**Title**
Estimation and Inference in High-dimensional Models



**Author**
Sahraee Ardakan, Mojtaba

**Publication Date**
2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

# Estimation and Inference in High-dimensional Models

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

by

Mojtaba Sahraee Ardakan

2022

ABSTRACT OF THE DISSERTATION

Estimation and Inference
in High-dimensional Models

by

Mojtaba Sahraee Ardakan

Doctor of Philosophy in Electrical and Computer Engineering
University of California, Los Angeles, 2022
Alyson K. Fletcher, Chair

A wide variety of problems that are encountered in different fields can be formulated as an inference problem. Common examples of such inference problems include estimating parameters of a model from some observations, inverse problems where an unobserved signal is to be estimated based on a given model and some measurements, or a combination of the two where hidden signals along with some parameters of the model are to be estimated jointly. For example, various tasks in machine learning such as image inpainting and super-resolution can be cast as an inverse problem over deep neural networks. Similarly, in computational neuroscience, a common task is to estimate the parameters of a nonlinear dynamical system from neuronal activities. Despite wide application of different models and algorithms to solve these problems, our theoretical understanding of how these algorithms work is often incomplete. In this work, we try to bridge the gap between theory and practice by providing theoretical analysis of three different estimation problems.

First, we consider the problem of estimating the input and hidden layer signals in a given multi-layer stochastic neural network with all the signals being matrix valued. Various problems such as multitask regression and classification, and inverse problems that use deep generative priors can be modeled as inference problem over multi-layer neural networks. We consider different types of estimators for such problems and exactly analyze the performance

of these estimators in a certain high-dimensional regime known as the large system limit. Our analysis allows us to obtain the estimation error of all the hidden signals in the deep neural network as expectations over low-dimensional random variables that are characterized via a set of equations called the state evolution.

Next, we analyze the problem of estimating a signal from convolutional observations via ridge estimation. Such convolutional inverse problems arise naturally in several fields such as imaging and seismology. The shared weights of the convolution operator introduces dependencies in the observations that makes analysis of such estimators difficult. By looking at the problem in the Fourier domain and using results about Fourier transform of a class of random processes, we show that this problem can be reduced to analysis of multiple ordinary ridge estimators, one for each frequency. This allows us to write the estimation error of the ridge estimator as an integral that depends on the spectrum of the underlying random process that generates the input features.

Finally, we conclude this work by considering the problem of estimating the parameters of a multi-dimensional autoregressive generalized linear model with discrete values. Such processes take a linear combination of the past outputs of the process as the mean parameter of a generalized linear model that generates the future values. The coefficients of the linear combination are the parameters of the model and we seek to estimate these parameters under the assumption that they are sparse. This model can be used for example to model the spiking activity of neurons. In this problem, we obtain a high-probability upper bound for the estimation error of the parameters. Our experiments further support these theoretical results.

The dissertation of Mojtaba Sahraee Ardakan is approved.

Lin Yang

Lieven Vandenberghe

Arash A. Amini

Alyson K. Fletcher, Committee Chair

University of California, Los Angeles

2022

*To Fateme...*

*without whose support*

*this work would never have been possible.*

# Contents

# List of Figures

**Acknowledgements**

This dissertation would not have been possible without the help of my adviser, Prof. Alyson K. Fletcher and her constant support throughout my studies in MS program in statistics and PhD program in electrical and computer engineering, as well the countless hours of discussion and helpful insight provided by Prof. Sundeep Rangan. I would like to thank Prof. Arash A. Amini who introduced me to the world statistics, Prof. Lieven Vandenberghe from whom I learned so much in the field of optimization, and Prof. Lin Yang who provided me with insightful comments and suggestion. I am thankful to Prof. P. Schniter who collaborated with us on many of our publications. My studies at University of California, Los Angeles were funded in part by NSF Grants 1738285 and 1738286, and ONR Grant N00014-15-1-2677.

## Vita

Education

| | |
|---|---|
| Sharif University of Technology | Sep 2010- Feb 2013 |
| MS in Electrical Engineering | |
| University of Tehran | Sep 2006- Sep 2010 |
| BS in Electrical Engineering | |

Internships

| | |
|---|---|
| Research Intern, Adobe Research | Jun 2020- Sep 2020 |
| Research Intern, Microsoft Research | Jul 2019- Sep 2019 |
| Research Intern, Microsoft Research | Jul 2018- Sep 2018 |

Awards

| | |
|---|---|
| Jack Keil Wolf ISIT Best Student Paper Award | 2019 |

# Chapter 1

# Introduction

Estimation and inverse problems where one is interested to make inference about unknown parameters or unobserved signals from a set of observations are widely encountered in various fields. Even though a wide variety of models and algorithms to solve such estimation problems are used in practice with great success, very often our theoretical understanding of why they work is limited. In this work we focus on three different estimation and inverse problems that are inspired by problems in machine learning and computational neuroscience, even though they are applicable to other areas as well.

The first problem we consider is to estimate the input in hidden signals in a given multi-layer neural network where all the signals are matrix valued. Next, we consider the inverse problem of estimating a signal from convolutional measurements which is also known as deconvolution. And finally we look at the problem of estimating parameters of an autoregressive generalized linear model. Our goal is to gain a better theoretical understanding of how different estimators perform in these problems. These theoretical results can help us understand different phenomena such as the now well-known double descent curve [Belkin et al., 2018, Belkin et al., 2019b, Mei and Montanari, 2019, Nakkiran et al., 2021] that we have observed empirically, aid us in model selection and hyper-parameter tuning, as well as guide us towards designing better models and learning algorithms.

In this chapter, we briefly discuss some background material on estimation problems and the statistical regimes in which the theoretical results are established. Then we briefly introduce the problems that we study in the subsequent chapters.

## 1.1 Estimation Problems

Estimation is the problem of making inference about an unknown signal based on some observations. Typical examples include:

- **Parameter estimation problems:** we are given a parametric model, e.g. a distribution $\mathbb{P}_\theta$ parameterized by $\boldsymbol{\theta}$ along with some samples from this distribution or a dynamical system with dynamics parameterized by $\boldsymbol{\theta}$ along with a trajectory of the states, and the goal is to estimate the true parameter $\boldsymbol{\theta}$.

- **Inverse problems:** we have an unknown signal $\mathbf{x}$ that goes through a (stochastic) model $\mathcal{M}$ and we observe $\mathbf{y} = \mathcal{M}(\mathbf{x})$ and the goal is to recover $\mathbf{x}$ from $\mathbf{y}$. Famous examples of inverse problems include compressed sensing where $\mathbf{x}$ is an unknown sparse signal that we would like to reconstruct from $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w}$ where $\mathbf{A}$ is known matrix and $\mathbf{w}$ is noise with known distribution or known moments up to a sufficient order.

- **Joint parameter and unknown signal reconstruction:** This is a combination of the above two problems where we have an unknown signal $\mathbf{x}$ that goes through a stochastic model $\mathcal{M}_{\boldsymbol{\theta}}$ parameterized by $\boldsymbol{\theta}$ and we aim to recover $\mathbf{x}$ as well as estimate the parameters $\boldsymbol{\theta}$ from the observations $\mathbf{y} = \mathcal{M}_{\boldsymbol{\theta}}(\mathbf{x})$. Examples of this problem include system identification where we want to jointly estimate the parameters of a system as well as the unobserved hidden states or the compressed sensing problem discussed above when the noise or signal $\mathbf{x}$ come from a parameterized family of distributions with unknown parameters.

In Chapters 3 and 4 we discuss two inverse problems whereas in Chapter 5 we consider a

parameter estimation problem.

## 1.2 Estimators

Even though we have classified estimation problems into different categories, they can all be modeled similarly. Here we discuss the inverse problem but the same formulation applies to the estimation problem and joint estimation and inverse problem with only slight modifications. Consider an unknown parameter $\boldsymbol{\theta}$ that we want to estimate from the observation $\mathbf{y}$. The relation between $\boldsymbol{\theta}$ and $\mathbf{y}$ can be modeled by a *likelihood function* $p(\mathbf{y}|\boldsymbol{\theta})$. In Bayesian inference, we further assume that the unknown signal $\boldsymbol{\theta}$ has a *prior* distribution $p(\boldsymbol{\theta})$. From a frequentist point of view, we only assume that $\boldsymbol{\theta}$ belongs to some set $\Omega$. Here for simplicity, we have assumed that all distributions have densities with respect to an underlying measure.

Given the statistical models, we can form different kinds of estimators and analyze their properties. Of particular importance are three estimators discussed below.

**Maximum likelihood estimator (MLE):** This estimator recovers the unknown signal by maximizing the likelihood function

$$\widehat{\boldsymbol{\theta}}_{\mathsf{MLE}} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{y}|\boldsymbol{\theta}).$$

**Bayesian estimators:** Bayesian estimators are obtained via the posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$. From Bayes rule the posterior can be written as

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})},$$

where $p(\mathbf{y}) = \int_{\boldsymbol{\theta}} p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}$. Common examples of Bayesian estimators are mode of the posterior which is known as the *maximum a posteriori* estimator $\widehat{\boldsymbol{\theta}}_{\mathsf{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathbf{y})$ and mean of the posterior known as the *minimum mean squared error (MMSE)* estimator

$\widehat{\boldsymbol{\theta}}_{\mathsf{MMSE}} = \mathbb{E}[\boldsymbol{\theta}|\mathbf{y}]$. More generally, given a loss $\mathcal{L}(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}})$, one can consider the minimizer of posterior loss

$$\widehat{\boldsymbol{\theta}} = \widehat{f}(\mathbf{y}), \quad \widehat{f} = \arg\min_{f \in \mathcal{F}} \mathbb{E}[\mathcal{L}(\boldsymbol{\theta}, f(\mathbf{y}))|\mathbf{y}],$$

where $\mathcal{F}$ is the class of all measurable functions of $\mathbf{y}$. $\widehat{\boldsymbol{\theta}}_{\mathsf{MMSE}}$ is a special case of this estimator with $\mathcal{L}(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}) = \left\|\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}\right\|_2^2$.

**M-estimators:** Given a loss function $\mathcal{L}(\boldsymbol{\theta}, \mathbf{y})$ and possibly a regularizer $\mathcal{R}(\boldsymbol{\theta})$, an M-estimator is defined as

$$\widehat{\boldsymbol{\theta}}_{\mathsf{M}} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \mathbf{y}) + \mathcal{R}(\boldsymbol{\theta}).$$

Both MLE and MAP estimators are special kinds of M-estimators with $\mathcal{L}(\boldsymbol{\theta}, \mathbf{y}) = -\log p(\mathbf{y}|\boldsymbol{\theta})$ and $\mathcal{R}(\boldsymbol{\theta}) = -\log p(\boldsymbol{\theta})$ for MAP and $\mathcal{R}(\boldsymbol{\theta}) = 0$ for MLE.

## 1.3 Statistical Regimes

Assume that we would like to estimate a parameter $\boldsymbol{\theta} \in \mathbb{R}^p$ from $n$ observations. Once we choose an estimator, we can ask about the performance of the estimator. Denoting the estimated $\boldsymbol{\theta}$ by $\widehat{\boldsymbol{\theta}}$, the performance is usually measured in terms of a risk $\mathcal{R}(\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}})$ where $\mathcal{L}$ is a loss that measures how close the estimated parameter is to the true parameter. Common loss examples include $\mathcal{L}(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}) = \|\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}\|_2^2$ and $\mathbb{I}\{\boldsymbol{\theta} \neq \widehat{\boldsymbol{\theta}}\}$ -where $\mathbb{I}\{\cdot\}$ denotes the indicator function- corresponding to the squared error and 0-1 loss respectively. In the Bayesian setting, one is usually interested in the average risk $\mathbb{E}_{\boldsymbol{\theta}}\mathcal{R}(\boldsymbol{\theta})$ where the expectation is taken with respect to the randomness of the parameter, i.e. the prior distribution, as well as the randomness of the data, whereas in a frequentest approach we might be interested in the worst case risk $\sup_{\boldsymbol{\theta} \in \Omega} \mathbb{E}\mathcal{R}(\boldsymbol{\theta})$ where $\Omega$ is the parameter space and expectation is only taken with respect to randomness of the data. These average or worst case risks are a function of the number of observations $n$. Except in very simple problems, in most problems of interest deriving the exact value of risk as a function of $n$ cannot be done. Therefore, we usually

either resort to computational approaches such as Markov Chain Monte Carlo (MCMC) methods or bootstrapping to estimate the risk curve, we obtain only bounds on the risk of the estimators, or we study them in more simplified settings.

Classically, estimators were studied in the setting where the number of parameters $p$ is fixed, but the number of samples $n$ is going to infinity. Properties of an estimator can then be studied under the limit of $n \to \infty$. This framework is usually called the *asymptotic theory* or the *large sample theory*. In the large sample limit, one can study asymptotic properties of estimators such as consistency, asymptotic risk of an estimator, or even better the asymptotic distribution of the estimator among others. The idea is that even though these results only hold in the limit of $n \to \infty$, the results might be approximately valid even for large enough sample sizes. At the same time, in this limit, we can take advantage of many tools such as laws of large numbers or central limit theorem (CLT) that are not available to us in the finite sample regime. Such tools makes the analysis of properties of estimators much easier, and looking at the asymptotic behavior of an estimator is usually the first thing that is done, but unfortunately, the results of large sample theory might not be a good predictor of their behavior in the finite sample regime.

In most problems of interest, the number of samples is comparable to the number of parameters, or even worse it could be far fewer than the number of parameters as is the case for example in compressed sensing sensing [Donoho, 2006]. If we plot a figure where one axis corresponds to the number of samples and the other to the number of parameters as in Figure 1.1, the regime where we are actually interested in corresponds to the green box where both the number of samples and number of parameters are finite. Unfortunately, in this regime, as mentioned earlier, it is usually not possible to obtain exact characterization of the performance of estimators. Therefore, instead of looking at the exact behavior, we are usually satisfied with high probability bounds on the risk. Obtaining such results are usually very difficult and require using tools such as concentration of measure. One example of this approach is discussed in Chapter 5 where we consider the problem of estimating the

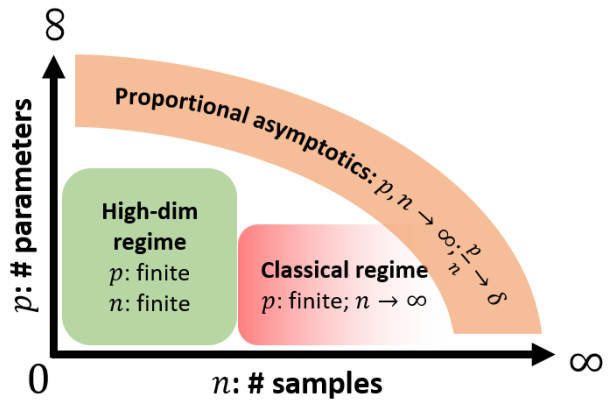



**Figure 1.1:** Different statistical regimes based on the number of parameters and the number of samples.

parameters of a nonlinear multi-dimensional autoregressive process.

More recently, a new regime has gained interest in which the number of samples is going to infinity, but at the same time the number of parameters is also going to infinity. In most of the works, both $n$ and $p$ go to infinity at a fixed ratio, i.e. $n \to \infty, n/p \to \delta \in (0, \infty)$ which we call the *proportional asymptotics*. But more generally, we can consider the case where $n \to \infty$ and $p = f(n)$ for some function $f$ which represents the rate at which the number of parameters is going to infinity. The most common form of $f$ is $p = n^\alpha$ for some fixed $\alpha \in \mathbb{R}$ but other rates of convergence can also be considered. The hope is that this regime might be the sweet spot between the classical large sample theory and the finite sample theory, where the number of samples being comparable to the number of parameters is more realistic for most modern machine learning tasks, but as we take the limit certain convergent behavior emerges from the problem that makes the analysis a lot easier. Therefore, the results are hopefully approximately correct even in the finite sample regime if both the number of parameters and sample are large enough as is the case in many machine learning problems. Examples of such convergent behavior are convergence of spectrum of large random matrices to known distributions among other results in random matrix theory. The results of the problems considered in both Chapter 3 and 4 are derived in the proportional asymptotic regime.

## 1.4 Organization of This Work

In this work, we consider three different problems that are presented in separate chapters. For each problem, we derive theoretical guarantees for the performance of specific estimators. To obtain such theoretical results, we use tools from high-dimensional statistics as well as approximate message passing. These background materials are reviewed in Chapter 2 and are used extensively throughout the subsequent chapters. We start by reviewing now-classic results in compressed sensing where we seek to solve linear inverse problems under the assumption of sparsity. We then review a general framework to analyze high-dimensional M-estimators. We conclude Chapter 2 by presenting two more recent algorithms: approximate message passing (AMP) and vector approximate message passing (VAMP). These algorithms analyze the performance of estimators in linear inverse problems in the proportional asymptotics regime. Then main problems are presented in Chapters 3 to 5. Each chapter is self-contained and can be read at any order or skipped altogether.

In Chapter 3 we consider the problem of estimating the input and hidden layer signals in a given multi-layer stochastic neural network with all the signals being matrix valued. This work is a generalization of our previous work [Pandit et al., 2020] where we considered the vector valued problem. We show that there are several interesting problems that can be modeled by the matrix valued generalization. We rigorously show that the estimated signals and the true signals jointly converge in a certain sense to some random variables the distribution of which we can compute through a set of recursive equations. These results hold in the proportional asymptotics regime where the number of hidden units in each layer as well as the output dimensions go to infinity at a fixed ratio and when all the weights and biases of the network are random as precisely defined in the next chapter. This enables us to exactly compute different types of losses between the true signals and the estimated signals such as the mean squared error.

Next, in Chapter 4, we consider the linear inverse problem of estimating a signal via convolutional measurements. Such convolutional inverse problems arise naturally in many

different fields such as signal processing. Similar to previous chapter, we show that if we use ridge regression to estimate the signal, the mean squared error can be calculated using a scalar integral that depends on the spectral properties of the true signal. This result also holds in a certain proportional asymptotic regime where the number of channels of the input signal and measurement signal are going to infinity at a fixed ratio. Here, channel denotes the definition that is widely accepted in machine learning community when using convolutional layers in neural network architectures. See Chapter 4 for the precise definition.

Finally, in Chapter 5, we consider the problem of estimating the parameters of a multi-dimensional autoregressive generalized linear model (GLM) where the data is discrete valued and the parameters are sparse. This model can be used for example to model the network of neurons in the brain and their spiking activity over time. Autoregressive (AR) GLMs are similar to the widely used Gaussian AR processes except that the distribution of the next state given the current state of the process is derived via a GLM rather than a Gaussian distribution. In this work, we bound the estimation error of $\ell_1$-regularized maximum likelihood estimator under the assumption that the parameters of the model are sparse. The results of this chapter are obtained in the finite sample, finite parameter regime.

# Chapter 2

# Background on Sparse Inverse Problems

Before presenting the problems that are considered in this work and our contributions, it is worth spending some time reviewing a few of the main theoretical results in sparse inverse problems. Such problems were a very hot area of research in the late 90s and early 2000s and sparse recovery approaches were commonly used for a wide range of tasks such as signal compression, denoising, image super-resolution, and compressed sensing to name a few. Nowadays, with improvements in hardware, availability of very large labeled and unlabeled datasets, as well as high-level deep learning libraries like PyTorch [Paszke et al., 2019] and TensorFlow [Abadi et al., 2015], deep learning methods have become the state of the art in most tasks and have replaced simple sparse recovery approaches in most cases. Nevertheless, much of the theory behind sparse recovery methods and the intuition that can be gained by studying them are still applicable to many of the problems we face today, and studying them would give us the tools that we will frequently use in the next few chapters.

In this chapter we first review some of the sparse recovery results for linear inverse problems using square loss in the noiseless and noisy settings. Next, we briefly mention the extension of these results to other loss functions which allows us to look at other M-estimators such as maximum likelihood estimator. Finally, we conclude this chapter by looking at a different approach to these inverse problems using approximate message passing. These

approaches include the approximate message passing framework as well as the closely related vector approximate message passing method.

## 2.1 Sparse linear inverse problems in high dimensions

Let $\boldsymbol{\theta}^* \in \mathbb{R}^p$ be the true parameter vector and consider the problem of estimating $\theta^*$ from linear observations

$$y^i = \langle \mathbf{x}^i, \boldsymbol{\theta}^* \rangle + \xi^i, \quad i = 1, \ldots, n. \tag{2.1}$$

Here, $\mathbf{x}^i \in \mathbb{R}^p$ is the feature vector or the vector of covariates for the $i$th data point and $\xi^i$ corresponds to the noise. This observation model can be rewritten in matrix form as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\theta}^* + \boldsymbol{\xi}, \tag{2.2}$$

where $\mathbf{X} \in \mathbb{R}^{n \times p}$ is the data matrix with $\mathbf{x}^i$ as its $i$ th row, and $\mathbf{y}, \boldsymbol{\xi} \in \mathbb{R}^n$ are vectors of observations and noise respectively. Our goal is to estimate $\boldsymbol{\theta}^*$ from the data $\{(\mathbf{x}^i, y^i)\}_{i=1}^n$. We would like to characterize the estimation error, often in the form of the squared error $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2^2$ or the mean squared error $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2^2/p$. Here, $\widehat{\boldsymbol{\theta}}$ denotes the estimated parameters. This characterization is either of the form of high-probability upper bounds for the error or the exact value of the error in certain high-dimensional asymptotics.

### 2.1.1 Estimation in the Noiseless Setting

First consider the simpler case of Equation (2.2) where there is no noise in the observation model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\theta}^*. \tag{2.3}$$

When, the number of samples, $n$, is greater than or equal to the number of parameters, $p$, so long as the data points are linearly independent, this equation has a unique solution and we could estimate $\boldsymbol{\theta}^*$ exactly. However, when $n < p$, we have an underdetermined system

of linear equations which leads to a whole subspace of solutions. It is clear that without making further assumption about $\boldsymbol{\theta}^*$, consistent estimation is no longer possible. Therefore, we need to assume that the true vector of parameters have some structure. A very common assumption is to assume that this vector is sparse (or approximately sparse), i.e. the support set of this vector defined as

$$S = \{i | \boldsymbol{\theta}_i^* \neq 0\} \tag{2.4}$$

has cardinality $s = |S| \ll p$. This assumption is justified by the observation that many signals can be sparsified by looking at them in an appropriate basis or frame. As an example, it is now well-known that natural images have an approximately sparse representation in discrete Fourier transform basis or certain wavelet bases. Assuming that the true parameter vector is at most $s$-sparse, we could look for the sparsest solution of the equation $\mathbf{y} = \mathbf{X}\boldsymbol{\theta}$. Defining the $\ell_0$ norm as $\|\boldsymbol{\theta}\|_0 = \sum_{i=1}^{p} \mathbb{I}\{\boldsymbol{\theta}_i \neq 0\}$, where $\mathbb{I}$ denotes the indicator function of a set, we can write the problem of finding the sparsest solution to a set of linear equations as

$$\min_{\boldsymbol{\theta}} \|\boldsymbol{\theta}\|_0 \qquad \text{s.t. } \mathbf{y} = \mathbf{X}\boldsymbol{\theta}. \tag{2.5}$$

Due to non-smoothness and non-convexity of this problem, the direct approach to solve this problem consists of exhaustively searching over the column span of all the combination of columns of $\mathbf{X}$ with cardinality less than $s$ and see if the constraints can be satisfied in the subspace. Unfortunately, the number of such subspaces grow exponentially in $s$ which makes this approach computationally infeasible. As is the case in many other optimization problems, a natural strategy here would be to look at the convex relaxation of this problem and replace the $\ell_0$ norm with the closest convex norm among $\ell_q$ norms. We would then obtain the problem

$$\min_{\boldsymbol{\theta}} \|\boldsymbol{\theta}\|_1 \qquad \text{s.t. } \mathbf{y} = \mathbf{X}\boldsymbol{\theta}. \tag{2.6}$$

This problem, known as the *basis pursuit* program [Chen and Donoho, 1994], is a convex problem and can be solved using various convex optimization methods. The main question

then would be: when can the solution to (2.5) be recovered by solving (2.6)?

Through the years, many sufficient conditions have been proven to establish this equivalence such the *pairwise incoherence* condition for the columns of $\mathbf{X}$. Here, we present a generalization of this condition known as the restricted isometry property (RIP) [Candes and Tao, 2005].

**Definition 1** (Restricted isometry property)**.** A matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ satisfies the restricted isometry property of order $s$ with constant $\delta_s(\mathbf{X})$ if

$$\|\mathbf{X}_S^\top \mathbf{X}_S/n - \mathbf{I}_S\|_2 \leqslant \delta_s(\mathbf{X}) \quad \text{for all subsets } S \text{ of dimension at most } s, \tag{2.7}$$

where $\mathbf{X}_S$ is the $n \times s$ consisting of the columns of $\mathbf{X}$ with indices in $S$, and $\mathbf{I}_s$ is the $s \times s$ identity matrix.

In other words, the RIP condition guarantees that the norm of all the vectors, $\boldsymbol{\theta}$, with at most $s$ nonzero entries is almost preserved under the linear transformation corresponding to the matrix $\mathbf{X}$

$$\big(1 - \delta_s(\mathbf{X})\big)\|\boldsymbol{\theta}\|_2^2 \leqslant \|\mathbf{X}\boldsymbol{\theta}\|_2^2 \leqslant \big(1 + \delta_s(\mathbf{X})\big)\|\boldsymbol{\theta}\|_2^2. \tag{2.8}$$

The RIP condition guarantees exact recovery of the sparsest solution as stated below.

**Proposition 1** (Exact recovery in noiseless setting [Candes and Tao, 2005])**.** *If the matrix* $\mathbf{X}$ *satisfies RIP condition of order* $2s$ *with constant* $\delta_s(\mathbf{X}) \leqslant \frac{1}{3}$*, then the unique solution of the basis pursuit program in* (2.6) *satisfies* $\widehat{\boldsymbol{\theta}} = \boldsymbol{\theta}^*$ *for any* $\boldsymbol{\theta}^*$ *with* $\|\boldsymbol{\theta}^*\|_0 \leqslant s$*.*

### 2.1.2 Estimation in the Noisy Setting

Next, we focus on the linear estimation with noisy observations as in (2.2). In this case, a widely used estimator is the *Lasso* [Tibshirani, 1996]

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 + \lambda_n \|\boldsymbol{\theta}\|_1, \tag{2.9}$$

where the first term corresponds to fidelity of the model with the observations, the second term is a regularization that encourages sparseness of the estimated parameters, and $\lambda_n > 0$ is the regularization parameter that can be tuned to adjust the importance of each term. By Lagrangian duality, the Lasso program is equivalent to

$$\min_{\boldsymbol{\theta}} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 \qquad \text{s.t. } \|\boldsymbol{\theta}\|_1 \leqslant R, \tag{2.10}$$

$$\min_{\boldsymbol{\theta}} \|\boldsymbol{\theta}\|_1 \qquad \text{s.t. } \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 \leqslant b^2, \tag{2.11}$$

for appropriate values of $R$ and $b$. Therefore, here we solely focus on the Lasso program in (2.9).

In presence of noise, exact recovery would no longer be possible and hence we seek to bound the $\ell_2$ error of the Lasso estimator.

It is worth spending some time here to gain some intuition about what makes it so hard to bound the estimation error in such high-dimensional problems where the number samples could be much smaller than the number of parameters in the model. Consider the constrained optimization in (2.10) with $R = \|\boldsymbol{\theta}^*\|_1$ such that the true parameter vector becomes feasible. The objective function in (2.10) is an average of the errors for all the data points for a given $\boldsymbol{\theta}$. As the number $n$ increases, we expect the $\boldsymbol{\theta}^*$ to be near minimizer of this objective, i.e. if we define $\mathcal{L}(\boldsymbol{\theta}) = 1/2n \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2$, we expect to have $\mathcal{L}(\widehat{\boldsymbol{\theta}}) \approx \mathcal{L}(\boldsymbol{\theta}^*)$. What makes it possible to control the error $\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*$ based on the value of the objective function in such convex programs is the curvature of the objective function around its minimizer. The curvature of the function is captured by the eigenvalues of the hessian matrix evaluated at the minimizer. As the objective function here is a quadratic, the Hessian is $\mathbf{H} = \mathbf{X}^\top \mathbf{X}/n \in \mathbb{R}^{p \times p}$ which (assuming the data is centered) is the empirical covariance matrix of the data. This matrix has rank at most $n$ which means that the the Hessian is flat in many directions if $n \ll p$. Therefore, the closeness of $\mathcal{L}(\widehat{\boldsymbol{\theta}})$ to $\mathcal{L}(\boldsymbol{\theta}^*)$ alone is not enough to control the error. Figure 2.1 shows an example of flatness of the objective function in high dimensions. What comes to the rescue

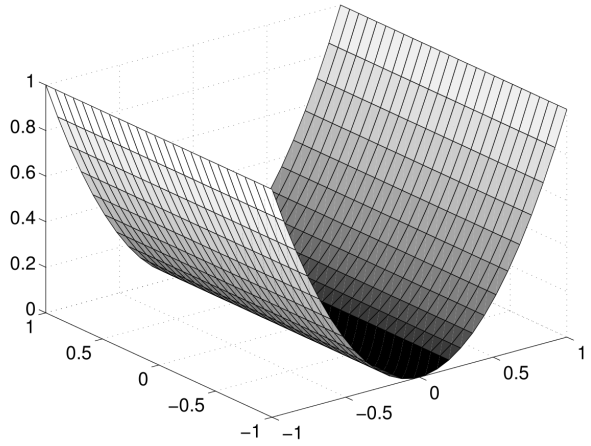



**Figure 2.1:** Curvature of the objective function in high-dimensional problems where the number of samples is fewer than the number of parameters [Wainwright, 2019]

then, is the regularization term which restricts the solution to lie in a set of the form

$$\mathbb{C}_\alpha(S) = \{\boldsymbol{\Delta} \in \mathbb{R}^p | \|\boldsymbol{\Delta}_{S^c}\|_1 \leqslant \alpha \|\boldsymbol{\Delta}_S\|_1\}, \tag{2.12}$$

where $S$ is a set of indices, $S^c$ is its complement, and $\boldsymbol{\Delta}_s$ corresponds to a vector formed by picking the entries of $\boldsymbol{\Delta}$ with indices in $S$ (and similarly for $\boldsymbol{\Delta}_{S^c}$) [Negahban et al., 2012]. If we can show that the objective function is curved over this set, we would be able to control the error based on how close the objective function is at $\boldsymbol{\theta}^*$ and $\widehat{\boldsymbol{\theta}}$. Towards that end, let us define the *restricted eigenvalue* (RE) condition.

**Definition 2** (Restricted eigenvalue condition)**.** A matrix $\mathbf{X}$ satisfies the restricted eigenvalue condition over $S$ with parameters $(\kappa, \alpha)$ if

$$\frac{1}{n} \|\mathbf{X}\boldsymbol{\Delta}\|_2^2 \leqslant \kappa \|\boldsymbol{\Delta}\|_2^2, \quad \text{for all } \boldsymbol{\Delta} \in \mathbb{C}_\alpha(S). \tag{2.13}$$

The following result gives us a bound on the square error for the Lasso estimator.

**Proposition 2** (Error of the Lasso estimator [Bickel et al., 2009])**.** *Let $\boldsymbol{\theta}^* \in \mathbb{R}^p$ be s-sparse, i.e. $\|\boldsymbol{\theta}^*\|_0 = s$. Assume that the design matrix $\mathbf{X}$ satisfies the restricted eigenvalue condition with parameters $(\kappa, 3)$. Then, any solution of the Lasso estimator in* (2.9) *with regularization*

*parameter* $\lambda_n \geqslant 2 \left\| \frac{\mathbf{X}^\top \boldsymbol{\xi}}{n} \right\|_\infty$ *satisfies the bound*

$$\left\| \widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^* \right\|_2 \leqslant \frac{3}{\kappa} \sqrt{s} \lambda_n. \tag{2.14}$$

Note that this result is a deterministic result and not a high-probability bound. The source of high-probability bounds come from showing that the restricted eigenvalue condition holds for different data ensembles. See Chapter 7 of [Wainwright, 2019] for a more detailed treatment of the subject of this section.

## 2.2 M-estimation with Decomposable Regularizers

In previous section, we focused on sparse linear problems with square error. Here, we summarize an extension of those results to general M-estimators. Please refer to Chapter 9 of [Wainwright, 2019] for more details. These results will be used extensively in Chapter 5. Let $\{\mathbf{z}^i\}_{i=1}^n$ be $n$ samples where $\mathbf{z}^i$ takes values in some space $\mathcal{Z}$. Each sample could be an input output pair $(\mathbf{x}^i, y^i)$ or simply of the form $\mathbf{x}^i$. The latter case happens for example when we have samples from a parametric family of distributions and aim to estimate the parameters. Let $\Omega$ be the parameter space. Given a loss function $\mathcal{L}_n : \Omega \times \mathcal{Z}^n \mapsto \mathbb{R}$, define the population loss and the target parameter as

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}[\mathcal{L}_n(\boldsymbol{\theta}, \{\mathbf{z}^i\}_1^n)], \qquad \boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}), \tag{2.15}$$

Where the expectation is with respect to the dataset $\{\mathbf{z}^i\}_{i=1}^n$. We seek to bound the estimation error of the regularized M-estimator

$$\widehat{\boldsymbol{\theta}} \in \arg\min_{\boldsymbol{\theta}} \mathcal{L}_n(\boldsymbol{\theta}, \{\mathbf{z}^i\}_1^n) + \lambda_n \mathcal{R}(\boldsymbol{\theta}), \tag{2.16}$$

where $\mathcal{R} : \Omega \mapsto \mathbb{R}$ is a regularizer. In practice, the loss function often has an additive form

$$\mathcal{L}_n(\boldsymbol{\theta}, \{\mathbf{z}^i\}_1^n) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}'(\boldsymbol{\theta}, \mathbf{z}^i). \tag{2.17}$$

Here, we consider regularizers that define a norm and are *decomposable* as defined below.

**Definition 3.** Given a pair of subspaces $\mathbb{M} \subseteq \bar{\mathbb{M}}$, a norm-based regularizer $\mathcal{R}$ is decomposable with respect to $(\mathbb{M}, \bar{\mathbb{M}})$ if

$$\mathcal{R}(\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2) = \mathcal{R}(\boldsymbol{\theta}_1) + \mathcal{R}(\boldsymbol{\theta}_2) \quad \text{for all } \boldsymbol{\theta}_1 \in \mathbb{M}, \boldsymbol{\theta}_2 \in \bar{\mathbb{M}}^{\perp}. \tag{2.18}$$

Many of the widely used regularizers in high-dimensional problems are decomposable. For example, if $\boldsymbol{\theta}^*$ is $s$-sparse with support in a set of indices $S$, then $\ell_1$ norm is decomposable with respect to the subspaces

$$\mathbb{M} = \{\boldsymbol{\theta} | \boldsymbol{\theta}_i = 0 \quad \text{for all} i \in S^c\}, \tag{2.19}$$

and $\bar{\mathbb{M}} = \mathbb{M}$. Similarly, one can show that group Lasso regularizers [Yuan and Lin, 2006, Kim et al., 2006], the overlapping group Lasso regularizer [Jacob et al., 2009], as well as many other regularizers are decomposable. See [Wainwright, 2019] for more details.

As was the case for the $\ell_1$ norm regularization discussed in the previous section, decomposability of a regularizer along with a suitable choice of regularization parameter $\lambda_n$ enforces the estimated parameters to lie in a very restricted set. Therefore, if we can show that the loss function is curved enough over this set, we will be able to obtain error bounds for the estimator using the bounds on the loss function.

If we define the dual norm of $\mathcal{R}$ via the variational formula

$$\mathcal{R}^*(\mathbf{u}^*) = \sup_{\mathcal{R}(\mathbf{u}) \leqslant 1} \langle \mathbf{u}, \mathbf{u}^* \rangle, \tag{2.20}$$

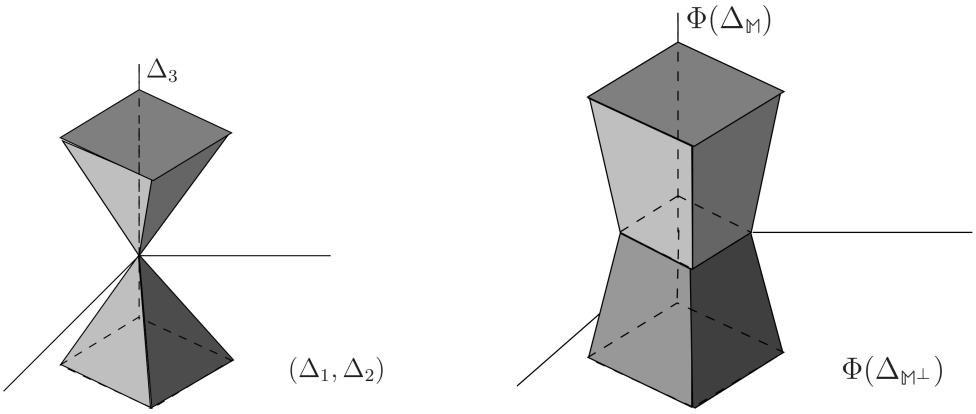



**Figure 2.2:** The set $\mathbb{C}_{\boldsymbol{\theta}*}$ for $\mathcal{R}(\boldsymbol{\Delta})$ for $\Delta \in \mathbb{R}^3$. Here, $\mathbb{M} = \bar{\mathbb{M}} = \{\boldsymbol{\Delta} | \boldsymbol{\Delta}_1 = \boldsymbol{\Delta}_2 = 0\}$. When $\boldsymbol{\theta}^* \in \mathbb{M}$ (left), the set is a cone. Otherwise, the set would not be a cone but would still be a star-shaped set (right) [Wainwright, 2019].

we have the following result.

**Proposition 3.** *If $\mathcal{L}_n$ is convex with respect to $\boldsymbol{\theta}$, and $\mathcal{R}$ is a norm which is decomposable over the pair of subspace $(\mathbb{M}, \bar{\mathbb{M}}^\perp)$, then if $\lambda_n \geqslant \mathcal{R}^*(\nabla \mathcal{L}_n(\boldsymbol{\theta}^*))$, the error $\boldsymbol{\Delta} = \widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}$ belongs to the set*

$$\mathbb{C}_{\boldsymbol{\theta}*} = \{\boldsymbol{\Delta} \in \Omega | \mathcal{R}(\boldsymbol{\Delta}_{\bar{\mathbb{M}}^\perp}) \leqslant 3\mathcal{R}(\boldsymbol{\Delta}_{\bar{\mathbb{M}}}) + 4\mathcal{R}(\theta^*_{\mathbb{M}^\perp})\}. \tag{2.21}$$

Here, for a subspace $V$, we are using the notation $\boldsymbol{\Delta}_V$ to denote the projection of $\boldsymbol{\Delta}$ onto $V$. Observe that if $\boldsymbol{\theta}^* \in \mathbb{M}$, then the second term on the right-hand side of (2.21) vanishes and the set would actually represent a convex cone. This set is illustrated in Figure 2.2 for the case of $\mathcal{R}(\boldsymbol{\Delta}) = \|\boldsymbol{\Delta}\|_1$.

Next, assuming that the loss is differentiable, if we consider the error of its first order Taylor expansion around $\boldsymbol{\theta}^*$

$$\mathcal{E}_n(\boldsymbol{\Delta}) := \mathcal{L}_n(\boldsymbol{\theta}^* + \boldsymbol{\Delta}) - \mathcal{L}_n(\boldsymbol{\theta}^*) - \langle \nabla \mathcal{L}_n(\boldsymbol{\theta}^*), \boldsymbol{\Delta} \rangle, \tag{2.22}$$

we can define a property analogous to restricted eigenvalue property.

**Definition 4** (Restricted strong convexity)**.** The cost function $\mathcal{L}_n$ satisfies restricted strong convexity (RSC) condition with curvature $\kappa > 0$ and tolerance $\tau_n^2$ with respect to a norm $\|\cdot\|$

and regularizer $\mathcal{R}(\cdot)$ if

$$\mathcal{E}_n(\boldsymbol{\Delta}) \geqslant \frac{\kappa}{2}\|\boldsymbol{\Delta}\|^2 - \tau_n^2 \mathcal{R}^2(\boldsymbol{\Delta}), \quad \text{for all } \boldsymbol{\Delta} \in \mathbb{C}_{\boldsymbol{\theta}^*}. \tag{2.23}$$

Finally, if we for a norm $\|\cdot\|$, a regularizer $\mathcal{R}(\cdot)$, and a given subspace $\mathbb{S}$ we define the subspace Lipschitz constant as

$$\Psi(\mathbb{S}) = \sup_{\mathbf{u} \in \mathbb{S}\setminus\{0\}} \frac{\mathcal{R}(\mathbf{u})}{\|\mathbf{u}\|}, \tag{2.24}$$

we have the following bound on the error of the regularized M-estimator in (2.16).

**Proposition 4** (Error of M-estimators [Negahban et al., 2012])**.** *Assume that the loss function is convex and satisfies the RSC condition with parameters $(\kappa, \tau_n^2)$. Further, assume that the regularizer decomposes over the pair of subspaces $(\mathbb{M}, \bar{\mathbb{M}})$. Then for any $\lambda_n \geqslant \mathcal{R}^*(\nabla \mathcal{L}_n(\boldsymbol{\theta}^*))$, if $\tau_n^2 \Psi(\bar{\mathbb{M}}) \leqslant \frac{\kappa}{64}$ we have*

$$\left\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right\|^2 \leqslant \underbrace{9\frac{\lambda_n^2}{\kappa^2}\Psi^2(\overline{\mathbb{M}})}_{\text{estimation error}} + \underbrace{\frac{8}{\kappa}[\lambda_n \mathcal{R}(\boldsymbol{\theta}^*_{\mathbb{M}^\perp}) + 16\tau_n^2 \mathcal{R}^2(\boldsymbol{\theta}^*_{\mathbb{M}^\perp})]}_{\text{approximation error}}. \tag{2.25}$$

Similar to Proposition 2, this is also a deterministic result. Probabilities come into play when one tries to prove that some value of $\lambda_n$ satisfies the bound $\lambda_n \geqslant \mathcal{R}^*(\nabla \mathcal{L}_n(\boldsymbol{\theta}^*))$, and to show that the loss function satisfies the RSC condition in (2.23).

