression for t_{ℓ_1}.

Note that the existence of this cap is a proxy for the equivalent statement that there exist vectors for which the sum-based algorithms are more powerful than the ℓ_2-based algorithm. While this existence is not the same as attributing uniformly greater power to the sum-based algorithms relative to the ℓ_2-based algorithm, the cap *not* existing is equivalent to the ℓ_2-based algorithm having uniformly greater power.

Essentially, we observe that for n and p sufficiently large, the cap will exist. Moreover, for $\mathbf{v}$ that is sufficiently dense ($\| \mathbf{v} \|_1$ is sufficiently large), θ_{lim} will lie inside the cap. Hence, in these situations we would prefer a sum-based algorithm over an ℓ_2-based algorithm.

Table 5.3: A summary of the six algorithms.

Algorithm	Threshold	Geometric Quantity
sum	$\\| \mathbf{v} \\|_1 \geq C_U \sqrt{\log p}$	$r - h = C_U \sqrt{\frac{\log p}{n}}$
HC-sum	$\\| \mathbf{v} \\|_1 \geq \sqrt{2\rho(\beta) \log p}$	$r - h = \sqrt{2\rho(\beta) \frac{\log p}{n}}$
FDR	$\\| \mathbf{v} \\|_1 \geq (1 - o(1)) \sqrt{\zeta}$ $\left(1 + \sqrt{2 \log \left(\nu p / \widehat{k}\right)}\right)$	$r - h = (1 - o(1)) \sqrt{\zeta}$ $\frac{1 + \sqrt{2 \log \left(\nu p / \widehat{k}\right)}}{\sqrt{n}}$
ℓ_1	$\\| \mathbf{v} \\|_1 \geq t_{\ell_1}$	$r - h = t_{\ell_1}$
ℓ_2	$\\| \mathbf{v} \\|_2 \geq \sqrt{C_2} \sqrt{\frac{\log ep}{\sqrt{n}}}$	$r = \sqrt{C_2} \sqrt{\frac{\log ep}{\sqrt{n}}}$
HC-ℓ_2	$\\| \mathbf{v} \\|_2 \geq 2\rho(\beta) \frac{2 \log p}{\sqrt{n}}$	$r = 2\rho(\beta) \frac{2 \log p}{\sqrt{n}}$

Table 5.4: The relative performance of ℓ_2-SEPCA.

Algorithm	$\cos \theta_{lim}$	Cap exists if
sum	$C_U / \sqrt{C_2} \sqrt{\frac{\log p}{(1+\log p)\sqrt{n}}}$	$n \geq C_U^4 / C_2^2, p \geq 1$
HC-sum	$\sqrt{2\rho(\beta)/C_2} \sqrt{\frac{\log p}{(1+\log p)\sqrt{n}}}$	$n \geq 1, p \geq 1$
FDR-sum	$\frac{1+\sqrt{2 \log \nu p / \widehat{k}}}{\sqrt{C_2 \sqrt{n} \log \nu p}}$	$p \geq 11, n \geq 1$

Table 5.5: The relative performance of HC-ℓ_2-SEPCA.

Algorithm	$\cos \theta_{lim}$	Cap exists if
sum	$\frac{C_U}{2\rho(\beta)\sqrt{\log p}}$	$p \geq \exp\left(\frac{C_U^2}{4\rho(\beta)^2}\right)$
HC-sum	$[2\rho(\beta) \log p]^{-1}$	$p \geq \exp\left(\frac{1}{2\rho(\beta)}\right)$
FDR-sum	$\frac{1+\sqrt{2 \log \nu p / \widehat{k}}}{2\rho(\beta) \log p}$	$\log p \geq$ $\frac{1}{4\rho(\beta)^2} \big(1 + 2\rho(\beta)$ $+ \sqrt{8\rho(\beta)^2 + 4\rho(\beta) + 1}\big)$

5.8.3 HC-ℓ_2-SEPCA versus ℓ_2-SEPCA

Now, we consider when HC-ℓ_2-SEPCA is more powerful than ℓ_2-SEPCA. The ratio of the radii is given by

$$\frac{2\rho(\beta)}{\sqrt{C_2} n^{1/4}} \frac{\log p}{\sqrt{1 + \log p}}. \tag{5.34}$$

If this ratio is smaller than 1, HC-ℓ_2-SEPCA is more powerful than ℓ_2-SEPCA. Note that the quantity

$$\frac{2\sqrt{2}}{\sqrt{C_2}}\sqrt{\frac{\log p}{\sqrt{n}}}$$

is an upper bound for (5.34), so that if

$$\frac{\log p}{\sqrt{n}} < \frac{e}{4\sqrt{2}},$$

the original ratio is smaller than 1 and HC-ℓ_2-SEPCA is preferable to ℓ_2-SEPCA.

5.8.4 Comparing the sum-based algorithms

Finally, we compare sum-, HC-sum-, and FDR-SEPCA. First, the ratio of the thresholds for HC-sum- and sum-SEPCA is

$$\frac{\sqrt{2\rho(\beta)}}{C_U}. \tag{5.35}$$

Noting that $\rho(\beta) \leq 1$ and that $C_U \geq \sqrt{2} + 1/3\sqrt{2}$, it is clear that this ratio is always smaller than 1 so that HC-sum-SEPCA is a strict improvement on sum-SEPCA.

Next, we compute the ratio of the thresholds for FDR- and sum-SEPCA:

$$\frac{1 + \sqrt{2\log \nu p/\widehat{k}}}{C_U\sqrt{\log p}}. \tag{5.36}$$

Using the lower bound on C_U, we find that if $\widehat{k} \geq 11$ (and $p \geq \widehat{k}$, naturally), FDR-SEPCA is always more powerful than sum-SEPCA. For smaller values of $\widehat{k}$, for sufficiently large values of p, the ratio will be smaller than 1.

Lastly, we compare FDR-SEPCA to HC-sum-SEPCA, wherein the ratio of the thresholds is (FDR to HC-sum):

$$\frac{1 + \sqrt{2\log \nu p/\widehat{k}}}{\sqrt{2\rho(\beta)}\sqrt{\log p}}. \tag{5.37}$$

Because of involvement of $\rho(\beta)$, this quantity is hard to analyze. If in an oracle manner, FDR-SEPCA obtained $\widehat{k}$ correctly as $p^{1-\beta}$, we would find that this ratio is always larger than 1 for $p > 1$. That is if $\widehat{k}$ assumes the the correct value, HC-sum-SEPCA is more powerful than FDR-SEPCA. Alternatively, we can note that $\rho(\beta) \in (0, 1]$ and ask when the ratio is larger than 1. Based on the ratio above, we can see that in the following scenarios

$$\begin{cases} \widehat{k} = 1 & \text{and} \quad p > 1, \\ 2 \leq \widehat{k} \leq 18 & \text{and} \quad p \geq \widehat{k}\,(\text{always}), \\ \widehat{k} \geq 19 & \text{and} \quad \log p \geq \frac{1}{8}\left(4(\log \widehat{k})^2 - 4\log \widehat{k} + 1\right), \end{cases} \tag{5.38}$$

HC-sum-SEPCA is more powerful than FDR-SEPCA.

To summarize, we prefer the FDR-controlling alternatives to sum-SEPCA, but depending on the output of FDR-SEPCA, HC-sum-SEPCA may be more powerful. However, as the simulations in Section 5.7 revealed (see Figure 5.2), the number of false positives with HC-sum-SEPCA may be higher than with FDR-SEPCA.

5.8.5 Overall Message

We have seen that for n and p sufficiently large and $\mathbf{v}$ that is sufficiently dense (in the sense of $\|\mathbf{v}\|_1$ being large), a sum-based statistic and algorithm leads to better performance. This is expected behavior, as by using a sum-based method, we are taking advantage of the equisigned nature of $\mathbf{v}$. Moreover, within the class of sum-based algorithms, controlling the FDR leads to greater power, as expected. It is difficult to clearly identity which of HC-sum- and FDR-SEPCA will have the greatest power, and the end result may come down to a practitioner's tolerance for false discoveries.

5.9 Conclusions

We have considered the setting where the left singular vector of the underlying rank one signal matrix plus noise data matrix is assumed to be sparse and the right singular vector is assumed to be equisigned. We have proposed six different SEPCA algorithms for estimating the sparse principal component based on different decision statistics and provided sparsistency conditions for the same. Our analysis reveals conditions where a coordinate selection scheme based on a sum-based decision statistic outperforms schemes that utilize the ℓ_1 and ℓ_2 decision statistics. Thereby, the proposed algorithm outperforms known schemes such as *diagonal thresholded PCA* [61] in terms of estimation of the singular vectors associated with the rank-1 component. We have derived lower bounds on the size of detectable coordinates of the principal left singular vector, utilized these lower bounds to derive lower bounds on the worst-case risk and verified our findings with numerical simulations. Finally, we have discussed the results of our simulations analytically, by providing a geometric interpretation of the differences in power among the algorithms.