There are two terms on the right hand side of (2.25). The first term corresponds to the statistical error incurred in the estimation, and the second error corresponds to approximation error of estimating the true parameter in a subspace in which it does not exactly belong. Indeed, if $\boldsymbol{\theta}^* \in \mathbb{M}$, we have $\boldsymbol{\theta}^*_{\mathbb{M}^\perp} = 0$ and the approximation error would be zero and we would only have the estimation error. As an example, for the $\ell_1$ regularizer and $s$-sparse $\boldsymbol{\theta}^*$, taking $\mathbb{M}$ to be the $s$-dimensional subspace that contain the $s$-sparse vectors with the same support as $\boldsymbol{\theta}^*$, we would have $\Psi(\mathbb{M}) = \sqrt{s}$, and we recover the result for the Lasso estimator in Proposition 2.

The result in Proposition 4 gives us a general framework to bound the error of many

high-dimensional estimators. The difficulty of proving such results often lies in proving that the cost function satisfies the RSC condition with high-probability for the class of problems in mind. We will use this framework to derive error bounds for estimating parameters of a class of discrete multi-dimensional autoregressive processes in Chapter 5.

## 2.3 Approximate Message Passing

In previous sections, we reviewed some standard results in high-dimensional statistics when applied to sparse recovery problems. These results were all finite sample results in the sense that so long as the number of samples were large enough -as determined by the size and other properties of the problem- we obtained error bounds that hold with high probability. Unfortunately, as discussed in Chapter 1, the analysis of problems in this regime is quite hard. As a result, in recent years, a new asymptotic regime has emerged in which both the number of samples and the number of parameters are going to infinity at a certain rate. This has allowed researchers to analyze new problems. Even though the analysis is done in an asymptotic regime, the results that are obtained in this regime often closely match what we observe in problems with finite size so long as the problems are large enough. The models that we utilize in practice seem to be large enough to agree quite well with the theoretical results derived in this regime. As opposed to high probability error bounds, in this asymptotic regime, we are able to derive formulae that allow us to exactly compute the error in different metrics.

In this section we briefly describe the approximate message passing (AMP) algorithm for linear inverse problems [Bayati and Montanari, 2011a]. This is one of the first works that has studied an algorithm in the proportional regime. Consider the problem of estimating $\mathbf{x}^0$ from linear observations

$$\mathbf{y} = \mathbf{X}\boldsymbol{\theta}^* + \boldsymbol{\xi}, \tag{2.26}$$

where $\mathbf{X} \in \mathbb{R}^{n \times p}$ is a known matrix and $\boldsymbol{\xi}$ is i.i.d. zero-mean Gaussian noise with variance $\sigma^2$.

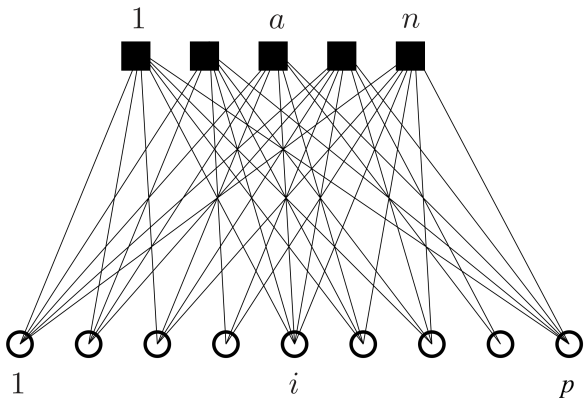



**Figure 2.3:** The factor graph corresponding to the linear model in (2.26) [Montanari et al., 2012].

Here for simplicity, we consider the case were $\boldsymbol{\theta}^*$ has a prior with i.i.d. distribution for its components with density (with respect to Lebesgue measure) $p_\theta(\cdot)$. The interested reader can refer to [Bayati and Montanari, 2011a] to see how these assumptions can be relaxed. With these assumptions, the posterior distribution of $\boldsymbol{\theta}^*$ given $\mathbf{y}$ factorizes as

$$p(\boldsymbol{\theta}^*|\mathbf{y}) \propto \prod_{i=1}^{p} p(\boldsymbol{\theta}_i^*) \prod_{j=1}^{n} \exp\left(-\frac{1}{2\sigma^2}(y^j - \mathbf{x}^{j\top}\boldsymbol{\theta}^*)^2\right). \tag{2.27}$$

This factorized structure can be represented by a graphical model shown in Figure 2.3. This is a bipartite graph where each variable is represented by a *variable node* $i \in [p]$ corresponding to the prior $p(\boldsymbol{\theta}_i^*)$ (represented by a circle), and each observation $y^j$ is represented by a *factor node* $j \in [n]$ (solid squares).

Similar to (2.9), many methods use regularized least squares to solve this inverse problem

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 + \lambda_n \mathcal{R}(\boldsymbol{\theta}). \tag{2.28}$$

We can also consider Bayesian estimators such as the Maximum a posteriori (MAP) estimator by taking $\mathcal{R}(\boldsymbol{\theta}) = -\log(p_{\boldsymbol{\theta}}(\boldsymbol{\theta}))$ with an appropriate choice of $\lambda_n$, or the minimum mean squared error (MMSE) estimator by considering the mean of the posterior distribution.

Approximate message passing is an iterative algorithm to solve this problem

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\eta}_t(\mathbf{X}^\mathsf{T}\mathbf{z}^t + \boldsymbol{\theta}^t) \tag{2.29}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{X}\boldsymbol{\theta}^t + \underbrace{\frac{1}{\delta}\mathbf{z}^{t-1}\langle \boldsymbol{\eta}'_{t-1}(\mathbf{X}^\mathsf{T}\mathbf{z}^{t-1} + \boldsymbol{\theta}^{t-1})\rangle}_{\text{Onsager correction}}, \tag{2.30}$$

where $\boldsymbol{\eta}_t(\cdot)$ is a denoiser that acts component-wise, $\boldsymbol{\eta}'_t(\cdot)$ is its component-wise derivative, $\langle\cdot\rangle$ is the empirical averaging operator defined for $\mathbf{a} \in \mathbb{R}^n$ as $\langle \mathbf{a} \rangle = 1/n \sum_{i=1}^n \mathbf{a}_i$, and $\delta = n/p$. This algorithm was first proposed by Donoho et al. in [Donoho et al., 2010b] and later rigorously analyzed in [Bayati and Montanari, 2011a]. Different choices of $\boldsymbol{\eta}$ allow us to implement different types of estimators. For example,

$$\boldsymbol{\eta}_t(\mathbf{r}) = \arg\min_{\boldsymbol{\theta}} \mathcal{R}(\boldsymbol{\theta}) + \frac{\gamma^t}{2}\|\boldsymbol{\theta} - \mathbf{r}\|_2^2 \tag{2.31}$$

with a fixed value of $\gamma^t$ (i.e. with no dependence on $t$) that can be derived from $\lambda_n$ would solve the problem in (2.28). Observe that if one removes the Onsager correction term from the AMP algorithm, the algorithm is equivalent to the proximal gradient descent method of optimization for solving (2.28). For example, for $\mathcal{R}(\cdot) = \|\cdot\|_1$, the proximal gradient descent algorithm is known as the *iterative soft thresholding algorithm* (ISTA) [Wright et al., 2009]. It is not hard to show that with an appropriate choice of $\gamma^t$, the fixed points of the AMP algorithm are the same as the fixed points of (2.28) and the Onsager correction term would not change the fixed points.

Similarly, taking $\mathcal{R}(\boldsymbol{\theta}) = -\log(p_{\boldsymbol{\theta}}(\boldsymbol{\theta}))$ with a suitable choice of $\gamma^t$ for each iteration would yield the MAP estimator. For the MMSE estimator, one can define a density

$$p_{\boldsymbol{\theta}}(\boldsymbol{\theta}|\mathbf{r}) = \frac{1}{Z}\left[\mathcal{R}(\boldsymbol{\theta}) + \frac{\gamma^t}{2}\|\boldsymbol{\theta} - \mathbf{r}\|_2^2\right], \tag{2.32}$$

where $Z$ is a normalizing factor and take $\boldsymbol{\eta}(\mathbf{r})$ to be the mean of this distribution. Again, a

specific choice of $\gamma^t$ should be used to obtain the MMSE estimate.

Therefore, AMP gives us a flexible algorithm that can be adapted to different types of estimators. The algorithm can be derived in many different ways, for example by approximating the messages in the loopy belief propagation algorithm on the factor graph in Figure 2.3 by Gaussian messages.

However, the key property of AMP algorithm is that when the sensing matrix $\mathbf{X}$ is large with i.i.d. sub-Gaussian entries, the behavior of the algorithm at each iteration can be exactly characterized via a *scalar* recursive equation called the *state evolution* (SE)

$$\tau_{t+1}^2 = \sigma^2 + \frac{1}{\delta}\mathbb{E}\left[(\eta_t(\theta_0 + \tau_t Z) - \theta_0)^2\right], \tag{2.33}$$

where $\theta_0 \sim p_{\theta_0}$ independent of $Z \sim \mathcal{N}(0,1)$. Here $p_{\theta_0}$ is the distribution to which the components of $\boldsymbol{\theta}^0$ are converging empirically. See Appendix B.2 for background on empirical convergence of sequences and some definitions we would use in this work. In particular, as $p, n \to \infty$ with fixed ratio $\delta := n/p$ we have

$$\begin{bmatrix} \boldsymbol{\theta}_0 \\ \widehat{\boldsymbol{\theta}}^t \end{bmatrix} \overset{PL(2)}{=} \begin{bmatrix} \theta_0 \\ \eta_{t-1}(\theta_0 + \tau_{t-1} Z) \end{bmatrix}, \tag{2.34}$$

where as in the state evolution we have $\theta_0 \sim p_{\theta_0}$ independent of $Z \sim \mathcal{N}(0,1)$. Define the joint empirical distribution of the components of $\boldsymbol{\theta}$ and $\widehat{\boldsymbol{\theta}}^t$ as

$$\mathbb{P}_n = \frac{1}{p}\sum_{i=1}^{p} \delta(\widehat{\boldsymbol{\theta}}_i^t, \boldsymbol{\theta}_i^*), \tag{2.35}$$

where $\delta$ denotes the Dirac measure. If we denote the distribution of the right hand side of (2.34) by $\mathbb{P}$, the PL(2) convergence is equivalent to weak convergence of $\mathbb{P}_n$ to $\mathbb{P}$ plus convergence of the second moments with respect to these distributions. It is also equivalent to convergence of $\mathbb{P}_n$ to $\mathbb{P}$ in Wasserstein-2 distance. See appendix B.2 for more details.

This convergence allows us to compute the estimation error in many separable metrics as an expectation. A loss function $\boldsymbol{\ell} : \Omega \times \Omega \mapsto \mathbb{R}$ is separable if we have

$$\boldsymbol{\ell}(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta}^*) = \frac{1}{p} \sum_{i=1}^{p} \ell(\widehat{\boldsymbol{\theta}}_i, \boldsymbol{\theta}_i^*) \tag{2.36}$$

for some function $\ell(\cdot, \cdot)$. Under the PL(2) convergence in (2.34), we can compute the error at any iteration of the AMP algorithm, for any such separable metric so long as $\ell(\cdot, \cdot)$ is bounded above by some quadratic function

$$\frac{1}{p} \sum_{i=1}^{p} \ell(\widehat{\boldsymbol{\theta}}_i^t, \boldsymbol{\theta}_i^*) = \mathbb{E}_{\theta_0, Z} \ell(\eta_{t-1}(\theta_0 + \tau_{t-1} Z), \theta_0) \qquad \text{almost surely,} \tag{2.37}$$

where $\theta_0 \sim p_{\theta_0}$ independent of $Z \sim \mathcal{N}(0, 1)$. For example, the mean squared error of the estimate at iteration $t$ defined as $\text{MSE} = 1/p \, \|\boldsymbol{\theta}^* - \boldsymbol{\theta}^t\|_2^2$ in the large system limit is

$$\text{MSE} = \mathbb{E}\left[(\eta_{t-1}(\theta_0 + \tau_{t-1} Z) - \theta_0)^2\right] \qquad \text{almost surely,} \tag{2.38}$$

where the expectation is over $\theta_0$ and $Z$.

It is worth spending some time here to compare the results of approximate message passing algorithms with those of Sections 2.1 and 2.2 obtained using methods from high-dimensional statistics. Unlike AMP results, those results are high-probability upper bounds that hold so long as the number of samples are large enough. Yet, the number of samples and parameters need not go to infinity and could remain finite. There are often unknown constants in those upper bounds that could be very large, therefore, such bounds are often interpreted as rates of convergence. On the other hand, the results that we obtain from AMP are exact, in the sense that we could exactly compute the estimation error, but they only hold in the asymptotic regime where the number of samples and number of parameters are both going to infinity at a fixed ratio. These results describe the behavior of models in large system limit by looking at them at macroscopic level. The behavior of the systems at this macroscopic level can be

described using distributions on low-dimensional random variables. This distributions are characterized by a set of equations called the state evolution which makes the interpretation of such results rather difficult.

The results of this section are used in Chapter 4 to derive the estimation error of ridge estimators in convolutional inverse problems.

## 2.4 Vector Approximate Message Passing

Vector approximate message passing (VAMP) is an algorithm that addresses some of the shortcomings of the AMP algorithm. As mentioned in the previous section, the AMP algorithm when applied to linear inverse problems that have large design matrices with i.i.d. sub-Gaussian entries, has the key property that at each iteration of the algorithm its behavior can be exactly characterized by a scalar state evolution equation. However, for generic design matrices, the AMP algorithm can fail completely and cause the iterates to diverge.

Vector AMP is an algorithm that uses similar ideas to AMP, but can succeed for a much larger class of design matrices. If the design matrix has the singular valued decomposition $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, the VAMP algorithm is guaranteed to converge so long as $\mathbf{\Sigma}$ has bounded singular values and $\mathbf{V}$ has a rotationally invariant distribution, i.e. for any orthogonal matrix $\mathbf{O}$, $\mathbf{VO}$ has the same distribution as $\mathbf{V}$. This happens when $\mathbf{V}$ is Haar distributed, i.e. uniform measure over the group of orthogonal matrices.

The VAMP iterations are shown in Algorithm 1. The algorithm requires two denoisers $\mathbf{g}^+$ and $\mathbf{g}^-$. Here, $\mathbf{g}^+$ plays the same role as the denoiser $\boldsymbol{\eta}$ in the AMP iterations in (2.29). $\mathbf{g}^-$ is a linear denoiser

$$\mathbf{g}^-(\mathbf{r}_k^+, \gamma_k^+) = \left(\sigma^2 \mathbf{X}^\top \mathbf{X} + \gamma_k^+ \mathbf{I}\right)^{-1} \left(\sigma^2 \mathbf{X}^\top \mathbf{y} + \gamma_k^+ \mathbf{r}_k^+\right), \tag{2.39}$$

which can be recognized as the minimum mean squared error denoiser under the prior $\boldsymbol{\theta} \sim \mathcal{N}(\mathbf{r}_k^+, \frac{1}{\gamma_k^+}\mathbf{I})$ and observation model $\mathbf{y} = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\xi}$ where $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$.

**Algorithm 1** Vector Approximate Message Passing (VAMP)

**Require:** Estimators $\mathbf{g}^+$ and $\mathbf{g}^-$ and number of iterations $N_{\text{it}}$

Set $\mathbf{r}_0^- = \mathbf{0} \in \mathbb{R}^p$ and initialize $\gamma_0^- > 0$.
**for** $k = 0, 1, \ldots, N_{\text{it}} - 1$ **do**
  // `Denoising`
  $\widehat{\boldsymbol{\theta}}_k^+ = \mathbf{g}^+(\mathbf{r}_k^-, \gamma_k^-)$
  $\lambda_k^+ = \gamma_k^- / \left\langle \frac{\partial \mathbf{g}^+}{\partial \mathbf{r}_k^-}(\mathbf{r}_k^-, \gamma_k^-) \right\rangle$,
  $\gamma_k^+ = \lambda_k^+ - \gamma_k^-$
  $\mathbf{r}_k^+ = (\lambda_k^+ \widehat{\mathbf{z}}_k^+ - \gamma_k^- \mathbf{r}_k^-)/\gamma_k^+$
  // `LMMSE estimation`
  $\widehat{\boldsymbol{\theta}}_k^- = \mathbf{g}^-(\mathbf{r}_k^+, \gamma_k^+)$
  $\lambda_k^- = \gamma_k^+ / \left\langle \frac{\partial \mathbf{g}^-}{\partial \mathbf{r}_k^+}(\mathbf{r}_k^+, \gamma_k^+) \right\rangle$,
  $\gamma_k^- = \lambda_k^- - \gamma_k^+$
  $\mathbf{r}_k^- = (\lambda_k^- \widehat{\mathbf{z}}_k^- - \gamma_k^+ \mathbf{r}_k^+)/\gamma_k^-$
**end for**

The VAMP iterations as presented in Algorithm 1 show an elegant symmetry. This is due to the fact that VAMP uses vector valued variable nodes along with variable splitting. More specifically, recall that the posterior in (2.27) factorized into two groups of terms on the right-hand side corresponding to the prior and likelihood models respectively. Let us rewrite this posterior in a more abstract form

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto p^+(\boldsymbol{\theta}) p^-(\mathbf{y}|\boldsymbol{\theta}). \tag{2.40}$$

Next, we can introduce a copy of $\boldsymbol{\theta}$ as

$$p(\boldsymbol{\theta}^-, \boldsymbol{\theta}^+|\mathbf{y}) \propto p^+(\boldsymbol{\theta}^-) \delta(\boldsymbol{\theta}^- - \boldsymbol{\theta}^+) p^-(\mathbf{y}|\boldsymbol{\theta}^+), \tag{2.41}$$

where $\delta(\cdot)$ represents the Dirac measure. The graphical model corresponding to this factorization of the posterior is shown in Figure 2.4. Unlike AMP which uses scalar valued variable nodes, VAMP uses vector valued variable nodes. The algorithm still uses Gaussian approximation of loopy belief propagation on this factor graph, but the factor graph here does not have any loops as opposed to the loopy bipartite graph of the AMP algorithm. A

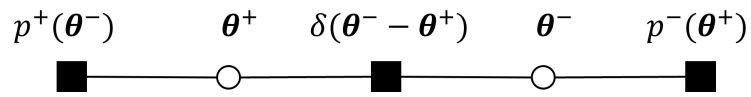



**Figure 2.4:** The graphical model corresponding to the factorization of the posterior in VAMP.

detailed derivation of the VAMP algorithm using this approach can be found in [Rangan et al., 2019a]. It should be noted that VAMP can also be derived using expectation consistent minimization of the Gibbs free energy. Please refer to [Fletcher et al., 2016] for details of this derivation under general prior and likelihood models.

As was the case for AMP, VAMP also enjoys the nice property that at each iteration of the algorithm, the estimates that it outputs can be exactly characterized by low-dimensional random variables that can be obtained through the state evolution equations. In particular, under certain conditions in the large system limit, as $n, p \to \infty$ with fixed ratio $\delta := n/p$, at each iteration $t$ of the algorithm we have the following convergence

$$\begin{bmatrix} \boldsymbol{\theta}^* \\ \widehat{\boldsymbol{\theta}}_t^+ \\ \mathbf{r}_t^- \end{bmatrix} \overset{PL(2)}{=} \begin{bmatrix} \theta^* \\ \widehat{\theta}_t^+ \\ R_t^- \end{bmatrix}, \tag{2.42}$$

where

$$R_t^- = \theta^* + P_t, \quad P_t \sim \mathcal{N}(0, \tau_t^-) \tag{2.43}$$

$$\widehat{\theta}_t^+ = g^+(R_t^-, \overline{\gamma_t^-}). \tag{2.44}$$

Here, $\overline{\gamma_t^-}$ and $\tau_t^-$ are parameters that are obtained using the state evolution equations. These equations are omitted here for brevity. The interested reader can refer to [Rangan et al., 2019b] to find the details of these equations as well as the technical conditions that are required for this convergence to hold.

Note that even though these equations seem hard to parse at first glance, they have a

rather simple interpretation: the estimated parameter at iteration $t$ can be thought of as the true parameter that has been perturbed by some Gaussian noise $P_t$ and gone through the denoiser $g^+(\cdot, \overline{\gamma}_t^-)$. As such, the VAMP algorithm can be interpreted as an algorithm that is iteratively denoising the a noisy version of the true parameter to get better and better estimates until the statistical error bound for the considered estimator is reached.

To summarize, the vector approximate message passing algorithm is an efficient algorithm to solve linear inverse problems. Similar to AMP, it has rigorous theoretical guarantees for the statistical error of the estimates that it generates at each iteration of the algorithm, and it converges for a much larger class of design matrices where AMP could easily fail. In Chapter 3, the VAMP algorithm is extended to estimation in multi-layer networks with matrix valued unknowns.

# Chapter 3

# Matrix Inference and Estimation in Multi-Layer Models

## 3.1 Introduction

Consider an $L$-layer stochastic neural network given by

$$\mathbf{Z}_\ell^0 = \mathbf{W}_\ell \mathbf{Z}_{\ell-1}^0 + \mathbf{B}_\ell + \mathbf{\Xi}_\ell^0, \qquad \ell = 1, 3, \ldots, L-1, \tag{3.1a}$$

$$\mathbf{Z}_\ell^0 = \boldsymbol{\phi}_\ell(\mathbf{Z}_{\ell-1}^0, \mathbf{\Xi}_\ell^0), \qquad \ell = 2, 4, \ldots, L, \tag{3.1b}$$

where, for $\ell = 0, 1, \ldots, L$, we have *true* activations $\mathbf{Z}_\ell^0 \in \mathbb{R}^{n_\ell \times d}$, weights $\mathbf{W}_\ell \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$, bias matrices $\mathbf{B}_\ell \in \mathbb{R}^{n_\ell \times d}$, and *true* noise realizations $\mathbf{\Xi}_\ell^0$. The activation functions $\boldsymbol{\phi}_\ell : \mathbb{R}^{n_{\ell-1} \times d} \to \mathbb{R}^{n_\ell \times d}$ are known non-linear functions acting row-wise on their inputs. See Fig. 3.1 (TOP). We use the superscript $^0$ in $\mathbf{Z}_\ell^0$ to indicate the true values of the variables, in contrast to estimated values denoted by $\widehat{\mathbf{Z}}_\ell$ discussed later. We model the true values $\mathbf{Z}_0^0$ as a realization of random $\mathbf{Z}_0$, where the rows $\mathbf{z}_{0,i:}^\mathsf{T}$ of $\mathbf{Z}_0$ are i.i.d. with distribution $p_0$: $p(\mathbf{Z}_0) = \prod_{i=1}^{n_0} p_0(\mathbf{z}_{0,i:})$. Similarly, we also assume that $\mathbf{\Xi}_\ell^0$ are realizations of random $\mathbf{\Xi}_\ell$ with i.i.d. rows $\boldsymbol{\xi}_{\ell,i:}^\mathsf{T}$. For odd $\ell$, the rows $\boldsymbol{\xi}_{\ell,i:}$ are zero-mean multivariate Gaussian with covariance matrix $\mathbf{N}_\ell^{-1} \in \mathbb{R}^{d \times d}$, whereas for even $\ell$, the rows $\boldsymbol{\xi}_{\ell,i:}$ can be arbitrarily distributed but i.i.d.

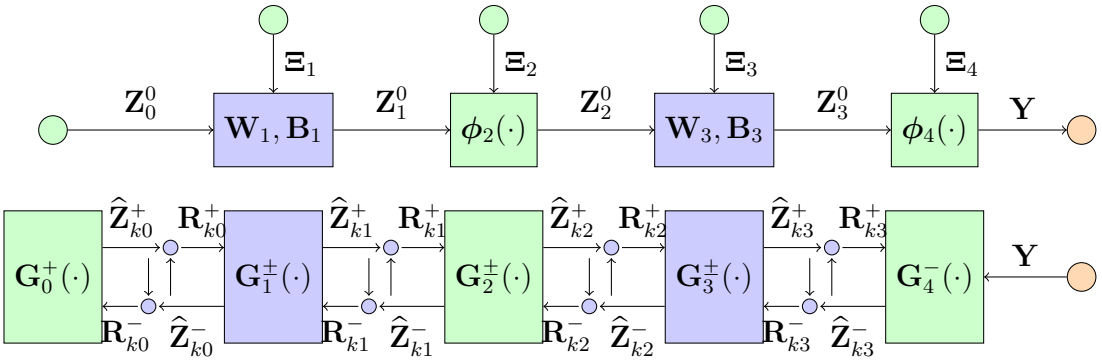



**Figure 3.1:** (TOP) The signal flow graph for *true* values of matrix variables $\{\mathbf{Z}_\ell^0\}_{\ell=0}^3$, given in eqn. (3.1) where $\mathbf{Z}_\ell^0 \in \mathbb{R}^{n_\ell \times d}$. (BOTTOM) Signal flow graph of the ML-MVAMP procedure in Algo. 2. The variables with superscript + and - are updated in the forward and backward pass respectively. ML-MVAMP (Algorithm 2) solves (3.2) by solving a sequence of simpler estimation problems over consecutive pairs $(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1})$.

Denoting by $\mathbf{Y} := \mathbf{Z}_L^0 \in \mathbb{R}^{n_L \times d}$ the output of the network, we consider the following matrix inference problem:

$$\text{Estimate } \mathbf{Z} := \{\mathbf{Z}_\ell\}_{\ell=0}^{L-1} \qquad \text{given } \mathbf{Y} := \mathbf{Z}_L^0 \text{ and } \{\mathbf{W}_{2k-1}, \mathbf{B}_{2k-1}, \boldsymbol{\phi}_{2k}\}_{k=1}^{L/2}. \tag{3.2}$$

A key feature of the problem we consider here is that the unknowns, $\mathbf{Z}_\ell$, are *matrix-valued* with $d$ columns with statistical dependencies between the columns. As we will see in Section 3.2, the matrix-valued case applies to several problems of broad interest such as matrix imputation, multi-task and mixed regression problems, sketched clustering. We also show that via this formulation we can analyze the learning in two layer neural networks under some architectural assumptions.

In many applications, the inference problem can be performed via minimization of an appropriate cost function. For example, suppose the network (3.1) has no noise $\boldsymbol{\Xi}_\ell$ for all layers except the final measurement layer, $\ell = L$. In this case, the $\mathbf{Z}_{L-1}^0 = \mathbf{g}(\mathbf{Z}_0^0)$ for some *deterministic function* $\mathbf{g}(\cdot)$ representing the action of the first $L-1$ layers. Inference can then

be conducted via a minimization of the form,

$$\widehat{\mathbf{Z}}_{L\text{–}1} := \mathbf{g}\left(\underset{\mathbf{Z}_0}{\arg\min}\ H_L(\mathbf{Y}, \mathbf{Z}_{L\text{–}1}) + H_0(\mathbf{Z}_0), \quad \text{subject to } \mathbf{Z}_{L\text{–}1} = \mathbf{g}(\mathbf{Z}_0)\right) \tag{3.3}$$

where the term $H_L(\mathbf{Y}, \mathbf{Z}_{L\text{–}1})$ penalizes the prediction error and $H_0(\mathbf{Z}_0)$ is an (optional) regularizer on the network input. For maximum a posteriori (MAP) estimation one takes, $H_L(\mathbf{Y}, \mathbf{Z}_{L\text{–}1}) = -\log p(\mathbf{Y}|\mathbf{Z}_{L\text{–}1})$, and $H_0(\mathbf{Z}_0) = -\log p(\mathbf{Z}_0)$, where the output probability $p(\mathbf{Y}|\mathbf{Z}_{L\text{–}1})$ is defined from the last layer of model (3.1b): $\mathbf{Y} = \mathbf{Z}_L = \boldsymbol{\phi}_L(\mathbf{Z}_{L\text{–}1}, \boldsymbol{\Xi}_L)$. The minimization (3.3) can then be solved using a gradient-based method. Encouraging results in image reconstruction have been demonstrated in [Yeh et al., 2016, Bora et al., 2017, Hand and Voroninski, 2017, Kabkab et al., 2018, Shah and Hegde, 2018, Tripathi et al., 2018, Mixon and Villar, 2018]. Markov-chain Monte Carlo (MCMC) algorithms and Langevin diffusion [Cheng et al., 2018, Welling and Teh, 2011] could also be employed for more complex inference tasks.

However, rigorous analysis of these methods is difficult due to the non-convex nature of the optimization problem. To address this issue, recent works [Manoel et al., 2017, Fletcher et al., 2018, Pandit et al., 2020] have extended Approximate Message Passing (AMP) methods to provide inference algorithms for the multi-layer networks. AMP was originally developed in [Donoho et al., 2009, Donoho et al., 2010a, Bayati and Montanari, 2011b, Kabashima, 2003] for compressed sensing. Similar to other AMP-type results, the performance of multi-layer AMP-based inference can be precisely characterized in certain high-dimensional random instances. In addition, the mean-squared error for inference of the algorithms match predictions for the Bayes-optimal inference predicted by various techniques from statistical physics [Reeves, 2017, Gabrié et al., 2018, Barbier et al., 2019]. Thus, AMP-based multi-layer inference provides a computationally tractable estimation framework with precise performance guarantees and testable conditions for optimality in certain high-dimensional random settings.

Prior multi-layer AMP works [He et al., 2017, Manoel et al., 2018, Fletcher et al., 2018, Pandit et al., 2020] have considered the case of vector-valued quantities with $d = 1$. The

main contribution of this work is to consider the *matrix-valued* case when $d > 1$. To handle the case when $d > 1$, we extend the Multi-Layer Vector Approximate Message Passing (ML-VAMP) algorithm of [Fletcher et al., 2018, Pandit et al., 2020] to the matrix case. The ML-VAMP method is based on VAMP method of [Rangan et al., 2019b], which is closely related to expectation propagation (EP) [Minka, 2001, Takeuchi, 2017], expectation-consistent approximate inference (EC) [Opper and Winther, 2005, Fletcher et al., 2016], S-AMP [Cakmak et al., 2014], and orthogonal AMP [Ma and Ping, 2017]. We will use "ML-Mat-VAMP" when referring to the matrix extension of ML-VAMP.

**Summary of Contributions** First, similar to the case of ML-VAMP, we analyze ML-Mat-VAMP in a large system limit, where $n_\ell \to \infty$ and $d$ is fixed, under rotationally invariant random weight matrices $\mathbf{W}_\ell$. In this large system limit, we prove that the mean-squared error (MSE) of the estimates of ML-Mat-VAMP can be exactly predicted by a deterministic set of equations called the *state evolution* (SE). The SE describes how the distribution of the true activations and pre-activations of the network as well as the estimated values generated by ML-Mat-VAMP evolve jointly from one iteration of the algorithm to the other. This extension of the SE equations to the matrix case is not trivial and requires considering correlation across multiple vectors. Indeed, in the case of ML-VAMP, the SE equations involve scalar quantities and $2 \times 2$ matrices. For ML-Mat-VAMP, the SE equations involve $d \times d$ and $2d \times 2d$ matrices.

Second, we show that the method can offer precise predictions in important estimation problems that are difficult to analyze via other means. The ML-VAMP was focused on deep reconstruction problems [Yeh et al., 2016, Bora et al., 2017]. The matrix version here can be applied to other classes of problems such as multi-task regression, matrix completion and learning the input layer of a neural network. Even though these networks are typically shallow (just $L = 2$ layers), there are no existing methods that can provide the same types of precise results. For example, in the case of learning the input layer of a neural network,

our results can exactly predict the test error as a function of the noise statistics, activations, number of training sample and other key modeling parameters.

**Notation:** Boldface uppercase letters $\mathbf{X}$ denote matrices. $\mathbf{X}_{n:}$ refers to the $n^{\text{th}}$ row of $\mathbf{X}$. Random vectors are row-vectors. For a function $f : \mathbb{R}^{1\times m} \to \mathbb{R}^{1\times k}$, its row-wise extension is represented by $\mathbf{f} : \mathbb{R}^{N\times m} \to \mathbb{R}^{N\times k}$, i.e., $[\mathbf{f}(\mathbf{X})]_{n:} = f(\mathbf{X}_{n:})$. We denote the Jacobian matrix of $f$ by $\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x}) \in \mathbb{R}^{m\times k}$, so that $[\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x})]_{ij} = \frac{\partial f_i}{\partial \boldsymbol{x}_j}(\boldsymbol{x})$. For its row-wise extension $\mathbf{f}$, we denote by $\langle \frac{\partial \mathbf{f}}{\partial \mathbf{X}}(\mathbf{X}) \rangle$ the average Jacobian, i.e., $\frac{1}{N}\sum_{n=1}^{N} \frac{\partial f}{\partial \mathbf{X}_{n:}}(\mathbf{X}_{n:}) \in \mathbb{R}^{m\times k}$

## 3.2 Example Applications

As we describe next, the matrix estimation problem 3.2 is of broad interest and several interesting applications can be formulated under this framework. We share a few examples below.

### 3.2.1 Multi-task and Mixed Regression Problems

A simple application of the matrix-valued multi-layer inference problem (3.2) is for *multi-task regression* [Obozinski et al., 2006]. Consider a generalized linear model of the form,

$$\mathbf{Y} = \boldsymbol{\phi}(\mathbf{XF}; \boldsymbol{\Xi}), \tag{3.4}$$

where $\mathbf{Y} \in \mathbb{R}^{N\times d}$ is a matrix of measured responses, $\mathbf{X} \in \mathbb{R}^{N\times p}$ is a known design matrix, $\mathbf{F} \in \mathbb{R}^{p\times d}$ are a set regression coefficients to be estimated, and $\boldsymbol{\Xi}$ is noise. The problem can be considered as $d$ separate regression problems – one for each column. However, in some applications, these design "tasks" are related in such a way that it benefits to *jointly* estimate the predictors. To do this, it is common to solve an optimization problem of the form

$$\arg\min_{\mathbf{F}} \left\{ \sum_{j=1}^{d}\sum_{i=1}^{N} L(y_{ij}, [\mathbf{XF}]_{ij}) + \lambda \sum_{k=1}^{p} \rho(\mathbf{F}_{k:}) \right\}, \tag{3.5}$$

where $L(\cdot)$ is a loss function, and $\rho(\cdot)$ is a regularizer that acts on the rows $\mathbf{F}_{k:}$ of $\mathbf{F}$ to couple the prediction coefficients across tasks. For example, the multi-task LASSO [Obozinski et al., 2006] uses loss $L(y,z) = (y-z)^2$ and regularization $\rho(\mathbf{F}_{k:}) = \|\mathbf{F}_{k:}\|_2$ to enforce row-sparsity in $\mathbf{F}$. In the compressive-sensing context, multi-task regression is known as the "multiple measurement vector" (MMV) problem, with applications in MEG reconstruction [Cotter et al., 2005], DoA estimation [Tzagkarakis et al., 2010], and parallel MRI [Liang et al., 2009]. An AMP approach to the MMV problem was developed in [Ziniel and Schniter, 2012]. The multi-task model (3.4) can be immediately written as a multi-layer network (3.1) by setting: $\mathbf{Z}_0 := \mathbf{F}, \mathbf{W}_0 := \mathbf{X}, \mathbf{Z}_1 := \mathbf{W}_0\mathbf{Z}_0 = \mathbf{XF}, \mathbf{Y} = \mathbf{Z}_2 := \boldsymbol{\phi}(\mathbf{Z}_1, \boldsymbol{\Xi})$. Also, by appropriately setting the prior $p(\mathbf{Z}_0)$, the multi-layer matrix MAP inference (3.3) will match the multi-task optimization (3.5).

In (3.5), the regularization couples the columns of $\mathbf{F}$ but the loss term couples its rows. In *mixed regression* problems, the loss couples the columns of $\mathbf{F}$. For example, consider designing predictors $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2]$ for *mixed linear regression* [Yi et al., 2014], i.e.,

$$y_i = q_i \mathbf{x}_i^\mathsf{T}\mathbf{f}_1 + (1-q_i)\mathbf{x}_i^\mathsf{T}\mathbf{f}_2 + v_i, \quad q_i \in \{0,1\}, \tag{3.6}$$

where $i = 1, \ldots, N$ and the $i$th response comes from one of two linear models, but which model is not known. This setting can be modeled by a different output mapping: As before, set $\mathbf{Z}_0 := \mathbf{F}$, $\mathbf{Z}_1 = \mathbf{XF}$ and let the noise in the output layer be $\boldsymbol{\Xi}_1 = [\mathbf{q}, \mathbf{v}]$ which includes the additive noise $v_i$ in (3.6) and the random selection variable $q_i$. Then, we can write (3.6) via an appropriate function, $\mathbf{y} = \boldsymbol{\phi}_1(\mathbf{Z}_1, \boldsymbol{\Xi}_1)$.

### 3.2.2 Sketched Clustering

A related problem arises in *sketched clustering* [Keriven et al., 2017a], where a massive dataset is nonlinearly compressed down to a short vector $\mathbf{y} \in \mathbb{R}^n$, from which cluster centroids $\mathbf{f}_k \in \mathbb{R}^p$, for $k = 1, \ldots, d$, are then extracted. This problem can be approached via the

optimization [Keriven et al., 2017b] $\min_{\boldsymbol{\alpha} \geqslant \mathbf{0}} \min_{\mathbf{F}} \sum_{i=1}^{n} \left| y_i - \sum_{j=1}^{d} \alpha_j e^{\sqrt{-1}\mathbf{x}_i^{\mathsf{T}} \mathbf{f}_j} \right|^2$ where $\mathbf{x}_i \in \mathbb{R}^p$ are known i.i.d. Gaussian vectors. An AMP approach to sketched clustering was developed in [Byrne et al., 2019]. For known $\boldsymbol{\alpha}$, the minimization corresponds to MAP estimation with the multi-layer matrix model with $\mathbf{Z}_0 = \mathbf{F}$, $\mathbf{W}_1 = \mathbf{X}$ $\mathbf{Z}_1 = \mathbf{XF}$ and using the output mapping, $\boldsymbol{\phi}_1(\mathbf{Z}_1, \boldsymbol{\Xi}) := \sum_{j=1}^{d} \alpha_j e^{\sqrt{-1}\mathbf{Z}_{1,:j}} + \boldsymbol{\Xi}$, where the exponential is applied elementwise and $\boldsymbol{\Xi}$ is i.i.d. Gaussian. The mapping $\phi_1$ operates row-wise on $\mathbf{Z}_1$ and $\boldsymbol{\Xi}$.

### 3.2.3 Learning the Input Layer of a Two-Layer Neural Network

The matrix inference problem (3.2) can also be applied to learning the input layer weights in a two-layer neural network (NN). Let $\mathbf{X} \in \mathbb{R}^{N \times N_{\text{in}}}$ and $\mathbf{Y} \in \mathbb{R}^{N \times N_{\text{out}}}$ be training data corresponding to $N$ data samples. Consider the two-layer NN model,

$$\mathbf{Y} = \sigma(\mathbf{X}\boldsymbol{F}_1)\boldsymbol{F}_2 + \boldsymbol{\Xi}, \tag{3.7}$$

with weight matrices $(\boldsymbol{F}_1, \boldsymbol{F}_2)$, componentwise activation function $\sigma(\cdot)$, and noise $\boldsymbol{\Xi}$. In (3.7), the bias terms are omitted for simplicity. We used the notation "$\boldsymbol{F}_\ell$" for the weights, instead of the standard notation "$\boldsymbol{W}_\ell$," to avoid confusion when (3.7) is mapped to the multi-layer inference network (3.2). Now, our critical assumption is that the weights in the second layer, $\boldsymbol{F}_2$, are known. The goal is to learn only the weights of the first layer, $\boldsymbol{F}_1 \in \mathbb{R}^{N_{\text{in}} \times N_{\text{hid}}}$, from a dataset of $N$ samples $(\mathbf{X}, \mathbf{Y})$.

If the activation is ReLU, i.e., $\sigma(\boldsymbol{H}) = \max\{\boldsymbol{H}, 0\}$ and $\mathbf{Y}$ has a single column (i.e. scalar output per sample), and $\boldsymbol{F}_2$ has all positive entries, we can, without loss of generality, treat the weights $\boldsymbol{F}_2$ as fixed, since they can always be absorbed into the weights $\boldsymbol{F}_1$. In this case, $\mathbf{y}$ and $\mathbf{F}_2$ are vectors and we can write the $i$th entry of $\mathbf{y}$ as

$$y_i = \sum_{j=1}^{d} F_{2j} \sigma([\mathbf{X}\boldsymbol{F}_1]_{ij}) + \xi_i = \sum_{j=1}^{d} \sigma([\mathbf{X}\boldsymbol{F}_1]_{ij} F_{2j}) + \xi_i \tag{3.8}$$

Thus, we can assume, without loss of generality, that $\mathbf{F}_2$ is all ones. The parameterization (3.8) is sometimes referred to as the *committee machine* [Tresp, 2000]. The committee machine has been recently studied by AMP methods [Aubin et al., 2018] and mean-field methods [Mei et al., 2018] as a way to understand the dynamics of learning.

To pose the two-layer learning problem as multi-layer inference, define $\boldsymbol{Z}_0 := \boldsymbol{F}_1, \quad \boldsymbol{W}_1 := \mathbf{X}, \quad \boldsymbol{Z}_1 := \mathbf{X}\boldsymbol{F}_1 \quad \boldsymbol{\Xi}_2 := \boldsymbol{\Xi}$, then $\boldsymbol{Y} = \boldsymbol{Z}_2$, where $\boldsymbol{Z}_2$ is the output of a 2-layer inference network of the form in (3.1):

$$\mathbf{Y} = \boldsymbol{Z}_2 = \boldsymbol{\phi}_2(\boldsymbol{Z}_1, \boldsymbol{\Xi}_2) := \sigma(\boldsymbol{Z}_1)\boldsymbol{F}_2 + \boldsymbol{\Xi}_2. \tag{3.9}$$

Note that $\boldsymbol{W}_1$ is known. Also, since we have assumed that $\boldsymbol{F}_2$ is known, the function $\boldsymbol{\phi}_2$ is known. Finally, the function $\boldsymbol{\phi}_2$ is row-wise separable on both inputs. Thus, the problem of learning the input weights $\boldsymbol{F}_1$ is equivalent to learning the input $\boldsymbol{Z}_0$ of the network (3.9).

### 3.2.4 Model-Based Matrix completion

Consider an observed matrix $\mathbf{Y} = \mathbf{Z}_L \in \mathbb{R}^{N_L \times d}$ with missing entries $\Omega^c \in [N_L] \times [d]$. The problem is to impute the missing entries of $\mathbf{Y}$. This is an important problem in several applications ranging from recommendation systems, genomics, bioinformatics and more broadly analysis of tabular data. There have been several approaches to solving this data imputation problem, right from 0 imputation and mean imputation to more sophisticated techniques based on generative models.

Consider a generative model based on a multi-layer perceptron as in (3.1) such that the output $\mathbf{Z}_{L-1}$ models the uncorrupted data matrix. Then the imputation problem can be posed as the solution of the MAP optimization problem:

$$\underset{\{\mathbf{Z}_\ell\}_{\ell=0}^{L}}{\text{minimize}} \, \|\mathbf{Y} - \mathbf{Z}_{L-1}\|_\Omega^2 - \log \mathbb{P}(\mathbf{Z}_{L-1}, \mathbf{Z}_{L-2}, \ldots, \mathbf{Z}_0) \tag{3.10}$$

where $\|\mathbf{Y} - \mathbf{Z}_{L-1}\|_{\Omega}^2 = \sum_{(i,j)\in\Omega}((\mathbf{Y})_{ij} - (\mathbf{Z}_{L-1})_{ij})^2$. One can also similarly construct Bayes estimators such as $\mathbb{E}[\mathbf{Z}_{L-1}|\mathbf{Z}_L]$.

Traditional approaches to matrix completion have looked at regularized convex minimization schemes just like (3.10) where $-\log \mathbb{P}(\mathbf{Z}_{L-1}) = \|\mathbf{Z}_{L-1}\|_*$, which is the nuclear norm, or some other structure inducing convex norms. While the term $-\log \mathbb{P}(\ldots)$ in (3.10) can be thought of as a more general regularization term, this formulation allows for more general application problems with heterogeneous variables.

For example, in imputation of tabular data, it is often the case that some columns correspond to continuous valued variables, whereas other variables are discrete valued modeling Yes/No answers or count data. In such scenarios the $-\log \mathbb{P}(\mathbf{Z}_{L-1}, \ldots)$ allows more flexibility towards modeling using GLMs and other exponential family distributions for every column separately. One simple instance of (3.10) would be a generative model $-\log \mathbb{P}(\mathbf{Z}_{L-1}, \ldots, \mathbf{Z}_0)$ which is trained on some fully observed data $\mathbf{Z}_{L-1}$ using unsupervised learning methods such as Variational Autoencoders (VAE) and Generative Adversarial Networks (GAN).

## 3.3 Multi-layer Matrix VAMP

### 3.3.1 MAP and MMSE inference

Observe that the equations (3.1) define a Markov chain over these signals and thus the posterior $p(\mathbf{Z}|\mathbf{Z}_L)$ factorizes as $p(\mathbf{Z}|\mathbf{Z}_L) \propto p(\mathbf{Z}_0) \prod_{\ell=1}^{L-1} p(\mathbf{Z}_\ell|\mathbf{Z}_{\ell-1})\, p(\mathbf{Y}|\mathbf{Z}_{L-1})$. where recall the notation $\mathbf{Z}$ from (3.2). The transition probabilities $p(\mathbf{Z}_\ell|\mathbf{Z}_{\ell-1})$ above are implicitly defined in equation (3.1) and depend on the statistics of noise terms $\mathbf{\Xi}_\ell$. We consider both maximum *a posteriori* (MAP) and minimum mean squared error (MMSE) estimation for this posterior:

$$\widehat{\mathbf{Z}}_{\mathsf{map}} = \arg\max_{\mathbf{Z}} \; p(\mathbf{Z}|\mathbf{Z}_L) \qquad \widehat{\mathbf{Z}}_{\mathsf{mmse}} = \mathbb{E}[\mathbf{Z}|\mathbf{Z}_L] = \int \mathbf{Z}\, p(\mathbf{Z}|\mathbf{Z}_L)\, \mathrm{d}\mathbf{Z} \tag{3.11}$$

### 3.3.2 Algorithm Details

The ML-Mat-VAMP for approximately computing the MAP and MMSE estimates is similar to the ML-VAMP method in [Fletcher et al., 2018, Pandit et al., 2019]. The specific iterations of ML-Mat-VAMP algorithm are shown in Algorithm 2. The algorithm produces estimates by a sequence of forward and backward pass updates denoted by superscripts $^+$ and $^-$ respectively. The estimates $\widehat{\mathbf{Z}}_\ell^\pm$ are constructed by solving sequential problems $\mathbf{Z} = \{\mathbf{Z}_\ell\}_{\ell=0}^{L-1}$ into a sequence of smaller problems each involving estimation of a single activation or preactivation $\mathbf{Z}_\ell$ via *estimation functions* $\{\mathbf{G}_\ell^\pm(\cdot)\}_{\ell=1}^{L-1}$ which are selected depending on whether one is interested in MAP or MMSE estimation.

To describe the estimation functions, we use the notation that, for a positive definite matrix $\mathbf{\Gamma}$, define the inner product $\langle \mathbf{A}, \mathbf{B} \rangle_\Gamma := \mathrm{Tr}(\mathbf{A}^\mathsf{T}\mathbf{B}\mathbf{\Gamma})$ and let $\|\mathbf{A}\|_\mathbf{\Gamma}$ denote the norm induced by this inner product. For $\ell = 1, \ldots, L-1$ define the approximate belief functions

$$b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1} | \mathbf{R}_\ell^-, \mathbf{R}_{\ell-1}^+, \mathbf{\Gamma}_\ell^-, \mathbf{\Gamma}_{\ell-1}^+) \propto p(\mathbf{Z}_\ell | \mathbf{Z}_{\ell-1}) e^{-\frac{1}{2}\left\|\mathbf{Z}_\ell - \mathbf{R}_\ell^-\right\|_{\mathbf{\Gamma}_\ell^-}^2 - \frac{1}{2}\left\|\mathbf{Z}_{\ell-1} - \mathbf{R}_{\ell-1}^+\right\|_{\mathbf{\Gamma}_{\ell-1}^+}^2}, \tag{3.12}$$

where $\mathbf{Z}_\ell, \mathbf{R}_\ell^\pm \in \mathbb{R}^{n_\ell \times d}$ and $\mathbf{\Gamma}_\ell^\pm \in \mathbb{R}^{d \times d}$ for all $\ell = 0, 1, \ldots L$. Define $b_0(\mathbf{Z}_0 | \mathbf{R}_0^-, \mathbf{\Gamma}_0^-)$ and $b_L(\mathbf{Z}_{L-1} | \mathbf{R}_{L-1}^+, \mathbf{\Gamma}_{L-1}^+)$ similarly. The MAP and MMSE estimation functions are then given by the MAP and MMSE estimates for these belief densities,

$$\mathbf{G}_{\ell,\mathsf{map}}^\pm = (\widehat{\mathbf{Z}}_\ell^+, \widehat{\mathbf{Z}}_{\ell-1}^-) = \text{argmax } b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}) \qquad \mathbf{G}_{\ell,\mathsf{mmse}}^\pm = (\widehat{\mathbf{Z}}_\ell^+, \widehat{\mathbf{Z}}_{\ell-1}^-) = \mathbb{E}[(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}) | b_\ell] \tag{3.13}$$

where the expectation is with respect to the normalized density proportional to $b_\ell$. Thus, the ML-Mat-VAMP algorithm reduces the joint estimation of the vectors $(\mathbf{Z}_0, \ldots, \mathbf{Z}_{L-1})$ to a sequence of simpler estimations on sub-problems with terms $(\mathbf{Z}_{\ell-1}, \mathbf{Z}_\ell)$. We refer to these subproblems as denoisers and denote their solutions by $\mathbf{G}_\ell^\pm$, so that $\widehat{\mathbf{Z}}_\ell^+ = \mathbf{G}_\ell^+$ and $\widehat{\mathbf{Z}}_{\ell-1}^- = \mathbf{G}_\ell^-$ corresponding to lines 9 and 20 of Algorithm 2. The denoisers $\mathbf{G}_0^+$ and $\mathbf{G}_L^-$, which provide updates to $\widehat{\mathbf{Z}}_0^+$ and $\widehat{\mathbf{Z}}_{L-1}^-$, are defined in a similar manner via $b_0$ and $b_L$ respectively.

**Algorithm 2** Multilayer Matrix VAMP (ML-Mat-VAMP)

**Require:** Estimators $\mathbf{G}_0^+$, $\mathbf{G}_L^-$, $\{\mathbf{G}_\ell^\pm\}_{\ell=1}^{L-1}$.

Set $\mathbf{R}_{0\ell}^- = \mathbf{0} \in \mathbb{R}^{n_\ell \times d}$ and initialize $\{\boldsymbol{\Gamma}_{0\ell}^-\}_{\ell=0}^{L-1} \in \mathbb{R}_{>0}^{d\times d}$.

**for** $k = 0, 1, \ldots, N_{\rm it} - 1$

// Forward Pass

$\widehat{\mathbf{Z}}_{k0}^+ = \mathbf{G}_0^+(\mathbf{R}_{k0}^-, \boldsymbol{\Gamma}_{k0}^-)$

$\boldsymbol{\Lambda}_{k0}^+ = \left\langle \frac{\partial \mathbf{G}_0^+}{\partial \mathbf{R}_0^-}(\mathbf{R}_{k0}^-, \boldsymbol{\Gamma}_{k0}^-) \right\rangle^{-1} \boldsymbol{\Gamma}_{k,0}^-$,

$\boldsymbol{\Gamma}_{k,0}^+ = \boldsymbol{\Lambda}_{k,0}^+ - \boldsymbol{\Gamma}_{k,0}^-$

$\mathbf{R}_{k,0}^+ = (\widehat{\mathbf{Z}}_{k,0}^+ \boldsymbol{\Lambda}_{k,0}^+ - \mathbf{R}_{k,0}^- \boldsymbol{\Gamma}_{k,0}^-)(\boldsymbol{\Gamma}_{k,0}^+)^{-1}$

**for** $\ell = 1, \ldots, L-1$ **do**

$\widehat{\mathbf{Z}}_{k\ell}^+ = \mathbf{G}_\ell^+(\mathbf{R}_{k\ell}^-, \mathbf{R}_{k,\ell-1}^+, \boldsymbol{\Gamma}_{k\ell}^-, \boldsymbol{\Gamma}_{k,\ell-1}^+)$

$\boldsymbol{\Lambda}_{k\ell}^+ = \left\langle \frac{\partial \mathbf{G}_\ell^+}{\partial \mathbf{R}_\ell^-}(\ldots) \right\rangle^{-1} \boldsymbol{\Gamma}_{k\ell}^-$,

$\boldsymbol{\Gamma}_{k\ell}^+ = \boldsymbol{\Lambda}_{k\ell}^+ - \boldsymbol{\Gamma}_{k\ell}^-$

$\mathbf{R}_{k\ell}^+ = (\widehat{\mathbf{Z}}_{k\ell}^+ \boldsymbol{\Lambda}_{k\ell}^+ - \mathbf{R}_{k\ell}^- \boldsymbol{\Gamma}_{k\ell}^-)(\boldsymbol{\Gamma}_{k\ell}^+)^{-1}$

**end for**

// Backward Pass

$\widehat{\mathbf{Z}}_{k,L-1}^- = \mathbf{G}_L^-(\mathbf{R}_{k,L-1}^+, \boldsymbol{\Gamma}_{k,L-1}^+)$

$\boldsymbol{\Lambda}_{k,L-1}^- = \left\langle \frac{\partial \mathbf{G}_L^-}{\partial \mathbf{R}_{L-1}^+}(\mathbf{R}_{k,L-1}^+, \boldsymbol{\Gamma}_{k,L-1}^+) \right\rangle^{-1} \boldsymbol{\Gamma}_{k,L-1}^+$,

$\boldsymbol{\Gamma}_{k,L-1}^- = \boldsymbol{\Lambda}_{k,L-1}^- - \boldsymbol{\Gamma}_{k,L-1}^+$

$\mathbf{R}_{k+1,L-1}^- = (\widehat{\mathbf{Z}}_{k,L-1}^- \boldsymbol{\Lambda}_{k,L-1}^- - \mathbf{R}_{k,0}^+ \boldsymbol{\Gamma}_{k,0}^+)(\boldsymbol{\Gamma}_{k,0}^-)^{-1}$

**for** $\ell = L-1, \ldots, 1$ **do**

$\widehat{\mathbf{Z}}_{k+1,\ell-1}^- = \mathbf{G}_\ell^-(\mathbf{R}_{k+1,\ell}^-, \mathbf{R}_{k,\ell-1}^+, \boldsymbol{\Gamma}_{k+1,\ell}^-, \boldsymbol{\Gamma}_{k,\ell-1}^+)$

$\boldsymbol{\Lambda}_{k+1,\ell-1}^- = \left\langle \frac{\partial \mathbf{G}_\ell^-}{\partial \mathbf{R}_{\ell-1}^+}(\cdots) \right\rangle^{-1} \boldsymbol{\Gamma}_{k,\ell-1}^+$,

$\boldsymbol{\Gamma}_{k+1,\ell}^- = \boldsymbol{\Lambda}_{k\ell}^- - \boldsymbol{\Gamma}_{k\ell}^+$

$\mathbf{R}_{k+1,\ell-1}^- = (\widehat{\mathbf{Z}}_{k\ell}^- \boldsymbol{\Lambda}_{k\ell}^- - \mathbf{R}_{k\ell}^+ \boldsymbol{\Gamma}_{k\ell}^+)(\boldsymbol{\Gamma}_{k+1,\ell}^-)^{-1}$

**end for**

**end for**

The estimation functions (3.13) can be easily computed for the multi-layer matrix network. An important characteristic of these estimators is that they can be computed using maps which are row-wise separable over their inputs and hence are easily parallelizable. To simplify notation, we denote the precision parameters for denoisers $\mathbf{G}_\ell^\pm$ in the $k^{\rm th}$ iteration by

$$\boldsymbol{\Theta}_{k\ell}^+ := (\boldsymbol{\Gamma}_{k\ell}^-, \boldsymbol{\Gamma}_{k,\ell-1}^+), \quad \boldsymbol{\Theta}_{k\ell}^- := (\boldsymbol{\Gamma}_{k+1,\ell}^-, \boldsymbol{\Gamma}_{k,\ell-1}^+), \qquad \boldsymbol{\Theta}_{k0}^+ := \boldsymbol{\Gamma}_{k0}^-, \qquad \boldsymbol{\Theta}_{kL}^- := \boldsymbol{\Gamma}_{k,L-1}^+. \tag{3.14}$$

**Non-linear layers:** For $\ell$ even, since the rows of $\boldsymbol{\Xi}_\ell$ are i.i.d., the belief density $b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}|\cdot)$ from (3.12) factors as a product across rows, $b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}) = \prod_n b_\ell([\mathbf{Z}_\ell]_{n:}, [\mathbf{Z}_{\ell-1}]_{n:})$. Thus, the MAP and MMSE estimates (3.13) can be performed over $d$-dimensional variables where $d$ is the number of entries in each row. There is no joint estimation across the different $n_\ell$ rows.

**Linear layers:** When $\ell$ is odd, the density $b_\ell(\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}|\cdot)$ in (3.12) is a Gaussian. Hence, the MAP and MMSE estimates agree and can be computed via least squares. Although for linear layers $[\mathbf{G}_\ell^+, \mathbf{G}_\ell^-](\mathbf{R}_\ell^-, \mathbf{R}_{\ell-1}^+, \boldsymbol{\Theta}_\ell)$ is not row-wise separable over $(\mathbf{R}_\ell^-, \mathbf{R}_{\ell-1})$, it can

be computed using another row-wise denoiser $[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-]$ via the SVD of the weight matrix $\mathbf{W}_\ell = \mathbf{V}_\ell \operatorname{diag}(\mathbf{S}_\ell)\mathbf{V}_{\ell-1}$ as follows. Note that the SVD is only needed to be performed once.:

$$
\begin{aligned}
&[\mathbf{G}_\ell^+, \mathbf{G}_\ell^-](\mathbf{R}_\ell, \mathbf{R}_{\ell-1}, \boldsymbol{\Theta}_\ell) = \underset{\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}}{\operatorname{argmax}} \|\mathbf{Z}_\ell - \mathbf{W}_\ell \mathbf{Z}_{\ell-1} - \mathbf{B}_\ell\|_{\mathbf{N}_\ell}^2 + \left\|\mathbf{Z}_\ell - \mathbf{R}_\ell^-\right\|_{\boldsymbol{\Gamma}_\ell^-}^2 + \left\|\mathbf{Z}_{\ell-1} - \mathbf{R}_{\ell-1}^+\right\|_{\boldsymbol{\Gamma}_{\ell-1}^+}^2 \\
&\overset{(a)}{=} \underset{\mathbf{Z}_\ell, \mathbf{Z}_{\ell-1}}{\operatorname{argmax}} \left\|\mathbf{V}_\ell^\mathsf{T}\mathbf{Z}_\ell - \operatorname{diag}(\mathbf{S}_\ell)\mathbf{V}_{\ell-1}\mathbf{Z}_{\ell-1} - \mathbf{V}_\ell^\mathsf{T}\mathbf{B}_\ell\right\|_{\mathbf{N}_\ell}^2 + \left\|\mathbf{V}_\ell^\mathsf{T}\mathbf{Z}_\ell - \mathbf{V}_\ell^\mathsf{T}\mathbf{R}_\ell^-\right\|_{\boldsymbol{\Gamma}_\ell^-}^2 + \left\|\mathbf{V}_{\ell-1}\mathbf{Z}_{\ell-1} - \mathbf{V}_{\ell-1}\mathbf{R}_{\ell-1}^+\right\|_{\boldsymbol{\Gamma}_{\ell-1}^+}^2 \\
&\overset{(b)}{=} [\mathbf{V}_\ell^\mathsf{T}\widetilde{\mathbf{G}}_\ell^+, \mathbf{V}_{\ell-1}\widetilde{\mathbf{G}}_\ell^-](\mathbf{V}_\ell^\mathsf{T}\mathbf{R}_\ell, \mathbf{V}_{\ell-1}\mathbf{R}_{\ell-1}, \boldsymbol{\Theta}_\ell)
\end{aligned}
$$

where (a) follows from the rotational invariance of the norm, and (b) follows from the definition of denoiser $[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-](\widetilde{\mathbf{R}}_\ell^-, \widetilde{\mathbf{R}}_{\ell-1}^+, \boldsymbol{\Theta}_\ell)$ given below

$$
[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-] := \underset{\widetilde{\mathbf{Z}}_\ell, \widetilde{\mathbf{Z}}_{\ell-1}}{\operatorname{argmax}} \left\|\widetilde{\mathbf{Z}}_\ell - \operatorname{diag}(\mathbf{S}_\ell)\widetilde{\mathbf{Z}}_{\ell-1} - \widetilde{\mathbf{B}}_\ell\right\|_{\mathbf{N}_\ell}^2 \left\|\widetilde{\mathbf{Z}}_\ell - \widetilde{\mathbf{R}}_\ell^-\right\|_{\boldsymbol{\Gamma}_\ell^-}^2 + \left\|\widetilde{\mathbf{Z}}_{\ell-1} - \widetilde{\mathbf{R}}_{\ell-1}^+\right\|_{\boldsymbol{\Gamma}_{\ell-1}^+}^2 \tag{3.15}
$$

Note that the optimization problem in (3.15), is decomposable accross the rows of variables $\widetilde{\mathbf{Z}}_\ell$ and $\widetilde{\mathbf{Z}}_{\ell-1}$, and hence $[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-]$ operates row-wise on its inputs.

**Fixed Points:** We note that the fixed points of the ML-Mat-VAMP algorithm can be shown to be KKT points of the variational formulations of (3.11), omitted here due to lack of space. This is a direct extention of results from Section 3 of [Pandit et al., 2020]. In particular, we can show that the ML-Mat-VAMP in the MAP inference case is a preconditioned *Peaceman-Rachford splitting* ADMM type algorithm [Themelis and Patrinos, 2020].

## 3.4 Analysis in the Large System Limit

We follow the analysis framework of the ML-VAMP work [Fletcher et al., 2018, Pandit et al., 2019], which is itself based on the original AMP analysis in [Bayati and Montanari, 2011b]. This analysis is based on considering the asymptotics of certain large random problem instances. We essentially show that under certain assumptions, as the dimension goes to infinity the behavior of the ML-Mat-VAMP algorithm can be characterized by a set

of equations that describe how the distribution of rows of hidden matrices evolve at each iteration of the algorithm for all the layers. Specifically, we consider a sequence of problems (3.1) indexed by $N$ such that for each problem the dimensions $n_\ell(N)$ are growing so that $\lim_{N\to\infty} \frac{n_\ell}{N} = \beta_\ell \in (0, \infty)$ are scalar constants. Note that $d$ is finite and does not grow with $N$.

**Distributions of weight matrices:** For $\ell = 1, 3, \ldots, L-1$, we assume that the weight matrices $\mathbf{W}_\ell$ are generated via the singular value decomposition, $\mathbf{W}_\ell = \mathbf{V}_\ell \operatorname{diag}(\mathbf{S}_\ell)\mathbf{V}_{\ell-1}$ where $\mathbf{V}_\ell \in \mathbb{R}^{n_\ell \times n_\ell}$ are Haar distributed over orthonormal matrices and $\mathbf{S}_\ell = (s_{\ell,1}, \ldots, s_{\ell,\min\{n_\ell, n_{\ell-1}\}})$. We will describe the distribution of the components $\mathbf{S}_\ell$ momentarily.

**Assumption on Denoisers:** We assume that the non-linear denoisers $\mathbf{G}_{2k}^{\pm}$ act row-wise on their inputs $(\mathbf{R}_{2k}^{-}, \mathbf{R}_{2k-1}^{+})$. Further these operators and their Jacobian matrices $\frac{\partial \mathbf{G}_{2k}^{+}}{\partial \mathbf{R}_{2k}^{-}}, \frac{\partial \mathbf{G}_{2k}^{-}}{\partial \mathbf{R}_{2k-1}^{+}}, \frac{\partial \mathbf{G}_{0}^{+}}{\partial \mathbf{R}_{0}^{-}}, \frac{\partial \mathbf{G}_{L}^{-}}{\partial \mathbf{R}_{L-1}^{+}}$ are *uniformly Lipschitz continuous*, the definition of which is provided in A.2.

**Assumption on initialization, true variables:** The distribution of the remaining variables is described by a weak limit: For a matrix sequence $\boldsymbol{X} := \boldsymbol{X}(N) \in \mathbb{R}^{N\times d}$, by the notation $\boldsymbol{X} \overset{2}{\Rightarrow} X$ we mean that there exists a random variable $X$ in $\mathbb{R}^d$ with $\mathbb{E}\|X\|^2 < \infty$ such that $\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} \psi(\boldsymbol{X}_{i:}) = \mathbb{E}\,\psi(X)$ almost surely, for any bounded continuous function $\psi : \mathbb{R}^d \to \mathbb{R}$, as well as for quadratic functions $\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x}$ for any $\boldsymbol{P} \in \mathbb{R}^{d\times d}_{\succeq 0}$. This is also referred to as Wasserstein-2 convergence [Montanari et al., 2019]. For e.g., this property is satisfied for a random $\boldsymbol{X}$ with i.i.d. rows with bounded second moments, but is more general, since it applies to deterministic matrix sequences as well. More details on this weak limit are given in A.2.

Let $\overline{\mathbf{B}}_\ell := \mathbf{V}_\ell^\intercal \mathbf{B}_\ell$, and $\overline{\mathbf{S}}_\ell \in \mathbb{R}^{n_\ell}$ be the zero-padded vector of singular values of $\mathbf{W}_\ell$, and let $\boldsymbol{\tau}_{0\ell}^{-} \in \mathbb{R}^{d\times d}_{\succ 0}$. Then we assume that the following empirical convergences hold.$(\boldsymbol{\Xi}_\ell, \mathbf{R}_{0\ell}^{-} - \mathbf{Z}_\ell^0) \overset{2}{\Rightarrow} (\Xi_\ell, Q_{0\ell}^{-})$ for even $\ell$ and $(\overline{\mathbf{S}}_\ell, \overline{\mathbf{B}}_\ell, \boldsymbol{\Xi}_\ell, \mathbf{V}_\ell^\top(\mathbf{R}_{0\ell}^{-} - \mathbf{Z}_\ell^0)) \overset{2}{\Rightarrow} (S_\ell, \overline{B}_\ell, \Xi_\ell, Q_{0\ell}^{-})$, for odd $\ell$. Here

$S_\ell \in \mathbb{R}_{\geqslant 0}$ is bounded, $\overline{B}_\ell \in \mathbb{R}^d$ is bounded, $\Xi_{2\ell-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_{2\ell-1}^{-1})$, and $Q_{0\ell}^- \sim \mathcal{N}(\mathbf{0}, \overline{\Gamma}_{0\ell}^-)$, for $\ell = 0, 1, \ldots, L-1$ are all pairwise independent random variables. Additionally, we assume that $\mathbf{Z}_0^0 \overset{2}{\Rightarrow} Z^0$ and that the sequence of initial matrices $\{\mathbf{\Gamma}_{0\ell}^-\}$ satisfies the following pointwise convergence

$$\mathbf{\Gamma}_{0\ell}^-(N) \to \overline{\mathbf{\Gamma}}_{0\ell}^-, \quad \ell = 0, 1, \ldots, L-1 \tag{3.16}$$

### 3.4.1 Main Result

The main result of this work concerns the empirical distribution of the rows $[\widehat{\mathbf{Z}}_\ell^\pm]_{n:}, [\mathbf{R}_\ell^\pm]_{n:}$ of the iterates of Algorithm 2. It characterizes the asymptotic behaviour of these empirical distributions in terms of $d$-dimensional random vectors which are either Gaussians or functions of Gaussians. Let $G_\ell^\pm$ denote maps $\mathbb{R}^{1\times d} \to \mathbb{R}^{1\times d}$, such that (3.13), i.e., $[\mathbf{G}_\ell^\pm(\mathbf{R}_\ell^-, \mathbf{R}_{\ell-1}^+, \mathbf{\Theta})]_{n:} = G_\ell^\pm([\mathbf{R}_\ell^-]_{n:}, [\mathbf{R}_{\ell-1}^+]_{n:}, \mathbf{\Theta})$. Having stated the requisite definitions and assumptions, we can now state our main result.

**Theorem 1.** *For a fixed iteration index $k \geqslant 0$, there exist deterministic matrices $\mathbf{K}_{k\ell}^+ \in \mathbb{R}_{>0}^{2d\times 2d}$, and $\boldsymbol{\tau}_{k\ell}^-, \overline{\mathbf{\Gamma}}_{k\ell}^+$ and $\overline{\mathbf{\Gamma}}_{k\ell}^-, \in \mathbb{R}_{>0}^{d\times d}$ such that for even $\ell$:*

$$\left(\mathbf{Z}_{\ell-1}^0, \mathbf{Z}_\ell^0, \widehat{\mathbf{Z}}_{k,\ell-1}^-, \widehat{\mathbf{Z}}_{k\ell}^+\right) \overset{2}{\Rightarrow} \left(\mathsf{A}, \widetilde{\mathsf{A}}, G_\ell^-(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \overline{\mathbf{\Gamma}}_{k\ell}^-, \overline{\mathbf{\Gamma}}_{k,\ell-1}^+), G_\ell^+(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \overline{\mathbf{\Gamma}}_{k\ell}^-, \overline{\mathbf{\Gamma}}_{k,\ell-1}^+)\right)$$

*where $(\mathsf{A}, \mathsf{B}) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k,\ell-1}^+)$, $\mathsf{C} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_{k\ell}^-)$, $\widetilde{\mathsf{A}} = \phi_\ell(\mathsf{A}, \Xi_\ell)$ and $(\mathsf{A}, \mathsf{B}), \mathsf{C}$ are independent. For $\ell = 0$, the same result holds where the $1^{\text{st}}$ and $3^{\text{rd}}$ terms are dropped, whereas for $\ell = L$, the $2^{\text{nd}}$ and $4^{\text{th}}$ terms are dropped. Similarly, for odd $\ell$:*

$$\left(\mathbf{V}_{\ell-1}^\mathsf{T}\mathbf{Z}_{\ell-1}^0,\ \mathbf{V}_{\ell-1}^\mathsf{T}\mathbf{Z}_\ell^0,\ \mathbf{V}_\ell\widehat{\mathbf{Z}}_{k,\ell-1}^-,\ \mathbf{V}_\ell\widehat{\mathbf{Z}}_{k\ell}^+\right) \overset{2}{\Rightarrow}$$
$$\left(\mathsf{A},\ \widetilde{\mathsf{A}},\ G_\ell^-(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \overline{\mathbf{\Gamma}}_{k\ell}^-, \overline{\mathbf{\Gamma}}_{k,\ell-1}^+),\ G_\ell^+(\mathsf{C}+\widetilde{\mathsf{A}}, \mathsf{B}+\mathsf{A}, \overline{\mathbf{\Gamma}}_{k\ell}^-, \overline{\mathbf{\Gamma}}_{k,\ell-1}^+)\right)$$

*where $(\mathsf{A}, \mathsf{B}) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k,\ell-1}^+)$, $\mathsf{C} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_{k\ell}^-)$, $\widetilde{\mathsf{A}} = S_\ell\,\mathsf{A} + \overline{B}_\ell + \Xi_\ell$ and $(\mathsf{A}, \mathsf{B}), \mathsf{C}$ are independent.*

*Furthermore for $\ell = 0, 1, \ldots L-1$, we have*

$$(\boldsymbol{\Gamma}_{k\ell}^{\pm}, \boldsymbol{\Lambda}_{k\ell}^{\pm}) \xrightarrow{a.s.} (\overline{\boldsymbol{\Gamma}}_{k\ell}^{\pm}, \overline{\boldsymbol{\Lambda}}_{k\ell}^{\pm}).$$

The parameters in the distribution, $\{\mathbf{K}_{k\ell}^{+}, \boldsymbol{\tau}_{k\ell}^{-}, \overline{\boldsymbol{\Gamma}}_{k\ell}^{\pm}, \overline{\boldsymbol{\Lambda}}_{k\ell}^{\pm}\}$ are deterministic and can be computed via a set of recursive equations called the *state evolution* or SE. The SE equations are provided in A.1 The result is similar to those for ML-VAMP in [Fletcher et al., 2018,Pandit et al., 2020] except that the SE equations for ML-Mat-VAMP involve $d \times d$ and $2d \times 2d$ matrix terms; the ML-VAMP SE only requires scalar and $2 \times 2$ matrix terms. The result holds for both MAP inference and MMSE inference, the only difference is implicit, i.e., the choice of denoiser $\mathbf{G}_{\ell}(\cdot)$ from eqn. (3.13).

The importance of Theorem 1 is that the rows of the iterates of the ML-Mat-VAMP Algorithm ($\widehat{\mathbf{Z}}_{k,\ell-1}^{-}, \widehat{\mathbf{Z}}_{k\ell}^{+}$ in Algorithm 2) and the rows of the corresponding true values, $\mathbf{Z}_{\ell-1}^{0}, \mathbf{Z}_{\ell}^{0}$, have a simple, asymptotic random vector description of a typical row. We will call this the "row-wise" model. According to this model, for even $\ell$, the rows of $\mathbf{Z}_{\ell-1}^{0}$ converge to a Gaussian $\mathsf{A} \in \mathbb{R}^{d}$ and the rows of $\mathbf{Z}_{\ell}^{0}$ converge to the output of the Gaussian through the row-wise function $\phi_{\ell}$, $\widetilde{\mathsf{A}} = \phi_{\ell}(\mathsf{A}, \Xi_{\ell})$. Then the rows of the estimates $\widehat{\mathbf{Z}}_{k,\ell-1}^{-}, \widehat{\mathbf{Z}}_{k\ell}^{+}$ asymptotically approach the outputs of row-wise estimation function $G^{-}(\cdot)$ and $G^{+}(\cdot)$ supplied by $\mathsf{A}$ and $\widetilde{\mathsf{A}}$ corrupted with Gaussian noise. A similar convergence holds for odd $\ell$.

This "row-wise" model enables exact an analysis of the performance of the estimates at each iteration. For example, to compute a weighted mean squared error (MSE) metric at iteration $k$, the convergence shows that,

$$\tfrac{1}{n_{\ell}} \left\| \widehat{\mathbf{Z}}_{k\ell}^{+} - \mathbf{Z}_{\ell}^{0} \right\|_{\mathbf{H}}^{2} \xrightarrow{a.s.} \mathbb{E} \|\mathbf{G}_{\ell}^{+}(\mathsf{C} + \widetilde{\mathsf{A}}, \mathsf{B} + \mathsf{A}, \boldsymbol{\Theta}_{k\ell}) - \widetilde{\mathsf{A}}\|_{\mathbf{H}}^{2},$$

for even $\ell$ and any positive semi-definite matrix $\mathbf{H} \in \mathbb{R}^{d \times d}$. The norm on the left-hand above acts row-wise, $\|\mathbf{Z}\|_{\mathbf{H}}^{2} := \sum_{i} \|\mathbf{Z}_{i:}\|_{\mathbf{H}}^{2}$. Hence, this asymptotic MSE can be evaluated via expectations of $d$-dimensional variables from the SE. Similarly, one can obtain exact answers

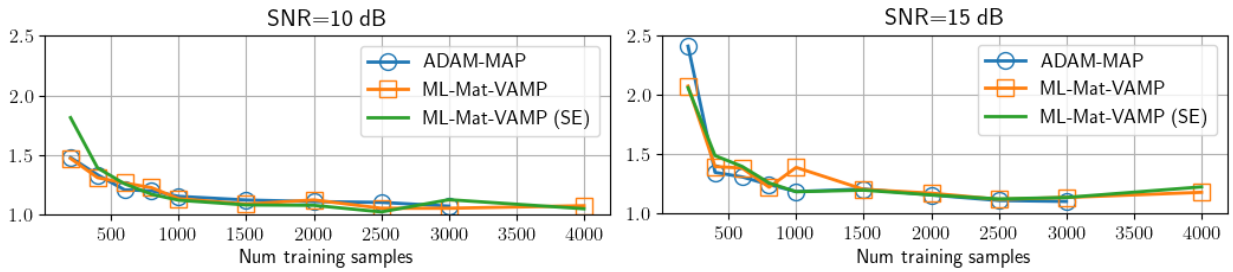



**Figure 3.2:** Test error in learning the first layer of a 2 layer neural network using ADAM-based gradient descent, ML-Mat-VAMP and its state evolution prediction.

for any other row-wise performance metric of $\{(\widehat{\mathbf{Z}}^{\pm}_{k\ell}, \mathbf{Z}^{0}_{\ell})\}_{\ell}$ for any $k$.

## 3.5 Numerical Experiments

We consider the problem of learning the input layer of a two layer neural network as described in Section 3.2.3. We learn the weights $\boldsymbol{F}_1$ of the first layer of a two-layer network by solving problem (3.9). The large system limit analysis in this case corresponds to the input size $n_{\text{in}}$ and number of samples $N$ going to infinity with the number of hidden units being fixed. Our experiment take $d = 4$ hidden units, $N_{\text{in}} = 100$ input units, $N_{\text{out}} = 1$ output unit, sigmoid activations and variable number of samples $N$. The weight vectors $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$ are generated as i.i.d. Gaussians with zero mean and unit variance. The input $\mathbf{X}$ is also i.i.d. Gaussians with variance $1/N_{\text{in}}$ so that the average pre-activation has unit variance. Output noise is added at two levels of 10 and 15 dB relative to the mean. We generate 1000 test samples and a variable number of training samples that ranges from 200 to 4000. For each trial and number of training samples, we compare three methods: (i) MAP estimation where the MAP loss function is minimized by the ADAM optimizer [Kingma and Ba, 2014] in the Keras package of Tensorflow; (ii) Algorithm 2 run for 20 iterations and (iii) the state evolution prediction. The ADAM algorithm is run for 100 epochs with a learning rate = 0.01. The expectations in the SE are estimated via Monte-Carlo sampling (hence there is some variation).

Given an estimate $\widehat{\mathbf{F}}_1$ and true value $\mathbf{F}^0_1$, we can compute the test error as follows: Given

a new sample $\mathbf{x}$, the true and predicted pre-activations will be $\mathbf{z}_1 = (\mathbf{F}_1^0)^\mathsf{T}\mathbf{x}$ and $\widehat{\mathbf{z}}_1 = \widehat{\mathbf{F}}_1^\mathsf{T}\mathbf{x}$. Thus, if the new sample $\mathbf{x} \sim \mathcal{N}(0, \frac{1}{N_{\text{in}}}\mathbf{I})$, the true and predicted pre-activations, $(\mathbf{z}_1, \widehat{\mathbf{z}}_1)$, will be jointly Gaussian with covariance equal to the empirical $2d \times 2d$ covariance matrix of the rows of $\mathbf{F}_1^0$ and $\widehat{\mathbf{F}}_1$:

$$\mathbf{K} := \tfrac{1}{N_{\text{in}}} \textstyle\sum_{k=1}^{N_{\text{in}}} \mathbf{u}_k^\mathsf{T}\mathbf{u}_k, \quad \mathbf{u}_k = \begin{bmatrix} \mathbf{F}_{1,k:} & \widehat{\mathbf{F}}_{1,k:} \end{bmatrix} \tag{3.17}$$

From this covariance matrix, we can estimate the test error, $\mathbb{E}|y - \widehat{y}|^2 = \mathbb{E}|\mathbf{F}_2^\mathsf{T}(\sigma(\mathbf{z}_1) - \sigma(\widehat{\mathbf{z}}_1)|^2$, where the expectation is taken over the Gaussian $(\mathbf{z}_1, \widehat{\mathbf{z}}_1)$ with covariance $\mathbf{K}$. Also, since (3.17) is a row-wise operation, it can be predicted from the ML-Mat-VAMP SE. Thus, the SE can also predict the asymptotic test error. The normalized test error for ADAM-MAP, ML-Mat-VAMP and the ML-Mat-VAMP SE are plotted in Fig. 3.2. The normalized test error is defined as the ratio of the MSE on the test samples to the optimal MSE. Hence, a normalized MSE of one is the minimum value.

Note that since ADAM and ML-Mat-VAMP are solving the same optimization problem, they perform similarly as expected. The main message of this work is not to develop an algorithm that outperforms ADAM, but rather an algorithm that has theoretical guarantees. The key property of ML-Mat-VAMP is that its asymptotic behavior at all the iterations can be exactly predicted by the state evolution equations. In this example, Fig. 3.2 shows that the normalized test MSE predicted via state evolution (plotted in green) matches the normalized MSE of ML-Mat-VAMP estimates (plotted in orange).

## 3.6 Conclusions

We have developed a general framework for analyzing inference in multi-layer networks with matrix valued quantities in certain high-dimensional random settings. For learning the input layer of a two layer network, the methods enables precise predictions of the expected test error as a function of the parameter statistics, numbers of samples and noise level. This analysis can be valuable in understanding key properties such as generalization error, for

example using ML-VAMP, Emami et al. [Emami et al., 2020] characterizes the generalization error of GLMs under a variety of feature distributions and train-test mismatch. Future work will look to extend these to more complex networks.

# Chapter 4

# Asymptotics of Ridge Regression in Convolutional Models

## 4.1 Introduction

Increasingly powerful hardware along with deep learning libraries that efficiently use these computational resources have allowed us to train ever larger neural networks. Modern neural networks are so over-parameterized that they can perfectly fit random noise [Zhang et al., 2016, Li et al., 2020]. With enough over-parameterization, they can also achieve zero loss over training data with their parameters moving only slightly away from the initialization [Allen-Zhu et al., 2018, Soltanolkotabi et al., 2018, Du et al., 2018a, Du et al., 2018b, Li and Liang, 2018, Zou et al., 2020]. Yet, these models generalize well on test data and are widely used in practice [Zhang et al., 2016]. In fact, some recent work suggest that it is best practice to use as large a model as possible for the tasks in hand [Huang et al., 2018]. This seems contrary to our classical understanding of generalization where increasing the complexity of the model space to the extent that the training data can be easily interpolated indicates poor generalization. Most statistical approaches explain generalization by controlling some notion of capacity of the hypothesis space, such as VC dimension, Rademacher complexity, or

metric entropy [Anthony and Bartlett, 2009]. Such approaches that do not incorporate the implicit regularization effect of the optimization algorithm fail to explain generalization of over-parameterized deep networks [Kalimeris et al., 2019, Oymak and Soltanolkotabi, 2019].

The so-called *double descent* curve where the test risk starts decreasing again by over-parameterizing neural networks beyond the interpolation threshold is widely known by now [Belkin et al., 2019a, Nakkiran et al., 2019]. Interestingly, such phenomenon is not unique to neural networks and have been observed even in linear models [Dobriban et al., 2018]. Recently, another line of work has also connected neural networks to linear models. In [Jacot et al., 2018], the authors show that infinitely wide neural networks trained by gradient descent behave like their linearization with respect to the parameters around their initialization. The problem of training such wide neural networks with square loss then turns into a kernel regression problem in a RKHS associated to a fixed kernel called the *neural tangent kernel* (NTK). The NTK results were later extended to many other architectures such as convolutional networks and recurrent neural networks [Li et al., 2019, Alemohammad et al., 2020, Yang, 2020]. Trying to understand the generalization in deep networks and explaining such phenomenon as the double descent curve has attracted a lot of attention to theoretical properties of kernel methods as well as simple machine learning models. Such models, despite their simplicity, can help us gain a better understanding of machine learning models and algorithms that might be hard to achieve just by looking at deep neural networks due their complex nature.