We note that while we have stated our results for Gaussian noise with identity covariance, we can extend the FWER-controlling results to any log-concave noise distribution, and the FDR-controlling procedures to Gaussian noise with certain non-identity covariances. Additionally, another way to view this work is that it proposes a two-stage procedure/framework for sparse PCA based around hypothesis testing of statistics associated to each row. Some natural extensions would be the inclusion or consideration of other testing frameworks, e.g., that in [90], where knowledge of the size of coordinates is taken into account.

5.A Proof of Theorem 5.1

a. Note that

$$\mathbb{P}\left(T_i \geq \tau\right) \leq \mathbb{P}\left(\max_{j \in I^c} T_j \geq \tau\right)$$

for $i \in I^c$. Taking the maximum over the left-hand side and noting that the right-hand side has limit zero yields the result. This follows from (5.5). □

b. We consider when true positives occur with probability approaching 1. We want to find the smallest coordinate (θu_i) such that the following probability approaches 1:

$$\mathbb{P}\left(T_i > \tau_{n,p}\right) = \mathbb{P}\left(\frac{T_i - \mathbb{E}T_i}{\sqrt{\text{Var } T_i}} > \frac{\tau_{n,p} - \mathbb{E}T_i}{\sqrt{\text{Var } T_i}}\right). \tag{5.39}$$

Note that if $(\tau_{n,p} - \mathbb{E}T_i)$ is negative and not tending to zero as n grows, and if the variance of T_i decays to zero as n grows, the quantity

$$\frac{\tau_{n,p} - \mathbb{E}T_i}{\sqrt{\text{Var } T_i}} \tag{5.40}$$

tends toward negative infinity. Hence, we will specify conditions so that Var T_i decays to zero as n grows and then compute when a coordinate is detectable by considering when $\tau_{n,p}$ is strictly less than $\mathbb{E}T_i$. For brevity, we omit the computations in solving $\tau_{n,p} < \mathbb{E}T_i$ for $|\theta u_i|$ and present verifications that the variance of T_i has limit 0. These results show that above the *decision boundary*, we have uniform detection.

In sum-SEPCA, T_i is a Gaussian random variable with mean $\frac{(\theta u_i)}{\sqrt{n}} \sum_k v_k$ and variance $\frac{\sigma^2}{n}$. Since σ does not grow with n, Var T_i always decays to zero.

In ℓ_2-SEPCA, T_i has

$$\mathbb{E}T_i = (\theta u_i)^2 + \sigma^2$$

and

$$\text{Var } T_i = \frac{2\sigma^2}{n} \left(\sigma^2 + 2\,(\theta u_i)^2\right).$$

Since σ and θ are fixed, the variance always decays to 0.

Let $x_{i,k} = \left(\sqrt{n}\frac{v_k(\theta u_i)}{\sigma}\right)$. In ℓ_1-SEPCA, T_i has

$$\begin{aligned}\text{Var } T_i = &\frac{\sigma^2}{n^2}\sum_k x_{i,k}^2 \left(1 - \left(\text{Erf }\left(\frac{x_{i,k}}{\sqrt{2}}\right)\right)^2\right)\\ &+ \frac{\sigma^2}{n}\left(1 - \frac{2}{n\pi}\sum_k \exp\left(-x_{i,k}^2\right)\right)\\ &- 2\sqrt{\frac{2}{\pi}}\frac{\sigma^2}{n^2}\sum_k x_{i,k}\exp\left(-\frac{x_{i,k}^2}{2}\right)\text{Erf }\left(\frac{x_{i,k}}{\sqrt{2}}\right),\end{aligned}$$

which is less than or equal to

$$\frac{(\theta u_i)^2}{n}\sum_k v_k^2 + \frac{\sigma^2}{n} + 2\sqrt{\frac{2}{\pi}}\frac{\sigma}{n\sqrt{n}}|\,(\theta u_i)\,|\sum_k |v_k|. \tag{5.41}$$

Since $\|\,\mathbf{v}\,\|_2 = 1$, the variance of T_i has limit 0. Because we cannot solve the inequality $\tau_{n,p} < \mathbb{E}T_i$ analytically, we leave the bound in the form given previously. □

In the proof above, note that if $(\tau_{n,p} - \mathbb{E}T_i)$ is positive and not tending to zero as n grows, the quantity in (5.40) tends to positive infinity when the variance decays to zero. Hence, modifying the proof by solving $\tau_{n,p} > \mathbb{E}T_i$ for $|\theta u_i|$ yields when a coordinate is not detectable with probability approaching 1: i.e., when $|\theta u_i|$ is smaller than the values given in (5.25). □

5.A.1 Proof of Corollary 5.1

The first three terms of (5.26) are characterized by Theorem 5.1(b) and by noting that $I_0 = I$ in the corollary. The last term can be characterized as follows. In particular, for an algorithm with row test statistics T_i and a threshold τ from Table 5.1, we may write

$$\sum_{i \notin I} \mathbb{P}\left(i \in \widehat{I}\right) = \sum_{i \notin I} \mathbb{P}\left(T_i \geq \tau\right). \tag{5.42}$$

For the ℓ_2- and ℓ_1-SEPCA algorithms, we may reindex the T_i according to their order statistics $T_{(i)}$, where

$$\left|T_{(1)}\right| \geq \left|T_{(2)}\right| \geq \cdots \geq \left|T_{(p-s)}\right|,$$

and write

$$\sum_{i \notin I} \mathbb{P}\left(i \in \widehat{I}\right) = \sum_{i=1}^{p-s} \mathbb{P}\left(T_{(i)} \geq \tau\right). \tag{5.43}$$

Note that there are $p - s$ null entries. For the ℓ_2- and ℓ_1-SEPCA algorithms, we have that (as a consequence of [68, Thm. 3])

$$\mathbb{P}\left(T_{(i)} \geq \tau\right) \leq \left(\frac{1}{ep}\right)^{\sqrt{i}},$$

so that

$$\sum_{i \notin I} \mathbb{P}\left(i \in \widehat{I}\right) = \sum_{i=1}^{p-s} \mathbb{P}\left(T_{(i)} \geq \tau\right) \leq \sum_{i=1}^{p-s} \left(\frac{1}{ep}\right)^{\sqrt{i}}. \tag{5.44}$$

The right-hand side of (5.44) has limit zero, as needed.

For the sum-SEPCA algorithm, from (5.10), it follows that there exists a non-zero constant $\epsilon > 0$ such that the threshold τ satisfies

$$\tau = \sigma\left(\sqrt{2} + \epsilon\right)\sqrt{\frac{\log p}{n}}.$$

In particular, from (5.10), we have that

$$\epsilon = \frac{1 + \sqrt{\log p}/3}{\sqrt{2}\mathrm{Erf}^{-1}\left(1 - 1/p\right)},$$

where for $p > 1$,

$$\frac{1}{3\sqrt{2}} < \epsilon < 2.$$

Hence, for any T_i such that $i \notin I$,

$$\mathbb{P}\left(T_i \geq \tau\right) \leq \exp\left(-\frac{\left(\sqrt{2}+\epsilon\right)^2}{2}\log p\right) = p^{-\left(1+\sqrt{2}\epsilon+\epsilon^2/2\right)},$$

where we have used a Gaussian tail bound [20, Sec. 2.3]. Then,

$$\begin{aligned}\sum_{i \notin I} \mathbb{P}\left(i \in \widehat{I}\right) &= \sum_{i \notin I} \mathbb{P}\left(T_i \geq \tau\right) \\ &\leq \left(p - s\right) p^{-\left(1+\sqrt{2}\epsilon+\epsilon^2/2\right)} \\ &\leq p^{-\left(\sqrt{2}\epsilon+\epsilon^2/2\right)}.\end{aligned} \tag{5.45}$$

Since ϵ is larger than $1/3\sqrt{2}$, the right-hand side of (5.45) is upper bounded by $p^{-13/36}$, which has limit zero, as desired.

5.B Proof of Theorem 5.2

5.B.1 Size of Detectable Coordinates

Sum: HC-SEPCA

If $\mathbf{v}$ is equisigned, summing across the rows of $\mathbf{X}$ yields a normally distributed quantity with mean $(\theta u_i)\|\,\mathbf{v}\,\|_1$ and variance σ^2. Dividing by σ and adopting the notation of HC, we have that under the alternative hypothesis, $\mu_i = \sqrt{2r\log p}$, so that

$$r = \left(\frac{|\theta u_i|\|\,\mathbf{v}\,\|_1}{\sqrt{2\log p}}\right)^2.$$

Rearranging the inequality $r > \rho(\beta)$ yields

$$|\theta u_i| > \sigma\sqrt{\rho(\beta)}\frac{\sqrt{2\log p}}{\|\,\mathbf{v}\,\|_1}. \tag{5.46}$$

Note that sum-SEPCA can detect coordinates of size

$$|\theta u_i| > \sigma C_U \frac{\sqrt{\log p}}{\| \mathbf{v} \|_1}. \tag{5.47}$$

However, C_U is strictly larger than $\sqrt{2} + 1/(3\sqrt{2})$. Thus, using HC yields a threshold of the same order, but with a strictly smaller scaling.