Double descent has been shown in linear models [Dobriban et al., 2018, Belkin et al., 2019b, Hastie et al., 2019], logistic regression [Deng et al., 2019], support vector machines [Montanari et al., 2019], generalized linear models [Emami et al., 2020], kernel regression [Liang et al., 2020], random features regression [Mei and Montanari, 2019, Hastie et al., 2019], and random Fourier feature regression [Liao et al., 2020] among others. Most of these works consider the problem in a doubly asymptotic regime where both the number of parameters and the number of observations go to infinity at a fixed ratio. This is in contrast to classical

statistics where either the number of parameters is assumed to be fixed and the samples go to infinity or vice versa. In practice, the number of parameters and number of samples are usually comparable and therefore the doubly asymptotic regime provides more value about the performance of different models and algorithms. In this work we study the performance of ridge estimators in convolutional linear inverse problems in this asymptotic regime. Unlike ordinary linear inverse problems, theoretical properties of the estimation problem in convolutional models has not been studied, despite their wide use in practice, e.g. in solving inverse problems with deep generative priors [Ulyanov et al., 2018].

Beyond machine learning, inverse problems involving convolutional measurement models are often called deconvolution and are encountered in many different fields. In astronomy, deconvolution is used for example to deblur, sharpen, and correct for optical aberrations in imaging [Starck et al., 2002]. In seismology, deconvolution is used to separate seismic traces into a source wavelet and an impulse response that corresponds to the layered structure of the earth [Treitel and Lines, 1982, Mueller, 1985]. In imaging, it is used to correct for blurs caused by the point spread function, sharpen out of focus areas in 3D microscopy [McNally et al., 1999], and to separate neuronal spikes for calcium traces in calcium imaging [Friedrich et al., 2017] among others. In practical applications, the convolution kernel might not be known and should either be estimated from the physics of the problem or jointly with the unknown signal using the data.

**Summary of Contributions.** We analyze the performance of ridge estimator for convolutional models in the proportional asymptotics regime. Our main result (Theorem 2) characterizes the limiting joint distribution of the true signal and its ridge estimate in terms of the spectral properties of the data. As a result of this theorem, we can provide an exact formula to compute the mean squared error of ridge estimator in the form of a scalar integral (Corollary 1). Our assumptions on the data are fairly general and include many random processes as an example as opposed to i.i.d. features only. Even though our theoretical

results hold only in a certain high dimensional limit, our experiments show that its prediction matches the observed error even for problems of moderate size. We show that our result can predict the double descent curve of the estimation error as we change the ratio of number of measurements to unknowns.

**Prior Work.** Asymptotic error of ridge regression for ordinary linear inverse problems (as opposed to convolutional linear inverse problem considered in this work) is studied in [Dicker et al., 2016] for isotropic features. Asymptotics of ridge regression for correlated features is studied in [Dobriban et al., 2018]. Error of ridgeless (minimum $\ell_2$-norm interpolant) regression for data generated from a linear or nonlinear model is obtained in [Hastie et al., 2019]. These works use results from random matrix theory to derive closed form formulae for the estimation or generalization error. For features with general i.i.d. prior other than Gaussian distribution, approximate message passing (AMP) [Donoho et al., 2010b,Bayati and Montanari, 2011a] or vector approximate message passing (VAMP) [Rangan et al., 2019a] can be used to obtain asymptotics of different types of error. Instead of a closed form formula, these works show that the asymptotic error can be obtained via a recursive equation that is called the *state evolution*. In [Deng et al., 2019], the authors use convex Gaussian min-max theorem to characterize the performance of maximum likelihood as well as SVM classifiers with i.i.d. Gaussian covariates. In [Emami et al., 2020], the problem of learning generalized linear models is reduced to an inference problem in deep networks and the results of [Pandit et al., 2020] are used to obtain the generalization error.

**Notation.** We use uppercase boldface letters for matrices and tensors, and lowercase boldface letters for vectors. For a matrix $\mathbf{A}$, its $i$th row and column is denoted by $\mathbf{A}_{i*}$ and $\mathbf{A}_{*i}$ respectively. A similar notation is used to show slices of tensors. The submatrix formed by columns $i$ through $j-1$ of $\mathbf{A}$ is shown by $\mathbf{A}_{*,i:j}$. Standard inner product for vectors, matrices, and tensors is represented by $\langle \cdot, \cdot \rangle$. $\mathcal{N}(0,1)$ and $\mathcal{CN}(0,1)$ denote standard normal and complex normal distributions respectively. Finally, $[n] = \{1, \dots, n\}$.

## 4.2 Problem Formulation

We consider the inverse problem of estimating $\mathbf{X}$ from $\mathbf{Y}$ in the convolutional model

$$\mathbf{Y} = \mathbf{K} * \mathbf{X} + \mathbf{\Xi}, \tag{4.1}$$

where $\mathbf{X} \in \mathbb{C}^{n_x \times T}$, $\mathbf{Y} \in \mathbb{C}^{n_y \times T}$, $\mathbf{K} \in \mathbb{C}^{n_y \times n_x \times k}$ with $\mathbf{K}_{i**} \in \mathbb{C}^{n_x \times k}$ being the $i$th convolutional kernel of width $k$, and $\mathbf{\Xi}$ is a noise matrix with the same shape as $\mathbf{Y}$ and i.i.d. zero-mean complex normal entries $\mathcal{CN}(0, \sigma^2)$. See Appendix B.1 for a brief overview of complex normal distribution. The circular convolution in equation (4.1) is defined as

$$\mathbf{Y}_{i*} = \mathbf{K}_{i**} * \mathbf{X} + \mathbf{\Xi}_{i*} \tag{4.2}$$

$$\mathbf{Y}_{it} = \langle \mathbf{K}_{i**}, \mathbf{X}_{*,t:t+k} \rangle + \mathbf{\Xi}_{it}, \quad i \in [n_y], t \in [T] \tag{4.3}$$

Note that in (4.3) we are not using the correct definition of the convolution operation where the kernel (or the signal $\mathbf{X}$) is reflected along the time axis, but rather we are using the common definition used in machine learning.

We consider the inference problem in the Bayesian setting where the signal $\mathbf{X}$ is assumed to have a prior that admits a density (with respect to Lebesgue measure) $p(\mathbf{X})$. Further, we assume that rows of $\mathbf{X}_i$ are i.i.d. such that this density factorizes as

$$p(\mathbf{X}) = \prod_{i=1}^{n_x} p(\mathbf{X}_{i*}). \tag{4.4}$$

The convolution kernel $\mathbf{K}$ is assumed to be known with i.i.d. entries drawn from $\mathcal{CN}(0, 1/(n_y k))$. Given this statistical model, the posterior is

$$p(\mathbf{X}|\mathbf{Y}) \propto p(\mathbf{X}, \mathbf{Y}) = p(\mathbf{X}) p(\mathbf{Y}|\mathbf{X}), \tag{4.5}$$

where with some abuse of notation, we are using $p(\cdot)$ to represent the densities of all random variables to simplify the notation. From the Gaussianity assumption on noise we have

$$\mathbf{Y}_{i*}|\mathbf{X} \sim \mathcal{CN}(\mathbf{K}_{i**} * \mathbf{X}, \sigma^2 \mathbf{I}), \tag{4.6}$$

where $\mathbf{I}$ is the identity matrix of size $T \times T$.

Given the model in (4.5), one can consider different types of estimators for $\mathbf{X}$. Of particular

interest are regularized M-estimators

$$\widehat{\mathbf{X}}_{\mathsf{m\text{-}est}} = \arg\min_{\mathbf{X}} \mathcal{L}(\mathbf{X}, \mathbf{Y}) + \mathcal{R}(\mathbf{X}), \tag{4.7}$$

where $\mathcal{L}$ is a loss function and $\mathcal{R}$ corresponds to the regularization term. Taking negative log-likelihood as the loss and negative log-prior as the regularization we get the *maximum a posteriori* (MAP) estimator

$$\widehat{\mathbf{X}}_{\mathsf{MAP}} := \arg\min_{\mathbf{X}} -\log p(\mathbf{Y}|\mathbf{X}) - \log p(\mathbf{X}), \tag{4.8}$$

which selects mode of the posterior as the estimate.

In this work we are interested in analyzing the performance of ridge-regularized least squares estimator which is another special case of the regularized M-estimator in (4.7) with square loss and $\ell_2$-norm regularization

$$\widehat{\mathbf{X}}_{\mathsf{ridge}} = \arg\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{K} * \mathbf{X}\|_{\mathrm{F}}^2 + \lambda \|\mathbf{X}\|_{\mathrm{F}}^2, \tag{4.9}$$

where $\lambda$ is the regularization parameter. By Gaussianity of the noise, this can also be thought of as $\ell_2$-regularized maximum likelihood estimtor. Given an estimate $\widehat{\mathbf{X}}$, one is usually interested in performance of the estimator based on some metric. In Bayesian setting, the metric is usually an average risk (averaged over the prior and the randomness of data)

$$R = \mathbb{E}\ell(\mathbf{X}, \widehat{\mathbf{X}}), \tag{4.10}$$

where $\ell$ is some loss function between the true parameters and the estimates. The most widely used loss is the squared error where $\ell(\mathbf{X}, \widehat{\mathbf{X}}) = \|\mathbf{X} - \widehat{\mathbf{X}}\|_{\mathrm{F}}^2$. Our theoretical results exactly characterize the mean squared error (MSE) of ridge estimator (4.9) in a certain high-dimensional regime described below.

## 4.3 Main Result

Similar to other works in this area [Bayati and Montanari, 2011a, Rangan et al., 2019a, Pandit et al., 2020], our goal is to analyze the average case performance of the ridge estimator in

(4.9) for the convolutional model in (4.1) in a certain high dimensional regime that is called *large system limit* (LSL).

### 4.3.1 Large System Limit

We consider a sequence of problems indexed by $T$ and $n_x$. We assume that $k := k(T)$ and $n_y := n_y(n_x)$ are functions of $T$ and $n_x$ respectively and

$$\lim_{n_x \to \infty} \frac{n_y(n_x)}{n_x} = \delta \in (0, \infty), \quad \lim_{T \to \infty} \frac{k(T)}{T} = \beta \in (0, 1].$$

This doubly asymptotic regime where both the number of parameters and unknowns are going to infinity at a fixed ratio is sometimes called proportional asymptotic regime in the literature. We assume that the entries of the convolution kernel $\mathbf{K}$ and the noise $\mathbf{\Xi}$ converge empirically with second order to random variables $K$ and $\Xi$, with distributions $\mathcal{CN}(0, \sigma_K^2/(kn_y))$ and $\mathcal{CN}(0, \sigma^2)$ respectively. See Appendix B.2 for definition of empirical convergence of random variables.

#### Assumptions on $\mathbf{X}_{i*}$

Next, based on [Peligrad et al., 2010], we state the distributional assumptions on the rows of $\mathbf{X}$ that we require for our theory to hold. Let $\{\xi_t\}_{t \in \mathbb{Z}}$ be a stationary ergodic Markov chain defined on a probability space $(S, \mathcal{F}, P)$ and let $\pi(\cdot)$ be the distribution of $\xi_0$. Note that stationarity implies all $\xi_t$s have the same marginal distribution. Define the space of functions

$$\mathcal{L}_0^2(\pi) := \left\{ h | \mathbb{E}_\pi[h(\xi_0)] = 0, \mathbb{E}_\pi[h^2(\xi_0)] < \infty \right\}. \tag{4.11}$$

Define $\mathcal{F}_k := \sigma(\{\xi_t\}_{t \leqslant k})$, the $\sigma$-algebra generated by $\xi_t$ up to time $k$ and let $x_t = h(\xi_t)$ for some $h \in \mathcal{L}_0^2(\pi)$. We assume that the process $\{x_t\}_{t \in \mathbb{Z}}$ satisfies the regularity condition

$$\mathbb{E}[x_0 | \mathcal{F}_{-\infty}] = 0, \quad P - \text{almost surely.} \tag{4.12}$$

The class of processes that satisfy these conditions is quite large and includes i.i.d. random processes as an example. It also includes causal functions of i.i.d. random variables of the

form $X_n = f(\xi_k, k \leqslant n)$ where $\xi_k$ is i.i.d. such as autoregressive (AR) processes and many Markov chains. See [Peligrad et al., 2010] for more examples satisfying these conditions.

We assume that each row $\mathbf{X}_{i*}$ of $\mathbf{X}$, is an i.i.d. sample of a process that satisfies the conditions mentioned. Let $\widetilde{\mathbf{X}}_i(\omega)$ be its (normalized) Fourier transform (defined in (4.18)). and define $g(\omega) := \lim_{T\to\infty} \mathbb{E}|\widetilde{\mathbf{X}}_i(\omega)|^2$. As shown in [Peligrad et al., 2010], since the rows are i.i.d., this limit is the same for all the rows. Also, $g(\omega)$ is proportional to the spectral density of the process that generates the rows of $\mathbf{X}$

$$g(\omega) = \frac{1}{2\pi} \sum_{t=-\infty}^{\infty} c_n \exp(-\mathrm{i}\omega t), \quad c_t = \mathbb{E}[\mathbf{X}_{i0}\mathbf{X}_{it}]. \tag{4.13}$$

As we will see in the next section, $g(\omega)$ plays a key role in characterization of estimation error of ridge estimator in convolutional linear inverse problems that have such processes as inputs.

### 4.3.2 Asymptotics of Ridge Estimator

The main result of this chapter characterizes the limiting distribution to which the the Fourier transform of the true signal $\widetilde{\mathbf{X}}_0(\omega)$ and Fourier transform of the estimated signal $\widehat{\widetilde{\mathbf{X}}}_{\mathsf{ridge}}(\omega)$ converge. The proof can be found in Section 4.4. In the following $\mathcal{B}$ and $\mu$ represent Borel $\sigma$-algebra and Lebesgue measure respectively.

**Theorem 2.** *Under the assumptions in Section 4.3.1, as $n_x, n_y, T, k \to \infty$, over the product space $([0, 2\pi] \times S^{n_x}, \mathcal{B} \times \mathcal{F}^{n_x}, \mu \times P^{n_x})$ the Ridge estimator satisfies*

$$\begin{bmatrix} \widetilde{\mathbf{X}}_0(\omega) \\ \widehat{\widetilde{\mathbf{X}}}_{\mathsf{ridge}}(\omega) \end{bmatrix} \overset{d,PL(2)}{=} \begin{bmatrix} \sqrt{g(U)}Z_0 \\ \alpha(\sqrt{g(U)}Z_0 + \tau(g(U))Z_1) \end{bmatrix},$$

*where $U \sim \mathrm{unif}([0, 2\pi]), Z_0, Z_1 \sim \mathcal{CN}(0, 1)$ where $\mathcal{CN}(0, 1)$ is the standard complex normal distribution, $U, Z_0$ and $Z_1$ are independent of each other, $\alpha$ is the smaller root of the quadratic equation*

$$\lambda = \frac{(1-\alpha)(1-\alpha/\delta)}{\alpha}, \tag{4.14}$$

*and*

$$\tau^2(g(U)) = \frac{\sigma^2 + (1-\alpha^2)g(U)/\delta}{1-\alpha^2/\delta}. \tag{4.15}$$

The convergence in this theorem is weak convergence for $\omega$ and PL(2) for $\widetilde{\mathbf{X}}_0(\omega)$ and $\widehat{\widetilde{\mathbf{X}}}_{\mathsf{ridge}}(\omega)$. This convergence result allows us to find the asymptotic mean squared error of ridge estimator for the convolutional model as an integral.

**Corollary 1.** *Under the same assumptions as in Theorem 2, ridge estimator satisfies*

$$\lim_{n_x\to\infty}\lim_{T\to\infty}\frac{1}{n_x T}\|\widehat{\widetilde{\mathbf{X}}}_{\mathsf{ridge}} - \widetilde{\mathbf{X}}\|_{\mathrm{F}}^2 = \int_0^{2\pi}\left((\alpha-1)^2 g(\omega) + \alpha^2\tau(g(\omega))\right)d\omega. \tag{4.16}$$

**Remark 1.** As shown in the proof Lemma 6, for $\lambda \geqslant 0$, the quadratic equation (4.14) always has two real positive solutions the smaller of which determines the error.

**Remark 2.** The $1/\sqrt{T}$ scaling in our definition of Fourier operator in Section 4.4.2 makes it a unitary operator, i.e. $\ell_2$ norm is preserved under the Fourier transform and its inverse. This implies

$$\|\widehat{\mathbf{X}}_{\mathsf{ridge}} - \mathbf{X}\|_{\mathrm{F}} = \|\widehat{\widetilde{\mathbf{X}}}_{\mathsf{ridge}} - \widetilde{\mathbf{X}}\|_{\mathrm{F}}. \tag{4.17}$$

Therefore, the result of Corollary 1 also holds in time domain.

**Remark 3.** If rows $\mathbf{X}_{i*}$ have zero mean i.i.d. entries, then the correlation coefficients $c_t$ in (4.13) are all zero except for $c_0$. Therefore, $g(\omega) = g$ where $g$ is constant. Hence, in this case, the integrand in (4.16) would be a constant and the estimation error across all the frequencies would be the same as the estimation error in the ordinary ridge regression as in Lemma 6, i.e. the error vs. $\delta$ would be exactly the same as the double descent curve in ordinary ridge regression with i.i.d. priors. In other words, the double descent curve for ordinary ridge regression carries over to the convolutional ridge regression for i.i.d. priors (see Figure 4.1).

## 4.4 Proof

In this section we present the proof of Theorem 2. Before presenting the details of the proof, it is helpful to see an overview of the proof.

### 4.4.1 Proof overview

We first show that convolutional models turn into ordinary linear models for each frequency in Fourier domain. We then show that ridge estimators in time domain can also be written as ridge estimators in frequency domain. This uses the fact that Fourier transform matrix, with appropriate normalization is a unitary matrix, and $\ell_2$ norm is preserved under unitary transformation, i.e. it is an isometry. Next we use properties of Fourier transform of random processes to show that under certain conditions, they asymptotically converge to a Gaussian process in frequency domain that is independent across different frequencies for almost every frequency. These together allow us to turn the ridge estimation in time domain into multiple ridge estimators in frequency domain, one for each frequency. We then use theoretical properties of ridge estimators to derive estimation error for each of these ridge estimators and integrate them over frequencies to derive our main result.

Our proof is based on previous results for asymptotic error of ridge estimators for ordinary linear inverse problems. This has been studied in many works [Dicker et al., 2016, Dobriban et al., 2018, Hastie et al., 2019] where the authors take advantage of the fact that ridge estimators have a closed form solution that can be analyzed, e.g. using results from random matrix theory. In this work we use approximate message passing [Donoho et al., 2010b, Bayati and Montanari, 2011a] to derive the asymptotic error of ridge estimators.

### 4.4.2 Main Technical Lemmas

In order to prove Theorem 2, we need several lemmas. We first characterize the convolutional model in (4.1) in Fourier domain. For $\omega \in \{0, 1 \cdot \frac{2\pi}{T}, \dots, (T-1) \cdot \frac{2\pi}{T}\} =: \Omega$, let $\widetilde{\mathbf{X}}_j(\omega)$ be the

discrete (circular) Fourier transform (DFT) of $\mathbf{X}_{j*}$

$$\widetilde{\mathbf{X}}_j(w) = \frac{1}{\sqrt{T}} \sum_{t=0}^{T-1} \mathbf{X}_{jt} \exp(-\mathrm{i}\omega t). \tag{4.18}$$

If we let $\mathbf{F}$ to denote the $T$-point unitary DFT matrix, i.e. $\mathbf{F}_{mn} = 1/\sqrt{T}\exp(-\mathrm{i}2\pi mn/T)$, then this equation can be written in matrix form as $\widetilde{\mathbf{X}}_i(*) = \mathbf{X}_{i*}\mathbf{F}$. Define $\widetilde{\mathbf{Y}}_i(\omega)$, $\widetilde{\mathbf{K}}_{ij}(\omega)$, and $\widetilde{\boldsymbol{\Xi}}(\omega)$ similarly. Note that in these definitions, we have included a $1/\sqrt{T}$ factor which is usually not included in the definition of Fourier transform, but since this makes the Fourier matrix unitary, it eases our notation slightly. With this definition we have $\mathbf{F}^{\mathsf{T}} = \mathbf{F}$ and $\mathbf{F}^*\mathbf{F} = \mathbf{I}$.

**Lemma 1** (Convolutional models in Fourier domain)**.** *The convolutional model in* (4.1) *can be written in Fourier domain as*

$$\widetilde{\mathbf{Y}}(\omega) = \sqrt{T}\widetilde{\mathbf{K}}_{**}(\omega)\widetilde{\mathbf{X}}(\omega) + \widetilde{\boldsymbol{\Xi}}(\omega), \quad \forall \omega \in \Omega. \tag{4.19}$$

*Proof.* Taking Fourier transform of Equation (4.2) and using the convolution theorem we get

$$\widetilde{\mathbf{Y}}_i(\omega) = \sqrt{T}\sum_{j=1}^{n_x} \widetilde{\mathbf{K}}_{ij}(\omega)\widetilde{\mathbf{X}}_j(\omega) + \widetilde{\boldsymbol{\Xi}}_i(\omega). \tag{4.20}$$

Note that the $\sqrt{T}$ factor on the right hand is due to our definition of Fourier transform where we have used a $1/\sqrt{T}$ factor to make the Fourier operator unitary. Rewriting this equation in matrix form gives us the desired result. ∎

The next lemma characterizes the ridge estimator in (4.9) in frequency domain.

**Lemma 2.** *The ridge estimator in* (4.9) *in frequency domain is equivalent to solving separate ordinary ridge regressions for each $\omega \in \Omega$:*

$$\widehat{\widetilde{\mathbf{X}}}_{\mathsf{ridge}}(\omega) \quad = \quad \underset{\widetilde{\mathbf{X}}(\omega)}{\arg\min} \left\| \widetilde{\mathbf{Y}}(\omega) - \sqrt{T}\widetilde{\mathbf{K}}(\omega)\widetilde{\mathbf{X}}(\omega) \right\|_2^2 \quad + \quad \lambda \left\| \widetilde{\mathbf{X}}(\omega) \right\|_2^2. \tag{4.21}$$

*Proof.* Since the Fourier matrix is unitary, we have

$$\|\mathbf{X}\|_{\mathrm{F}} = \|\mathbf{X}\mathbf{F}\|_{\mathrm{F}} = \|\widetilde{\mathbf{X}}\|_{\mathrm{F}}$$

$$\|\mathbf{Y} - \mathbf{K} * \mathbf{X}\|_{\mathrm{F}} = \|(\mathbf{Y} - \mathbf{K} * \mathbf{X})\mathbf{F}\|_{\mathrm{F}} = \|\widetilde{\mathbf{Y}} - \sqrt{T}\widetilde{\mathbf{K}}\widetilde{\mathbf{X}}\|_{\mathrm{F}},$$

where $\widetilde{\mathbf{K}}\widetilde{\mathbf{X}}$ is a tensor-matrix product defined as $(\widetilde{\mathbf{K}}\widetilde{\mathbf{X}})(\omega) = \widetilde{\mathbf{K}}(\omega)\widetilde{\mathbf{X}}(\omega)$. Then, a change of variable $\widetilde{\mathbf{X}} = \mathbf{XF}$ in (4.9) proves the lemma. ∎

**Lemma 3.** *If the kernel* $\mathbf{K}$ *has i.i.d.* $\mathcal{CN}(0, \sigma_K^2/(kn_y))$ *entries, then for each* $\omega \in \Omega$, $\sqrt{T}\widetilde{\mathbf{K}}(\omega)$ *has i.i.d. complex normal entries* $\mathcal{CN}(0, \sigma_K^2/n_y)$.

*Proof.* The DFT of the kernel is

$$\widetilde{\mathbf{K}}_{ij}(\omega) = \frac{1}{\sqrt{T}} \sum_{t=0}^{k-1} \mathbf{K}_{ijt} \exp(-\mathrm{i}\omega t). \tag{4.22}$$

This is a linear combination of complex Gaussian random variables and therefore, $\widetilde{\mathbf{K}}$ is a tensor with jointly complex Gaussian entries. Clearly, $\mathbb{E}(\widetilde{\mathbf{K}}) = \mathbf{0}$ and for $(i,j) \neq (i',j')$, using independence of $\mathbf{K}_{ij*}$ and $\mathbf{K}_{i'j'*}$ we have

$$\mathbb{E}\widetilde{\mathbf{K}}_{ij}(\omega)\widetilde{\mathbf{K}}_{i'j'}(\omega') = 0, \quad \mathbb{E}\widetilde{\mathbf{K}}_{ij}(\omega)\widetilde{\mathbf{K}}^*_{i'j'}(\omega') = 0 \quad \forall \omega, \omega',$$

which proves that for any $\omega$, $\widetilde{\mathbf{K}}(\omega)$ has independent entries and all the dependence in $\widetilde{\mathbf{K}}$ is across different frequencies.

It remains to find the variance and relation of each entry of $\widetilde{\mathbf{K}}(\omega)$. Let $\mathbf{k} := \mathbf{K}_{ij*}$ for some $i$ and $j$ be a row vector, and let $\widetilde{\mathbf{K}} := \widetilde{\mathbf{K}}_{ij}(*)$. Then we have $\widetilde{\mathbf{k}} = \mathbf{kF}$ and the variance of $\sqrt{T}\widetilde{\mathbf{k}}_m$ is

$$\gamma = T\mathbb{E}[\widetilde{\mathbf{k}}_m\widetilde{\mathbf{k}}^*_m] = T\mathbb{E}[\mathbf{kF}_{*m}\mathbf{F}^*_{*m}\mathbf{k}^\mathsf{T}] \tag{4.23}$$

$$= T\mathbf{F}_{m*}\mathbb{E}[\mathbf{k}^\mathsf{T}\mathbf{k}]\mathbf{F}^*_{m*} = \frac{T\sigma_K^2}{kn_y}\mathbf{F}_{m*}\begin{bmatrix}\mathbf{I}_k & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\mathbf{F}^*_{m*} \tag{4.24}$$

$$= \frac{\sigma_K^2}{kn_y}\sum_{t=0}^{k-1}\exp(\frac{2\pi\mathrm{i}tm}{T})\exp(\frac{-2\pi\mathrm{i}tm}{T}) \tag{4.25}$$

$$= \frac{\sigma_K^2}{n_y}. \tag{4.26}$$

Similarly, the relation is

$$\begin{aligned} c(m) &= T\mathbb{E}[\widetilde{\mathbf{k}}_m \widetilde{\mathbf{k}}_m^{\mathsf{T}}] = T\mathbb{E}[\mathbf{k}\mathbf{F}_{*m}\mathbf{F}_{m*}\mathbf{k}^{\mathsf{T}}] \\ &= T\mathbf{F}_{m*}\mathbb{E}[\mathbf{k}^{\mathsf{T}}\mathbf{k}]\mathbf{F}_{*m} = \frac{\sigma_K^2}{k}\mathbf{F}_{m*}\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\mathbf{F}_{*m} \\ &= 0. \end{aligned}$$

Therefore, for each $\omega$, $\widetilde{\mathbf{K}}_{ij}(\omega) \sim \mathcal{CN}(0, \sigma_K^2/n_y)$ and they are i.i.d. for all $i, j$. ∎

**Remark 4.** Observe that the scaling of variance of entries of $\mathbf{K}$ with $1/k$ is crucial to get a non-trivial distribution for entries of $\sqrt{T}\widetilde{\mathbf{K}}(\omega)$ as we take the limit $T \to \infty$.

**Lemma 4.** *If noise $\boldsymbol{\Xi}$ has i.i.d. $\mathcal{CN}(0, \sigma^2)$ entries, then $\widetilde{\boldsymbol{\Xi}}$ has i.i.d. complex normal entries $\widetilde{\boldsymbol{\Xi}}_{ij}(\omega) \sim \mathcal{CN}(0, \sigma^2)$, i.e.*

$$\boldsymbol{\Xi}_{ij}(\omega) \overset{d}{=} \frac{\sigma^2}{2}Z_1 + \frac{\sigma^2}{2}Z_2, \quad Z_1, Z_2 \sim \mathcal{N}(0,1), Z_1 \perp\!\!\!\perp Z_2.$$

*Proof.* The proof is similar to the proof of Lemma 3 with $k = T$. ∎

Lemma 4 is the complex analogue of the fact that distribution of vectors with i.i.d. Gaussian entries is invariant under orthogonal transformations.

As stated in Appendix B.2, for a Gaussian random sequence, convergence in the first and second moments implies convergence in Wasserstein distance which is equivalent to $PL(2)$ convergence. Therefore, Lemma 3 and 4 also imply that the entries of kernel and noise for each frequency are converging empirically with second order to i.i.d. complex Gaussian random variables. As shown in the appendix, this convergence is stronger than convergence in distribution.

Next, we mention a result about Fourier transform of random processes from [Peligrad et al., 2010].

**Lemma 5** (Fourier transform of random processes [Peligrad et al., 2010])**.** *Let $\{X_t\}_{t\in\mathbb{Z}}$ be a stationary ergodic process that satisfies the assumptions in Section 4.3.1. Let $\widetilde{X}(\omega)$ be its*

*(normalized) Fourier transform and* $g(\omega) := \lim_{T\to\infty} \mathbb{E}|\widetilde{X}(\omega)|^2$. *Then on the product space* $([0, 2\pi] \times S, \mathcal{B} \times \mathcal{F}, \mu \times P)$ *we have*

$$\widetilde{X}(\omega) \overset{d}{=} \sqrt{g(U)}\mathcal{CN}(0, 1), \tag{4.27}$$

*where* $U \sim \text{unif}([0, 2\pi])$ *is independent of* $\mathcal{CN}(0, 1)$.

Lemma 5 allows us to characterize the limiting distribution of each row $\mathbf{X}_{i*}$ of the input in the frequency domain, i.e. distribution of $\widetilde{\mathbf{X}}_i(\omega)$ as $T \to \infty$.

All the lemmas so far allow us to look at the convolutional model in the frequency domain. We need one last lemma to characterize the asymptotics of ridge regression in high dimensions.

**Lemma 6** (Asymptotics of ridge regression)**.** *Consider the linear model* $\mathbf{y} = \mathbf{A}\mathbf{x}_0 + \boldsymbol{\xi}$, *where* $\mathbf{x} \in \mathbb{R}^{n_x}, \boldsymbol{\xi} \in \mathbb{R}^{n_y}$, *and* $\mathbf{A} \in \mathbb{R}^{n_y \times n_x}$ *all have i.i.d. components with* $\mathbf{x}_i \sim \mathcal{N}(0, \sigma_x^2)$, $\boldsymbol{\xi}_i \sim \mathcal{N}(0, \sigma^2)$, *and* $\mathbf{A}_{ij} \sim \mathcal{N}(0, 1/n_y)$. *Then the ridge estimator*

$$\widehat{\mathbf{x}}_{\mathsf{ridge}} = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_2^2 \tag{4.28}$$

*as* $n_x, n_y \to \infty$ *with* $n_y/n_x \to \delta$ *satisfy*

$$\begin{bmatrix} \mathbf{x}_0 \\ \widehat{\mathbf{x}}_{\mathsf{ridge}} \end{bmatrix} \overset{PL(2)}{=} \begin{bmatrix} X_0 \\ \alpha(X_0 + \tau(\sigma_x)Z) \end{bmatrix}, \tag{4.29}$$

*where* $X_0 \sim p_X$, $Z \sim \mathcal{N}(0, 1)$ *independent of* $X_0$, $\alpha$ *is the smaller root of the quadratic equation*

$$\lambda = \frac{(1-\alpha)(1-\alpha/\delta)}{\alpha}, \tag{4.30}$$

*and*

$$\tau^2(\sigma_x^2) = \frac{\sigma^2 + (1-\alpha^2)\sigma_x^2/\delta}{1 - \alpha^2/\delta}. \tag{4.31}$$

*Therefore, we almost surely have*

$$\lim_{n_x\to\infty} 1/n_x \|\mathbf{x}_{\mathsf{ridge}} - \mathbf{x}_0\|_2^2 = (\alpha - 1)^2\sigma_X^2 + \alpha^2\tau^2(\sigma_x).$$

*Proof.* This is a consequence of using *approximate message passing* (AMP) algorithm to solve (4.28). See Section 2.3 for an introduction to AMP algorithm. Here we briefly mention the

sketch of the proof for this lemma. In Appendix B.3.1 we show how the AMP algorithm can be used to solve the ridge regression problem. In particular, we show that in order to perform the ridge regression, the denoiser in AMP algorithm should be a linear denoiser of the form $\boldsymbol{\eta}(\mathbf{x}) = \alpha\mathbf{x}$. The correct value of $\alpha$ to perform ridge regression depends on the regularization parameter $\lambda$ and can be obtained by analyzing the fixed points of AMP algorithm and matching them to the ridge regression solution which can be found in closed form. Once we have found the exact form of the denoiser that solves the ridge regression problem, we can use the AMP state evolution to obtain its statistical error in the large system limit.

As shown in the appendix, the AMP recursions to solve ridge regression in (4.28) are

$$\mathbf{x}^{t+1} = \alpha(\mathbf{A}^\top \mathbf{z}^t + \mathbf{x}^t), \tag{4.32}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t + \frac{\alpha}{\delta}\mathbf{z}^{t-1}, \tag{4.33}$$

where $\alpha$ is the smaller root of the quadratic equation in (4.30). In Appendix B.3.2, we prove that for $\lambda \geqslant 0$, the roots of this equation are real and positive, and only for the smaller root the AMP algorithm converges. Finally, using the state evolution, we obtain that the ridge estimator and true values of $\mathbf{x}$ jointly converge as in (4.29), and $\tau^2(\sigma_x)$ in (4.31) is the fixed point value of state evolution recursion in (B.22). ∎

This lemma allows us to find asymptotics of ridge regression for real linear inverse problems. Even though the $\tau$ in (4.31) depends also on $\alpha, \delta$, and noise variance, we have only made the dependence on $\sigma_x$ explicit, as all the other parameters will be fixed for the ridge regression problem for each frequency, but $\sigma_x$ could change as a function of frequency.

**Remark 5.** The exact same result holds for complex valued ridge regression *mutatis mutandis*, i.e. by changing normal distributions $\mathcal{N}(0, \cdot)$ to complex normal distributions $\mathcal{CN}(0, \cdot)$.

**Remark 6.** The requirements of Lemma 6 can be relaxed. Rather than requiring $\mathbf{x}, \mathbf{A}$, or $\boldsymbol{\xi}$ to have i.i.d. Gaussian entries, we only need them to converge PL(2) to random variables with these distributions.

### 4.4.3 Proof of Theorem 2

We now have all the ingredients to prove Theorem 2. Lemma 2 allows us to find Fourier transform of Ridge estimator in 4.9 using a series of ridge regressions in Fourier domain. Lemmas 3 and 4 show that the Fourier transform of the convolution kernel and the noise and the signal asymptotically have complex Gaussian distributions with i.i.d. components for each frequency. Then, Lemma 5 shows that

$$\widetilde{\mathbf{X}}_i(\omega) \stackrel{d}{=} \sqrt{g(U)}\mathcal{CN}(0,1), \tag{4.34}$$

where $g(\cdot)$ is defined in the Lemma.

Next, the complex version of Lemma 6 would give us the asymptotic error of ridge estimator on the product space $([0,2\pi] \times S, \mathcal{B} \times \mathcal{F}, \lambda \times P)$ in the limit

$$\begin{bmatrix} \widetilde{\mathbf{X}}_0(\omega) \\ \widetilde{\mathbf{X}}(\omega) \end{bmatrix} \stackrel{PL(2)}{=} \begin{bmatrix} \sqrt{g(U)}Z_0 \\ \alpha(\sqrt{g(U)}Z_0 + \tau(g(U))Z_1) \end{bmatrix}, \tag{4.35}$$

where $U \sim \text{unif}([0,2\pi])$, and $Z_0, Z_1 \sim \mathcal{CN}(0,1)$, and $\tau(\cdot)$ is the function in (4.31). Note that the variance of $\widetilde{\mathbf{X}}_0(\omega)$ is $g(\omega)$, hence the term $\tau(g(U))Z_1$. As mentioned earlier, this variance is the only variable that changes with frequency while all the other parameters are the same for all frequencies. Using this convergence, in the limit, the error is

$$\lim_{n_x\to\infty}\lim_{T\to\infty}\frac{1}{n_x T}\|\widehat{\widetilde{\mathbf{X}}}_{\mathsf{ridge}} - \widetilde{\mathbf{X}}\|_{\mathrm{F}}^2 = \int_0^{2\pi}\left((\alpha-1)^2 g(\omega) + \alpha^2\tau(g(\omega))\right)d\omega. \tag{4.36}$$

As mentioned in Section 4.4.2, our scaling of the Fourier operator makes it a unitary operator. Therefore, $\ell_2$ norm is preserved under our definition of Fourier transform and its inverse. This implies that the same result as in (4.36) also holds in *time* domain.

## 4.5 Experiments

In this section we validate our theoretical results on simulated data. We generate data using a ground truth convolutional model of the form (4.3). We use i.i.d. complex normal convolution kernel and noise with different variances. For the data matrix $\mathbf{X}$, we consider two different

models: i) i.i.d. complex normal data; and ii) a non-Gaussian autoregressive process of order 1 (an AR(1) process). In both cases we take $T = 256$, $n_y = 500$ and use different values of $n_x$ to create plots of estimation error with respect to $\delta = n_y/n_x$.

AR(1) process is a process that evolves in time as

$$x_t = ax_{t-1} + \xi_t, \tag{4.37}$$

where $\xi_t$ is some zero mean random noise and $a$ is a fixed constant. Note that our assumptions on the process require it to be a stationary and ergodic process. These assumptions are only satisfied when the noise is i.i.d. and $|a| < 1$. The parameter $a$ controls how fast the process is mixing. The case were $a = 0$ results in an i.i.d. process, whereas values with magnitude close to 1 would result in a process that has strong correlations over a long period of time. This process has the form of one of the examples given in Section 4.3.1 and hence satisfies all the assumptions required for our theory to hold.

If the noise $\xi_t$ has zero mean Gaussian distribution, the process will be a centered Gaussian process which is completely characterized by an auto-correlation function

$$R[t] = \mathbb{E}[x_{t'+t}x_{t'}], \tag{4.38}$$

where we have used the fact that stationarity of the process implies this auto-correlation only depends on the time difference and not on the actual time. Since for Gaussian processes, the Fourier transform is another Gaussian process, we do not need to use the results of [Peligrad et al., 2010] to analyze them. As such, a more interesting example would be to use a non-Gaussian noise $\xi$. Therefore, besides the Gaussian AR process, we also take $\xi_t \sim \text{unif}(\{-s, s\})$ where $s$ is an step size that controls the variance of the process. The variance and auto-correlation of a univariate AR(1) process can be found as follows. Squaring both sides of (4.37) and taking the expectation we obtain

$$\mathbb{E}[x^2] = \frac{\mathbb{E}[\xi^2]}{1 - a^2}. \tag{4.39}$$

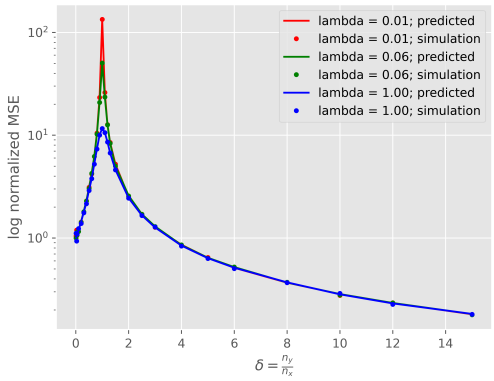



**Figure 4.1:** Log of normalized error for i.i.d. Gaussian features with respect to $\delta = n_y/n_x$ for three different values of $\lambda$. Solid lines show the predictions of our theory whereas the dots show the observed error on synthetic data.

Similarly, it is also easy to show that the auto-correlation of this process is

$$R[t] = \frac{\mathbb{E}[\xi^2]}{1-a^2} a^{|t|}. \tag{4.40}$$

The auto-correlation function of the AR process only depends on the variance of the noise and not its distribution. Our main result, Theorem 2, shows that the asymptotic error of ridge estimator depends on the the underlying process only through the function $g(\omega)$ which as stated earlier, is proportional to the spectral density of the process, i.e. norm of Fourier transform of the auto-correlation function. Therefore, so long as the zero mean noise has the same variance, irrespective of its distribution, we expect to see the same asymptotic error in the convolutional ridge regression when the rows of $\mathbf{X}$ are i.i.d. samples of such processes. To show this, we use both a Gaussian AR(1) process with $\text{var}(\xi_t^2) = 0.1$ as well as $\xi_t \, \text{unif}(\{-s, s\})$ with $s = \sqrt{0.1}$ to match the variances. In both cases we take $a = 0.9$ and measurement noise variance $\sigma^2 = 0.1$.

We first present the results for the i.i.d. Gaussian covariates. In this case the variance of signal and noise are 0.004 and 1 respectively. Figure 4.1 shows the log of normalized estimation error with respect to $\delta = n_y/n_x$ for three different values of the regularization

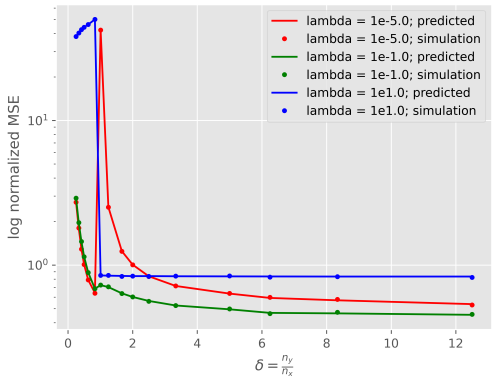



**Figure 4.2:** Log of normalized error for the AR(1) features with the process noise $\text{unif}(\{-s, s\})$, with respect to $\delta = n_y/n_x$ for three different values of $\lambda$. Solid lines show the predictions of our theory and the dots show the observed error on synthetic data. The plots for Gaussian AR process is essentially identical. to this plot.

parameter $\lambda$. Normalized estimation error is defined as

$$\text{NMSE} = \frac{\mathbb{E}\|\widehat{\mathbf{X}}_{\mathsf{ridge}} - \mathbf{X}_0\|_2^2}{\mathbb{E}\,\|\mathbf{X}_0\|_2^2}. \tag{4.41}$$

The solid curves correspond to what our theory predicts and the dots correspond to what we observe on synthetic data. Even though our results hold in the limit of $n_x, n_y, k, T \to \infty$ at proportional ratio, we can see that already at this problem size, there is an almost perfect match between our predictions and the error that is observed in practice. This suggests that the errors concentrate around these asymptotic values. The figure also shows the double descent phenomenon where as the number of parameters increases beyond the interpolation threshold, the error starts decreasing again. It can be seen that regularization helps with pulling the estimation error down in vicinity of the interpolation threshold. The interpolation threshold is where we have just enough parameters to fit the observations perfectly. This happens at $\delta = 1$, i.e. $n_x = n_y$.