Sum of squares: HC-ℓ_2-SEPCA

If we sum the squares of the entries of rows of $\mathbf{X}$, abusing notation slightly and using $\mathcal{N}(\mu, \sigma^2)$ to indicate a Gaussian random variable with mean μ and variance σ^2, the statistic for the i^{th} coordinate is of the form

$$\sum_{k=1}^{n} \left(\frac{\sigma}{\sqrt{n}} \mathcal{N}\left(\frac{\theta u_i}{\sigma} v_k \sqrt{n}, 1 \right) \right)^2 .$$

Assuming oracular knowledge of σ, the statistic

$$\frac{n}{\sigma^2} \sum_k X_{ik}^2$$

places us in the setting of (5.15). The non-centrality parameter δ is given by

$$\delta = \sqrt{\sum_{k=1}^{n} \left(\frac{\theta u_i}{\sigma} v_k \sqrt{n} \right)^2} = \left| \frac{\theta u_i}{\sigma} \right| \sqrt{n}.$$

Setting $\delta = 2r \log p$ and solving $r > \rho(\beta)$ yields

$$|\theta u_i| > \sigma \rho(\beta) \frac{2 \log p}{\sqrt{n}}. \tag{5.48}$$

We have that ℓ_2-SEPCA can detect coordinates with

$$|\theta u_i| > \sigma \sqrt{e\sqrt{2}} \sqrt{\frac{1 + \log p}{\sqrt{n}}}. \tag{5.49}$$

Using HC offers a significant improvement over ℓ_2-SEPCA. However, we also expect HC with the χ_n^2 statistic to have a smaller detectable coordinate: $\| \mathbf{v} \|_1 \leq \sqrt{n}$, so that for fixed

β and p, the threshold in (5.46) is asymptotically larger than that in (5.48) (but potentially of the same order). This result is strange in context of the non-FDR results. In any case, HC improves on ℓ_2-SEPCA.

FDR-SEPCA

Recall that taking sums across the rows of $\mathbf{X}$, we obtain a vector $\mathbf{y}$ where $y_i = \mu_i + \sigma z_i$, with $\mu_i = (\theta u_i)\| \mathbf{v} \|_1$. Moreover, we have noted that

$$t_k \approx \sqrt{\zeta}(1 + \sqrt{2\log(\nu p/k)}),$$

where t_k is the level at which $\mathbf{y}$ is thresholded. It follows that, entries of $\mathbf{y}$ that are of size at least

$$y_i > (1 - o(1))\sqrt{\zeta}\sigma\left(1 + \sqrt{2\log(\nu p/\widehat{k})}\right)$$

are selected, or, since $\mu_i = (\theta u_i)\| \mathbf{v} \|_1$ (when v is equisigned), if we select $\widehat{k}$ coordinates, we expect to detect

$$\begin{aligned} |\theta u_i| &> (1 - o(1))\sqrt{\zeta}\sigma\frac{\left(1 + \sqrt{2\log(\nu p/\widehat{k})}\right)}{\|v\|_1} \\ &= O\left(\sigma\frac{\sqrt{2\log\left(\nu p/\widehat{k}\right)}}{\| \mathbf{v} \|_1}\right). \end{aligned} \tag{5.50}$$

Relative to HC and sum-SEPCA, the gain here is found when there are many smaller coordinates of $\mathbf{u}$ and $\widehat{k}$ is large.

5.B.2 Proofs for the Higher Criticism-Based Methods

a. From (2.8) in [47],

$$\mathbb{P}\left(T_i \geq \tau\right) \leq \mathbb{P}\left(\max_{j \in I^c} T_j \geq \tau\right)$$

has limit zero. □

b. Let $I_1 \subseteq I$ be the set of coordinates with signal larger than the detection limit ($i \in I$ such that $|\theta u_i| > \beta_{crit}(1 + \epsilon)$), and let $I_2 \subseteq I$ contain the rest of the coordinates ($i \in I$

such that $|\theta u_i| < \beta_{crit}(1-\epsilon)$). By Theorem 1 in [6], the asymptotic power for detecting signals below the detection limit is one, and that for signals below the limit is zero. Hence, for $i \in I_1$,

$$\min_{i\in I \,:\, |\theta u_i|>\beta_{crit}(1+\epsilon)} \mathbb{P}\left(i \text{ selected}\right) \to 1,$$

and for $i \in I_2$,

$$\max_{i\in I \,:\, |\theta u_i|<\beta_{crit}(1-\epsilon)} \mathbb{P}\left(i \text{ selected}\right) \to 0.$$

As with Theorem 1, we omit the computation of β_{crit}, as it follows from the discussion in Section 5.4.1. □

5.B.3 FDR-SEPCA

The details of these computations are in Appendix 5.D.1, so we will summarize the properties here.

a. The choice of $\nu = 2^{1/\omega}$ controls the FDR at level ω [62]. Choosing $\omega = \omega(p) \to 0$ as $p \to \infty$ leads to an asymptotic FDR of zero. I.e., for $i \in I^c$,

$$\max_{i\in I^c} \qquad \mathbb{P}\left(i \text{ selected}\right) \to 0. \quad \square$$

b. Noting that the consistency of estimating the mean vector $\boldsymbol{\mu} = (\theta\|v\|_1)\,\mathbf{u}$ encompasses the estimation of the support of $\mathbf{u}$, risk bounds for the estimation of $\boldsymbol{\mu}$ yield the result. To be precise, if the expected risk $\mathbb{E}\|\,\boldsymbol{\mu} - \widehat{\boldsymbol{\mu}}\|_2^2 \le B$ for some bound B, we expect to detect coordinates of size larger than B and to not detect those smaller than B. □

5.C Risk bounds under ℓ_q sparsity

In this section, we simultaneously generalize our setting to approximate sparsity, and specify the risk lower-bounds. We omit the ℓ_1-SEPCA algorithm from consideration.

Let $\mathbf{u} \in \mathbb{R}^p$ have unit ℓ_2-norm and belong to an ℓ_q ball with radius C for $q \in (0, 2]$. I.e.,

$$\sum_{i=1}^{p} |u_i|^q \le C^q. \tag{5.51}$$

When $q = 0$, we replace C^q with s, the level of 'hard' sparsity. We have the following result:

Theorem 5.3. *Let*

$$L(\widehat{\mathbf{u}}, \mathbf{u}) = \|\mathbf{u} - sign(\langle \mathbf{u}, \widehat{\mathbf{u}} \rangle) \widehat{\mathbf{u}}\|_2^2 \tag{5.52}$$

be the risk of the estimator $\widehat{\mathbf{u}}$ *of* $\mathbf{u}$, *where* $\mathbf{u}$ *is as specified in (5.1) and the estimators are the six algorithms that we have previously described. Then,*

a. sum-, HC-sum, and FDR-SEPCA have expected risks lower-bounded by

$$\mathbb{E}L(\widehat{\mathbf{u}}, \mathbf{u}) \geq O\left([C^q - 1] \| \mathbf{v} \|_1^{-(2-q)} \right). \tag{5.53}$$

b. ℓ_2*-SEPCA has a risk lower-bounded by*

$$\mathbb{E}L(\widehat{\mathbf{u}}, \mathbf{u}) \geq O\left([C^q - 1] n^{-\frac{1}{2}(1-q/2)} \right). \tag{5.54}$$

c. HC-ℓ_2*-SEPCA has a risk lower-bounded by*

$$\mathbb{E}L(\widehat{\mathbf{u}}, \mathbf{u}) \geq O\left([C^q - 1] n^{-(1-q/2)} \right). \tag{5.55}$$

The rest of this section contains the proof of Theorem 5.3.

5.C.1 Proof of Theorem 5.3

We construct a 'worst-case' sparse $\mathbf{u}$. Note that $C^q \geq 1$ necessarily, and that if $C \geq p^{1-q/2}$, every unit norm vector is in the ℓ_q ball. Hence, we take $C \in [1, p^{1-q/2})$.

Let θ and σ be fixed. We want a sparse vector with several coordinates guaranteed to be missed (the probability of not detecting them is asymptotically 1). For this vector $\mathbf{u}$ to be sparse and for the loss to not be 1, set u_1 to be $\sqrt{1 - r_n^2}$, where $r_n^2 = o(1)$, and take $u_2, \cdots, u_{m_n+1}$ to be $r_n / \sqrt{m_n}$. The other coordinates of u are 0, so that $\mathbf{u}$ has unit ℓ_2-norm.

We assume that u_1 is detected with probability 1 as $n \to \infty$, and want to set $u_2, \cdots, u_{m_n+1}$

so that the expected loss is lower bounded by:

$$\begin{aligned}\mathbb{E}L(\mathbf{u},\widehat{\mathbf{u}}) &\geq \sum_{k=1}^{p} |u_k|^2 \mathbb{P}(\text{Not Selecting Coordinate k}) \\ &\geq \sum_{k=2}^{m_n+1} |u_k|^2 \mathbb{P}(\text{Not Selecting Coordinate k}).\end{aligned} \tag{5.56}$$

If coordinates of size $\frac{r_n}{\sqrt{m_n}}$ are not detected, the expected loss is lower bounded by r_n^2.

Let $m_n = \lfloor m \rfloor$ where

$$m = \delta n^{\phi} r_n^{\psi} \|v\|^{\eta}.$$

Note that we have not specified the norm used in $\|v\|$: we will choose the norm at the very end of the calculation. Let

$$r_n = [C^q - 1]^{\alpha} n^{\beta+\gamma q} \|v\|^{\kappa},$$

so that,

$$\frac{r_n}{\sqrt{m_n}} \approx \frac{r_n}{\sqrt{m}} = \frac{1}{\delta} n^{-\phi/2} \|v\|^{\kappa-\eta/2} r_n^{1-\psi/2}.$$

We will choose $\delta, \phi, \eta, \alpha, \beta, \gamma, \kappa, \psi$ so that the ℓ_q sparsity constraint is met and the lower bound r_n^2 is maximized. The sparsity constraint requires:

$$\sum_{i=1}^{p} |u_i|^q = (1-r_n)^{q/2} + m_n^{1-q/2} r_n^q \leq 1 + m^{1-q/2} r_n^q \leq C^q. \tag{5.57}$$

First, we will assume (for now) that $r_n = o(1)$ and that via other parameters we may control the scaling of the coordinate sizes; hence, we set $\psi = 2$. Then,

$$r_n^q m^{1-q/2} = \delta^{1-q/2} n^{2(\beta+\gamma q)+(1-q/2)\phi} [C^q - 1]^{2\alpha} \|v\|^{2\kappa+\eta(1-q/2)}. \tag{5.58}$$

We need this quantity to be smaller than $C^q - 1$. To eliminate the n dependence, we set $\beta = \frac{-\phi}{2}$ and $\gamma = \frac{\phi}{4}$. We choose $\alpha = \frac{1}{2}$ to match powers of $[C^q - 1]$ on both sides of the inequality. Defining another parameter ρ, let $\delta = \rho \| \mathbf{v} \|^{-\eta}$. Then, the inequality is

$$\rho^{1-q/2} \| \mathbf{v} \|^{2\kappa} [C^q - 1] \leq [C^q - 1].$$

Choosing $\rho \leq \| \mathbf{v} \|^{-2\kappa/(1-q/2)}$ is enough.

With these choices of parameters,

$$r_n = \sqrt{[C^q - 1]} n^{-\frac{1}{2}\phi(1-q/2)} \| \mathbf{v} \|^{\kappa},$$

and

$$m = \rho n^{\phi} r_n^2,$$

so that

$$\frac{r_n}{\sqrt{m}} = \frac{1}{\sqrt{\rho}} n^{-\phi/2}.$$

Noting that

$$\frac{1}{\sqrt{\rho}} \geq \| \mathbf{v} \|^{\kappa/(1-q/2)},$$

choosing $\rho = \| \mathbf{v} \|^{-2\kappa/(1-q/2)}$ leads to the smallest possible choice of coordinate.

In summary:

$$r_n = \sqrt{[C^q - 1]} n^{-\frac{1}{2}\phi(1-q/2)} \| \mathbf{v} \|^{\kappa}, \tag{5.59}$$

$$r_n^2 = [C^q - 1] n^{-\phi(1-q/2)} \| \mathbf{v} \|^{2\kappa}, \tag{5.60}$$

$$m = \| \mathbf{v} \|^{-2\kappa/(1-q/2)} n^{\phi} r_n^2, \tag{5.61}$$

and

$$\frac{r_n}{\sqrt{m}} = \| \mathbf{v} \|^{\kappa/(1-q/2)} n^{-\phi/2}. \tag{5.62}$$

So, for a given algorithm, it remains to choose ϕ and κ so that the worst-case risk is lower-bounded by r_n^2. Sum-SEPCA misses coordinates of size $O\left(\frac{\sqrt{\log p}}{\| \mathbf{v} \|_1}\right)$ and ℓ_2-SEPCA misses coordinates of size $O\left(\frac{\sqrt{\log p}}{n^{1/4} \| \mathbf{v} \|_2}\right)$. For sum-SEPCA, $\kappa = \frac{q-2}{2}$, and for ℓ_2-SEPCA, κ is irrelevant, as $\| \mathbf{v} \|_2 = 1$. Sum-SEPCA uses $\phi = 0$ and ℓ_2-SEPCA uses $\phi = \frac{1}{2}$. Hence, sum-SEPCA has a risk lower-bounded by

$$O\left([C^q - 1] \| \mathbf{v} \|_1^{-(2-q)} \right). \tag{5.63}$$

Noting that $\| \mathbf{v} \|_2 = 1$, ℓ_2-SEPCA has

$$\begin{aligned} &O\left([C^q - 1] n^{-\frac{1}{2}(1-q/2)} \| \mathbf{v} \|_2^{-(1-q/2)} \right) = \\ &O\left([C^q - 1] n^{-\frac{1}{2}(1-q/2)} \right). \end{aligned} \tag{5.64}$$

In the ℓ_0 case, i.e., when $\mathbf{u}$ has no more than s non-zero entries, the preceding analysis goes through with C^q replaced by s and q set to zero.

FDR Algorithms

For HC-sum-SEPCA, the β_{crit} is of the same order as that for sum-SEPCA. Similarly, for FDR-SEPCA, if $\widehat{k}$ is much smaller than p, β_{crit} is of roughly the same order. Hence, these two algorithms have the same risk bound as sum-SEPCA. For HC-ℓ_2-SEPCA, $\kappa = 0$ and $\phi = 1$. The risk is therefore lower-bounded by

$$O\left([C^q - 1]n^{-(1-q/2)}\right). \tag{5.65}$$

5.D FDR-SEPCA: Further Details

Let $y_i = \mu_i + \sigma z_i$, where $i \in \{1, \cdots, p\}$, the vector $\mathbf{z}$ of the z_i is normally distributed with mean 0 and covariance $\boldsymbol{\Sigma}$, and $\boldsymbol{\Sigma}$ satisfies

$$\xi_o \mathcal{I}_p \leq \boldsymbol{\Sigma} \leq \xi_1 \mathcal{I}_p.$$

Here, ξ_0 is the smallest eigenvalue of $\boldsymbol{\Sigma}$ and ξ_1 is the largest. The mean vector $\boldsymbol{\mu}$ of the μ_i is assumed to be sparse; the goal is to estimate $\boldsymbol{\mu}$. The following penalized least squares formulation yields an estimator for $\boldsymbol{\mu}$:

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \| \mathbf{y} - \boldsymbol{\mu} \|_2^2 + \sigma^2 \text{pen}\left(\| \boldsymbol{\mu} \|_0\right), \tag{5.66}$$

where pen(k) is defined as

$$\text{pen}(k) = \xi_1 \zeta k \left(1 + \sqrt{2L_{p,k}}\right)^2, \tag{5.67}$$

with $\zeta > 1$ and

$$L_{p,k} = (1 + 2\beta)\log(\nu p/k). \tag{5.68}$$

The parameter β may be set to 0 here, and ν is chosen to be no smaller than $e^{1/(1+2\beta)}$. We define $\| \boldsymbol{\mu} \|_0$ to be number of non-zero coordinates of $\boldsymbol{\mu}$.

The solution to (5.66) is given by hard-thresholding. Let $|y|_{(i)}$ be the i^{th} order statistic of $|y_i|$, namely $|y|_{(1)} \geq \cdots \geq |y|_{(p)}$. Then if

$$\widehat{k} = \arg\min_{k \geq 0} \sum_{i>k} |y|_{(i)}^2 + \sigma^2 \text{pen}(k), \tag{5.69}$$

defining

$$t_k^2 = \text{pen}(k) - \text{pen}(k-1), \tag{5.70}$$

the solution is to hard threshold at $t_{\widehat{k}}$.