As mentioned in Remark 3 an i.i.d. process has a white spectrum in frequency domain, meaning that $\boldsymbol{g}(\omega)$ is a constant. Therefore, for these processes, the integral in Corollary 1 would be proportional to the integrand. The integrand in turn is the asymptotic error of an ordinary ridge regression problem with i.i.d. Gaussian features. As such, this figure

is essentially the same as the figures in papers that have looked at asymptotics of ridge regression, some of which we have mentioned in the Introduction and prior work.

Figure 4.2 shows the same plot for the case where the rows of $\mathbf{X}$ are i.i.d. samples of an AR(1) process with process noise unif($\{-s, s\}$). The asymptotic error for processes that have dependencies over time can be significantly different from i.i.d. random features. The red curve is similar to the red curve in Figure 4.1, but the other curves show very different behavior. The double descent phenomenon is still present here. The plot for AR(1) process with Gaussian noise is essentially identical to Figure 4.2 and we have moved it to the appendix (Figure B.1). This supports our theoretical result that error in this asymptotic regime only depends on the spectral density of the process.

## 4.6 Conclusion

**Summary.** We characterized the performance of ridge estimator for convolutional models in proportional asymptotics regime. By looking at the problem in Fourier domain, we showed that the asymptotic mean squared error of ridge estimator can be found from a scalar integral that depends on the spectral properties of the true signal. Our experiments show that our theoretical predictions match what we observe in practice even for problems of moderate size.

**Future work.** The results of this work only apply to ridge regression estimator for convolutional linear inverse problem. The key property of ridge regularization is that it is invariant under unitary transforms, and hence we could instead solve the problem in frequency domain. Proving such result for general estimators and regularizers allows us to extend this work to inference in deep convolutional neural networks similar to [Pandit et al., 2020]. Such work would allow us to obtain the estimation error inverse problems use of deep convolutional generative priors.

# Chapter 5

# Generalized Autoregressive Linear Models for Discrete High-dimensional Data

## 5.1 Introduction

We consider the problem of learning a $p$-lag autoregressive (AR) generalized linear model (GLM) for a multivariate time series involving $N$-variables: $\boldsymbol{x}^t = (x_i^t) \in \mathbb{R}^N$, where $x_i^t \in \mathcal{X}_i \subseteq \mathbb{R}$ for all $i \in [N]$, $t \in \mathbb{Z}$. A particular case of the model we consider is of the form,

$$x_i^t \mid z_i^t \sim \mathbb{Q}_i(\,\cdot \mid z_i^t), \quad z_i^t = f_i\left(\langle \Theta_i^*, \mathbf{X}^{t-1} \rangle\right), \tag{5.1}$$

where the inner product corresponds to $\mathbb{R}^{N \times p}$, for $t = 1, 2, \ldots$ and $i = 1, 2, \ldots, N$ where $\mathbf{X}^{t-1} = [\boldsymbol{x}^{t-1}\ \boldsymbol{x}^{t-2}\ \ldots\ \boldsymbol{x}^{t-p}] \in \mathbb{R}^{N \times p}$ is the $p$-lag history of the process up to time $t-1$, and $\mathbb{Q}_i(\,\cdot \mid z_i^t)$ is a probabilistic link function. The problem is to estimate the unknown parameters $\Theta_i^* \in \mathbb{R}^{N \times p}$ for $i = 1, 2, \ldots, N$, given observations of $n$ time samples $\boldsymbol{x}^t$, $t = 1, \ldots, n$. The conditional distributions $\mathbb{Q}_i(\,\cdot \mid z_i^t)$ and link functions $f_i$ are assumed to be known.

Modeling problems of this form appear in a wide-range of applications with time-series

data. For example, in neural modeling, $\boldsymbol{x}^t$ can represent a vector of spike counts or some other measure of activity from $N$ neurons or brain regions. In this case, estimation of the tensor $\Theta^*$ in (5.1) can provide insight into the neural connectivity. Other applications include genomics, econometrics [De Mol et al., 2008], data science, sociology, business management, financial markets [Timmermann, 1996, DeJong and Whiteman, 1991] and natural language processing.

A key challenge in estimating the multivariate AR($p$) models is the large number of unknown parameters to estimate, particularly as the dimension of the process, $N$, and number of time lags, $p$, grows. However, in many cases, one can assume some sparsity constraint in the connectivity tensor $\Theta^*$. For example, in neural modeling, there are physically limited numbers of direct connections between brain regions. Under a sparsity assumption, it is common to estimate $\Theta^*$ via an $\ell_1$-regularized M-estimator of the form,

$$\begin{aligned}\widehat{\Theta} := \underset{\Theta \in \mathbb{R}^{N\times N\times p}}{\operatorname{argmin}} \; \tfrac{1}{n}\sum_{i=1}^{N}\sum_{t=1}^{n} \mathcal{L}_{it}\left(x_i^t; \langle \Theta_i, \mathbf{X}^{t-1}\rangle\right) \\ + \lambda_n \left\|\Theta\right\|_{1,1,1},\end{aligned} \tag{5.2}$$

where $\mathcal{L}_{it} : \mathcal{X}_i \times \mathbb{R} \to \mathbb{R}$ are loss functions and $\lambda_n |\!|\!| \Theta |\!|\!|_{1,1,1}$ is an $\ell_1$ regularizer (precise definitions will be given in the Section 5.2 below). The broad goal of this work is to analyze the sample complexity of such $\ell_1$-regularized M-estimators. That is, given a sparsity constraint on $\Theta^*$, and the number of measurements, $n$, how well can we estimate $\Theta^*$?

**Summary of Contributions** We consider the case where $\{\mathcal{X}_i\}_{i=1}^N$ are bounded countable subsets of $\mathbb{R}$. We analyze the $\ell_1$-regularized M-estimator (5.2) when the loss functions $v \mapsto \mathcal{L}_{it}(u; v)$ are strongly convex, for all $u \in \mathcal{X}_i$. We assume that the connectivity tensor can be approximated by a sparse tensor with at most $s_{\max}$ non-zero values in each slice $\Theta_i^*$. Under these assumptions, our main result in Theorem 3 establishes the consistency of the regularized M-estimator (5.2) in the high-dimensional regime of $n = \mathsf{poly}\big(s_{\max}\log(N^2 p)\big)$ under some regularity conditions.

In proving our main result, we establish the so-called restricted strong convexity (RSC) [Ne-

gahban et al., 2012] for a large class of loss functions, for a dependent non-Gaussian discrete-valued multivariate process. Our proof of the RSC property requires showing a restricted eigenvalue condition, which is nontrivial due to the non-Gaussian and highly-correlated nature of the design matrix. What makes the problem more challenging is the existence of feedback from more than just the immediate past (the case $p > 1$).

We establish the RSC for general $p \geqslant 1$ using the novel approach of viewing the $p$-block version of the process as a Markov chain. The problem becomes significantly more challenging when going from $p = 1$ to even $p = 2$. The difficulty with this *higher-order* Markov chain is that its *Dobrushin contraction coefficient* is trivially 1. We develop techniques to get around this issue which could be of independent interest (cf. Section 5.7). Our techniques hold for all $p \geqslant 1$.

Much of the previous work towards proving the RSC condition has either focused on the independent sub-Gaussian case [Raskutti et al., 2011, Zhang et al., 2008] or the dependent Gaussian case [Basu et al., 2015, Raskutti et al., 2019] for which powerful Gaussian concentration results such as the Hanson–Wright inequality [Rudelson et al., 2013] are still available. Our approach is to use concentration results for Lipschitz functions of Markov chains over countable spaces, and strengthen them to uniform results using metric entropy arguments. In doing so, we circumvent the use of empirical processes which require additional assumptions for estimation [Rakhlin et al., 2015]. Moreover, our approach allows us to identify key properties of the model that allow for sample-efficient estimation.

Although discrete time series are often modeled using the specific link functions such as `logit` or `softmax`, our result allows more flexibility to choose the link functions. For example in the Bernoulli AR($p$) and Truncated-Poisson AR($p$) cases discussed in Section 5.3.2, any Lipschitz continuous, log-convex link function can be used. The analysis also brings out crucial properties of the link function, and the role it plays in determining the estimation error and sample complexity.

Our model also allows for each individual time series $x_i^t$ to lie in distinct spaces $\mathcal{X}_i$ which

is desirable in practical applications with heterogeneous types of data.

### 5.1.1 Previous work

There is a vast literature on recovering sparse vectors in under-sampled settings [Candes and Tao, 2006, Candes et al., 2006, Donoho, 2006, Eldar and Kutyniok, 2012]. The generic results show that if a vector $\theta$ is $s$-sparse in a $p$-dimensions, it can be estimated in $n = \Omega(s \log(p))$ measurements. However, these results typically do not have feedback as in the AR process considered here.

The estimation of sparse Gaussian VAR($p$) processes with linear feedback has been considered only more recently [Basu et al., 2015, Cai et al., 2016, McMurry et al., 2015, Mei and Moura, 2017, Ahelegbey et al., 2016]. For these models, a restricted eigenvalue condition can be established fairly easily, by reducing the problem, even in the time-correlated setting, to the concentration of quadratic functionals of Gaussian vectors for which powerful inequalities exist [Rudelson et al., 2013]. These techniques do not extend to non-Gaussian setups.

In the non-Gaussian setting, Hall et al. [Hall et al., 2018, Zhou and Raskutti, 2018] recently considered a multivariate time series evolving as a GLM driven by the history of the process similar to our model. The Bernoulli AR(1) and Poisson AR(1) with $p = 1$ lags were considered as special cases of this model. They provide statistical guarantees on the error rate for the $\ell_1$ regularized estimator. More importantly, their results are restricted to the case $p = 1$ which does not allow the explicit encoding of long-term dependencies. More recently, Mark et al. [Mark et al., 2018, Mark et al., 2017] considered a model closer to ours for multivariate AR(p) processes with lags $p = 1$ or $p = 2$.

A key contribution of ours is to bring out the explicit dependence on $p$ in the AR($p$) models, allowing for a general $p \geqslant 1$. In the special cases we consider: the Bernoulli AR($p$) and the Truncated-Poisson AR($p$), we show how the scaling of the sample complexity and the error rate with $p$ can be controlled by the properties of the link function $f_i$ and a certain

norm of the parameter tensor.

Our results improve upon those in [Hall et al., 2018, Mark et al., 2018] when applied to the Bernoulli AR($p$) and Truncated-Poisson AR($p$). Due to the key observation that an AR($p$) over a countable space can be viewed as a higher order Markov chain, our analysis relaxes several assumptions made by [Hall et al., 2018, Mark et al., 2018]. In doing so, we achieve better sample complexities with explicit dependence on $p$. Our analysis borrows from martingale-based concentration inequalities for Lipschitz functions of Markov chains [Kontorovich et al., 2008].

The univariate Bernoulli AR($p$) process for $p \geqslant 1$ was considered by Kazemipour et. al. [Kazemipour, 2018, Kazemipour et al., 2017] where they analyzed a multilag Bernoulli process for a single neuron. Their analysis does not extend to the $N > 1$ case. Even for $N = 1$, their analysis is restricted to the biased process with $\mathbb{P}(x_1^t = 1|\mathbf{X}^{t-1}) < \frac{1}{2}$ for all $t$. Mixing times of the Bernoulli AR(1) have been considered in [Katselis et al., 2018]. However, their discussion is again limited to $p = 1$.

The rest of this chapter is organized as follows. In Section 5.2, we introduce the generalized discrete VAR($p$) model and the associated class of regularized M-estimators. Section 5.3 presents our main result, Theorem 3, on the consistency of the regularized M-estimator and discusses its assumptions and implications. Applications of Theorem 3 to the special cases of Binomial and Truncated-Poisson processes are detailed in Section 5.3.2. In Section 5.4, we provide simulation results corroborating our theoretical predictions. Section 5.5 provides an overview of the proof of Theorem 3. In Section 5.7, we present new techniques for deriving concentration inequalities for dependent multivariate processes. We conclude with a discussion and point to some open problems and directions for solving them in Section 5.8.

**Notation.** For two sequence $\{a_n\}$ and $\{b_n\}$, we write either of $a_n \gtrsim b_n$ or $b_n \lesssim a_n$ or $b_n = O(a_n)$ or $a_n = \Omega(a_n)$ to mean that there is a constant $C > 0$ such that $a_n \geqslant Cb_n$ for all $n$. We write $a_n \asymp b_n$ if both $a_n \gtrsim b_n$ and $b_n \gtrsim a_n$. We write $a_n \gg b_n$ or $b_n \ll a_n$ or

$b_n = o(a_n)$ if $b_n/a_n \to 0$ as $n \to \infty$. We use $[N]$ to denote the set $\{1, 2, \ldots, N\}$. For a subset $\mathcal{X}$ of a vector space, we write $\mathcal{X}^{\times p}$ for the set of matrices with $p$ columns from $\mathcal{X}$. Formally $\mathcal{X}^{\times p} := \{(x_1, x_2, \ldots, x_p) \mid x_i \in \mathcal{X},\ i \in [p]\}$. For example, $(\mathbb{R}^N)^{\times p}$ is the same as the set of real-valued $N \times p$ matrices. In addition, Table C.1 in the Appendices provides a list of all notations used in this chapter.

## 5.2 Models and methods

To state our results in their full generality, we consider a slightly more general model than (5.1). We assume that the multivariate time series $\boldsymbol{x}^t = (x_i^t) \in \mathcal{X} \subset \mathbb{R}^N$ evolves as,

$$x_i^t \mid z_i^t \sim \mathbb{Q}_i(\,\cdot \mid z_i^t) \tag{5.3a}$$

$$z_i^t = f_i\big(\langle \Theta_i^*, \mathbf{X}^{t-1}\mathbf{D}\rangle_{\mathbb{R}^{N\times L}}\big) \tag{5.3b}$$

$$x_i^t \perp\!\!\!\perp x_j^t \mid \boldsymbol{x}^{t-1}, \boldsymbol{x}^{t-2}, \ldots \tag{5.3c}$$

for $t = 1, 2, \ldots$ and $i = 1, 2, \ldots, N$. The key difference here is that we have added a matrix $\mathbf{D} = [\boldsymbol{d}_1\ \boldsymbol{d}_2\ \ldots\ \boldsymbol{d}_L] \in \mathbb{R}^{p\times L}$, a known dictionary of filters $\{\boldsymbol{d}_\ell\}_{\ell=1}^L$. When $\mathbf{D} = \mathbf{I}_{p\times p}$, we obtain the special case (5.1). The role of this dictionary will be explained below. To model the discrete-valued nature of the states, we assume that $\boldsymbol{x}^t \in \mathcal{X} := \prod_{i=1}^N \mathcal{X}_i$ where each $\mathcal{X}_i$ is a bounded countable subset of $\mathbb{R}$. The matrix $\mathbf{X}^{t-1} = [\boldsymbol{x}^{t-1}\ \boldsymbol{x}^{t-2}\ \ldots\ \boldsymbol{x}^{t-p}] \in \mathbb{R}^{N\times p}$ is the $p$-lag history of the process up to time $t-1$, and $\mathbb{Q}_i(\,\cdot \mid z)$ is a distribution on $\mathcal{X}_i$ parameterized by $z$. For example an exponential family distribution with mean parameter $z$. The matrices $\Theta_i^* \in \mathbb{R}^{N\times L}$, $i \in [N]$ are the (unknown) model parameters and $\langle\cdot,\cdot\rangle_{\mathbb{R}^{N\times L}}$ is the inner product. A process of this form will be denoted GVAR($p$).

The distribution $\mathbb{Q}_i(\,\cdot \mid z_i^t)$ represents the conditional distribution of $x_i^t$ given the past $\boldsymbol{x}^{t-1}, \boldsymbol{x}^{t-2}, \ldots$. Functions $f_i : \mathbb{R} \to \mathbb{R}$ are similar to the inverse-link functions in GLMs, and can be nonlinear in general. It is worth noting that $\mathcal{X}_i$ and $\mathbb{Q}_i$ can vary for every variable $i \in [N]$ making the model extremely flexible to include heterogeneous types of discrete data.

The inner product $\langle \cdot, \cdot \rangle_{\mathbb{R}^{N \times L}}$ in (5.3) is the Hilbert-Schmidt inner product on $\mathbb{R}^{N \times L}$, and can be expanded as:

$$\langle \Theta_i^*, \mathbf{X}^{t-1}\mathbf{D} \rangle_{\mathbb{R}^{N \times L}} = \sum_{j=1}^{N} \sum_{\ell=1}^{L} \Theta_{ij\ell}^* \langle \mathbf{x}_j^{t-*}, \boldsymbol{d}_\ell \rangle_{\mathbb{R}^p} \tag{5.4}$$

where $\mathbf{x}_j^{t-*} := [x_j^{t-1} \; x_j^{t-2} \; \dots \; x_j^{t-p}]$ is the $p$-lag history of variable $j$ up to time $t-1$, i.e., the $j^{\text{th}}$ row of $\mathbf{X}^{t-1}$. Note that $(\mathbf{X}^{t-1}\mathbf{D})_{j\ell} = \langle \mathbf{x}_j^{t-*}, \boldsymbol{d}_\ell \rangle_{\mathbb{R}^p}$. The parameter $(\Theta_i^*)_{j\ell} = \Theta_{ij\ell}^* \in \mathbb{R}$ captures the dependence of variable $x_i^t$ on the past activity of variable $j$, via $\mathbf{x}_j^{t-*}$. The vectors $\boldsymbol{d}_\ell \in \mathbb{R}^p$ act as filters that modulate the mean of variable $x_i^t$ based on the past activity of all the variables, that is, $x_j^k$ for $j \in [N]$, and $t - p \leqslant k < t$.

### 5.2.1 Dictionary and network interpretations

The filters $\{\boldsymbol{d}_\ell\}$ serve two main purposes: (i) interpretability and (ii) dimension reduction. For example, in neuroscience applications where the types of spiking behaviors are limited, the presence of a dictionary causes the model to favor specific forms of interactions between the spiking activities of two neurons. We refer to [Weber and Pillow, 2017] which explores these filters for various interactive behaviors among neurons such as bursting, tonic spiking, phasic spiking, etc. The dictionary increases the interpretability of the parameter $\Theta_i^*$—one interprets $(\Theta_i^*)_{j\ell}$ as measuring the effect of the activity of neuron $i$ on neuron $j$, as explained by interaction type $\ell$. Thus, the sparsity of $\Theta_i^*$ is more meaningful in the presence of a dictionary. An earlier version of this work [Pandit et al., 2019a] considered modeling the interaction with the past as $\langle \Theta_i^*, \mathbf{X}^{t-1} \rangle$ where $\Theta_i^*$ lies in $\mathbb{R}^{N \times p}$, corresponding to taking $\mathbf{D} = \mathbf{I}_{p \times p}$, the identity matrix, in (5.3c). The formulation with a general dictionary $\mathbf{D}$ has the added advantage of potentially reducing the number of free parameters from $Np$ to $NL$. When $L \ll p$, this leads to a massive dimension reduction. The bilinear term $\langle \Theta_i^*, \mathbf{X}^{t-1}\mathbf{D} \rangle_{\mathbb{R}^{N \times L}} = \langle \Theta_i^* \mathbf{D}^\top, \mathbf{X}^{t-1} \rangle_{\mathbb{R}^{N \times p}}$ can also be thought of as a low-rank approximation to the parameter, forcing one factor to be fixed by $\mathbf{D}$. By adding pre-existing knowledge of temporal interactions between variables, the dictionary allows for a rich model with fewer parameters, leading to more (sample) efficient

estimators for $\Theta^*$.

The parameter $\Theta^*$ can be interpreted as representing a network among variables $x_i^t$, $i \in [N]$. A slice $\Theta_{**\ell}$ can be thought of as an adjacency matrix for the *influence network* explained by coupling behaviour $\ell$. If neurons $i$ and $j$ are not connected, then $\Theta_{ij\ell} = 0$ for all $\ell \in [L]$. For example, in the neural spike train application, one can reveal a latent network among the neurons (i.e., who influences whose firing) just from the observations of patterns of neural activity, a task which is of significant interest in neuroscience [Okatan et al., 2005, Smith and Brown, 2003, Brown et al., 2004]. Similarly, in the context of social networks, one might be interested in who is influencing whom [Raginsky et al., 2012].

### 5.2.2 Examples

The GVAR($p$) process of the form (5.3) can be applied in a wide range of applications. For example, letting $\mathbb{Q}_i(\,\cdot \,|\, z) = \text{Ber}(z)$ and $f_i(u) = (1 + e^{-u})^{-1}$ recovers the Bernoulli autoregressive process in [Pandit et al., 2019a]. Similarly, $\mathbb{Q}_i(\,\cdot \,|\, z) = \text{Binomial}(K_i, z)$ and $f_i(u) = (1 + e^{-u})^{-1}$ models a Binomial process with $K_i$ trials (for coordinate $i$) and success probability $z$. Such a model can be suitable for modeling count data. Another common model for point processes in neuroscience [Smith and Brown, 2003] is the Truncated-Poisson autoregressive process given by $\mathbb{Q}_i(\,\cdot \,|\, z) = \mathbb{P}(\min(M_i, Z) \in \,\cdot\,)$ where $Z \sim \text{Poi}(z)$, and $f_i(u) = \exp(u)$ or $f_i(u) = \log(1 + e^u)$ for some integer $M_i$ [Hall et al., 2018, Mark et al., 2018]. Although we focus on single-parameter discrete distributions in this work, the ideas can be easily extended to distributions with multiple parameters. For example, one can construct a categorical or multinomial process, by allowing $z_i^t$ to be vector-valued and taking $f_i$ to be the `softmax` function.

### 5.2.3 Regularized $M$-Estimation

We are primarily interested in parameter estimation in the high-dimensional regime where $n \ll N$. To make the estimation feasible, we assume that the activity of each variable $i$ depends on the past activity of only a few number of variables, $s_i \ll N$. We refer to $s_i$ as the *in-degree* of variable $i$. Our main result provides sufficient conditions under which parameter $\Theta^*$ can be estimated in the high-dimensional setting where $n = \mathsf{poly}(\{s_i\}_{i=1}^N, \log(NLp))$.

Given a collection of loss functions $\mathcal{L}_{it} : \mathcal{X}_i \times \mathbb{R} \to \mathbb{R}$, for $i \in [N]$ and $t \in \mathbb{Z}$, we consider the following $\ell_1$-regularized M-estimator

$$\begin{aligned} \widehat{\Theta} &:= \operatorname*{argmin}_{\Theta \in \mathbb{R}^{N \times N \times L}} \sum_{i=1}^{N} \mathcal{L}_i(\Theta_i) + \lambda_n \|\Theta\|_{1,1,1}. \\ \mathcal{L}_i(\Theta_i) &:= \tfrac{1}{n} \sum_{t=1}^{n} \mathcal{L}_{it}\left(x_i^t ; \langle \Theta_i, \mathbf{X}^{t-1}\mathbf{D} \rangle\right) \end{aligned} \tag{5.5}$$

where we use the notation

$$\|\boldsymbol{M}\|_{p,q,r} := \left( \sum_{i=1}^{a} \left\{ \sum_{j=1}^{b} \left( \sum_{k=1}^{c} |M_{ijk}|^r \right)^{\frac{q}{r}} \right\}^{\frac{p}{q}} \right)^{\frac{1}{p}} \tag{5.6}$$

to denote a norm of a $a \times b \times c$ tensor $\boldsymbol{M}$ (when $p, q, r > 1$). We also use a similar norm notation for matrices $\|\boldsymbol{M}\|_{p,q} := \sum_{i=1}^{a} (\sum_{j=1}^{b} |M_{ij}|^q)^{\frac{p}{q}}$. For $p = q = r = 2$, we denote the norm subscript by $F$.

Since both the loss function and the $\ell_1$ penalty are decomposable, we can solve each of the $N$ problems in (5.5) indexed by $i$ separately,

$$\widehat{\Theta}_i := \operatorname*{argmin}_{\Theta_i \in \mathbb{R}^{N \times L}} \mathcal{L}_i(\Theta_i) + \lambda_n \|\Theta_i\|_{1,1} \quad \forall\, i \in [N]. \tag{5.7}$$

The possible dependence of $\mathcal{L}_{it}$ on $t$ in the $M$-estimator (5.5) allows for the incorporation of time-discounting factors such as $\gamma^t$ for some $\gamma < 1$. We consider a large class of loss functions later stated explicitly in assumptions (A2) and (A3). This class always includes the negative-log likelihood function for exponential family distributions $\mathbb{Q}_i(\,\cdot \,|\, f_i(v))$ with log-concave link $f_i$, and pseudo-likelihood functions in some cases. When $\mathcal{L}_{it}$ are chosen to be convex, the whole problem (5.5) is unconstrained, convex, with a coercive objective

function, whereby the solution $\widehat{\Theta}$ is unique. Furthermore, the estimator (5.5) can be solved efficiently using any non-smooth convex optimization solver, such as the subgradient methods or proximal gradient descent methods [Bertsekas, 2011]. An implementation for the general problem in (5.5) is available at [Sahraee-Ardakan et al., 2020] which implements both the subgradient method as well as the proximal gradient method.

Each iteration of both of these methods involve computation of the gradient of the loss function followed by finding the sub-gradient or proximal mapping for the regularization. Computing the gradient of the loss is the most expensive step. The gradient of the loss is

$$\nabla \mathcal{L}(\Theta_i) = \frac{1}{n} \sum_{t=1}^{n} \mathcal{L}'_{it} \left( x_i^t \, ; \langle \Theta_i, \mathbf{X}^{t-1}\mathbf{D} \rangle \right) \mathbf{X}^{t-1}\mathbf{D}, \tag{5.8}$$

where in $\mathcal{L}'_{it}(\cdot\,;\cdot)$ the derivative is with respect to the second argument. To compute the gradient, $\mathbf{X}^{t-1}\mathbf{D}$ can be precomputed once by multiplying $\mathbb{X} := \{\boldsymbol{x}^t\}_{t=-p+1}^{n}$ and $\mathbf{D}$. Hence, the complexity of obtaining the gradient $\nabla \mathcal{L}(\Theta_i)$ at each iteration is dominated by that of computing $\langle \Theta_i, \mathbf{X}^{t-1}\mathbf{D} \rangle$ for all $i$, that is, $O(nNL)$. To solve the optimization problem, one can then use the subgradient method with a provable convergence rate of $1/\sqrt{k}$ after $k$ steps. This relatively slow rate is due to the non-smoothness of the objective function. Alternatively, we can use the proximal gradient method that converges at a rate of $1/k$. Then, the overall computational complexity of obtaining an $\varepsilon$-optimal solution is $O(nNL/\varepsilon)$. The parallel implementation in (5.7) allows for massive speed-ups in computation when using GPUs. The main result of this work concerns the statistical complexity of the estimator and is agnostic to the choice of the optimization solver.

Our main result establishes the statistical properties of estimator (5.5) such as consistency, sample complexity and error rate. Our analysis also highlights desirable properties of the loss functions $\mathcal{L}_{it}$ and the nonlinearities $f_i$ for achieving consistency. The result also shows the effect of the dictionary $\mathbf{D}$ in increasing the sample-efficiency of the estimator.

## 5.3 Main Results

Our main result concerns the estimation error of the parameters $\{\widehat{\Theta}_i\}_{i=1}^N$, obtained by solving (5.7). We implicitly assume $\Theta_i^*$ to be approximately $s_i$-sparse. This assumption is encoded via the $\ell_1$-approximation errors

$$\omega_i := \min_{\boldsymbol{\beta} \in \mathbb{R}^{N \times L}} \{\|\boldsymbol{\beta} - \Theta_i^*\|_1 \mid \|\boldsymbol{\beta}\|_{0,0} \leqslant s_i\}. \tag{5.9}$$

We also impose the following assumptions:

(A1) The process is wide-sense stationary and stable, i.e., the power spectral density matrix exists:

$$\mathscr{X}(\omega) := \sum_{\ell=-\infty}^{\infty} \mathrm{Cov}(\boldsymbol{x}^t, \boldsymbol{x}^{t-\ell}) e^{-j\omega\ell} \in \mathbb{C}^{N \times N},$$

$$\min_{\omega \in [-\pi, \pi)} \lambda_{\min}\left(\mathscr{X}(\omega)\right) \geqslant C_{\mathscr{X}}^2 > 0.$$

(A2) The loss function $v \mapsto \mathcal{L}_{it}(u, v)$ is twice differentiable and strongly convex for all $u$, with curvature $\kappa_i > 0$, i.e., $\partial_v^2 \mathcal{L}_{it}(u; v) \geqslant \kappa_i$ for all $u \in \mathcal{X}_i, v \in \mathbb{R}, i \in [N], t \in \mathbb{N}_+$.

(A3) $|\partial_v \mathcal{L}_{it}(u, v)| \leqslant C_{\mathcal{L}}$, and for all $v \in \mathbb{R}$, $i \in [N], t \in \mathbb{N}_+$ we have

$$U \sim \mathbb{Q}_i(\,\cdot \mid f_i(v)) \implies \mathbb{E}\big[\partial_v \mathcal{L}_{it}(U; v)\big] = 0.$$

Assumption (A3) guarantees that $\Theta^*$ is the minimizer of the population loss, and is necessary for the consistency of the $M$-estimator. The second half of the assumption is generally satisfied if the loss is taken to be the log-likelihood function. The next example verifies this for single-parameter exponential families.

**Example 1.** Assume that $Q_i(\,\cdot \mid z)$ is an exponential family with density $x \mapsto \exp(xz - \phi(z))$, for all $i$. Here, $z$ is the so-called natural parameter of the family and $\phi$ is the log-partion function. Let $U \sim Q(\,\cdot \mid f_i(v))$ and take $\mathcal{L}_{it}(x, v)$ to be the log-likelihood of this model, that is,

$$\mathcal{L}_{it}(x; v) = -x f_i(v) + \phi(f_i(v)).$$

This class includes Bernoulli, Poisson, and Gaussian (with known variance) AR processes among others. We have

$$\partial_v \mathcal{L}_{it}(U; v) = -U f_i'(v) + \phi'(f_i(v)) f_i'(v).$$

By a standard property of the exponential family $\mathbb{E}[U] = \phi'(f_i(v))$, hence $\mathbb{E}[\partial_v \mathcal{L}_{it}(U; v)] = 0$ verifying the second half of (A3). If, in addition, the family has bounded support and both $\phi$ and $f_i$ are Lipschitz, then the entire (A3) holds. Distributions such as Poisson and Gaussian violate the boundedness assumption. However, the truncated version of these distributions belong to the exponential family and satisfy the boundedness condition.

**Example 2.** Under the same exponential family distribution as in Example 1, the second half of (A3) also holds for the squared error loss

$$\mathcal{L}_{it}(x; v) = \big[x - \phi'(f_i(v))\big]^2.$$

To verify this, it is enough to observe that

$$\partial \mathcal{L}_{it}(U; v) = 2\big[U - \phi'(f_i(v))\big] \cdot \phi''(f_i(v)) f_i'(v),$$

and use $\mathbb{E}[U] = \phi'(f_i(v))$.

These two examples show that (A3) is satisfied for commonly used loss functions. As for (A2), we recall that in an exponential family with the natural parameterization, the log-partition function $\phi(\cdot)$ is convex. Assumption (A2), however, requires the map $v \mapsto \mathcal{L}_{it}(u, v)$ to be strongly convex. Extra care should be taken in choosing the loss and $f_i(\cdot)$ to ensure that this assumption is satisfied. The stability assumption (A1) is further discussed in the remarks following the main result.

Let us now define a few constants necessary to state our main result. Let

$$\begin{aligned} C_{\mathbf{D}} &:= \max_{\ell} \|\boldsymbol{d}_\ell\|_1, \\ G = G(\Theta^*) &:= 64 C_{\mathbf{D}}^4 B^4 \Big(1 + p^2 \psi\big(\tau_1(\Theta^*)\big)\Big), \end{aligned} \tag{5.10}$$

where $\psi(x) = (1 - x^{-1})^{-2}$ and

$$\begin{aligned} \tau_1(\Theta^*) &:= \sup_{\mathbf{z},\boldsymbol{y} \in \mathcal{X}^{\times p}} \|\mathbb{P}_{\mathbf{z}} - \mathbb{P}_{\boldsymbol{y}}\|_{\mathrm{TV}} < 1, \\ \mathbb{P}_{\mathbf{z}} &:= \mathbb{P}(\mathbf{X}^{t+p} = \cdot \mid \mathbf{X}^t = \mathbf{z}), \quad \mathbf{z} \in \mathcal{X}^{\times p}. \end{aligned} \tag{5.11}$$

Here, $\mathcal{X}^{\times p} \subset \mathbb{R}^{N \times p}$ denotes the set of matrices consisting of $p$ columns, each from $\mathcal{X}$. Note that $\mathbb{P}_{\mathbf{z}}$ is $t$-invariant. Fix $\mathcal{U} \subset [N]$ and let us write

$$\begin{aligned} s_{\max} &:= \max_{i \in \mathcal{U}} s_i, \quad s_+ := \sum_{i \in \mathcal{U}} s_i, \quad \overline{\kappa} := \max_{i \in \mathcal{U}} \kappa_i \\ \underline{\kappa} &:= \frac{C_{\mathscr{X}}^2}{8} \min_{i \in \mathcal{U}} \kappa_i, \quad \text{and} \quad \widetilde{\omega}_+ := \sum_{i \in \mathcal{U}} \overline{\kappa} \frac{\omega_i^2}{s_i} + 4\omega_i, \end{aligned} \tag{5.12}$$

where $\kappa_i$ and $C_{\mathscr{X}}$ are specified in (A2) and (A1). We are now ready to state the main result:

**Theorem 3.** *Suppose that $\{\boldsymbol{x}^t\}_{t=-p+1}^n$ are samples from process (5.3), with each $\mathcal{X}_i$ being a countable subsets of $[-B, B]$ for some $B > 0$, and satisfying (A1). Fix a subset $\mathcal{U} \subseteq [N]$ and let $\{\widehat{\Theta}_i\}_{i \in \mathcal{U}}$ be the solutions of (5.7) with loss functions $\mathcal{L}_{it}$ satisfying (A2)-(A3). Fix $c_1 > 2$ and let $c = c_1/2 - 1$. If*

$$\begin{aligned} &\lambda_n = 2BC_{\mathcal{L}} C_{\mathbf{D}} \sqrt{c_1 \log(|\mathcal{U}|NL)/n}, \quad \textit{and} \\ &n \gtrsim \frac{G}{C_{\mathscr{X}}^6} s_{\max}^3 \log(NL), \end{aligned}$$

*then, with probability at least $1 - (NL)^{-Cs_{\max}} - (|\mathcal{U}|NL)^{-c}$,*

$$\sum_{i \in \mathcal{U}} \|\widehat{\Theta}_i - \Theta_i^*\|_F^2 \;\leqslant\; \frac{9}{\underline{\kappa}^2} s_+ \lambda_n^2 + \frac{\widetilde{\omega}_+}{\underline{\kappa}} \lambda_n. \tag{5.13}$$

*where $C = O(C_{\mathscr{X}}^{-2})$ only depends on $C_{\mathscr{X}}$.*

The error bound in (5.13) can be written, up to constants, as:

$$\sum_{i \in \mathcal{U}} \|\widehat{\Theta}_i - \Theta_i^*\|_F^2 \lesssim \frac{s_+ \log(NL)}{n} + \widetilde{\omega}_+ \sqrt{\frac{\log(NL)}{n}}. \tag{5.14}$$

The two terms in the bound correspond to the estimation and approximation errors, respectively. The estimation error scales at the so-called *fast rate* $\log(NL)/n$, while the approximation error scales at the slower rate $\sqrt{\log(NL)/n}$. For the exact sparsity model, where $\omega_i = 0$ for all $i$, the approximation error vanishes and the estimator achieves the fast rate. For simplicity, assume that $C_{\mathcal{L}}, C_{\mathbf{D}} \lesssim 1 \lesssim C_{\mathscr{X}}$. Then, the overall (excess) sample

complexity for consistent estimation is

$$n \gg \max\left\{Gs_{\max}^3,\ s_+,\ (\widetilde{\omega}_+)^2\right\}\log(NL). \tag{5.15}$$

By consistency, we mean that the estimator converges to the true parameter when $n$ grows to infinity, as long as the above condition holds, even when the rest of the parameters $s, p, L$ and $N$ grow to infinity alongside $n$. We discuss the meaning of the "excess" qualification for the sample complexity in the remarks below.

Bound (5.14) has a logarithmic dependence on $N$, the number of variables in the process, which is a notable feature of our work. Compared to some of the previous work [Kazemipour et al., 2017], we overcome the $N > 1$ barrier for the BAR model while allowing for $p > 1$ dependence on the past. The bound also depends logarithmically on $L$. This means that dictionary $\mathbf{D}$ can be overcomplete, allowing for $\Theta^*$ to be sparse, for nearly no additional cost.

### 5.3.1 Remarks on Theorem 3

Let us make a few comments on the various choices in Theorem 3:

**Choice of the loss $\mathcal{L}$** Theorem 3 holds for any loss function satisfying conditions (A2) and (A3). For the Bernoulli AR process, the negative log-likelihood $\mathcal{L}_{i,t}(u, v) = -u \log f_i(v) - (1-u)\log(1 - f_i(v))$ satisfies these assumptions for any log-concave $f_i$; see [Pandit et al., 2019a]. For the Truncated-Poisson AR process, the negative log-likelihood takes the form $\mathcal{L}_{it}(u, v) = f_i(v) - u \log f_i(v) + \log(u!)$ and satisfies the assumptions for $f_i(v) = \exp(v)$ or $f_i(v) = \log(1 + e^v)$.

**Choice of $\mathcal{U}$** The result in Theorem 3 has been stated for a general $\mathcal{U} \subseteq [N]$. Taking $\mathcal{U} = [N]$, gives a bound on the Frobenius norm of the entire tensor $\left\|\widehat{\Theta} - \Theta^*\right\|_{\mathrm{F}}^2$. On the other extreme, we can take $\mathcal{U} = \{i\}$ to obtain bounds on each slice of the tensor with better scaling with sparsity. For example, in the exact sparsity setting, we obtain $\|\widehat{\Theta}_i - \Theta_i^*\|_F^2 \lesssim s_i \log(NL)/n$,

avoiding the extra price of $(\sum_{j\neq i} s_j)\log(NL)/n$ that we pay for the entire tensor.

**Scaling with sparsity** Considering the exact sparsity setting, the scaling of the sample complexity (5.15) with sparsity is $n = \Omega(s_+ \vee s_{\max}^3)$. In the worst case, $s_+ = s_{\max}$ and we get a cubic dependence on sparsity which is not ideal. However, when $s_+ \gtrsim s_{\max}^3$, Theorem 3 requires $n = \Omega(s_+)$ which is the optimal scaling with sparsity. (This can be seen by noting that in the linear independent setting, one cannot do better than $n = \Omega(s_+)$.) Our result also holds for the more general case of $\omega_i \neq 0$. For example, for the $\ell_q$ ball sparsity with $q \in (0,1)$, we have $\omega_i = O(s_i^{1-1/q})$ hence $\omega_i^2/s_i + w_i = O(\omega_i) = O(s_i)$ and $\widetilde{\omega}_+ = O(s_+)$ and the same sample complexity as the exact sparsity case holds.

It is not clear if the worst-case cubic dependence on the sparsity can be improved without imposing restrictive assumptions. It is worth noting that in our proof, the additional $s_i^2$ factor comes from concentration inequality (5.33) in Lemma 11. This additional factor can be removed if one were able to show sub-Gaussian concentration for deviations of the order of $\|\boldsymbol{\beta}\|_F^2$ instead of $\|\boldsymbol{\beta}\|_{1,1}^2$, in Lemma 11. It remains open whether such concentration is possible and under what additional assumptions. Section 5.7 provides a more detailed discussion on this concentration inequality. Figure 5.4a in Section 5.4 suggests a superlinear dependence on $s$, hinting that the situation may not be as simple as the i.i.d. case.

For $p = 1$, a sample complexity of $\rho^3 \log(N)$ was reported in [Hall et al., 2018, Cor. 1]. One can verify that $\rho$ in their model is equal to $s_{\max}$ in ours, hence they obtain the same $s_{\max}^3$ dependence on sparsity. Similarly for $p = 2$, the result in [Mark et al., 2018, Thm 4.4] requires $(s/r_\rho^2)\log(N)$ samples where $s$ and $r_\rho$ are sparsity parameters defined therein and $r_\rho$ is inversely related to $s_{\max}$ in the worst case, yielding a similar cubic dependence on sparsity as ours. Furthermore, it appears that their analysis only holds for $s_{\max} = \mathcal{O}(1)$, whereas we make no such assumption. In short, to our knowledge, no prior work has broken the $s_{\max}^3$ barrier in the non-Gaussian AR setting.

**Scaling with lag $p$** Our result is the first to provide sufficient conditions for a sample complexity logarithmic in $p$ in the case of the identity dictionary, for any value of $N$. As will be discussed in Section 5.3.2, the dependence of the (excess) sample size $n$ on $p$ could be as good as $O(\log L)$ for a general dictionary, under certain tail and normalization conditions. In these cases, one could obtain an $O(1)$ growth of $n$ as function of $p$ in the best case (when $L = O(1)$) and an $O(\log p)$ growth in the worse case (the identity dictionary). In contrast, [Kazemipour et al., 2017, Thm. 1] requires $s^{2/3}p^{2/3}\log(p)$ samples, for the identity dictionary, and their proof relies heavily on $N = 1$.