In this set-up, we have that

$$t_k \approx \lambda_{p,k} = \sqrt{\xi_1 \zeta}(1 + \sqrt{2L_{p,k}}),$$

with $|t_k - \lambda_{p,k}| < c/\lambda_k$. More precisely, Lemma 11.7 of [60] says that

$$\lambda_{p,k} - \frac{4\zeta b}{\lambda_{p,k}} \leq t_k \leq \lambda_{p,k}.$$

When $\nu \geq e^2$, we may take $b = (1 + 2\beta)$. In any case, if $k = o(n)$, $\lambda_{p,k} \asymp \sqrt{\log p}$. Hence, entries of $\mathbf{y}$ that are of size at least

$$y_i > (1 - o(1))\sqrt{\xi_1 \zeta}\sigma \left(1 + \sqrt{2\log(\nu p/\widehat{k})}\right)$$

are selected, or, since $\mu_i = (\theta u_i)\| \mathbf{v} \|_1$ (when $\mathbf{v}$ is equisigned), if we select $\widehat{k}$ coordinates, we expect to detect

$$\begin{aligned} |\theta u_i| &> (1 - o(1))\sqrt{\xi_1 \zeta}\sigma \frac{\left(1 + \sqrt{2\log(\nu p/\widehat{k})}\right)}{\| \mathbf{v} \|_1} \\ &= O\left(\sigma \sqrt{\xi_1} \frac{\sqrt{2\log\left(\nu p/\widehat{k}\right)}}{\| \mathbf{v} \|_1}\right). \end{aligned} \tag{5.71}$$

5.D.1 Risk Behavior

Recalling that (5.66) solves a penalized least squares problem for $\widehat{\mathbf{y}}$ close to $\mathbf{y}$, we may discuss the statistical behavior of this estimator. The following discussion follows and reproduces that in [62]

First, note that for $\beta = 0$, the parameter ν directly controls the FDR (where a false positive corresponds to selecting a zero coordinate in $\mathbf{y}$): a choice of $\nu = 2^{1/\omega}$ for $\omega \in (0,1)$ bounds the FDR at a level ω.

Second, the expected risk, $\mathbb{E}\| \mathbf{y} - \widehat{\mathbf{y}}\|_2^2$, is bounded as follows. By Proposition 4.1 in [62],

$$\mathbb{E}\| \mathbf{y} - \widehat{\mathbf{y}}\|_2^2 \leq D\left[2M_p'\xi_1\sigma^2 + \mathcal{R}(\mathbf{y},\sigma)\right], \tag{5.72}$$

where D is a constant $2\zeta(\zeta+1)^3(\zeta-1)^{-3} = \Theta(1)$, we assume that $\xi_1 = 1$, and $0 \leq M_p' \leq C_\beta p^{-2\beta}\nu^{-1}$, for some $C_\beta > 0$. Since $\beta = 0$, $M_p' = O(1/\nu) = O(\omega)$, if we control the FDR at level ω.

The second term in (5.72) is the ideal risk, or, the infimum of the penalized least squares objective. If $\mathbf{y}$ belongs to an ℓ_q ball with radius C and $0 < q < 2$, and we define

$$r_{p,q}(C) = \begin{cases} C^2 & \text{if } C \leq \sqrt{1+\log p}, \\ C^q[1+\log(p/C^q)]^{1-q/2} & \text{if } \sqrt{1+\log p} \leq C \leq p^{1/q}, \\ p & \text{if } C \geq p^{1/q}, \end{cases} \tag{5.73}$$

the ideal risk is bounded as

$$\sup_{\mathbf{y}\in\mathbb{R}^p:\sum_i |y_i|^q \leq C^q} \mathcal{R}(\mathbf{y},\sigma) \leq c(\log\nu)\sigma^2 r_{p,q}(C/\sigma), \tag{5.74}$$

for some $c > 0$. The supplementary results in [62] yield that $\mathcal{R}(\mathbf{y},\sigma)$ is bounded by $C^2\log\nu$, and by C^2 when $C \leq \sqrt{1+\log p}$.

As in Appendix 5.C, we may replace q with 0 and C^q with s in the case of hard sparsity with s non-zero coordinates. Doing so leads to the bound:

$$\mathbb{E}\| \mathbf{y} - \widehat{\mathbf{y}}\|_2^2 \leq s\sigma\log\nu\log\frac{\sigma p\nu}{s} + \sigma_1\sigma^2\frac{2}{\nu}. \tag{5.75}$$

Note that we have recovered the factor of $\log\nu p/s$ in β_{crit}.

Chapter 6

Afterword

In this work, we have studied two methods, both of which are eigenvalue problems, for learning, inferring, and unmixing signals in the presence of noise. We have seen that when the signals are structured, the effect of noise can be mitigated and we can recover the latent signal.

We have studied the Dynamic Mode Decomposition algorithm in the noise-free, missing data, and noisy data settings. A major novelty and advance of this work is the demonstration that DMD can solve the blind source separation problem [29]. Moreover, we have demonstrated that a truncated SVD denoising step can improve performance of the algorithm. Additionally, we have presented different perspectives on the algorithm, including that of convex optimization.

We have also presented a novel perspective for sparse PCA, based off hypothesis testing. We have incorporated FDR control into the sparse PCA problem, and have compared the effectiveness of various decision statistics in the rank-1, non-negative coordinates setting. Moreover, this work generalizes to more than just Gaussian-distributed noise.

6.1 Future Work and Open Questions

There are some unanswered questions, and it is our hope that this thesis can serve as a starting point toward answering those.

First, the performance of DMD without the truncated SVD is an open question. So far, we have shown that we may express the DMD eigenvalue problem of a rank-1 plus noise data matrix as a rank-4 perturbation of the noise-only setting. However, analysis of the perturbation is an open problem, as is the analysis of how the perturbation affects the result. Moreover, we have not extended our work to higher rank signal matrices, and it is not clear what the relationship between signal rank and the perturbation is. Second, the analysis of DMD with the truncated SVD for the non-orthogonal model is open. This is the subject of ongoing work, and the design of the optimal weighting estimator is still open.

Taking a step back, the behavior of DMD in the noise-only setting is still an open question. We have conjectures that we can numerically substantiate, but a formal proof is still to be written down. In particular, we need to show convergence of the empirical eigenvalue density to the conjectured density, and we have undertaken preliminary moment calculations.

Returning to DMD, another interesting direction that one could take this work would be the incorporation of multiple lags: a novelty of this work was the demonstration that DMD could be performed at lags other than one. Much like the SOBI algorithm, it may be advantageous to use more than one lag to recover the latent signals/eigenvectors [83]. The SOBI algorithm performs a joint diagonalization of covariance matrices from different lags, and one might imagine a similar approach for DMD (jointly diagonalizing several non-Hermitian $\widehat{A}$ matrices). Taking a step back from the incorporation of multiple lags, there is still the question of choosing the optimal lag (much like with SOBI). Our investigations have shown that lags that have the greatest separation of autocorrelations (from each other and from zero) while having the lowest cross-correlations of the latent signals lead to the best performance. However, without oracle knowledge of the latent signals, it is not necessarily clear how one should choose the best lag. Finally, non-linear extensions of this work, particularly in the design and analysis of provably convergent DMD-based unmixing on non-linearly mixed ergodic time series are of interest and would complement related works on non-linear ICA [3, 39, 57, 78, 55, 22, 58, 46, 4, 133] and non-linear DMD [130, 124].

In Chapter 4, we presented several vignettes that extended the themes in our DMD analysis. The first problem that we considered was the convex optimization framework that mimicked the DMD eigenvalue problem and offered the ability to impose sparsity. We have presented an algorithm to solve the problem and some results to characterize the resulting solutions, but our method is computationally demanding and not tractable for larger problems. An immediate next step would be to either reformulate this problem or improve the algorithm to handle larger problems. One immediate improvement to the tractability would be to replace the parameter grid search with some more intelligent, like a

Bayesian Hyperparameter Search [112] or even a simple random search [15]. Additionally, it would be interesting to formalize the performance analysis of the algorithm for noisy data.

Two other ideas in Chapter 4 were DMD performed on Hilbert-transformed data and a two-dimensional, spatial DMD analogue. It would be interesting to study the behavior and effects of the algorithm on non-sinusoidal data. For the two-dimensional DMD algorithm, a careful study in the same theme as that in Chapter 2 would be of interest.

Bibliography

[1] Gernot Akemann et al. "Universal microscopic correlation functions for products of truncated unitary matrices". In: *Journal of Physics A: Mathematical and Theoretical* 47.25 (2014), p. 255202.

[2] Zeyuan Allen-Zhu and Yuanzhi Li. "LazySVD: even faster SVD decomposition yet without agonizing pain". In: *Advances in Neural Information Processing Systems*. 2016, pp. 974–982.

[3] Luís B Almeida. "MISEP–Linear and Nonlinear ICA Based on Mutual Information". In: *J. Mach. Learn. Res.* 4.Dec (2003), pp. 1297–1318.

[4] Shun-ichi Amari, Andrzej Cichocki, and Howard H Yang. "Recurrent neural networks for blind separation of sources". In: *Proc. Int. Symp. NOLTA*. 1995.

[5] Joakim Andén and José Luis Romero. "Multitaper estimation on arbitrary domains". In: *arXiv preprint arXiv:1812.03225* (2018).

[6] Ery Arias-Castro, Emmanuel J Candès, and Yaniv Plan. "Global testing under sparse alternatives: ANOVA, multiple comparisons and the higher criticism". In: *Annals of Statistics* (2011), pp. 2533–2556.

[7] Behtash Babadi and Emery N Brown. "A review of multitaper spectral analysis". In: *IEEE Trans. Biomed. Eng.* 61.5 (2014), pp. 1555–1564.

[8] Shervin Bagheri. "Koopman-mode decomposition of the cylinder wake". In: *J. Fluid Mech.* 726 (2013), pp. 596–623.

[9] Zhe Bai et al. "Dynamic Mode Decomposition for compressive system identification". In: *arXiv preprint arXiv:1710.07737* (2017).

[10] Emilio Barocio et al. "A Dynamic Mode Decomposition framework for global power system oscillation analysis". In: *IEEE Trans. Power Sys.* 30.6 (2015), pp. 2902–2912.

[11] Adel Belouchrani et al. "A Blind Source Separation technique using second-order statistics". In: *IEEE Trans. Signal Process.* 45.2 (1997), pp. 434–444.

[12] Florent Benaych-Georges and Raj Rao Nadakuditi. "The singular values and vectors of low rank perturbations of large rectangular random matrices". In: *Journal of Multivariate Analysis* 111 (Oct. 2012), pp. 120–135. ISSN: 0047-259X. DOI: 10.1016/j.jmva.2012.04.019. URL: http://dx.doi.org/10.1016/j.jmva.2012.04.019.

[13] Florent Benaych-Georges and Jean Rochet. “Outliers in the single ring theorem”. In: *Probability Theory and Related Fields* 165.1-2 (2016), pp. 313–363.

[14] Erik Berger et al. “Estimation of perturbations in robotic behavior using Dynamic Mode Decomposition”. In: *Adv. Robot.* 29.5 (2015), pp. 331–343.

[15] James Bergstra and Yoshua Bengio. “Random search for hyper-parameter optimization”. In: *Journal of Machine Learning Research* 13.Feb (2012), pp. 281–305.

[16] Quentin Berthet, Philippe Rigollet, et al. “Optimal detection of sparse principal components in high dimension”. In: *Annals of Statistics* 41.4 (2013), pp. 1780–1815.

[17] Aharon Birnbaum et al. “Minimax bounds for sparse PCA with noisy high-dimensional data”. In: *Annals of Statistics* 41.3 (2013), p. 1055.

[18] Alex Bloemendal et al. “Isotropic local laws for sample covariance and generalized Wigner matrices”. In: *Electron. J. Probab* 19.33 (2014), pp. 1–53.

[19] Sergey G Bobkov and Fedor L Nazarov. “On convex bodies and log-concave probability measures with unconditional basis”. In: *Geometric aspects of functional analysis*. Springer, 2003, pp. 53–69.

[20] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford University Press, 2013.

[21] Stéphane Boucheron and Maud Thomas. “Concentration inequalities for order statistics”. In: *Electronic Communications in Probability* 17 (2012), no. 51, 1–12. ISSN: 1083-589X. DOI: `10.1214/ECP.v17-2210`. URL: `http://ecp.ejpecp.org/article/view/2210`.

[22] Philemon Brakel and Yoshua Bengio. “Learning independent features with adversarial nets for non-linear ICA”. In: *arXiv preprint arXiv:1710.05050* (2017).

[23] Matthew Brennan, Guy Bresler, and Wasim Huleihel. “Reducibility and computational lower bounds for problems with planted sparse structure”. In: *arXiv preprint arXiv:1806.07508* (2018).

[24] Cristina Butucea et al. “Variable selection with Hamming loss”. In: *The Annals of Statistics* 46.5 (2018), pp. 1837–1875.

[25] Emmanuel J Candes and Yaniv Plan. “A probabilistic and RIPless theory of compressed sensing”. In: *Information Theory, IEEE Transactions on* 57.11 (2011), pp. 7235–7254.

[26] Jean-François Cardoso and Antoine Souloumiac. “Blind beamforming for non-Gaussian signals”. In: *IEE proceedings F (radar and signal processing)*. Vol. 140. 6. IET. 1993, pp. 362–370.

[27] Aiyou Chen, Peter J Bickel, et al. “Efficient independent component analysis”. In: *The Annals of Statistics* 34.6 (2006), pp. 2825–2855.

[28] Kevin K Chen, Jonathan H Tu, and Clarence W Rowley. “Variants of Dynamic Mode Decomposition: Boundary condition, Koopman, and Fourier analyses”. In: *J. Nonlinear Sci.* 22.6 (2012), pp. 887–915.

[29] Seungjin Choi et al. “Blind Source Separation and Independent Component Analysis: A review”. In: *NIP-LR* 6.1 (2005), pp. 1–57.

[30] Nelida Črnjarić-Žic, Senka Maćešić, and Igor Mezić. “Koopman Operator Spectrum for Random Dynamical System”. In: *arXiv preprint arXiv:1711.03146* (2017).

[31] Alexandre d’Aspremont et al. “A direct formulation for sparse PCA using semidefinite programming”. In: *SIAM Review* 49.3 (2007), pp. 434–448.

[32] Mark A Davenport and Justin Romberg. “An overview of low-rank matrix recovery from incomplete observations”. In: *IEEE Sel. Top. Signal Proc.* 10.4 (2016), pp. 608–622.

[33] James W Demmel. *Applied numerical linear algebra.* Vol. 56. SIAM, 1997.

[34] Chris HQ Ding, Tao Li, and Michael I Jordan. “Convex and semi-nonnegative matrix factorizations”. In: *IEEE Transactions on Pattern analysis and Machine Intelligence* 32.1 (2010), pp. 45–55.

[35] David Donoho, Matan Gavish, et al. “Minimax risk of matrix denoising by singular value thresholding”. In: *The Annals of Statistics* 42.6 (2014), pp. 2413–2440.

[36] David Donoho and Jiashun Jin. “Higher criticism for detecting sparse heterogeneous mixtures”. In: *Annals of Statistics* (2004), pp. 962–994.

[37] David Donoho, Jiashun Jin, et al. “Higher criticism for large-scale inference, especially for rare and weak effects”. In: *Statistical Science* 30.1 (2015), pp. 1–25.

[38] Carl Eckart and Gale Young. “The approximation of one matrix by another of lower rank”. In: *Psychometrika* 1.3 (1936), pp. 211–218.

[39] Jan Eriksson and Visa Koivunen. “Blind identifiability of class of nonlinear instantaneous ICA models”. In: *Proceedings of the 11th EUSIPCO.* IEEE. 2002, pp. 1–4.

[40] Steve Fisk. “A very short proof of Cauchy’s interlace theorem for eigenvalues of Hermitian matrices”. In: *Am. Math. Mon.* 112.math. CA/0502408 (2005), p. 118.

[41] François Le Gall and Florent Urrutia. “Improved rectangular matrix multiplication using powers of the Coppersmith-Winograd tensor”. In: *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms.* SIAM. 2018, pp. 1029–1046.

[42] Chen Gao, Brian E Moore, and Raj Rao Nadakuditi. “Augmented robust PCA for foreground-background separation on noisy, moving camera video”. In: *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP).* IEEE. 2017, pp. 1240–1244.

[43] Matan Gavish and David L Donoho. "The optimal hard threshold for singular values is $4/\sqrt{3}$". In: *IEEE Transactions on Information Theory* 60.8 (2014), pp. 5040–5053.

[44] Donald Goldfarb and Shiqian Ma. "Fast multiple-splitting algorithms for convex optimization". In: *SIAM Journal on Optimization* 22.2 (2012), pp. 533–556.

[45] Nina Golyandina, Vladimir Nekrutkin, and Anatoly A Zhigljavsky. *Analysis of time series structure: SSA and related techniques*. Chapman and Hall/CRC, 2001.

[46] Emad M Grais, Mehmet Umut Sen, and Hakan Erdogan. "Deep neural networks for single channel source separation". In: *Proceedings of the ICASSP*. IEEE. 2014, pp. 3734–3738.

[47] Peter Hall, Jiashun Jin, et al. "Innovated higher criticism for detecting sparse signals in correlated noise". In: *Annals of Statistics* 38.3 (2010), pp. 1686–1732.