Our bound scales with $p$ through $G$ which is defined in terms of the contraction coefficient $\tau_1(\Theta^*)$ in (5.11). The contraction coefficient only depends on $\Theta^*$ and is always less than 1. Intuitively, if $\Theta^*$ is too large, then for two different initializations $\mathbf{z}$ and $\boldsymbol{y}$, the distributions $\mathbb{P}(\mathbf{X}^{t+p} = \cdot \mid \mathbf{X}^t = \boldsymbol{y})$ and $\mathbb{P}(\mathbf{X}^{t+p} = \cdot \mid \mathbf{X}^t = \mathbf{z})$ may significantly differ. A clear sufficient condition for $G = O(1)$ is to have $\tau_1(\Theta^*) = O(p^{-1})$ as well as $C_{\mathbf{D}} \lesssim 1$. The challenge is to control $\tau_1(\Theta^*)$ in terms of the size of $\Theta^*$. Section 5.3.2 further discusses sufficient conditions under which $G = O(1)$. There, we show that for certain exponential families, the scaling depends on the behavior of the tail of $k \mapsto |(\boldsymbol{d}_\ell)_k|$, that is, how fast the *influence from the past* dies down in the filters $\{\boldsymbol{d}_\ell\}$.

A subtle point worth noting here, which does not arise in ordinary $M$-estimation with i.i.d. measurements, is that $n$ is in fact the *excess* sample-size one needs beyond the $p$ initial samples. It is clear that at least $p$ initial samples are needed for estimating a $p$-lag process. Examples discussed in Section 5.3.2 provide conditions that guarantee that the excess sample size, $n$, needed for consistent estimation is $O(\log L)$ as $p$ grows, the smallest order one could hope for.

**Stability assumption (A1)** We use assumption (A1) to guarantee that the strong convexity holds for the population loss $\Theta \mapsto \mathbb{E}\,\mathcal{L}(\Theta)$. This is key in guaranteeing that any parameter tensor $\widehat{\Theta}$ that maximizes the regularized loss function in (5.5) does not deviate far from the

true parameter $\Theta^*$.

Assumption (A1) is by now standard in time-series estimation literature [Raskutti et al., 2019, Basu et al., 2015, Lütkepohl, 2005]. The quantity $C_{\mathscr{X}}$ is fundamental to multivariate time-series analysis, however, its behavior as a function of the parameters of the model is not yet fully understood. Intuitively, $C_{\mathscr{X}}$ is related to the *flatness* of the power spectral density (PSD) $\mathscr{X}$, and the stability of the process. For the $N = 1$ case, $C_{\mathscr{X}} > 0$ implies that the process does not have zeros on the unit circle in the spectral domain.

In general, $C_{\mathscr{X}}$ could potentially depend on $N$, indirectly via $\Theta^*$. In subsequent discussions of Theorem 3, we have assumed that $C_{\mathscr{X}}$ stays uniformly bounded away from zero as $N$ grows. This assumption is explicitly stated as $C_{\mathscr{X}} \gtrsim 1$. Our main result (Theorem 3), however, holds for all positive values of $C_{\mathscr{X}}$, regardless of its growth rate. Even if $C_{\mathscr{X}} = o(1)$ with respect to $N$, Theorem 3 still gives a consistency result, albeit with a worse dependence on $N$.

The dependence of $C_{\mathscr{X}}$ on $N$ occurs through the scaling of the true parameter $\Theta^*$. That $C_{\mathscr{X}}$ is in general bounded below by a constant (or has a slow decay as a function of $N$) is part of the folklore of the time series literature. It is reasonable to assume that this holds for certain structured $\Theta^*$. However, obtaining exact conditions on $\Theta^*$ for $C_{\mathscr{X}} \gtrsim 1$ to hold is, in general, a non-trivial open problem, even for univariate Gaussian AR($p$) processes. The main difficulty is that the relation between the power spectral density of the process and its parameter is indirect and via the Z-transform. Nevertheless, conditions are known in special cases. See for example the discussion surrounding Proposition 2.2 in [Basu et al., 2015], where explicit conditions are given on the parameter matrix of a VAR(1) Gaussian process, for $C_{\mathscr{X}}$ to stay bounded away from zero.

### 5.3.2 Special Cases

Let us now look at the applications of Theorem 3 to two special cases often considered in discrete-valued time series modeling — Binomial and Poisson AR processes. We take $\mathcal{U} = [N]$

throughout this section. To apply the theorem, we need to upper-bound $G(\Theta^*)$ in each case. Since the $\psi$ function in (5.10) is non-decreasing on $[0, 1)$, it is enough to control $\tau_1(\Theta^*)$. In fact, a sufficient condition for $G(\Theta^*) = O(1)$ is to have $\tau_1(\Theta^*) = O(\frac{1}{p})$ and $C_{\mathbf{D}} = O(1)$.

The quantity $\tau_1(\Theta^*)$ is the maximum total variation distance between the $p$-step conditional distributions of the process, starting from two initial states $\boldsymbol{y}$ and $\mathbf{z}$. The Pinsker's inequality [Csiszar and Körner, 2011, p. 44] can be used to further control the total variation distance by the KL divergence, which is the natural choice for comparing two exponential family distributions with independent coordinates.

Recall $\mathcal{X} = \prod_{i=1}^{N} \mathcal{X}_i \subset [-B, B]^N$ and the notation $\mathbb{P}_{\mathbf{z}}$ from (5.11). Pinsker's inequality yields

$$\tau_1^2(\Theta^*) \leqslant \sup_{\mathbf{z},\boldsymbol{y} \in \mathcal{X}^{\times p}} \tfrac{1}{2} D_{\mathrm{KL}}(\mathbb{P}_{\mathbf{z}} \| \mathbb{P}_{\boldsymbol{y}}), \tag{5.16}$$

where $D_{\mathrm{KL}}(\cdot\|\cdot)$ is the KL-divergence. We now state upper bounds on $D_{\mathrm{KL}}(\mathbb{P}_{\mathbf{z}} \| \mathbb{P}_{\boldsymbol{y}})$ for the two cases of the Binomial and Poisson processes. A quantity of interest is the tail decay of the dictionary elements $\{\boldsymbol{d}_\ell\}_{\ell=1}^{L}$, measured by

$$\gamma_{t\ell} := \sum_{m=t}^{p} |(\boldsymbol{d}_\ell)_m|. \tag{5.17}$$

Let us define the following norm on $\Theta$,

$$\|\Theta\|_\star := \Big( \sum_{i,t} L_i^2 \Big[ \sum_{j,\ell} \gamma_{t\ell} |\Theta_{ij\ell}| \Big]^2 \Big)^{1/2}$$

where $L_i$ is the Lipschitz constant of the link function $f_i$, and the summations run over $(i, t, j, \ell) \in [N] \times [p] \times [N] \times [L]$. One can often establish a bound of the form

$$D_{\mathrm{KL}}(\mathbb{P}_{\mathbf{z}} \| \mathbb{P}_{\boldsymbol{y}}) \leqslant C_f B^2 \|\Theta^*\|_\star^2 \tag{5.18}$$

where $C_f$ depends on $\{f_i\}$ and $\Theta^*$ is the true parameter generating the samples.

**Lemma 7.** *Consider a Binomial AR process given by* (5.3) *with* $\mathcal{X}_i = \{0, 1, \ldots, K_i\}$*, where* $K_i \leqslant B$*, and* $\mathbb{Q}_i(\,\cdot \mid z) = \mathrm{Bin}(K_i, z)$*. Assume that* $f_i$ *is* $L_i$*-Lipschitz, and for some* $\varepsilon \in (0, \frac{1}{2})$*,* $f_i : \mathbb{R} \to [\varepsilon, 1 - \varepsilon]$ *for all* $i$*. Then,* (5.18) *holds with* $C_f = 6/\varepsilon$*.*

The case of $B = 1$ recovers the result for the Bernoulli Autoregressive Process in [Pandit

et al., 2019a].

**Lemma 8.** *Consider a Truncated Poisson AR process given by* (5.3) *with* $\mathcal{X}_i = \{0, 1, \ldots, K_i\}$ *and* $\mathbb{Q}_i(\,\cdot\,|\,z) = \mathbb{P}(\min(K_i, Z) \in \cdot\,)$ *where* $Z \sim \mathrm{Poi}(z)$ *and* $K_i \leqslant B$. *Assume that* $f_i$ *is* $L_i$*-Lipschitz, and for some* $\varepsilon > 0$, $f_i : \mathbb{R} \to [\varepsilon, \infty)$ *for all* $i$. *Then,* (5.18) *holds with* $C_f = 4/\varepsilon$.

Combining with (5.16), we have the following corollary.

**Corollary 2.** *Under the assumptions of Lemma 7 or 8,*

$$\tau_1(\Theta^*) \lesssim \frac{B}{\sqrt{\varepsilon}} \left\|\Theta^*\right\|_\star .$$

*In particular, if* $C_{\mathcal{L}}, C_{\mathbf{D}} \lesssim 1 \lesssim C_{\mathscr{X}}$ *and* $\left\|\Theta^*\right\|_\star = O(1/p)$, *then* $G = O(1)$ *and the following is sufficient for consistency:*

$$n \gg \max\left\{s_{\max}^3,\ s_+,\ (\widetilde{\omega}_+)^2\right\} \log(NL).$$

In other words, Corollary 2 provides conditions under which consistent estimation is possible with (excess) sample complexity that grows at most logarithmically in $L$.

Let us consider some examples for which $\left\|\Theta^*\right\|_\star = O(1/p)$. For the purpose of illustration, let us separate the tail decay of $\Theta^*$, along the lag dimension, by assuming that

$$|\Theta^*_{ij\ell}| \leqslant R_{ij} h_\ell, \quad \forall\, (i, j, \ell) \in [N] \times [N] \times [L].$$

for some sequence $\{h_\ell\}_{\ell=1}^\infty$ such that $\sum_{\ell=1}^\infty h_\ell < \infty$ and a matrix $R = (R_{ij})$. Assume that $\Theta^*_{ij\ell}$ is normalized so that $\|R\|_{2,1} = O(1)$. Moreover, assume that $\max_i L_i = O(1/p)$. Since in model (5.3), the input to each $f_i$ involves terms $\langle \mathbf{x}_j^{t-*}, \boldsymbol{d}_\ell \rangle_{\mathbb{R}^p}$, each of which is essentially a sum of $p$ terms (cf. (5.4)), the aforementioned assumption on the Lipschitz constant is a natural normalization that prevents the saturation of the nonlinearities $f_i$ as $p$ grows. Equivalently, we can make this condition more explicit by replacing $f_i(\cdot)$ in the definition of model (5.3) with $\widetilde{f}_i(\frac{1}{p}\cdot)$ and assuming that $\widetilde{f}_i$ have Lipschitz constants uniformly bounded by a constant.

Under the above modeling assumptions, consider the following two dictionaries:

*Case (a):* The identity dictionary, where $L = p$ and $(\boldsymbol{d}_\ell)_m = 1\{m = \ell\}$. In this case,

$\gamma_{t\ell} = 1\{t \leqslant \ell\}$. Then,

$$\|\Theta\|_\star \lesssim \frac{1}{p}\|R\|_{2,1}\Big[\sum_{t=1}^{p}\Big(\sum_{\ell=t}^{p} h_\ell\Big)^2\Big]^{1/2} = O\Big(\frac{1}{p}\Big)$$

assuming that $\sum_{t=1}^{\infty}(\sum_{\ell=t}^{\infty} h_\ell)^2 < \infty$ which holds, for example, if $h_\ell$ decays at least as fast as $\ell^{-1-\alpha/2}$ for some $\alpha > 1$. Note that in this case $C_{\mathbf{D}} \asymp 1$ is trivially satisfied.

*Case (b):* A general dictionary, with filters satisfying the decay rate $\max_\ell |(\boldsymbol{d}_\ell)_m| \lesssim m^{-\alpha-1}$ for some $\alpha > 1$. Then, $\max_\ell \gamma_{t\ell} \lesssim t^{-\alpha}$ and

$$\|\Theta\|_\star \lesssim \frac{1}{p}\|R\|_{2,1}\Big(\sum_{t=1}^{p} t^{-2\alpha}\Big)^{1/2}\sum_{\ell=1}^{p} h_\ell = O\Big(\frac{1}{p}\Big)$$

using $\sum_{t=1}^{\infty} t^{-2\alpha} < \infty$ and $\sum_{\ell=1}^{\infty} h_\ell < \infty$. Moreover, since we have $C_{\mathbf{D}} \lesssim \sum_{m=1}^{p} m^{-\alpha-1}$, it follows that $C_{\mathbf{D}} = O(1)$ as $p$ grows.

Thus in both cases, Corollary 2 guarantees that the excess sample size $n$ needed for consistency grows at most logarithmically in $L$. This translates to an $O(\log p)$ growth in the case the identity dictionary but could be as low as $O(1)$ for a dictionary with the number of filters $L$ not growing with $p$. Note that the summability condition on $h_\ell$ in case (b) is milder than that in case (a), showing the trade-off between the tail decay of $\Theta$ (along the lag dimension) and the tail decay of the dictionary filters. Having fast decaying filters relaxes the decay requirement on the tails of $\Theta$.

## 5.4 Simulations

In this section, we evaluate the performance of the estimator in (5.5) using simulated data. We generate the data using the model in (5.3). In all the examples, we first randomly generate $\Theta^*$ and $\mathbf{D}$. To generate $\Theta^*$, we select the support of $\Theta_i^*$ for each $i$ uniformly at random based on the sparsity $s_i$. We then fill the support with i.i.d. draws of the normal distribution, and finally normalize such that $\|\Theta_i^*\|_{1,1}$ is a constant.

To report the performance of (5.5), we use the metric normalized squared error (NSE)

defined as:

$$\mathrm{NSE}(\Theta^*, \widehat{\Theta}) = \frac{\|\Theta^* - \widehat{\Theta}\|_F^2}{\|\Theta^*\|_F^2}. \tag{5.19}$$

to normalize variations in the size of the parameter across independent instances of $\Theta^*$. An implementation is provided at [Sahraee-Ardakan et al., 2020]. We consider the following 3 processes:

### 5.4.1 Poisson AR($p$) process without dictionary

We evaluate the performance of the regularized maximum likelihood and the regularized least-squares estimators on a Poisson process with no dictionary, i.e., $\mathbf{D} = \boldsymbol{I}_p$. For the Poisson process, we use the inverse link function $f_i(z) = \log(1 + e^z)$. Then, these estimators have the form of (5.5) with

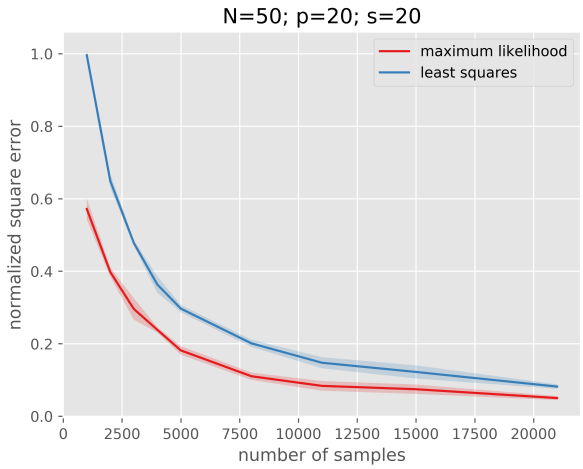
$$\mathcal{L}_{it}^{\mathrm{ML}}\left(x_i^t\,;\, z_i^t\right) = z_i^t - x_i^t \log(z_i^t), \tag{5.20a}$$

$$\mathcal{L}_{it}^{\mathrm{LS}}\left(x_i^t\,;\, z_i^t\right) = (x_i^t - z_i^t)^2, \tag{5.20b}$$

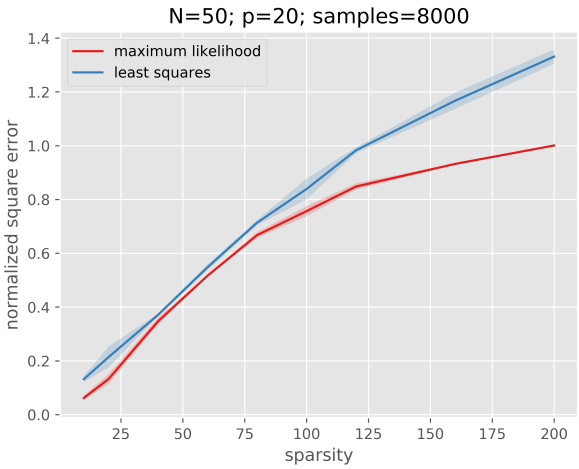
where $z_i^t = f(\langle \Theta_i^*, \mathbf{X}^{t-1} \rangle)$, since $\mathbf{D} = \boldsymbol{I}_p$. Note that the M-estimation problem in (5.5) corresponding to (5.20a) is convex, whereas it is non-convex for (5.20b) (we report a local minimum). Here, we generate the ground truth parameters as mentioned before with $N = 50$ and $p = 20$ and we use $\lambda_n = 0.05/\sqrt{n}$. When comparing $NSE$ v/s $n$, each $\Theta_i$ has sparsity 20. The results are shown in Figure 5.1. The error shades correspond to one standard deviation over 5 independent instances of $(\Theta^*, \widehat{\Theta})$. With the NSE metric, the regularized maximum likelihood estimator appears to perform better for the Poisson AR($p$) process, for the random ensemble of problems generated in these examples.

### 5.4.2 Poisson AR($p$) process with dictionary

We choose $\mathbf{D}$ to be entrywise i.i.d. Gaussian with standard deviation $\sigma/p$ for a constant $\sigma$, so that the $\ell_1$-norm of all columns of $\mathbf{D}$ are close to a constant for large $p$ (the constant

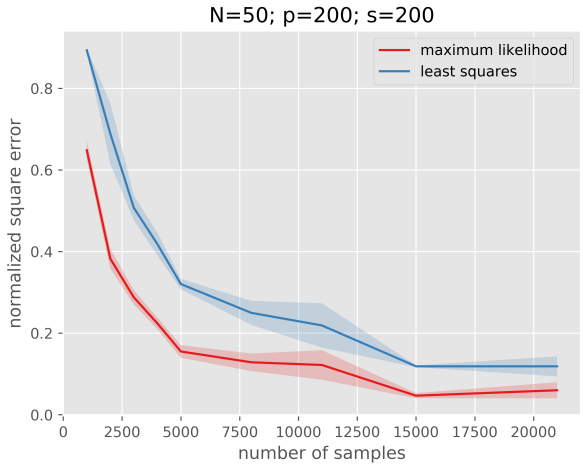



**(a)** NSE vs. sample size for a Poisson process without dictionary.

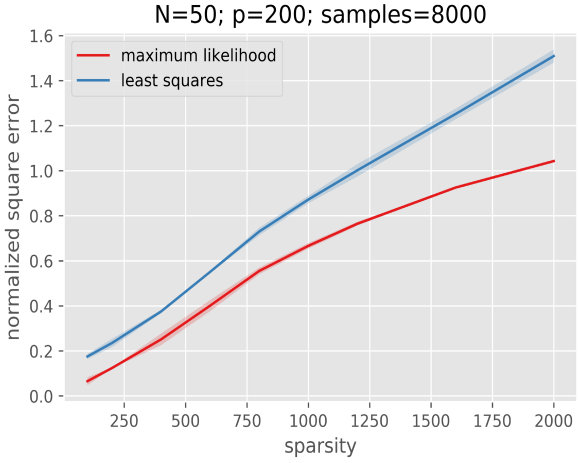



**(b)** NSE vs. sparsity for a Poisson process without dictionary.

**Figure 5.1:** Poisson AR($p$) process without a dictionary (i.e., $\mathbf{D} = \boldsymbol{I}_p$).




**(a)** NSE vs. sample size for a Poisson process with dictionary.




**(b)** NSE vs. sparsity for a Poisson process with dictionary.

**Figure 5.2:** Poisson AR($p$) process with dictionary of size $L = 20$.

being the mean of a folded normal distribution). The process is generated as in the previous example using (5.3). We take $N = 50, p = 200$, and $L = 20$ such that the process has very long range dependencies. We again consider the two regularized M-estimators: the regularized maximum likelihood and the regularized least-squares with the inverse link function $f(z) = \log(1 + e^z)$. These estimators are identical to the ones in (5.20a) and (5.20b), except that $z_i^t = f(\langle \Theta_i, \mathbf{X}^{t-1}\mathbf{D} \rangle)$ with $\mathbf{D} \neq \boldsymbol{I}_p$.

The results are shown in Figure 5.2. They are very similar to Figure 5.1. In accordance with our theoretical results, these figures suggest that for an AR processes with very long range dependencies, estimating the parameter is easier in the presence of a dictionary.

### 5.4.3 Bernoulli AR($p$) process without dictionary

Finally, we look at a Bernoulli autoregressive process. We use the sigmoid function, $f(z) = 1/(1+e^{-z})$, as the inverse link function. We compare the performance of regularized maximum likelihood estimator to regularized least-squares estimator. Both of these estimators have the form of (5.5) with

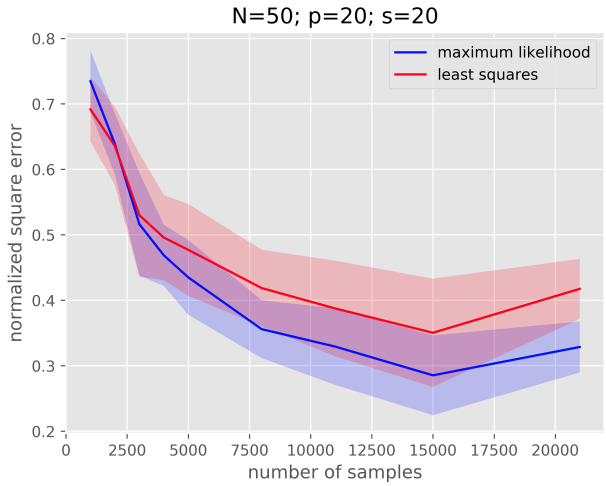
$$\mathcal{L}_{it}^{\mathrm{ML}}\left(x_i^t\,;\,z_i^t\right) = -z_i^t \log(x_i^t) - (1-x_i^t)\log(1-z_i^t) \tag{5.21a}$$

$$\mathcal{L}_{it}^{\mathrm{LS}}\left(x_i^t\,;\,z_i^t\right) = (x_i^t - z_i^t)^2, \tag{5.21b}$$

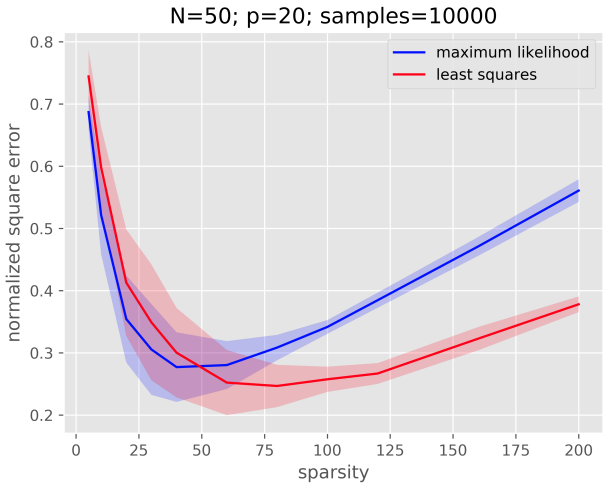
where $z_i^t = f(\langle \Theta_i, \mathbf{X}^{t-1}\rangle)$ is the mean parameter of the dimension $i$ of the Bernoulli process at time $t$. Note that due to inverse link function, despite convexity of square loss with respect to $z_i^t$, the optimization problem corresponding to least square estimator is non-convex and our results do not apply to it. Nevertheless, we observe that its performance is similar to maximum likelihood estimator.

Figure 5.3 shows different measures of performance of the regularized maximum likelihood estimator. We have set $N = 50, p = 20$ and $\lambda_n = 0.05/\sqrt{n}$ as recommended by Theorem 3, in these examples. Figure 5.3a shows how the normalized estimation error changes with respect to the number of training samples.

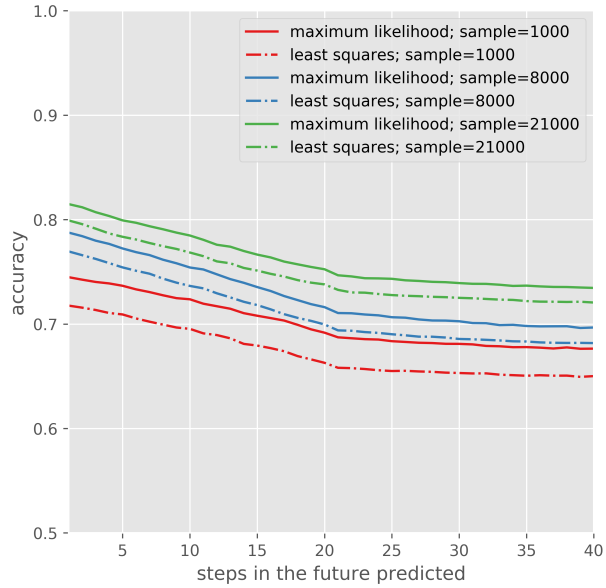
The sparsity is 20 for each $\Theta_i$. Note that we are using the same regularization parameter for both estimators and not the optimal $\lambda_n$, i.e.without any cross-validation. The error shades correspond to one standard deviation. Figure 5.3b shows the normalized square error for different sparsity levels. For small values of sparsity, the denominator $\Theta^*$ has a small norm which causes high normalized error, however for higher values of sparsity, we see the linear dependence on sparsity as predicted by Theorem 3.



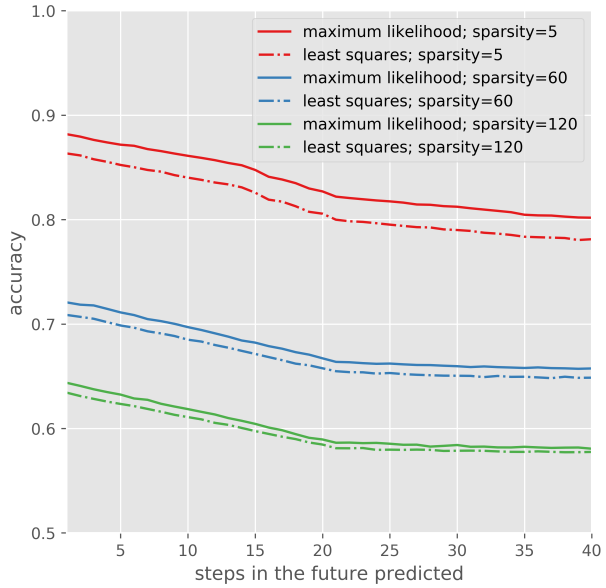

**(a)** NSE vs. sample size for sparsity $s_i = 20$ for all $i$.




**(b)** NSE vs. sparsity for sample size $n = 10,000$



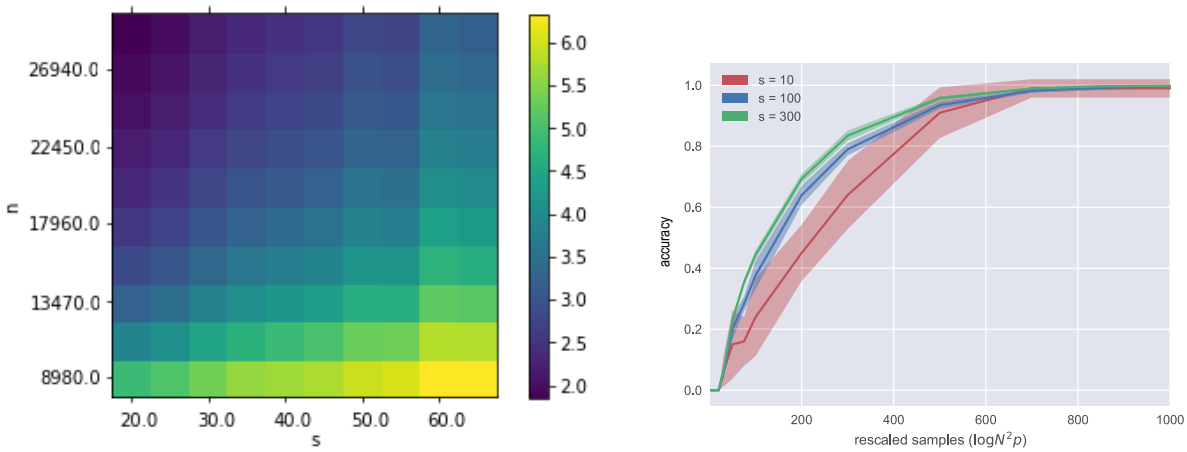

**(c)** Accuracy vs. steps predicted in the future for different $n$.




**(d)** Accuracy vs. steps predicted in the future for different $s$.

**Figure 5.3:** Bernoulli AR($p$) process without dictionary.

The next two figures correspond to generalization error as opposed to estimation error in the first two figures. Here, we use the estimated parameters $\widehat{\Theta}$ to predict the process in the future and calculate the accuracy of prediction. We use 5 MCMC runs of the process to estimate the accuracy. The plot shows average accuracy over all $N$ variables of the process. Figure 5.3c shows the accuracy vs. steps in the future for different training sample sizes and




**(a)** Average Frobenius norm of the error over 20 runs with $N = 20$, $p = 20$. Each pixel corresponds to a pair $(s, n)$ for $\Theta^*$.

**(b)** Fraction of support recovered by taking the largest $s$ entries of $\widehat{\Theta}$ as the estimator of support. Here $N = 100$, $p = 1$.

**Figure 5.4:** Simulation results for Bernoulli AR($p$) process.

Figure 5.3d shows it for different levels of sparsity. There is a prominent change in in the accuracy plots at 21 steps. This corresponds to $p = 20$ where the future of the process is being estimated purely based on simulated samples using the estimated parameter. Prior to this point, parts of the samples being used to make the predictions are True values and not estimated ones. As expected, the accuracies improve as the number of training samples increase with sparsity fixed, and they decrease as sparsity level increases with number of training samples fixed. Figure 5.4a shows the estimation error for different sample sizes and sparsity levels.

Finally, we also use the regularized maximum likelihood estimator to perform support recovery, i.e. assuming that the true parameter tensor is exactly $s$-sparse, how does the support estimated from $\widehat{\Theta}$ compare to the support of $\Theta^*$? To do so, we need to estimate the support from $\widehat{\Theta}$. If we know the sparsity $s$, we can estimate the support by taking the indices corresponding to the $s$ largest entries of $\widehat{\Theta}$ in magnitude. If we do not know the sparsity in advance, we can estimate the support based on a threshold chosen by cross-validation. Given a threshold $\gamma$, the estimated support would be

$$\widehat{\text{supp}}(\Theta) := \{(j, k, \ell) : |\widehat{\Theta}_{jk\ell}| \geqslant \gamma\}.$$

Note that our theoretical results do not give any guarantees for support recovery. In order to guarantee support recovery, a stronger result bounding the error uniformly for each entry of $\widehat{\Theta}$ is required, i.e., we need to control $\left\|\widehat{\Theta} - \Theta^*\right\|_{\infty,\infty,\infty}$ with high probability. Therefore, more work is needed to obtain theoretical guarantees for support recovery. Nevertheless, our simulations show that the estimator is able to recover the support very well. Figure 5.4b shows the results for a process with $p = 1, N = 100$ and three different sparsities. For recovering the support, we assumed that the sparsity $s$ is known, and took the indices corresponding to the $s$ largest entries of $\widehat{\Theta}$ as the recovered support. The fraction of the correctly recovered indices is plotted against the sample size. Figure 5.4b shows that if the sample size is below some threshold, no entries of the support are recovered, while above the threshold, the recovered fraction gradually increases to 1.

## 5.5 Proof Sketch for Theorem 3

We now outline the proof of Theorem 3. Our analysis applies the framework of Negahban et al. [Negahban et al., 2012]. Let

$$\mathcal{L}_i(\boldsymbol{\beta}) := \frac{1}{n}\sum_{t=1}^{n} \mathcal{L}_{it,}(x_i^t; \langle \boldsymbol{\beta}, \mathbf{X}^{t-1}\mathbf{D}\rangle), \quad \boldsymbol{\beta} \in \mathbb{R}^{N\times L}.$$

Fix $\mathcal{U} \subseteq [N]$ and set $\Theta_{\mathcal{U}} := (\Theta_i)_{i\in\mathcal{U}}$ and similarly $\Theta^*_{\mathcal{U}} := (\Theta^*_i)_{i\in\mathcal{U}}$ and $\widehat{\Theta}_{\mathcal{U}} := (\widehat{\Theta}_i)_{i\in\mathcal{U}}$, all tensors in $\mathbb{R}^{|\mathcal{U}|\times N\times L}$. We also write $\mathcal{L}_{\mathcal{U}}(\Theta_{\mathcal{U}}) = \sum_{i\in\mathcal{U}} \mathcal{L}_i(\Theta_i)$. We have

$$\widehat{\Theta}_{\mathcal{U}} = \underset{\Theta_{\mathcal{U}}\in\mathbb{R}^{|\mathcal{U}|\times N\times L}}{\arg\min}\ \mathcal{L}_{\mathcal{U}}(\Theta_{\mathcal{U}}) + \|\Theta_{\mathcal{U}}\|_{1,1,1}\,. \tag{5.22}$$

In the sequel, $\nabla\mathcal{L}_{\mathcal{U}}$ and $\nabla^2\mathcal{L}_{\mathcal{U}}$ are the gradient and Hessian of $\mathcal{L}_{\mathcal{U}}$ with respect to variable $\Theta_{\mathcal{U}}$. When $n \ll |\mathcal{U}|NL$, the empirical Hessian, $\nabla^2\mathcal{L}_{\mathcal{U}}(\Theta^*_{\mathcal{U}})$, is rank-deficient, hence the loss function is flat in many directions around $\Theta^*_{\mathcal{U}}$. The approach of Negahban et al. [Negahban et al., 2012] is to guarantee that $\mathcal{L}_{\mathcal{U}}$ is positively curved in certain directions, including $\widehat{\Delta}_{\mathcal{U}} := \widehat{\Theta}_{\mathcal{U}} - \Theta^*_{\mathcal{U}}$.

In particular, if the regularization parameter $\lambda_n$ is large enough, specifically

$$\lambda_n \geqslant 2 \left\| \nabla \mathcal{L}_{\mathcal{U}}(\Theta^*_{\mathcal{U}}) \right\|_{\infty,\infty,\infty}, \tag{5.23}$$

then, the error tensor $\widehat{\Delta}_{\mathcal{U}}$ lies in a small *cone-like* subset $\mathscr{C}(\mathscr{S}; \Theta^*_{\mathcal{U}})$—to be defined below—and on this set, $\mathcal{L}_{\mathcal{U}}$ is "nearly" strongly convex, i.e., $\nabla^2 \mathcal{L}_{\mathcal{U}}(\Theta^*_{\mathcal{U}})$ is uniformly quadratically bounded below.

For a set $S \subseteq [N] \times [L]$, let $\boldsymbol{\beta}_S$ denote the projection of $\boldsymbol{\beta}$ on the subspace of matrices with support $S$. For $\boldsymbol{\beta}^*$ define:

$$\mathbb{C}(S; \boldsymbol{\beta}^*) := \{\boldsymbol{\beta} : \|\boldsymbol{\beta}\|_{1,1} \leqslant 3 \|\boldsymbol{\beta}_S\|_{1,1} + 4 \|\boldsymbol{\beta}^*_{S^c}\|_{1,1}\}. \tag{5.24}$$

Note that this is a *cone-like* subset of $\mathbb{R}^{N \times L}$ around $\boldsymbol{\beta}^*$. See [Negahban et al., 2012] for a visualization. Let $\mathscr{S} := \bigcup_{i \in \mathcal{U}} \{i\} \times S_i$ where $S_i \subseteq [N] \times [L]$ for $i \in \mathcal{U}$. Equivalently, $\mathscr{S} = \bigsqcup_{i \in \mathcal{U}} S_i$ using the notation of *disjoint union*. With some abuse of notation, we write $\mathscr{S}^c := \bigcup_{i \in \mathcal{U}} \{i\} \times S_i^c$. The cone-like set $\mathscr{C}(\mathscr{S}; \Theta^*_{\mathcal{U}})$ is defined as follows:

$$\mathscr{C}(\mathscr{S}; \Theta^*_{\mathcal{U}}) := \{(\Delta_i)_{i \in \mathcal{U}} : \Delta_i \in \mathbb{C}(S_i; \Theta^*_i),\ \forall i \in \mathcal{U}\}. \tag{5.25}$$

For loss functions $\mathcal{L}_i$, $i \in \mathcal{U}$, and for $\boldsymbol{\delta}, \boldsymbol{\beta}^* \in \mathbb{R}^{N \times L}$, let

$$R\mathcal{L}_i(\boldsymbol{\delta}; \boldsymbol{\beta}^*) := \mathcal{L}_i(\boldsymbol{\beta}^* + \boldsymbol{\delta}) - \mathcal{L}_i(\boldsymbol{\beta}^*) - \langle \nabla \mathcal{L}_i(\boldsymbol{\beta}^*), \boldsymbol{\delta} \rangle, \tag{5.26}$$

be the remainder of the first-order Taylor expansion of $\mathcal{L}_i$ around $\boldsymbol{\beta}^*$. Following [Negahban et al., 2012], we say that $\mathcal{L}_{\mathcal{U}}$ satisfies restricted strong convexity (RSC) at $\Theta^*_{\mathcal{U}}$ with curvature $\kappa > 0$ and tolerance $\tau^2$ if for all $\Delta \in \mathscr{C}(\mathscr{S}; \Theta^*_{\mathcal{U}})$, we have,

$$\sum_{i \in \mathcal{U}} R\mathcal{L}_i(\Delta_i; \Theta^*_i) \geqslant \kappa \sum_{i \in \mathcal{U}} \|\Delta_i\|_F^2 - \tau^2. \tag{5.27}$$

The left-hand side is the remainder of the first-order Taylor expansion of $\mathcal{L}_{\mathcal{U}}$ around $\Theta^*_{\mathcal{U}}$, that is, $R\mathcal{L}_{\mathcal{U}}(\Delta_{\mathcal{U}}; \Theta^*_{\mathcal{U}})$—defined similar to (5.26).

Now, assume that (5.23) and (5.27) hold. Then, [Negahban et al., 2012, Theorem 1] implies that $\widehat{\Theta}_{\mathcal{U}} - \Theta^*_{\mathcal{U}} \in \mathscr{C}(\mathscr{S}; \Theta^*_{\mathcal{U}})$, and that

$$\|\widehat{\Theta}_{\mathcal{U}} - \Theta^*_{\mathcal{U}}\|_F^2 \leqslant \tfrac{9\lambda_n^2}{\kappa^2} |\mathscr{S}| + \tfrac{\lambda_n}{\kappa} (2\tau^2 + 4 \|(\Theta^*_{\mathcal{U}})_{\mathscr{S}^c}\|_{1,1,1}). \tag{5.28}$$

The above inequality provides a family of bounds, one for each choice of $\mathscr{S} = \bigsqcup_{i \in \mathcal{U}} S_i$. Decreasing $|\mathscr{S}|$ reduces the first term, but potentially increases $\|(\Theta^*_{\mathcal{U}})_{\mathscr{S}^c}\|_{1,1,1}$. We choose $\mathscr{S}$ to balance the two. Let $S_i^* \subset [N] \times [L]$ be the support of the minimizer in (5.9), so that $|S_i^*| = s_i$. We take $\mathscr{S} = \mathscr{S}^* = \bigsqcup_{i \in \mathcal{U}} S_i^*$. Consequently, $|\mathscr{S}^*| = \sum_{i \in \mathcal{U}} s_i$ and $\|(\Theta^*_{\mathcal{U}})_{\mathscr{S}^{*c}}\|_{1,1,1} = \sum_{i \in \mathcal{U}} \omega_i$. For this choice of $\mathscr{S}$, Proposition 5 below shows that (5.27) holds, with high probability. To state the concentration inequality, recall the definitions (5.12).

**Proposition 5.** *Under assumptions (A1) and (A2), if we have,*

$$n \gtrsim \frac{G}{C_{\mathscr{X}}^6} s_{\max}^3 \log(NL) \tag{5.29}$$

*then, the RSC property* (5.27) *for* $\mathscr{S} = \mathscr{S}^*$ *holds with curvature* $\kappa = \underline{\kappa}$ *and tolerance* $\tau^2 = \frac{\overline{\kappa}}{2} \sum_{i \in \mathcal{U}} \omega_i^2 / s_i$*, with probability at least* $1 - (NL)^{-Cs_{\max}}$ *where* $C = O(C_{\mathscr{X}}^{-2})$.

Lemma 20 in Appendix C.1 shows that $\Theta^*_{\mathcal{U}}$ is in fact the minimizer of the expected loss $\mathbb{E}\mathcal{L}_{\mathcal{U}}(\cdot)$. Lemma 21 in Appendix C.1 shows that taking $\lambda_n = O(\sqrt{\log(|\mathcal{U}|NL)/n})$ is enough for (5.23) to hold with high probability. Putting the pieces together proves Theorem 3. The next section sketches a proof of Proposition 5.

## 5.6 Restricted Strong Convexity: Proof of Proposition 5

Showing the RSC property (5.27) for a particular choice of $\mathscr{S}$ is a major contribution of our work. This is a nontrivial task since it involves uniformly controlling a dependent non-Gaussian empirical process. Even for i.i.d. samples, the task is challenging since the quantity to be controlled, $\Delta \mapsto R\mathcal{L}(\Delta; \Theta^*)$, is a *random function* that needs to be uniformly bounded below. Controlling the behavior of this function becomes significantly harder without the independence assumption.

We proceed by a establishing a series of intermediate lemmas which are proved in Appendix

C.1. First, we show that $\boldsymbol{\beta} \mapsto R\mathcal{L}_i(\boldsymbol{\beta}; \Theta_i^*)$ is lower-bounded by the following quadratic form:

$$\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) := \frac{1}{n} \sum_{t=1}^{n} i + \boldsymbol{\beta}, \mathbf{X}^{t-1} \mathbf{D}^2, \tag{5.30}$$

where $\mathbb{X} := \{\boldsymbol{x}^t\}_{t=-p+1}^{n}$.