[48] Edward J Hannan. "The uniform convergence of autocovariances". In: *Ann. Statist.* (1974), pp. 803–806.

[49] Alfred Hanssen. "Multidimensional multitaper spectral estimation". In: *Signal Proc.* 58.3 (1997), pp. 327–332.

[50] Jinane Harmouche et al. "The sliding singular spectrum analysis: A data-driven nonstationary signal decomposition tool". In: *IEEE Transactions on Signal Processing* 66.1 (2017), pp. 251–263.

[51] Maziar S Hemati et al. "De-biasing the Dynamic Mode Decomposition for applied Koopman spectral analysis of noisy datasets". In: *Theor. Comput. Fluid Dyn.* 31.4 (2017), pp. 349–368.

[52] An Hong-Zhi, Chen Zhao-Guo, and Edward J Hannan. "Autocorrelation, autoregression and autoregressive approximation". In: *Ann. Statist.* (1982), pp. 926–936.

[53] Patrik O Hoyer. "Non-negative matrix factorization with sparseness constraints". In: *Journal of Machine Learning Research* 5.Nov (2004), pp. 1457–1469.

[54] Kejun Huang, Nicholas D Sidiropoulos, and Ananthram Swami. "Non-negative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition". In: *IEEE Transactions on Signal Processing* 62.1 (2014), pp. 211–224.

[55] A. J. Hyvarinen and H. Morioka. "Nonlinear ICA of temporally dependent stationary sources". In: *Proceedings of Machine Learning Research*. 2017.

[56] Aapo Hyvarinen, Juha Karhunen, and Erkki Oja. *Independent Component Analysis, A Wiley-Interscience Publication*. 2001.

[57] Aapo Hyvarinen and Hiroshi Morioka. "Unsupervised feature extraction by time-contrastive learning and nonlinear ICA". In: *Advances in Neural Information Processing Systems*. 2016, pp. 3765–3773.

[58] Aapo Hyvarinen, Hiroaki Sasaki, and Richard E Turner. “Nonlinear ICA using auxiliary variables and generalized contrastive learning”. In: *arXiv preprint arXiv:1805.08651* (2018).

[59] Charles R Johnson and Roger A Horn. *Matrix analysis*. Cambridge University Press, 1985.

[60] Iain M Johnstone. “Gaussian estimation: Sequence and wavelet models”. In: *Unpublished manuscript* (2017). `http://statweb.stanford.edu/~imj/GE_08_09_17.pdf`.

[61] Iain M Johnstone and Arthur Yu Lu. “On consistency and sparsity for principal components analysis in high dimensions”. In: *Journal of the American Statistical Association* 104.486 (2009), p. 682.

[62] Iain M Johnstone and Debashis Paul. “Adaptation in some linear inverse problems”. In: *Stat* 3.1 (2014), pp. 187–199.

[63] D Cryer Jonathan and Chan Kung-Sik. “Time series analysis with applications in R”. In: *SpringerLink, Springer eBooks* (2008).

[64] Mihailo R Jovanović, Peter J Schmid, and Joseph W Nichols. “Sparsity-promoting Dynamic Mode Decomposition”. In: *Physics of Fluids* 26.2 (2014), p. 024103.

[65] M. Karker et al. “Evaluation of sparse sampling approaches for 3D functional MRI”. In: *Proc. Intl. Soc. Mag. Res. Med.* 2019, p. 0370. URL: `https://index.mirasmart.com/ISMRM2019/PDFfiles/0370.html`.

[66] Gaetan Kerschen et al. “The method of Proper Orthogonal Decomposition for dynamical characterization and order reduction of mechanical systems: An overview”. In: *Nonlinear dynamics* 41.1-3 (2005), pp. 147–169.

[67] J. Kutz et al. *Dynamic Mode Decomposition*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2016.

[68] Rafał Latała. “Order statistics and concentration of norms for log-concave vectors”. In: *Journal of Functional Analysis* 261.3 (2011), pp. 681 –696. ISSN: 0022-1236. DOI: `http://dx.doi.org/10.1016/j.jfa.2011.02.013`. URL: `http://www.sciencedirect.com/science/article/pii/S002212361100070X`.

[69] Rafał Latała. “Some estimates of norms of random matrices”. In: *Proc. Am. Math. Soc.* 133.5 (2005), pp. 1273–1282.

[70] Te-Won Lee. “Independent Component Analysis”. In: *Independent component analysis*. Springer, 1998, pp. 27–66.

[71] Zeng Li, Qinwen Wang, Jianfeng Yao, et al. “Identifying the number of factors from singular values of a large sample auto-covariance matrix”. In: *The Annals of Statistics* 45.1 (2017), pp. 257–288.

[72] Claire Yilin Lin and Jeffrey A Fessler. "Efficient Dynamic Parallel MRI Reconstruction for the Low-Rank Plus Sparse Model". In: *IEEE Transactions on Computational Imaging* 5.1 (2018), pp. 17–26.

[73] Haifeng Liu et al. "Constrained nonnegative matrix factorization for image representation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.7 (2012), pp. 1299–1311.

[74] Bethany Lusch, J Nathan Kutz, and Steven L Brunton. "Deep learning for universal linear embeddings of nonlinear dynamics". In: *Nature communications* 9.1 (2018), p. 4950.

[75] Zongming Ma et al. "Sparse principal component analysis and iterative thresholding". In: *Annals of Statistics* 41.2 (2013), pp. 772–801.

[76] Jordan Mann and J Nathan Kutz. "Dynamic Mode Decomposition for financial trading strategies". In: *Quantitative Finance* 16.11 (2016), pp. 1643–1655.

[77] Markus Matilainen, Klaus Nordhausen, and Joni Virta. "On the number of signals in multivariate time series". In: *LVA/ICA*. Springer. 2018, pp. 248–258.

[78] Takeru Matsuda and Aapo Hyvarinen. "Estimation of Non-Normalized Mixture Models and Clustering Using Deep Representation". In: *arXiv preprint arXiv:1805.07516* (2018).

[79] Carl D Meyer and Gilbert W Stewart. "Derivatives and perturbations of eigenvectors". In: *SIAM J. Numer. Anal.* 25.3 (1988), pp. 679–691.

[80] Carl D Meyer Jr. "Generalized inversion of modified matrices". In: *SIAM Journal on Applied Mathematics* 24.3 (1973), pp. 315–323.

[81] Igor Mezić. "Analysis of fluid flows via spectral properties of the Koopman operator". In: *Annu. Rev. Fluid Mech.* 45 (2013), pp. 357–378.

[82] Jari Miettinen, Klaus Nordhausen, and Sara Taskinen. "Blind Source Separation Based on Joint Diagonalization in R: The Packages JADE and BSSasymp". In: *J. Stat. Software* 76.2 (2017), pp. 1–31.

[83] Jari Miettinen et al. "Separation of uncorrelated stationary time series using autocovariance matrices". In: *J. Time Series Anal.* 37.3 (2016), pp. 337–354.

[84] Yoshiki Mitsui et al. "Blind Source Separation based on independent low-rank matrix analysis with sparse regularization for time-series activity". In: *Proceedings of the ICASSP*. IEEE. 2017, pp. 21–25.

[85] Andrea Montanari and Emile Richard. "Non-negative principal component analysis: Message passing algorithms and sharp asymptotics". In: *IEEE Transactions on Information Theory* 62.3 (2015), pp. 1458–1484.

[86] Brian Moore, Chen Gao, and Raj Rao Nadakuditi. "Panoramic robust PCA for foreground-background separation on noisy, free-motion camera video". In: *IEEE Transactions on Computational Imaging* (2019).

[87] Juan Luis Morera and LLandel Veguilla. *Rakata*. Universal Music Group, 2005. URL: `https://www.youtube.com/watch?v=C3lXxoQPqoA`.

[88] Raj Rao Nadakuditi. "OptShrink: An algorithm for improved low-rank signal matrix denoising by optimal, data-driven singular value shrinkage". In: *IEEE Trans. Inform. Theory* 60.5 (2014), pp. 3002–3018.

[89] Boaz Nadler. "Nonparametric detection of signals by information theoretic criteria: Performance analysis and an improved estimator". In: *IEEE Transactions on Signal Processing* 58.5 (2010), pp. 2746–2756.

[90] Mohamed Ndaoud. "Interplay of minimax estimation and minimax support recovery under sparsity". In: *arXiv preprint arXiv:1810.05478* (2018).

[91] Alan V Oppenheim and Ronald W Schafer. *Discrete Time Signal Processing*. Prentice-Hall, 2010.

[92] Sean O'Rourke, Van Vu, and Ke Wang. "Random perturbation of low rank matrices: Improving classical bounds". In: *Linear Algebra Appl.* 540 (2018), pp. 26–59.