**Lemma 9** (Quadratic lower bound)**.** *Under assumption (A2),*

$$R\mathcal{L}_i(\boldsymbol{\beta}; \Theta_i^*) \;\geqslant\; \frac{\kappa_i}{2} \mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) \tag{5.31}$$

*for all* $\boldsymbol{\beta} \in \mathbb{R}^{N \times L}$ *and* $i \in [N]$.

Notice that $\boldsymbol{\beta} \mapsto \mathcal{E}(\boldsymbol{\beta}; \mathbb{X})$ is a random function due to the randomness in $\mathbb{X}$. Importantly, $\mathcal{E}(\,\cdot\,; \mathbb{X})$ does not depend on the choice of $i$. The following set of results establish some important properties of the random function $\mathcal{E}(\,\cdot\,; \mathbb{X})$.

**Lemma 10** (Strong convexity at the population level)**.** *Under assumption (A1),*

$$\mathbb{E}\, \mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) \geqslant C_{\mathscr{X}}^2 \; \|\boldsymbol{\beta}\|_F^2 \,, \qquad \text{for all } \boldsymbol{\beta} \in \mathbb{R}^{N \times L}. \tag{5.32}$$

Next, we show that for a fixed $\boldsymbol{\beta}$, the quantity $\mathcal{E}(\boldsymbol{\beta}; \mathbb{X})$ concentrates around its mean. Section 5.7 provides a sketch of the proof of the following concentration inequality:

**Lemma 11** (Concentration inequality)**.** *For any* $\boldsymbol{\beta} \in \mathbb{R}^{N \times L}$*, if* $\mathbb{X}$ *is generated as* (5.3)*, then with probability at least* $1 \;-\; 2\exp\left(-nt^2/G\right)$*, we have*

$$\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) > \mathbb{E}\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) - t \, \|\boldsymbol{\beta}\|_{1,1}^2 \,. \tag{5.33}$$

Finally, for a fixed $i \in [N]$ we use the structural properties of set $\mathbb{C}(S_i^*; \Theta_i^*)$ along with Lemmas 10 and 11 to give a uniform quadratic lower bound on $\mathcal{E}(\boldsymbol{\beta}; \mathbb{X})$, which holds with high probability:

**Lemma 12.** *Fix* $i \in \mathcal{U}$*. For constants* $C_1, C_2 > 0$*, if* $s_i \geqslant \frac{C_{\mathscr{X}}^2}{C_1}$*, then with probability* $\geqslant 1 - \exp(\frac{C_2}{C_{\mathscr{X}}^2} s_i \log(NL) - \frac{nC_{\mathscr{X}}^4}{16Gs_i^2})$*,*

$$\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) \geqslant \tfrac{C_{\mathscr{X}}^2}{4} \, \|\boldsymbol{\beta}\|_F^2 - \omega_i^2/s_i, \quad \forall\, \boldsymbol{\beta} \in \mathbb{C}(S_i^*; \Theta_i^*).$$

The proof of Lemma 12 (cf. Appendix C.2) makes use of a discretization argument. Proving uniform laws are challenging when the parameter space is not finite. The discretization of the set $\mathbb{C}(S^*;\Theta^*)$ uses estimates of the *entropy numbers* for absolute convex hulls of collections of points (Lemma 22). These estimates are well-known in approximation theory and have been previously adapted to the analysis of regression problems in [Raskutti et al., 2011]. The following technical lemma allows us to put the above results together:

**Lemma 13.** *For all $i \in \mathcal{U}$, let $a_i, b_i, d_i, p_i$ be positive constants, and consider random variables $X_i, Y_i \in \mathbb{R}$ which satisfy $Y_i \geqslant a_i X_i$, and $\mathbb{P}(X_i < b_i - d_i) \leqslant p_i$ for all $i \in \mathcal{U}$. Then with probability at least $1 - |\mathcal{U}| \max_{i\in\mathcal{U}} p_i$, we have,*

$$\sum_{i\in\mathcal{U}} Y_i > (\min_{i\in\mathcal{U}} a_i) \sum_{i\in\mathcal{U}} b_i - (\max_{i\in\mathcal{U}} a_i) \sum_{i\in\mathcal{U}} d_i$$

Proposition 5 follows by taking $Y_i = R\mathcal{L}_i(\Delta_i;\Theta_i^*)$, $X_i = \mathcal{E}(\Delta_i,\mathbb{X})$, $a_i = \frac{\kappa_i}{2}$, $b_i = \frac{C_{\mathcal{X}}^2}{4}\|\Delta_i\|_F^2$, and $d_i = \omega_i^2/s_i$.

## 5.7 Concentration under dependence: Proof of Lemma 11.

In this section, we sketch the proof of Lemma 11 which is a concentration inequality for $\boldsymbol{\beta} \mapsto \mathcal{E}(\boldsymbol{\beta};\mathbb{X})$, a quadratic empirical process based on dependent non-Gaussian variables with long-term dependence. For independent sub-Gaussian variables $\{\mathbf{X}^{t-1}\}$, such a concentration result is often called the Hanson–Wright inequality [Rudelson et al., 2013, Thm. 1]. Providing similar inequalities for dependent random variables is significantly more challenging. For dependent Gaussian variables, the machinery of the Hanson–Wright inequality can still be adapted to derive the desired result [Basu et al., 2015, Prop. 2.4]. However, these arguments do not extend easily to non-Gaussian dependent variables and hence other techniques are needed to provide such concentration inequalities.

Recent results [Fan et al., 2018, Chung et al., 2012] on the concentration of empirical processes derived from Markov chains could provide improvements on the results we present

here. However, since we are dealing with a non-Markovian process (when $p > 1$), such results are not directly applicable. A key observation, discussed in Section C.3.2, is that process (5.3) can be represented as a discrete-space *p-Markov chain*. This allows us to use concentration results for dependent processes in countable metric spaces. There are several results for such processes; see [Kontorovich et al., 2008, Marton et al., 1996, Samson et al., 2000] and [Kontorovich, 2012] for a review. Here, we apply that of Kontorovich et. al. [Kontorovich et al., 2008]. These concentration inequalities are stated in terms of various mixing and contraction coefficients of the underlying process. The challenge is to control the contraction coefficients in terms of the process parameter $\Theta^*$, which in our case is done using quantities $\tau_1(\Theta^*)$ and $G(\Theta^*)$. Some results developed in this section hold more generally for any $p-$Markov process, even those outside the current autoregressive framework.

We start by stating the result of Kontorovich et. al. [Kontorovich et al., 2008] for a process $\{X^t\}_{t\in[n]}$ consisting of (possibly dependent) random variables taking values in a countable space $\mathcal{X}$. For any $\ell \geqslant k \geqslant 1$, define the *mixing coefficient*

$$\eta_{k\ell} \stackrel{\Delta}{=} \sup_{w,w',y} \Big\| \mathbb{P}\left(X_\ell^n = \cdot \mid X_k = w', X_1^{k-1} = y\right) - \mathbb{P}\left(X_\ell^n = \cdot \mid X_k = w, X_1^{k-1} = y\right) \Big\|_{\mathrm{TV}}, \tag{5.34}$$

where the supremum is over $w, w' \in \mathcal{X}$ and $y \in \mathcal{X}^{k-1}$. Here, $X_u^v := (X^t, u \leqslant t \leqslant v)$ is viewed either as a member of $\mathcal{X}^{\times(v-u+1)}$ (the set of a matrices with $v-u+1$ columns from $\mathcal{X}$) or simply as a vector in $\mathcal{X}^{v-u+1}$. Let $\mathbf{H} \in \mathbb{R}^{n\times n}$ be an upper triangular matrix with entries $\eta_{k\ell}$ for $\ell \geqslant k$ and zero otherwise. Let $\|\mathbf{H}\|_\infty := \max_k \sum_{\ell\geqslant k} \eta_{k\ell}$ be the $\ell_\infty$ operator norm of $\mathbf{H}$.

**Proposition 6.** *[Kontorovich et al., 2008, Theorem 1.1] Let $\phi : \mathcal{X}^n \to \mathbb{R}$ be an $L_\phi$-Lipschitz function of $\{X^t\}_{t=1}^n$ with respect to the Hamming norm, then for all $\varepsilon > 0$, with probability at least $1 - 2\exp(-\frac{\varepsilon^2}{2nL_\phi^2\|\mathbf{H}\|_\infty^2})$, we have*

$$|\phi(\{X^t\}_{t=1}^n) - \mathbb{E}\phi(\{X^t\}_{t=1}^n)| < \varepsilon. \tag{5.35}$$

We apply the above result to $\phi = \mathcal{E}(\boldsymbol{\beta}; \mathbb{X})$ by finding an upper bound for the Lipschitz constant $L_\phi$ of the map $\mathbb{X} \mapsto \mathcal{E}(\boldsymbol{\beta}, \mathbb{X})$ with respect to the Hamming distance over $\mathcal{X}^{\times(n+p-1)} =$

$(\prod_{i=1}^{N} \mathcal{X}_i)^{\times(n+p-1)}$. Lemma 24 in Appendix C.3 shows that $L_\phi \leqslant (4B^2C_{\mathbf{D}}^2/n)\,\|\boldsymbol{\beta}\|_{1,1}^2$, whereas Lemma 25 in Appendix C.3 shows that $\|\mathbf{H}\|_\infty^2 \leqslant 2(1+p^2\psi_1(\Theta^*))$, where the quantity $\psi_1(\Theta^*)$ is defined below equation (5.10). Lemma 25 is a general result that applies to any $p$-lag Markov chain, including the GVAR($p$) processes considered in this chapter. In Appendix C.3 we also develop some tools for controlling $\|\mathbf{H}\|_\infty$ in terms of the contraction coefficient of another related Markov chain obtained via a non-standard state augmentation.

Applying Proposition 6 with $\varepsilon = t\,\|\boldsymbol{\beta}\|_{1,1}^2$, and using the upper bounds for $L$ and $\|\mathbf{m}H\|_\infty^2$ concludes the proof.

## 5.8 Discussion

Fitting autoregressive AR($p$) models with multiple lags is of broad interest in multivariate time series analysis. We consider a large class of multivariate discrete-valued AR($p$) processes with nonlinear feedback. We study statistical properties of a general $\ell_1$ regularized M-estimator for this model, and provide sufficient conditions on the model hyperparameters under which consistent estimation is possible. Under assumptions of approximate sparsity, our result shows that a sample complexity $\Omega(\mathsf{poly}(s), \log(Np))$ is achievable. Our experiments validate the theoretical results on simulated data. Commonly occurring special cases of discrete-valued processes such as Bernoulli AR($p$) and Truncated-Poisson AR($p$) are explored in detail. The proof technique develops concentration inequalities and identifies mixing properties of higher order Markov chains which may be of independent interest. These techniques were previously unknown to the best of our knowledge.

Several open questions remain to be uncovered for the general AR($p$) model. For example the current model explores the case of bounded, discrete valued data. Getting around this assumption requires finding concentration inequalities for random averages of the form in Lemma 11 for real-valued random processes. Also, it remains unknown whether the dependence on the sparsity hyperparameter $s$ is optimal, since there is a small gap between

our upper bound and the naive lower bound. Finally, it would be interesting to study parameter estimation, and potentially even controls, for the case of partial observability, i.e., when we observe $g(\boldsymbol{x}^t)$ and not $\boldsymbol{x}^t$ fully, akin to partially-observed Markov decision processes (POMDPs).

# Appendix A

# Appendix for Matrix Inference and Estimation in Multi-Layer Models

## A.1 State Evolution Equations

The state evolution equations given in Algo. 3 define an iteration indexed by $k$ of constant matrices $\{\mathbf{K}^{+}_{k\ell}, \boldsymbol{\tau}^{-}_{kl}, \overline{\boldsymbol{\Gamma}}^{\pm}_{kl}\}_{\ell=0}^{L}$. These constants appear in the statement of the main result in Theorem 1. The iterations in Algo. 3 also iteratively define a few $\mathbb{R}^{1\times d}$ valued random vectors $\{Q^0_\ell, P^0_\ell, Q^{\pm}_{k\ell}, P^{\pm}_{k\ell}\}$ which are either multivariate Gaussian or functions of Multivariate Gaussians. In order to state Algorithm 3, we need to define certain random variables and functions appearing therein which are described below. Let $\mathcal{L}_{\text{odd}} = \{1, 3, \ldots, L-1\}$ and $\mathcal{L}_{\text{even}} = \{2, 4, \ldots, L-2\}$.

Define $\{\overline{\boldsymbol{\Theta}}^{\pm}_{k\ell}\}$ similar to $\boldsymbol{\Theta}^{\pm}_{k\ell}$ from equation (3.14) using $\{\overline{\boldsymbol{\Gamma}}^{\pm}_{k\ell}\}$. Further, for $\ell = 1, 2, \ldots, L-1$ define

$$\overline{\boldsymbol{\Omega}}^{+}_{k\ell} := (\overline{\boldsymbol{\Lambda}}^{+}_{k\ell}, \overline{\boldsymbol{\Gamma}}^{+}_{k\ell}, \overline{\boldsymbol{\Gamma}}^{-}_{k\ell}),\ \overline{\boldsymbol{\Omega}}^{-}_{k\ell} := (\overline{\boldsymbol{\Lambda}}^{+}_{k,\ell-1}, \overline{\boldsymbol{\Gamma}}^{-}_{k,\ell-1}, \overline{\boldsymbol{\Gamma}}^{-}_{k,\ell-1}),$$

**Algorithm 3** State Evolution for ML-Mat-VAMP (Algo. 2)

**Require:** Functions $\{f_\ell^0\}$ from (A.2), $\{h_\ell^\pm\}$ from (A.3), and $\{f_\ell^\pm\}$ from (A.4). Perturbation random variables $\{W_\ell\}$ from (A.1). Initial random vectors $\{Q_{0\ell}^-\}_{\ell=0}^{L-1}$ with Initial covariance matrices $\{\boldsymbol{\tau}_{0\ell}^-\}_{\ell=0}^{L-1}$ from Section 4. Initial matrices $\{\overline{\boldsymbol{\Gamma}}_{0\ell}^-\}_{\ell=0}^{L}$ from (3.16).

// `Initial Pass`
$Q_0^0 = W_0$, $\boldsymbol{\tau}_0^0 = \mathrm{Cov}(Q_0^0)$ and $P_0^0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_0^0)$
**for** $\ell = 1, \ldots, L-1$ **do**
  $Q_\ell^0 = f_\ell^0(P_{\ell-1}^0, W_\ell)$
  $P_\ell^0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_\ell^0), \qquad \boldsymbol{\tau}_\ell^0 = \mathrm{Cov}(Q_\ell^0)$
**end for**

**for** $k = 0, 1, \ldots$ **do**
  // `Forward Pass`
  $\widehat{Q}_{k0}^+ = h_0^+(Q_{k0}^-, W_0, \overline{\boldsymbol{\Theta}}_{k0}^+)$
  $\overline{\boldsymbol{\Lambda}}_{k0}^+ = (\mathbb{E}\frac{\partial \widehat{Q}_{k0}^+}{\partial Q_0^-})^{-1}\overline{\boldsymbol{\Gamma}}_{k,0}^-$
  $\overline{\boldsymbol{\Gamma}}_{k0}^+ = \overline{\boldsymbol{\Lambda}}_{k0}^+ - \overline{\boldsymbol{\Gamma}}_{k0}^-$
  $Q_{k0}^+ = f_0^+(Q_{k0}^-, W_0, \overline{\boldsymbol{\Omega}}_{k0}^+)$
  $(P_0^0, P_{k0}^+) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k0}^+), \qquad \mathbf{K}_{k0}^+ := \mathrm{Cov}(Q_0^0, Q_{k0}^+)$

  **for** $\ell = 1, \ldots, L-1$ **do**
    $\widehat{Q}_{k\ell}^+ = h_\ell^+(P_{\ell-1}^0, P_{k,\ell-1}^+, Q_{k\ell}^-, W_\ell, \overline{\boldsymbol{\Theta}}_{k\ell}^+)$
    $\overline{\boldsymbol{\Lambda}}_{k\ell}^+ = (\mathbb{E}\frac{\partial \widehat{Q}_{k\ell}^+}{\partial Q_{k\ell}^-})^{-1}\overline{\boldsymbol{\Gamma}}_{k\ell}^-$
    $\overline{\boldsymbol{\Gamma}}_{k\ell}^+ = \overline{\boldsymbol{\Lambda}}_{k\ell}^+ - \overline{\boldsymbol{\Gamma}}_{k\ell}^-$
    $Q_{k\ell}^+ = f_\ell^+(P_{\ell-1}^0, P_{k,\ell-1}^+, Q_{k\ell}^-, W_\ell, \overline{\boldsymbol{\Omega}}_{k\ell}^+)$
    $(P_\ell^0, P_{k\ell}^+) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k\ell}^+), \quad \mathbf{K}_{k\ell}^+ := \mathrm{Cov}(Q_\ell^0, Q_{k\ell}^+)$
  **end for**

  // `Backward Pass`
  $\widehat{P}_{k+1,L-1}^- = h_L^-(P_{L-1}^0, P_{k,L-1}^+, W_L, \overline{\boldsymbol{\Theta}}_{k+1,L}^-)$
  $\overline{\boldsymbol{\Lambda}}_{k+1,L}^- = (\mathbb{E}\frac{\partial \widehat{P}_{k+1,L-1}^-}{\partial P_{L-1}^+})^{-1}\overline{\boldsymbol{\Gamma}}_{kL}^+$
  $\overline{\boldsymbol{\Gamma}}_{k+1,L-1}^- = \overline{\boldsymbol{\Lambda}}_{k+1,L-1}^- - \overline{\boldsymbol{\Gamma}}_{k,L-1}^+,$
  $P_{k+1,L-1}^- = f_L^-(P_{L-1}^0, P_{k,L-1}^+, W_L, \overline{\boldsymbol{\Omega}}_{k+1,L}^-)$
  $Q_{k+1,L-1}^- \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_{k+1,L-1}^-), \ \boldsymbol{\tau}_{k+1,L-1}^- := \mathrm{Cov}(P_{k+1,L-1}^-)$
  **for** $\ell = L-2, \ldots, 0$ **do**
    $\widehat{P}_{k+1,\ell}^- = h_\ell^-(P_\ell^0, P_{k\ell}^+, Q_{k+1,\ell+1}^-, W_\ell, \overline{\boldsymbol{\Theta}}_{k+1,\ell}^-)$
    $\overline{\boldsymbol{\Lambda}}_{k+1,\ell}^- = (\mathbb{E}\frac{\partial \widehat{P}_{k+1,\ell}^-}{\partial P_{k,\ell}^+})^{-1}\overline{\boldsymbol{\Gamma}}_{k,\ell}^+$
    $\overline{\boldsymbol{\Gamma}}_{k+1,\ell}^- = \overline{\boldsymbol{\Lambda}}_{k+1,\ell}^- - \overline{\boldsymbol{\Gamma}}_{k,\ell}^+,$
    $P_{k+1,\ell}^- = f_\ell^-(P_\ell^0, P_{k\ell}^+, Q_{k+1,\ell+1}^-, W_\ell, \overline{\boldsymbol{\Omega}}_{k+1,\ell}^-)$
    $Q_{k+1,\ell}^- \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\tau}_{k+1,\ell}^-), \quad \boldsymbol{\tau}_{k+1,\ell}^- := \mathrm{Cov}(P_{k+1,\ell}^-)$
  **end for**
**end for**

and $\overline{\mathbf{\Omega}}_{k0}^{+}$ and $\overline{\mathbf{\Omega}}_{kL}^{-}$. Now define random variables $W_\ell$ as

$$\begin{aligned} &W_0 = Z_0^0, \quad W_L = (Y, \Xi_L), \quad W_\ell = \Xi_\ell, \quad \forall\, \ell \in \mathcal{L}_{\text{even}}, \\ &W_\ell = (S_\ell, \overline{B}_\ell, \Xi_\ell), \qquad\qquad\qquad\qquad \forall\, \ell \in \mathcal{L}_{\text{odd}}. \end{aligned} \tag{A.1}$$

Define functions $\{f_\ell^0\}_{\ell=1}^L$ as

$$\begin{aligned} f_\ell^0(P_{\ell-1}^0, W_\ell) &:= S_\ell P_{\ell-1}^0 + \overline{B}_\ell + \Xi_\ell, \quad \forall\, \ell \in \mathcal{L}_{\text{odd}}, \\ f_\ell^0(P_{\ell-1}^0, W_\ell) &:= \phi_\ell(P_{\ell-1}^0, \Xi_\ell), \quad \forall\, \ell \in \mathcal{L}_{\text{even}} \cup \{L\}. \end{aligned} \tag{A.2}$$

and using (3.14) define functions $\{h_\ell^{\pm},\}_{\ell=1}^L$, $h_0^+$ and $h_L^-$ as

$$\begin{aligned} &h_\ell^{\pm}(P_{\ell-1}^0, P_{\ell-1}^+, Q_\ell^-, W_\ell, \mathbf{\Theta}_{k\ell}^{\pm}) = G_\ell^{\pm}(Q_\ell^- + Q_\ell^0, P_{\ell-1}^+ + P_{\ell-1}^0, \mathbf{\Theta}_{k\ell}^{\pm}), \quad \forall\, \ell \in \mathcal{L}_{\text{even}}, \\ &h_\ell^{\pm}(P_{\ell-1}^0, P_{\ell-1}^+, Q_\ell^-, W_\ell, \mathbf{\Theta}_{k\ell}^{\pm}) = \widetilde{G}_\ell^{\pm}(Q_\ell^- + Q_\ell^0, P_{\ell-1}^+ + P_{\ell-1}^0, \mathbf{\Theta}_{k\ell}^{\pm}), \quad \forall\, \ell \in \mathcal{L}_{\text{odd}} \\ &h_0^+(Q_0^-, W_0, \mathbf{\Theta}_{k0}^+) = G_0^+(Q_0^- + W_0, \mathbf{\Theta}_{k0}^+), \\ &h_L^-(P_{L-1}^0, P_{L-1}^+, W_L, \mathbf{\Theta}_{kL}^-) = G_L^-(P_{L-1}^+ + P_{L-1}^0, \mathbf{\Theta}_{kL}^-). \end{aligned} \tag{A.3}$$

Note that $[G_\ell^+, G_\ell^-]$ and $[\widetilde{G}_\ell^+, \widetilde{G}_\ell^-]$ are maps from $\mathbb{R}^{1\times d} \to \mathbb{R}^{1\times d}$ such that their row-wise extensions are the denoisers $[\mathbf{G}_\ell^+, \mathbf{G}_\ell^-]$ and $[\widetilde{\mathbf{G}}_\ell^+, \widetilde{\mathbf{G}}_\ell^-]$ respectively. Using (A.3) define functions $\{f_\ell^{\pm}\}_{\ell=1}^{L-1}$, $f_0^+$ and $f_L^-$ as

$$\begin{aligned} &f_\ell^+(P_{\ell-1}^0, P_{\ell-1}^+, Q_\ell^-, W_\ell, \mathbf{\Omega}_{k\ell}^+) = \left[\left(h_\ell^+ - Q_\ell^0\right)\mathbf{\Lambda}_{k\ell}^+ - Q_\ell^-\mathbf{\Gamma}_{k\ell}^-\right](\mathbf{\Gamma}_{k\ell}^+)^{-1}, \\ &f_\ell^-(P_{\ell-1}^0, P_{\ell-1}^+, Q_\ell^-, W_\ell, \mathbf{\Omega}_{k\ell}^-) = \left[\left(h_\ell^- - P_{\ell-1}^0\right)\mathbf{\Lambda}_{k,\ell-1}^- - P_{\ell-1}^+\mathbf{\Gamma}_{k,\ell-1}^+\right](\mathbf{\Gamma}_{k,\ell-1}^-)^{-1}. \\ &f_0^+(Q_0^-, W_0, \mathbf{\Omega}_{k0}^+) = \left[\left(h_0^+ - W_0\right)\mathbf{\Lambda}_{k0}^+ - Q_0^-\mathbf{\Gamma}_{k0}^-\right](\mathbf{\Gamma}_{k0}^+)^{-1}, \\ &f_L^-(P_{L-1}^0, P_{L-1}^+, W_L, \mathbf{\Omega}_{kL}^-) = \left[\left(h_L^- - P_{L-1}^0\right)\mathbf{\Lambda}_{k,L-1}^- - P_{L-1}^+\mathbf{\Gamma}_{k,L-1}^+\right](\mathbf{\Gamma}_{k,L-1}^-)^{-1}. \end{aligned} \tag{A.4}$$

## A.2 Large System Limit Details

The analysis of Algorithm 2 in the large system limit is based on [Bayati and Montanari, 2011b] and is by now standard in the theory of AMP-based algorithms. The goal is to characterize ensemble row-wise averages of iterates of the algorithm using *simpler* finite-dimensional random variables which are either Gaussians or functions of Gaussians. To that end, we start by defining some key terms needed in this analysis.

**Definition 5** (Pseudo-Lipschitz continuity)**.** For a given $p \geqslant 1$, a map $\mathbf{g} : \mathbb{R}^{1\times d} \to \mathbb{R}^{1\times r}$ is called pseudo-Lipschitz of order $p$ if for any $\mathbf{r}_1, \mathbf{r}_2 \in \mathbb{R}^d$ we have,

$$\|\mathbf{g}(\mathbf{r}_1) - \mathbf{g}(\mathbf{r}_2)\| \leqslant C\|\mathbf{r}_1 - \mathbf{r}_2\| \left(1 + \|\mathbf{r}_1\|^{p-1} + \|\mathbf{r}_2\|^{p-1}\right)$$

**Definition 6** (Empirical convergence of rows of a matrix sequence)**.** Consider a matrix-sequence $\{\mathbf{X}^{(N)}\}_{N=1}^{\infty}$ with $\mathbf{X}^{(N)} \in \mathbb{R}^{N\times d}$. For a finite $p \geqslant 1$, let $X \in (\mathbb{R}^d, \mathcal{R}^d)$ be a $\mathcal{R}^d$-measurable random variable with bounded moment $\mathbb{E}\|X\|_p^p < \infty$. We say the rows of matrix sequence $\{\mathbf{X}^{(N)}\}$ *converge empirically to $X$ with $p^{th}$ order moments* if for all pseudo-Lipschitz continuous functions $f(\cdot)$ of order $p$,

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(\mathbf{X}_{n:}^{(N)}) = \mathbb{E}[f(X)] \quad \text{a.s.} \tag{A.5}$$

Note that the sequence $\{\mathbf{X}^{(N)}\}$ could be random or deterministic. If it is random, however, then the quantities on the left hand side are random sums and the almost sure convergence must take this randomness into account as well.

The above convergence is equivalent to requiring weak convergence as well as convergence of the $p^{\text{th}}$ moment, of the empirical distribution $\frac{1}{N}\sum_{n=1}^{N} \delta_{\mathbf{X}_{n:}^{(N)}}$ of the rows of $\mathbf{X}^{(N)}$ to $X$. This is also referred to convergence in the Wasserstein-$p$ metric [Villani, 2008, Chap. 6].

In the case of $p = 2$, the condition is equivalent to requiring (A.5) to hold for all continuously bounded functions $f$ as well as for all $f_q(\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{x}$ for all positive definite matrices $\boldsymbol{Q}$.

**Definition 7** (Uniform Lipschitz continuity)**.** For a positive definite matrix $\mathbb{M}^{n,d}(1)$, the map $\phi(\mathbf{r}; \mathbb{M}^{n,d}(1)) : \mathbb{R}^d \to \mathbb{R}^d$ is said to be uniformly Lipschitz continuous in $\mathbf{r}$ at $\mathbb{M}^{n,d}(1) = \overline{\mathbb{M}^{n,d}(1)}$ if there exist non-negative constants $L_1$, $L_2$ and $L_3$ such that for all $\mathbf{r} \in \mathbb{R}^d$

$$\|\phi(\mathbf{r}_1; \mathbb{M}^{n,d}(1)_0) - \phi(\mathbf{r}_2; \mathbb{M}^{n,d}(1)_0)\| \leqslant L_1\|\mathbf{r}_1 - \mathbf{r}_2\|$$

$$\|\phi(\mathbf{r}; \mathbb{M}^{n,d}(1)_1) - \phi(\mathbf{r}; \mathbb{M}^{n,d}(1)_2)\| \leqslant L_2(1 + \|\mathbf{r}\|)\rho(\mathbb{M}^{n,d}(1)_1, \mathbb{M}^{n,d}(1)_2)$$

for all $\mathbb{M}^{n,d}(1)_i$ such that $\rho(\mathbb{M}^{n,d}(1)_i, \overline{\mathbb{M}^{n,d}(1)}) < L_3$ where $\rho$ is a metric on the cone of positive semidefinite matrices.

We are now ready to prove Theorem 1.

## A.3 Proof of Theorem 1

The proof of Theorem 1 is a special case of a more general result on multi-layer recursions given in Theorem 4. This result is stated in A.4, and proved in A.5. The rest of this section identifies certain relevant quantities from Theorem 1 in order to apply Theorem 4.

Consider the SVD given of weight matrices $\mathbf{W}_\ell$ of the network given by,

$$\mathbf{W}_\ell = \mathbf{V}_\ell \text{diag}(\mathbf{S}_\ell)\mathbf{V}_\ell - 1$$

as explained in Section 3.4 of Chapter 3. We analyze Algo. 2 using *transformed* versions of the true signals $\mathbf{Z}_\ell^0$ and input errors $\mathbf{R}_\ell^\pm - \mathbf{Z}_\ell^0$ to the denoisers $\mathbf{G}_\ell^\pm$. For $\ell = 0, 2, \ldots L-2$, define

$$\mathbf{q}_\ell^0 = \mathbf{Z}_\ell^0 \qquad \mathbf{q}_{\ell+1}^0 = \mathbf{V}_{\ell+1}^\top \mathbf{Z}_{\ell+1}^0 \tag{A.6a}$$

$$\mathbf{p}_\ell^0 = \mathbf{V}_\ell \mathbf{Z}_\ell^0 \qquad \mathbf{p}_{\ell+1}^0 = \mathbf{Z}_{\ell+1}^0 \tag{A.6b}$$

which are depicted in Fig. A.1 (TOP). Similarly, define the following *transformed* versions of errors in the inputs $\mathbf{R}_\ell^\pm$ to the denoisers $\mathbf{G}_\ell^\pm$

$$\mathbf{q}_\ell^- = \mathbf{R}_\ell^- - \mathbf{Z}_\ell^0 \qquad \mathbf{q}_{\ell+1}^- = \mathbf{V}_{\ell+1}^\top (\mathbf{R}_{\ell+1}^- - \mathbf{Z}_{\ell+1}^0) \tag{A.7a}$$

$$\mathbf{p}_\ell^+ = \mathbf{V}_\ell (\mathbf{R}_\ell^+ - \mathbf{Z}_\ell^0) \qquad \mathbf{p}_{\ell+1}^+ = \mathbf{R}_{\ell+1}^+ - \mathbf{Z}_{\ell+1}^0 \tag{A.7b}$$

These quantities are depicted as inputs to function blocks $\mathbf{f}_\ell^\pm$ in Fig. A.1 (MIDDLE). Define perturbation variables $\mathbf{w}_\ell$ as

$$\mathbf{w}_0 = \mathbf{Z}_0^0, \quad \mathbf{w}_{_L} = (\mathbf{Y}, \boldsymbol{\Xi}_{_L}), \quad \mathbf{w}_\ell = \boldsymbol{\Xi}_\ell, \qquad \forall\, \ell \in \mathcal{L}_{\text{even}} \tag{A.8a}$$

$$\mathbf{w}_\ell = (\mathbf{S}_\ell, \overline{\mathbf{B}}_\ell, \boldsymbol{\Xi}_\ell), \qquad \forall\, \ell \in \mathcal{L}_{\text{odd}} \tag{A.8b}$$

Finally, we define $\mathbf{q}_\ell^+$ and $\mathbf{p}_\ell^-$ for $\ell = 1, 2, \ldots, L-1$ as

$$\mathbf{q}_\ell^+ = \mathbf{f}_\ell^+(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{\ell-1}^+, \mathbf{q}_\ell^-, \mathbf{w}_\ell, \Omega_\ell) \tag{A.9a}$$

$$\mathbf{p}_{\ell-1}^- = \mathbf{f}_\ell^-(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{\ell-1}^+, \mathbf{q}_\ell^-, \mathbf{w}_\ell, \Omega_\ell), \tag{A.9b}$$

which are outputs of function blocks in Fig. A.1 (MIDDLE). Similarly, define the quantities $\mathbf{q}_0^+ = \mathbf{f}_0^+(\mathbf{q}_0^-, \mathbf{Z}_0, \Omega_0)$ and $\mathbf{p}_{L-1}^- = \mathbf{f}_L^+(\mathbf{p}_{L-1}^0, \mathbf{p}_{L-1}^+, \mathbf{Y}, \Omega_L)$.

**Lemma 14.** *Algorithm 2 is a special case of Algorithm 4 with the definitions* $\{\mathbf{q}_\ell^0, \mathbf{p}_\ell^0, \mathbf{q}_\ell^\pm, \mathbf{p}_\ell^\pm\}_{\ell=0}^{L-1}$ *given in equations* (A.6),(A.7)*, and* (A.9)*, functions* $\mathbf{f}_\ell^\pm$ *are row-wise extensions of* $f_\ell^\pm$ *defined using equations* (A.4) *and* (A.3)*.*

**Lemma 15.** *Assumptions 1 and 2 required for applying Theorem 4 are satisfied by the conditions in Theorem 1.*

*Proof.* The proofs of the above lemmas are identical to the case of $d = 1$, which was shown in [Pandit et al., 2019b]. For details see [Pandit et al., 2019b, Appendix F]. ∎

## A.4 General Multi-Layer Recursions

To analyze Algorithm 2, we consider a more general class of recursions as given in Algorithm 4 and depicted in Fig. A.1. The Gen-ML recursions generates (i) a set of *true matrices* $\mathbf{q}_\ell^0$ and $\mathbf{p}_\ell^0$ and (ii) *iterated matrices* $\mathbf{q}_{k\ell}^\pm$ and $\mathbf{p}_{k\ell}^\pm$. Each of these matrices have the same number of columns, denoted by $d$.

The true matrices are generated by a single forward pass, whereas the iterated matrices are generated via a sequence of forward and backward passes through a multi-layer system. In proving the State Evolution for the ML-Mat-VAMP algorithm (Algo. 2, one would then associate the terms $\mathbf{q}_{k\ell}^\pm$ and $\mathbf{p}_{k\ell}^\pm$ with certain error quantities in the ML-Mat-VAMP recursions. To account for the effect of the parameters $\mathbf{\Gamma}_{k\ell}^\pm$ and $\mathbf{\Lambda}_{k\ell}^\pm$ in ML-Mat-VAMP, the Gen-ML algorithm describes the parameter updates through a sequence of *parameter lists* $\Upsilon_{k\ell}^\pm$. The parameter lists are ordered lists of parameters that accumulate as the algorithm progresses.

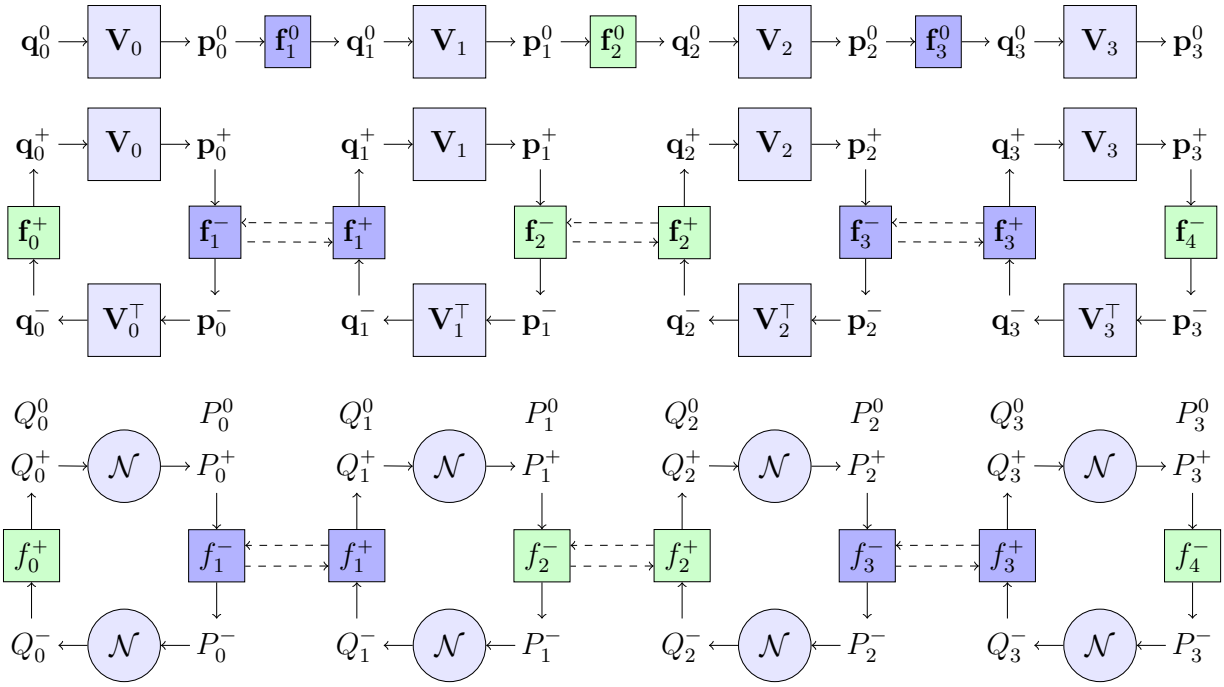


**Figure A.1:** (TOP) The equations (3.1) with equivalent quantities defined in (A.6), and $\mathbf{f}_\ell^0$ defined using (A.2).
(MIDDLE) The Gen-ML-Mat recursions in Algorithm 4. These are also equivalent to ML-Mat-VAMP recursions from Algorithm 2 (See Lemma 14) if $\mathbf{q}^\pm, \mathbf{p}^\pm$ are as defined as in equations (A.7) and (A.9), and $\mathbf{f}_\ell^\pm$ given by equations (A.4) and (A.3).
(BOTTOM) Quantities in the GEN-ML-SE recursions. These are also equivalent to ML-Mat-VAMP SE recursions from Algorithm 3 (See Lemma 14)
The iteration indices $k$ have been dropped for notational simplicity.

The true and iterated matrices from Algorithm 4 are depicted in the signal flow graphs on the (TOP) and (MIDDLE) panel of Fig. A.1 respectively. The iteration index $k$ for the iterated vectors $\mathbf{q}_{k\ell}, \mathbf{p}_{k\ell}$ has been dropped for simplifying notation.

The functions $\mathbf{f}_\ell^0(\cdot)$ that produce the true matrices $\mathbf{q}_\ell^0, \mathbf{p}_\ell^0$ are called *initial matrix functions* and use the initial parameter list $\Upsilon_{01}^-$. The functions $\mathbf{f}_{k\ell}^\pm(\cdot)$ that produce the matrices $\mathbf{q}_{k\ell}^+$ and $\mathbf{p}_{k\ell}^-$ are called the *matrix update functions* and use parameter lists $\Upsilon_{kl}^\pm$. The initial parameter lists $\Upsilon_{01}^-$ are assumed to be provided. As the algorithm progresses, new parameters $\lambda_{k\ell}^\pm$ are computed and then added to the lists in lines 12, 18, 25 and 31. The matrix update functions $\mathbf{f}_{k\ell}^\pm(\cdot)$ may depend on any sets of parameters accumulated in the parameter list. In lines 11, 17, 24 and 30, the new parameters $\lambda_{k\ell}^\pm$ are computed by: (1) computing average values $\mu_{k\ell}^\pm$ of

*row-wise* functions $\boldsymbol{\varphi}_{k\ell}^{\pm}(\cdot)$; and (2) taking functions $T_{k\ell}^{\pm}(\cdot)$ of the average values $\mu_{k\ell}^{\pm}$. Since the average values $\mu_{k\ell}^{\pm}$ represent statistics on the rows of $\boldsymbol{\varphi}_{k\ell}^{\pm}(\cdot)$, we will call $\boldsymbol{\varphi}_{k\ell}^{\pm}(\cdot)$ the *parameter statistic functions*. We will call the $T_{k\ell}^{\pm}(\cdot)$ the *parameter update functions*. The functions $\mathbf{f}_{\ell}^{0}, \mathbf{f}_{k\ell}^{\pm}, \boldsymbol{\varphi}_{\ell}^{\pm}$ also take as input some perturbation vectors $\mathbf{w}_{\ell}$.

Similar to the analysis of the ML-Mat-VAMP Algorithm, we consider the following large-system limit (LSL) analysis of Gen-ML. Specifically, we consider a sequence of runs of the recursions indexed by $N$. For each $N$, let $N_{\ell} = N_{\ell}(N)$ be the dimension of the matrix signals $\mathbf{p}_{\ell}^{\pm}$ and $\mathbf{q}_{\ell}^{\pm}$ as we assume that $\lim_{N\to\infty} \frac{N_{\ell}}{N} = \beta_{\ell} \in (0,\infty)$ is a constant so that $N_{\ell}$ scales linearly with $N$. Note however that the number of columns of each of the matrices $\{\mathbf{q}_{\ell}^{0}, \mathbf{p}_{\ell}^{0}, \mathbf{q}_{k\ell}^{\pm}, \mathbf{p}_{k\ell}^{\pm}\}$ is equal to a finite integer $d > 0$, which remains fixed for all $N$. We then make the following assumptions. See A.2 for an overview of empirical convergence of sequences which we use in the assumptions described below.

**Assumption 1.** For vectors in the Gen-ML Algorithm (Algorithm 4), we assume:

(a) The matrices $\mathbf{V}_{\ell}$ are Haar distributed on the set of $N_{\ell} \times N_{\ell}$ orthogonal matrices and are independent from one another and from the matrices $\mathbf{q}_{0}^{0}$, $\mathbf{q}_{0\ell}^{-}$, perturbation variables $\mathbf{w}_{\ell}$.