[93] Emanuel Parzen. "On consistent estimates of the spectrum of a stationary time series". In: *Ann. Math. Stat.* (1957), pp. 329–348.

[94] Damien Passemier, Zhaoyuan Li, and Jianfeng Yao. "On estimation of the noise variance in high dimensional probabilistic principal component analysis". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 79.1 (2017), pp. 51–67.

[95] Damien Passemier and Jian-Feng Yao. "On determining the number of spikes in a high-dimensional spiked population model". In: *Random Matrices: Theory and Applications* 1.01 (2012), p. 1150002.

[96] Dominique Pastor and Francois-Xavier Socheleau. "Robust estimation of noise standard deviation in presence of signals with unknown distributions and occurrences". In: *IEEE transactions on Signal Processing* 60.4 (2012), pp. 1545–1555.

[97] S Pendergrass et al. "Dynamic Mode Decomposition for Background Modeling". In: *Proceedings of the ICCVW*. IEEE. 2017, pp. 1862–1870.

[98] Amelia Perry et al. "Optimality and sub-optimality of PCA I: Spiked random matrix models". In: *The Annals of Statistics* 46.5 (2018), pp. 2416–2451.

[99] Massimo Piccardi. "Background subtraction techniques: A review". In: *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*. Vol. 4. IEEE. 2004, pp. 3099–3104.

[100] Arvind Prasadan, Asad Lodhia, and Raj Rao Nadakuditi. "Phase Transitions in the Dynamic Mode Decomposition Algorithm". In: *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2019 IEEE Workshop on*. IEEE. 2019, pp. 1–5.

[101] Arvind Prasadan and Raj Rao Nadakuditi. "The Finite Sample Performance of Dynamic Mode Decomposition". In: *Signal and Information Processing (GlobalSIP), 2018 IEEE Global Conference on*. IEEE. 2018, pp. 1–5.

[102] Arvind Prasadan and Raj Rao Nadakuditi. "Time Series Source Separation using Dynamic Mode Decomposition". In: *arXiv preprint arXiv:1903.01310* (2019, In Review).

[103] Arvind Prasadan, Raj Rao Nadakuditi, and Debashis Paul. "Sparse Equisigned PCA: Algorithms and Performance Bounds in the Noisy Rank-1 Setting". In: *arXiv preprint arXiv:1905.09369, to appear, Electronic Journal of Statistics* (2020).

[104] Pradeep Ravikumar, Martin J Wainwright, and et al. Lafferty John D. "High-dimensional Ising model selection using ℓ_1-regularized logistic regression". In: *Annals of Statistics* 38.3 (2010), pp. 1287–1319.

[105] Saiprasad Ravishankar, Raj Rao Nadakuditi, and Jeffrey A Fessler. "Efficient sum of outer products dictionary learning (SOUP-DIL) and its application to inverse problems". In: *IEEE transactions on computational imaging* 3.4 (2017), pp. 694–709.

[106] Galen Reeves and Michael Gastpar. "Sampling bounds for sparse support recovery in the presence of noise". In: *2008 IEEE International Symposium on Information Theory*. IEEE. 2008, pp. 2187–2191.

[107] Bin Ren et al. "Non-negative matrix factorization: robust extraction of extended structures". In: *The Astrophysical Journal* 852.2 (2018), p. 104.

[108] Paul Ross. *Stars*. `https://archive.org/details/Stars_2D`. Accessed: 2016 November 16.

[109] Clarence W Rowley et al. "Spectral analysis of nonlinear flows". In: *J. Fluid Mech.* 641 (2009), pp. 115–127.

[110] Peter J Schmid. "Dynamic Mode Decomposition of numerical and experimental data". In: *J. Fluid Mech.* 656 (2010), pp. 5–28.

[111] Ari Shapiro, Yong Cao, and Petros Faloutsos. "Style components". In: *Proceedings of Graphics Interface 2006*. Canadian Information Processing Society. 2006, pp. 33–39.

[112] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. "Practical Bayesian optimization of machine learning algorithms". In: *Advances in neural information processing systems*. 2012, pp. 2951–2959.

[113] Francois-Xavier Socheleau and Dominique Pastor. "Testing the energy of random signals in a known subspace: An optimal invariant approach". In: *IEEE Signal Processing Letters* 21.10 (2014), pp. 1182–1186.

[114] Raj Tejas Suryaprakash and Raj Rao Nadakuditi. “Consistency and MSE performance of MUSIC-based DOA of a single source in white noise with randomly missing data”. In: *IEEE Transactions on Signal Processing* 63.18 (2015), pp. 4756–4770.

[115] Naoya Takeishi et al. “Bayesian Dynamic Mode Decomposition”. In: *Proceedings of the Twenty-Sixth IJCAI*. 2017, pp. 2814–2821.

[116] Akaysha C Tang, Jing-Yu Liu, and Matthew T Sutherland. “Recovery of correlated neuronal sources from EEG: the good and bad ways of using SOBI”. In: *Neuroimage* 28.2 (2005), pp. 507–519.

[117] Terence Tao. *Topics in random matrix theory*. Vol. 132. American Mathematical Soc., 2012.

[118] Leo Taslaman and Björn Nilsson. “A framework for regularized non-negative matrix factorization, with application to the analysis of gene expression data”. In: *PloS one* 7.11 (2012), e46331.

[119] Pierre-Antoine Thouvenin, Nicolas Dobigeon, and Jean-Yves Tourneret. “Hyperspectral unmixing with spectral variability using a perturbed linear mixing model”. In: *IEEE Transactions on Signal Processing* 64.2 (2015), pp. 525–538.

[120] Petr Tichavskỳ et al. “A computationally affordable implementation of an asymptotically optimal BSS algorithm for AR sources”. In: *14th EUSIPCO*. IEEE. 2006, pp. 1–5.

[121] Lang Tong et al. “AMUSE: A new blind identification algorithm”. In: *ISCAS*. IEEE. 1990, pp. 1784–1787.

[122] Nikolaus F Troje. “Decomposing biological motion: A framework for analysis and synthesis of human gait patterns”. In: *Journal of vision* 2.5 (2002), pp. 2–2.

[123] Nikolaus F Troje. “The little difference: Fourier based gender classification from biological motion”. In: *Dynamic perception* (2002), pp. 115–120.

[124] Jonathan H Tu et al. “On Dynamic Mode Decomposition: Theory and applications”. In: *J. Comput. Dyn.* 1.2 (2014), pp. 391–421.

[125] Munetoshi Unuma, Ken Anjyo, and Ryozo Takeuchi. “Fourier principles for emotion-based human figure animation”. In: *Proceedings of the 22nd PACMCGIT*. Citeseer. 1995, pp. 91–96.

[126] Jacobus Johannes Van Der Leeuw. *The Conquest of Illusion*. Pickle Partners Publishing, 2017.

[127] Namrata Vaswani et al. “Robust subspace learning: Robust PCA, robust subspace tracking, and robust subspace recovery”. In: *IEEE signal processing magazine* 35.4 (2018), pp. 32–55.

[128] Yu-Xiong Wang and Yu-Jin Zhang. “Nonnegative matrix factorization: A comprehensive review”. In: *IEEE Transactions on Knowledge and Data Engineering* 25.6 (2013), pp. 1336–1353.

[129] James Hardy Wilkinson. *The algebraic eigenvalue problem*. Vol. 87. Clarendon Press Oxford, 1965.

[130] Matthew O Williams, Ioannis G Kevrekidis, and Clarence W Rowley. “A data–driven approximation of the Koopman operator: Extending Dynamic Mode Decomposition”. In: *J. Nonlinear Sci.* 25.6 (2015), pp. 1307–1346.

[131] Hao Wu and Raj Rao Nadakuditi. “Free Component Analysis: Theory, Algorithms & Applications”. In: *arXiv preprint arXiv:1905.01713* (2019).

[132] Yangyang Xu and Wotao Yin. “A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion”. In: *SIAM Journal on imaging sciences* 6.3 (2013), pp. 1758–1789.

[133] Bo Yang et al. “Learning Nonlinear Mixtures: Identifiability and Algorithm”. In: *arXiv preprint arXiv:1901.01568* (2019).

[134] Xiao-Tong Yuan and Tong Zhang. “Truncated power method for sparse eigenvalue problems”. In: *Journal of Machine Learning Research* 14.Apr (2013), pp. 899–925.

[135] Ron Zass and Amnon Shashua. “Nonnegative sparse PCA”. In: *Advances in neural information processing systems*. 2007, pp. 1561–1568.

[136] Hao Zhang et al. “Online Dynamic Mode Decomposition for time-varying systems”. In: *Bulletin Am. Phys. Soc.* 62 (2017).