(b) The rows of the initial matrices $\mathbf{q}_{0\ell}^{-}$, and perturbation variables $\mathbf{w}_{\ell}$ converge jointly empirically with limits,

$$\mathbf{q}_{0\ell}^{-} \overset{2}{\Rightarrow} Q_{0\ell}^{-}, \quad \mathbf{w}_{\ell} \overset{2}{\Rightarrow} W_{\ell}, \tag{A.10}$$

where $Q_{0\ell}^{-}$ are random vectors in $\mathbb{R}^{1\times d}$ such that $(Q_{00}^{-}, \cdots, Q_{0,L-1}^{-})$ is jointly Gaussian. For $\ell = 0, \ldots, L-1$, the random variables $W_{\ell}, P_{\ell-1}^{0}$ and $Q_{0\ell}^{-}$ are all independent. We also assume that the initial parameter list converges as

$$\lim_{N\to\infty} \Upsilon_{01}^{-}(N) \xrightarrow{a.s.} \overline{\Upsilon}_{01}^{-}, \tag{A.11}$$

to some list $\overline{\Upsilon}_{01}^{-}$. The limit (A.11) means that every element in the list $\lambda(N) \in \Upsilon_{01}^{-}(N)$ converges to a limit $\lambda(N) \to \overline{\lambda} \in \overline{\Upsilon}_{01}^{-}$ as $N \to \infty$ almost surely.

(c) The *matrix update functions* $\mathbf{f}_{k\ell}^{\pm}(\cdot)$ and *parameter update functions* $\boldsymbol{\varphi}_{k\ell}^{\pm}(\cdot)$ act row-wise. For e.g., in the $k^{\text{th}}$ forward pass, at stage $\ell$, we assume that for each output row $n$,

$$\left[\mathbf{f}_{k\ell}^{+}(\mathbf{p}_{\ell-1}^{0}, \mathbf{p}_{k,\ell-1}^{+}, \mathbf{q}_{k\ell}^{-}, \mathbf{w}_{\ell}, \Upsilon_{k\ell}^{+})\right]_{n:} = f_{k\ell}^{+}(\mathbf{p}_{\ell-1,n:}^{0}, \mathbf{p}_{k,\ell-1,n:}^{+}, \mathbf{q}_{k\ell,n:}^{-}, \mathbf{w}_{\ell,n:}, \Upsilon_{k\ell}^{+})$$

$$\left[\boldsymbol{\varphi}_{k\ell}^{+}(\mathbf{p}_{\ell-1}^{0}, \mathbf{p}_{k,\ell-1}^{+}, \mathbf{q}_{k\ell}^{-}, \mathbf{w}_{\ell}, \Upsilon_{k\ell}^{+})\right]_{n:} = \varphi_{k\ell}^{+}(\mathbf{p}_{\ell-1,n:}^{0}, \mathbf{p}_{k,\ell-1,n:}^{+}, \mathbf{q}_{k\ell,n:}^{-}, \mathbf{w}_{\ell,n:}, \Upsilon_{k\ell}^{+}),$$

for some $\mathbb{R}^{1\times d}$-valued functions $f_{k\ell}^{+}(\cdot)$ and $\varphi_{k\ell}^{+}(\cdot)$. Similar definitions apply in the reverse directions and for the initial vector functions $\mathbf{f}_{\ell}^{0}(\cdot)$. We will call $f_{k\ell}^{\pm}(\cdot)$ the *matrix update row-wise functions* and $\varphi_{k\ell}^{\pm}(\cdot)$ the *parameter update row-wise functions*.

Next we define a set of *deterministic* constants $\{\mathbf{K}_{k\ell}^{+}, \boldsymbol{\tau}_{k\ell}^{-}, \overline{\mu}_{k\ell}^{\pm}, \overline{\Upsilon}_{kl}^{\pm}, \boldsymbol{\tau}_{\ell}^{0}\}$ and $\mathbb{R}^{1\times d}$-valued random vectors $\{Q_{\ell}^{0}, P_{\ell}^{0}, Q_{k\ell}^{\pm}, P_{\ell}^{\pm}\}$ which are recursively defined through Algorithm 5, which we call the *Gen-ML-Mat State Evolution* (SE). These recursions in Algorithm closely mirror those in the Gen-ML-Mat algorithm (Algorithm 4). The matrices $\mathbf{q}_{k\ell}^{\pm}$ and $\mathbf{p}_{k\ell}^{\pm}$ are replaced by random vectors $Q_{k\ell}^{\pm}$ and $P_{k\ell}^{\pm}$; the matrix and parameter update functions $\mathbf{f}_{k\ell}^{\pm}(\cdot)$ and $\boldsymbol{\varphi}_{k\ell}^{\pm}(\cdot)$ are replaced by their row-wise functions $f_{k\ell}^{\pm}(\cdot)$ and $\varphi_{k\ell}^{\pm}(\cdot)$; and the parameters $\lambda_{k\ell}^{\pm}$ are replaced by their limits $\overline{\lambda}_{k\ell}^{\pm}$. We refer to $\{Q_{\ell}^{0}, P_{\ell}^{0}\}$ as *true random vectors* and $\{Q_{k\ell}^{\pm}, P_{kl}^{\pm}\}$ as *iterated random vectors*. The signal flow graph for the true and iterated random variables in Algorithm 5 is given in the (BOTTOM) panel of Fig. A.1. The iteration index $k$ for the iterated random variables $\{Q_{k\ell}^{\pm}, P_{kl}^{\pm}\}$ to simplify notation.

We also assume the following about the behaviour of row-wise functions around the quantities defined in Algorithm 5. The iteration index $k$ has been dropped for simplifying notation.

**Assumption 2.** For row-wise functions $f, \varphi$ and parameter update functions $T$ we assume:

(a) $T_{k\ell}^{\pm}(\mu_{k\ell}^{\pm}, \cdot)$ are continuous at $\mu_{k\ell}^{\pm} = \overline{\mu}_{k\ell}^{\pm}$

(b) $f_{k\ell}^{+}(p_{\ell-1}^{0}, p_{k,\ell-1}^{+}, q_{k\ell}^{-}, w_{\ell}, \Upsilon_{k\ell}^{+})$, $\frac{\partial f_{k\ell}^{+}}{\partial q_{k\ell}^{-}}(p_{\ell-1}^{0}, p_{k,\ell-1}^{+}, q_{k\ell}^{-}, w_{\ell}, \Upsilon_{k\ell}^{+})$ and $\varphi_{k\ell}^{+}(p_{\ell-1}^{0}, p_{k,\ell-1}^{+}, q_{k\ell}^{-}, w_{\ell}, \Upsilon_{k,\ell-1}^{+})$ are uniformly Lipschitz continuous in $(p_{\ell-1}^{0}, p_{k,\ell-1}^{+}, q_{k\ell}^{-}, w_{\ell})$ at $\Upsilon_{k\ell}^{+} = \overline{\Upsilon}_{k\ell}^{+}$, $\Upsilon_{k,\ell-1}^{+} = \overline{\Upsilon}_{k,\ell-1}^{+}$. Similarly,

$f^-_{k+1,\ell}(p^0_{\ell-1}, p^+_{k,\ell-1}, q^-_{k+1,\ell}, w_\ell, \Upsilon^-_{k\ell})$, $\frac{\partial f^-_{k\ell}}{\partial p^+_{k,\ell-1}}(p^0_{\ell-1}, p^+_{k,\ell-1}, q^-_{k+1,\ell}, w_\ell, \Upsilon^-_{k\ell})$, and $\varphi^-_{k\ell}(p^0_{\ell-1}, p^+_{k,\ell-1}, q^-_{k+1,\ell}, w_\ell, \Upsilon^-_{k+1,\ell}$ are uniformly Lipschitz continuous in $(p^0_{\ell-1}, p^+_{k,\ell-1}, q^-_{k+1,\ell}, w_\ell)$ at $\Upsilon^-_{k\ell} = \overline{\Upsilon}^-_{k\ell}$, $\Upsilon^-_{k+1,\ell+1} = \overline{\Upsilon}^-_{k+1,\ell+1}$.

(c) $f^0_\ell(p^0_{\ell-1}, w_\ell, \Upsilon^-_{01})$ are uniformly Lipschitz continuous in $(p^0_{k,\ell-1}, w_\ell)$ at $\Upsilon^-_{k+1,\ell} = \overline{\Upsilon}^-_{k+1,\ell}$.

(d) Matrix update functions $\mathbf{f}^\pm_{k\ell}$ are *asymptotically divergence free* meaning

$$\lim_{N\to\infty} \left\langle \tfrac{\partial \mathbf{f}^+_{k\ell}}{\partial \mathbf{q}^-_{k\ell}}(\mathbf{p}^+_{k,\ell-1}, \mathbf{q}^-_{k\ell}, \mathbf{w}_\ell, \overline{\Upsilon}^+_{k\ell}) \right\rangle = \mathbf{0}, \quad \lim_{N\to\infty} \left\langle \tfrac{\partial \mathbf{f}^-_{k\ell}}{\partial \mathbf{p}^+_{k,\ell-1}}(\mathbf{p}^+_{k,\ell-1}, \mathbf{q}^-_{k+1,\ell}, \mathbf{w}_\ell, \overline{\Upsilon}^-_{k\ell}) \right\rangle = \mathbf{0} \tag{A.12}$$

We are now ready to state the general result regarding the empirical convergence of the true and iterated vectors from Algorithm 4 in terms of random variables defined in Algorithm 5.

**Theorem 4.** *Consider the iterates of the Gen-ML recursion (Algorithm 4) and the corresponding random variables and parameter limits defined by the SE recursions (Algorithm 5) under Assumptions 1 and 2. Then,*

*(a) For any fixed $k \geqslant 0$ and fixed $\ell = 1, \ldots, L-1$, the parameter list $\Upsilon^+_{k\ell}$ converges as*

$$\lim_{N\to\infty} \Upsilon^+_{k\ell} = \overline{\Upsilon}^+_{k\ell} \tag{A.13}$$

*almost surely. Also, the rows of $\mathbf{w}_\ell$, $\mathbf{p}^0_{\ell-1}$, $\mathbf{q}^0_\ell$, $\mathbf{p}^+_{0,\ell-1}, \ldots, \mathbf{p}^+_{k,\ell-1}$ and $\mathbf{q}^\pm_{0\ell}, \ldots, \mathbf{q}^\pm_{k\ell}$ almost surely jointly converge empirically with limits,*

$$(\mathbf{p}^0_{\ell-1}, \mathbf{p}^+_{i,\ell-1}, \mathbf{q}^-_{j\ell}, \mathbf{q}^0_\ell, \mathbf{q}^+_{j\ell}) \overset{2}{\Longrightarrow} (P^0_{\ell-1}, P^+_{i,\ell-1}, Q^-_{j\ell}, Q^0_\ell, Q^+_{j\ell}), \tag{A.14}$$

*for all $0 \leqslant i, j \leqslant k$, where the variables $P^0_{\ell-1}$, $P^+_{i,\ell-1}$ and $Q^-_{j\ell}$ are zero-mean jointly Gaussian random variables independent of $W_\ell$ and with covariance matrix given by*

$$\mathrm{Cov}(P^0_{\ell-1}, P^+_{i,\ell-1}) = \mathbf{K}^+_{i,\ell-1}, \quad \mathbb{E}(Q^-_{j\ell})^2 = \boldsymbol{\tau}^-_{j\ell}, \quad \mathbb{E}(P^{+\mathsf{T}}_{i,\ell-1} Q^-_{j\ell}) = \mathbf{0}, \quad \mathbb{E}(P^{0\mathsf{T}}_{\ell-1} Q^-_{j\ell}) = \mathbf{0}, \tag{A.15}$$

*and $Q^0_\ell$, $Q^+_{j\ell}$ are the random variable in lines 4, 19,*i.e.,

$$Q^0_\ell = f^0_\ell(P^0_{\ell-1}, W_\ell), \quad Q^+_{j\ell} = f^+_{j\ell}(P^0_{\ell-1}, P^+_{j,\ell-1}, Q^-_{j\ell}, W_\ell, \overline{\Upsilon}^+_{j\ell}). \tag{A.16}$$

An identical result holds for $\ell = 0$ with all the variables $\mathbf{p}^{+}_{i,\ell-1}$ and $P^{+}_{i,\ell-1}$ removed.

(b) For any fixed $k \geqslant 1$ and fixed $\ell = 1, \ldots, L-1$, the parameter lists $\Upsilon^{-}_{k\ell}$ converge as

$$\lim_{N\to\infty} \Upsilon^{-}_{k\ell} = \overline{\Upsilon}^{-}_{k\ell} \tag{A.17}$$

almost surely. Also, the rows of $\mathbf{w}_\ell$, $\mathbf{p}^{0}_{\ell-1}$, $\mathbf{p}^{\pm}_{0,\ell-1}, \ldots, \mathbf{p}^{\pm}_{k-1,\ell-1}$, and $\mathbf{q}^{-}_{0\ell}, \ldots, \mathbf{q}^{-}_{k\ell}$ almost surely jointly converge empirically with limits,

$$(\mathbf{p}^{0}_{\ell-1}, \mathbf{p}^{+}_{i,\ell-1}, \mathbf{q}^{-}_{j\ell}, \mathbf{p}^{-}_{j,\ell-1}) \overset{2}{\Longrightarrow} (P^{0}_{\ell-1}, P^{+}_{i,\ell-1}, Q^{-}_{j\ell}, P^{-}_{j,\ell-1}), \tag{A.18}$$

for all $0 \leqslant i \leqslant k-1$ and $0 \leqslant j \leqslant k$, where the variables $P^{0}_{\ell-1}$, $P^{+}_{i,\ell-1}$ and $Q^{-}_{j\ell}$ are zero-mean jointly Gaussian random variables independent of $W_\ell$ and with covariance matrix given by equation (A.15) and $P^{-}_{j\ell}$ is the random variable in line 32:

$$P^{-}_{j\ell} = f^{-}_{j\ell}(P^{0}_{\ell-1}, P^{+}_{j-1,\ell-1}, Q^{-}_{j\ell}, W_\ell, \overline{\Upsilon}^{-}_{j\ell}). \tag{A.19}$$

An identical result holds for $\ell = L$ with all the variables $\mathbf{q}^{-}_{j\ell}$ and $Q^{-}_{j\ell}$ removed.

For $k = 0$, $\Upsilon^{-}_{01} \to \overline{\Upsilon}^{-}_{01}$ almost surely, and the rows $\{(\mathbf{w}_{\ell,n:}, \mathbf{p}^{0}_{\ell-1,n:}, \mathbf{q}^{-}_{j\ell,n:})\}_{n=1}^{N}$ empirically converge to independent random variables $(W_\ell, P^{0}_{\ell-1}, Q^{-}_{0\ell})$.

*Proof.* A.5 is dedicated to proving this result. ∎

## A.5 Proof of Theorem 4

The proof proceeds using mathematical induction. It largely mimics the proof for the case of $d = 1$ which were the convergence results in [Pandit et al., 2019b, Thm. 5]. However, in the case of $d > 1$, we observe that several quantities which were scalars in proving [Pandit et al., 2019b, Thm. 5] are now matrices. Due to the non-commutativity of these matrix quantities, we re-state the whole prove, while modifying the requisite matrix terms appropriately.

### A.5.1 Overview of the Induction Sequence

The proof is similar to that of [Rangan et al., 2019b, Theorem 4], which provides a SE analysis for VAMP on a single-layer network. The critical challenge here is to extend that proof to multi-layer recursions. Many of the ideas in the two proofs are similar, so we highlight only the key differences between the two.

Similar to the SE analysis of VAMP in [Rangan et al., 2019b], we use an induction argument. However, for the multi-layer proof, we must index over both the iteration index $k$ and layer index $\ell$. To this end, let $\mathcal{H}^+_{k\ell}$ and $\mathcal{H}^-_{k\ell}$ be the hypotheses:

- $\mathcal{H}^+_{k\ell}$: The hypothesis that Theorem 4(a) is true for a given $k$ and $\ell$, where $0 \leqslant \ell \leqslant L-1$.
- $\mathcal{H}^-_{k\ell}$: The hypothesis that Theorem 4(b) is true for a given $k$ and $\ell$, where $1 \leqslant \ell \leqslant L$.

We prove these hypotheses by induction via a sequence of implications,

$$\{\mathcal{H}^-_{0\ell}\}_{\ell=1}^{L} \cdots \Rightarrow \mathcal{H}^-_{k1} \Rightarrow \mathcal{H}^+_{k0} \Rightarrow \cdots \Rightarrow \mathcal{H}^+_{k,L-1} \Rightarrow \mathcal{H}^-_{k+1,L} \Rightarrow \cdots \Rightarrow \mathcal{H}^-_{k+1,1} \Rightarrow \cdots, \tag{A.20}$$

beginning with the hypotheses $\{\mathcal{H}^-_{0\ell}\}$ for all $\ell = 1, \ldots, L-1$.

### A.5.2 Base Case: Proof of $\{\mathcal{H}^-_{0\ell}\}_{\ell=1}^{L}$

The base case corresponds to the hypotheses $\{\mathcal{H}^-_{0\ell}\}_{\ell=1}^{L}$. Note that Theorem 4(b) states that for $k = 0$, we need $\Upsilon^-_{01} \to \overline{\Upsilon}^-_{01}$ almost surely, and $\{(\mathbf{w}_{\ell,n:}, \mathbf{p}^0_{\ell-1,n:}, \mathbf{q}^-_{j\ell,n:})\}_{n=1}^{N}$ empirically converge to independent random variables $(W_\ell, P^0_{\ell-1}, Q^-_{0\ell})$. These follow directly from equations (A.10) and (A.11) in Assumption 1 (a).

### A.5.3 Inductive Step: Proof of $\mathcal{H}^+_{k,\ell+1}$

Fix a layer index $\ell = 1, \ldots, L-1$ and an iteration index $k = 0, 1, \ldots$. We show the implication $\cdots \implies \mathcal{H}^+_{k,\ell+1}$ in (A.20). All other implications can be proven similarly using symmetry arguments.

**Definition 8** (Induction hypothesis)**.** The hypotheses prior to $\mathcal{H}^+_{k,\ell+1}$ in the sequence (A.20), but not including $\mathcal{H}^+_{k,\ell+1}$, are true.

The inductive step then corresponds to the following result.

**Lemma 16.** *Under the induction hypothesis, $\mathcal{H}^+_{k,\ell+1}$ holds*

Before proving the inductive step in Lemma 16, we prove two intermediate lemmas. Let us start by defining some notation. Define $\mathbf{P}^+_{k\ell} := \left[\mathbf{p}^+_{0\ell} \cdots \mathbf{p}^+_{k\ell}\right] \in \mathbb{R}^{N_\ell \times (k+1)d}$, be a matrix whose column blocks are the first $k+1$ values of the matrix $\mathbf{p}^+_\ell$. We define the matrices $\mathbf{P}^-_{k\ell}$, $\mathbf{Q}^+_{k\ell}$ and $\mathbf{Q}^-_{k\ell}$ in a similar manner with values of $\mathbf{p}^-_\ell, \mathbf{q}^+_\ell$ and $\mathbf{q}^-_\ell$ respectively.

Note that except the initial matrices $\{\mathbf{w}_\ell, \mathbf{q}^-_{0\ell}\}_{\ell=1}^L$, all later iterates in Algorithm 4 are random due to the randomness of $\mathbf{V}_\ell$. Let $\mathfrak{G}^\pm_{k\ell}$ denote the collection of random variables associated with the hypotheses, $\mathcal{H}^\pm_{k\ell}$. That is, for $\ell = 1, \ldots, L-1$,

$$\mathfrak{G}^+_{k\ell} := \left\{\mathbf{w}_\ell, \mathbf{p}^0_{\ell-1}, \mathbf{P}^+_{k,\ell-1}, \mathbf{q}^0_\ell, \mathbf{Q}^-_{k\ell}, \mathbf{Q}^+_{k\ell}\right\}, \qquad \mathfrak{G}^-_{k\ell} := \left\{\mathbf{w}_\ell, \mathbf{p}^0_{\ell-1}, \mathbf{P}^+_{k-1,\ell-1}, \mathbf{q}^0_\ell, \mathbf{Q}^-_{k\ell}, \mathbf{P}^-_{k,\ell-1}\right\}.$$

For $\ell = 0$ and $\ell = L$ we set, $\mathfrak{G}^+_{k0} := \left\{\mathbf{w}_0, \mathbf{Q}^-_{k0}, \mathbf{Q}^+_{k0}\right\}, \quad \mathfrak{G}^-_{kL} := \left\{\mathbf{w}_L, \mathbf{p}^0_{L-1}, \mathbf{P}^+_{k-1,L-1}, \mathbf{P}^-_{k,L-1}\right\}.$

Let $\overline{\mathfrak{G}}^+_{k\ell}$ be the sigma algebra generated by the union of all the sets $\mathfrak{G}^\pm_{k'\ell'}$ as they have appeared in the sequence (A.20) up to and including the final set $\mathfrak{G}^+_{k\ell}$. Thus, the sigma algebra $\overline{\mathfrak{G}}^+_{k\ell}$ contains all *information* produced by Algorithm 4 immediately *before* line 20 in layer $\ell$ of iteration $k$. Note also that the random variables in Algorithm 5 immediately before defining $P^+_{k,\ell}$ in line 20 are all $\overline{\mathfrak{G}}^+_{k\ell}$ measurable.

Observe that the matrix $\mathbf{V}_\ell$ in Algorithm 4 appears only during matrix-vector multiplications in lines 20 and 32. If we define the matrices, $\mathbf{A}_{k\ell} := \left[\mathbf{p}^0_\ell, \mathbf{P}^+_{k-1,\ell}\ \mathbf{P}^-_{k\ell}\right], \quad \mathbf{B}_{k\ell} := \left[\mathbf{q}^0_\ell, \mathbf{Q}^+_{k-1,\ell}\ \mathbf{Q}^-_{k\ell}\right]$, all the matrices in the set $\overline{\mathfrak{G}}^+_{k\ell}$ will be unchanged for all matrices $\mathbf{V}_\ell$ satisfying the linear constraints

$$\mathbf{A}_{k\ell} = \mathbf{V}_\ell \mathbf{B}_{k\ell}. \tag{A.21}$$

Hence, the conditional distribution of $\mathbf{V}_\ell$ given $\overline{\mathfrak{G}}^+_{k\ell}$ is precisely the uniform distribution on the set of orthogonal matrices satisfying (A.21). The matrices $\mathbf{A}_{k\ell}$ and $\mathbf{B}_{k\ell}$ are of dimensions

$N_\ell \times (2k+2)d$. From [Rangan et al., 2019b, Lemmas 3,4], this conditional distribution is given by

$$\mathbf{V}_\ell|_{\overline{\mathfrak{G}}^+_{k\ell}} \stackrel{d}{=} \mathbf{A}_{k\ell}(\mathbf{A}_{k\ell}^\mathsf{T}\mathbf{A}_{k\ell})^{-1}\mathbf{B}_{k\ell}^\mathsf{T} + \mathbf{U}_{\mathbf{A}^\perp_{k\ell}}\widetilde{\mathbf{V}}_\ell\mathbf{U}_{\mathbf{B}^\perp_{k\ell}}^\mathsf{T}, \tag{A.22}$$

where $\mathbf{U}_{\mathbf{A}^\perp_{k\ell}}$ and $\mathbf{U}_{\mathbf{B}^\perp_{k\ell}}$ are $N_\ell \times (N_\ell - (2k+2)d)$ matrices whose columns are an orthonormal basis for $\text{Range}(\mathbf{A}_{k\ell})^\perp$ and $\text{Range}(\mathbf{B}_{k\ell})^\perp$. The matrix $\widetilde{\mathbf{V}}_\ell$ is Haar distributed on the set of $(N_\ell - (2k+2)d) \times (N_\ell - (2k+2)d)$ orthogonal matrices and is independent of $\overline{\mathfrak{G}}^+_{k\ell}$.

Next, similar to the proof of [Rangan et al., 2019b, Thm. 4], we can use (A.22) to write the conditional distribution of $\mathbf{p}^+_{k\ell}$ (from line 20 of Algorithm 4) given $\overline{\mathfrak{G}}^+_{k\ell}$ as a sum of two terms

$$\mathbf{p}^+_{k\ell}|_{\overline{\mathfrak{G}}^+_{k\ell}} = \mathbf{V}_\ell|_{\overline{\mathfrak{G}}^+_{k\ell}}\,\mathbf{q}^+_{k\ell} \stackrel{d}{=} \mathbf{p}^{+\text{det}}_{k\ell} + \mathbf{p}^{+\text{ran}}_{k\ell}, \tag{A.23a}$$

$$\mathbf{p}^{+\text{det}}_{k\ell} := \mathbf{A}_{k\ell}(\mathbf{B}_{k\ell}^\mathsf{T}\mathbf{B}_{k\ell})^{-1}\mathbf{B}_{k\ell}^\mathsf{T}\mathbf{q}^+_{k\ell} \tag{A.23b}$$

$$\mathbf{p}^{+\text{ran}}_{k\ell} := \mathbf{U}_{\mathbf{B}^\perp_k}\widetilde{\mathbf{V}}_\ell^\mathsf{T}\mathbf{U}_{\mathbf{A}^\perp_k}^\mathsf{T}\mathbf{q}^+_{k\ell}. \tag{A.23c}$$

where we call $\mathbf{p}^{+\text{det}}_{k\ell}$ the *deterministic* term and $\mathbf{p}^{+\text{ran}}_{k\ell}$ the *random* term. The next two lemmas characterize the limiting distributions of the deterministic and random terms.

**Lemma 17.** *Under the induction hypothesis, the rows of the "deterministic" term* $\mathbf{p}^{+\text{det}}_{k\ell}$ *along with the rows of the matrices in* $\overline{\mathfrak{G}}^+_{k\ell}$ *converge empirically. In addition, there exists constant* $d \times d$ *matrices* $\beta^+_{0\ell}, \ldots, \beta^+_{k-1,\ell}$ *such that*

$$\mathbf{p}^{+\text{det}}_{k\ell} \stackrel{2}{\Longrightarrow} P^{+\text{det}}_{k\ell} := P^0_\ell\beta^0_\ell + \sum_{i=0}^{k-1} P^+_{i\ell}\beta_{i\ell}, \tag{A.24}$$

*where* $P^{+\text{det}}_{k\ell} \in \mathbb{R}^{1\times d}$ *is the limiting random vector for the rows of* $\mathbf{p}^{\text{det}}_{k\ell}$.

*Proof.* The proof is similar that of [Rangan et al., 2019b, Lem. 6], but we go over the details as there are some important differences in the multi-layer matrix case. Define $\widetilde{\mathbf{P}}^+_{k-1,\ell} = \left[\mathbf{p}^0_\ell,\ \mathbf{P}^+_{k-1,\ell}\right], \widetilde{\mathbf{Q}}^+_{k-1,\ell} = \left[\mathbf{q}^0_\ell,\ \mathbf{Q}^+_{k-1,\ell}\right]$, which are the matrices in $\mathbb{R}^{N_\ell\times(k+1)d}$. We can then write $\mathbf{A}_{k\ell}$ and $\mathbf{B}_{k\ell}$ from (A.21) as

$$\mathbf{A}_{k\ell} := \left[\widetilde{\mathbf{P}}^+_{k-1,\ell}\ \mathbf{P}^-_{k\ell}\right], \quad \mathbf{B}_{k\ell} := \left[\widetilde{\mathbf{Q}}^+_{k-1,\ell}\ \mathbf{Q}^-_{k\ell}\right], \tag{A.25}$$

We first evaluate the asymptotic values of various terms in (A.23b). By definition of $\mathbf{B}_{k\ell}$ in (A.25),

$$\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{B}_{k\ell} = \begin{bmatrix} (\widetilde{\mathbf{Q}}_{k-1,\ell}^{+})^{\mathsf{T}}\widetilde{\mathbf{Q}}_{k-1,\ell}^{+} & (\widetilde{\mathbf{Q}}_{k-1,\ell}^{+})^{\mathsf{T}}\mathbf{Q}_{k\ell}^{-} \\ (\mathbf{Q}_{k\ell}^{-})^{\mathsf{T}}\widetilde{\mathbf{Q}}_{k-1,\ell}^{+} & (\mathbf{Q}_{k\ell}^{-})^{\mathsf{T}}\mathbf{Q}_{k\ell}^{-} \end{bmatrix}$$

We can then evaluate the asymptotic values of these terms as follows: For $0 \leqslant i, j \leqslant k-1$ the asymptotic value of the $(i+2, j+2)^{\text{nd}}$ $d \times d$ block of the matrix $(\widetilde{\mathbf{Q}}_{k-1,\ell}^{+})^{\mathsf{T}}\widetilde{\mathbf{Q}}_{k-1,\ell}^{+}$ is

$$\begin{aligned} \lim_{N\to\infty} \tfrac{1}{N_\ell} \left[(\widetilde{\mathbf{Q}}_{k-1,\ell}^{+})^{\mathsf{T}}\widetilde{\mathbf{Q}}_{k-1,\ell}^{+}\right]_{i+2,j+2} &\overset{(a)}{=} \lim_{N\to\infty} \frac{1}{N_\ell}(\mathbf{q}_{i\ell}^{+})^{\mathsf{T}}\mathbf{q}_{j\ell}^{+} \\ &= \lim_{N\to\infty} \tfrac{1}{N_\ell} \sum_{n=1}^{N_\ell} [\mathbf{q}_{i\ell}^{+}]_{n:}[\mathbf{q}_{j\ell}^{+}]_{n:}^{\mathsf{T}} \overset{(b)}{=} \mathbb{E}\left[Q_{i\ell}^{+\mathsf{T}}Q_{j\ell}^{+}\right] \end{aligned}$$

where (a) follows since the $(i+2)^{\text{th}}$ column block of $\widetilde{\mathbf{Q}}_{k-1,\ell}^{+}$ is $\mathbf{q}_{i\ell}^{+}$, and (b) follows due to the empirical convergence assumption in (A.14). Also, since the first column block of $\widetilde{\mathbf{Q}}_{k-1,\ell}^{+}$ is $\mathbf{q}_{\ell}^{0}$, we obtain that

$$\begin{aligned} &\lim_{N_\ell\to\infty} \tfrac{1}{N_\ell}(\widetilde{\mathbf{Q}}_{k-1,\ell}^{+})^{\mathsf{T}}\widetilde{\mathbf{Q}}_{k-1,\ell}^{+} = \mathbf{R}_{k-1,\ell}^{+} \qquad \text{and} \\ &\lim_{N_\ell\to\infty} \tfrac{1}{N_\ell}(\mathbf{Q}_{k\ell}^{-})^{\mathsf{T}}\mathbf{Q}_{k\ell}^{-} = \mathbf{R}_{k\ell}^{-}, \end{aligned} \tag{A.26}$$

where $\mathbf{R}_{k-1,\ell}^{+} \in \mathbb{R}^{(k+1)d\times(k+1)d}$ is the covariance matrix of $\left[Q_{\ell}^{0}\ Q_{0\ell}^{+}\ \ldots\ Q_{k-1,\ell}^{+}\right]$, and $\mathbf{R}_{k\ell}^{-} \in \mathbb{R}^{(k+1)d\times(k+1)d}$ is the covariance matrix of $\left[Q_{0\ell}^{-}\ Q_{1\ell}^{-}\ \ldots\ Q_{k\ell}^{-}\right]$. For the matrix $(\widetilde{\mathbf{Q}}_{k-1,\ell}^{+})^{\mathsf{T}}\mathbf{Q}_{k\ell}^{-}$, first observe that the limit of the divergence free condition (A.12) implies

$$\mathbb{E}\left[\frac{\partial f_{i\ell}^{+}(P_{i,\ell-1}^{+}, Q_{i\ell}^{-}, W_\ell, \overline{\Upsilon}_{i\ell})}{\partial Q_{i\ell}^{-}}\right] = \lim_{N_\ell\to\infty} \left\langle \frac{\partial \mathbf{f}_{i\ell}^{+}(\mathbf{p}_{i,\ell-1}^{+}, \mathbf{q}_{i\ell}^{-}, \mathbf{w}_\ell, \overline{\Upsilon}_{i\ell}^{+})}{\partial \mathbf{q}_{i\ell}^{-}} \right\rangle = \mathbf{0}, \tag{A.27}$$

for any $i$. Also, by the induction hypothesis $\mathcal{H}_{k\ell}^{+}$,

$$\mathbb{E}(P_{i,\ell-1}^{+\mathsf{T}}Q_{j\ell}^{-}) = \mathbf{0}, \quad \mathbb{E}(P_{\ell-1}^{0\mathsf{T}}Q_{j\ell}^{-}) = \mathbf{0}, \tag{A.28}$$

for all $0 \leqslant i, j \leqslant k$. Therefore using (A.16), the cross-terms $\mathbb{E}(Q_{i\ell}^{+\mathsf{T}}Q_{j\ell}^{-})$ are given by

$$\begin{aligned} \mathbb{E}(f_{i\ell}^{+}(P_{\ell-1}^{0}, P_{i,\ell-1}^{+}, Q_{i\ell}^{-}, W_\ell, \overline{\Upsilon}_{i\ell})^{\mathsf{T}}Q_{j\ell}^{-}) &\overset{(a)}{=} \mathbb{E}\left[\tfrac{\partial f_{i\ell}^{+}(P_{\ell-1}^{0}, P_{i,\ell-1}^{+}, Q_{i\ell}^{-}, W_\ell, \overline{\Upsilon}_{i\ell}^{+})}{\partial P_{\ell-1}^{0}}\right] \mathbb{E}(P_{\ell-1}^{0\mathsf{T}}Q_{j\ell}^{-}) \\ &\quad + \mathbb{E}\left[\tfrac{\partial f_{i\ell}^{+}(P_{\ell-1}^{0}, P_{i,\ell-1}^{+}, Q_{i\ell}^{-}, W_\ell, \overline{\Upsilon}_{i\ell}^{+})}{\partial P_{i,\ell-1}^{+}}\right] \mathbb{E}(P_{i,\ell-1}^{+\mathsf{T}}Q_{j\ell}^{-}) \\ &\quad + \mathbb{E}\left[\tfrac{\partial f_{i\ell}^{+}(P_{\ell-1}^{0}, P_{i,\ell-1}^{+}, Q_{i\ell}^{-}, W_\ell, \overline{\Upsilon}_{i\ell}^{+})}{\partial Q_{i\ell}^{-}}\right] \mathbb{E}(Q_{i\ell}^{-\mathsf{T}}Q_{j\ell}^{-}) \overset{(b)}{=} \mathbf{0}, \end{aligned} \tag{A.29}$$

(a) follows from a multivariate version of Stein's Lemma [Liu, 1994, eqn.(2)]; and (b) follows

from (A.27), and (A.28). Consequently,

$$\lim_{N_\ell\to\infty}\tfrac{1}{N_\ell}\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{B}_{k\ell}=\begin{bmatrix}\mathbf{R}_{k-1,\ell}^{+} & \mathbf{0}\\ \mathbf{0} & \mathbf{R}_{k\ell}^{-}\end{bmatrix},\quad \text{and}\ \lim_{N_\ell\to\infty}\tfrac{1}{N_\ell}\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{q}_{k\ell}^{+}=\begin{bmatrix}\mathbf{b}_{k\ell}^{+}\\ \mathbf{0}\end{bmatrix}, \tag{A.30}$$

where $\mathbf{b}_{k\ell}^{+}:=\left[\mathbb{E}(Q_{0\ell}^{+\mathsf{T}}Q_{k\ell}^{+})\ \mathbb{E}(Q_{1\ell}^{+\mathsf{T}}Q_{k\ell}^{+})\ \cdots\ \mathbb{E}(Q_{k-1,\ell}^{+\mathsf{T}}Q_{k\ell}^{+})\right]^{\mathsf{T}}$, is the matrix of correlations. We again have $\mathbf{0}$ in the second term because $\mathbb{E}[Q_{i\ell}^{+\mathsf{T}}Q_{j\ell}^{-}]=\mathbf{0}$ for all $0\leqslant i,j\leqslant k$. Hence we have

$$\lim_{N_\ell\to\infty}(\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{B}_{k\ell})^{-1}\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{q}_{k\ell}^{+}=\begin{bmatrix}\boldsymbol{\beta}_{k\ell}^{+}\\ \mathbf{0}\end{bmatrix},\quad \boldsymbol{\beta}_{k\ell}^{+}:=\begin{bmatrix}\mathbf{R}_{k-1,\ell}^{+}\end{bmatrix}^{-1}\mathbf{b}_{k\ell}^{+}. \tag{A.31}$$

Therefore, $\mathbf{p}_{k\ell}^{+\det}$ equals

$$\begin{aligned}\mathbf{A}_{k\ell}(\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{B}_{k\ell})^{-1}\mathbf{B}_{k\ell}^{\mathsf{T}}\mathbf{q}_{k\ell}^{+} &= \begin{bmatrix}\widetilde{\mathbf{P}}_{k-1,\ell}^{+}\ \mathbf{P}_{k,\ell}^{-}\end{bmatrix}\begin{bmatrix}\boldsymbol{\beta}_{k\ell}^{+}\\ \mathbf{0}\end{bmatrix}+O\left(\tfrac{1}{N_\ell}\right)\\ &= \mathbf{p}_{\ell}^{0}\beta_{\ell}^{0}+\sum_{i=0}^{k-1}\mathbf{p}_{i\ell}^{+}\beta_{i\ell}^{+}+O\left(\tfrac{1}{N_\ell}\right),\end{aligned} \tag{A.32}$$

where $\beta_\ell^0$ and $\beta_{i\ell}^+$ are $d\times d$ block matrices of $\boldsymbol{\beta}_{k\ell}^+$ and the term $O(\frac{1}{N_\ell})$ means a matrix sequence, $\boldsymbol{\varphi}(N)\in\mathbb{R}^{N_\ell}$ such that $\lim_{N\to\infty}\frac{1}{N}\|\boldsymbol{\varphi}(N)\|^2=0$. A continuity argument then shows the empirical convergence (A.24). ∎

**Lemma 18.** *Under the induction hypothesis, the components of the "random" term* $\mathbf{p}_{k\ell}^{+\mathrm{ran}}$ *along with the components of the vectors in* $\overline{\mathfrak{G}}_{k\ell}^{+}$ *almost surely converge empirically. The components of* $\mathbf{p}_{k\ell}^{+\mathrm{ran}}$ *converge as*

$$\mathbf{p}_{k\ell}^{+\mathrm{ran}}\overset{2}{\Rightarrow}U_{k\ell}, \tag{A.33}$$

*where* $U_{k\ell}$ *is a zero mean Gaussian random vector in* $\mathbb{R}^{1\times d}$ *independent of the limiting random variables corresponding to the variables in* $\overline{\mathfrak{G}}_{k\ell}^{+}$.

*Proof.* The proof is identical to that of [Rangan et al., 2019b, Lemmas 7,8]. ∎

We are now ready to prove Lemma 16.

*Proof of Lemma 16.* Using the partition (A.23a) and Lemmas 17 and 18, we see that the components of the vector sequences in $\overline{\mathfrak{G}}_{k\ell}^{+}$ along with $\mathbf{p}_{k\ell}^{+}$ almost surely converge jointly

empirically, where the components of $\mathbf{p}^+_{k\ell}$ have the limit

$$\mathbf{p}^+_{k\ell} = \mathbf{p}^{\text{det}}_{k\ell} + \mathbf{p}^{\text{ran}}_{k\ell} \overset{2}{\Longrightarrow} P^0_\ell \beta^0_\ell + \sum_{i=0}^{k-1} P^+_{i\ell}\beta^+_{i\ell} + U_{k\ell} =: P^+_{k\ell}. \tag{A.34}$$

Note that the above Wasserstein-2 convergence can be shown using the same arguments involved in showing that if $X_N|\mathcal{F} \overset{d}{\Longrightarrow} X|\mathcal{F}$, and $Y_N|\mathcal{F} \overset{d}{\Longrightarrow} c$, then $(X_N, Y_N)|\mathcal{F} \overset{d}{\Longrightarrow} (X, c)|\mathcal{F}$ for some constant $c$ and sigma-algebra $\mathcal{F}$.

We first establish the Gaussianity of $P^+_{k\ell}$. Observe that by the induction hypothesis, $\mathcal{H}^-_{k,\ell+1}$ holds whereby $(P^0_\ell, P^+_{0\ell}, \ldots, P^+_{k-1,\ell}, Q^-_{0,\ell+1}, \ldots, Q^-_{k,\ell+1})$, is jointly Gaussian. Since $U_k$ is Gaussian and independent of $(P^0_\ell, P^+_{0\ell}, \ldots, P^+_{k-1,\ell}, Q^-_{0,\ell+1}, \ldots, Q^-_{k,\ell+1})$, we can conclude from (A.34) that $(P^0_\ell, P^+_{0\ell}, \ldots, P^+_{k-1,\ell}, P^+_{k\ell}, Q^-_{0,\ell+1}, \ldots, Q^-_{k,\ell+1})$ is jointly Gaussian.

We now need to prove the correlations of this jointly Gaussian random vector are as claimed by $\mathcal{H}^+_{k,\ell+1}$. Since $\mathcal{H}^-_{k,\ell+1}$ is true, we know that (A.15) is true for all $i = 0, \ldots, k-1$ and $j = 0, \ldots, k$ and $\ell = \ell + 1$. Hence, we need only to prove the additional identity for $i = k$, namely the equations: $\text{Cov}(P^0_\ell, P^+_{k\ell})^2 = \mathbf{K}^+_{k\ell}$ and $\mathbb{E}(P^+_{k\ell} Q^-_{j,\ell+1}) = 0$. First observe that

$$\mathbb{E}(P^{+\mathsf{T}}_{k\ell} P^+_{k\ell})^2 \overset{(a)}{=} \lim_{N_\ell \to \infty} \tfrac{1}{N_\ell} \mathbf{p}^{+\mathsf{T}}_{k\ell} \mathbf{p}^+_{k\ell} \overset{(b)}{=} \lim_{N_\ell \to \infty} \tfrac{1}{N_\ell} \mathbf{q}^{+\mathsf{T}}_{k\ell} \mathbf{q}^+_{k\ell} \overset{(c)}{=} \mathbb{E}\left(Q^{+\mathsf{T}}_{k\ell} Q^+_{k\ell}\right)^2$$

where (a) follows from the fact that the rows of $\mathbf{p}^+_{k\ell}$ converge empirically to $P^+_{k\ell}$; (b) follows from line 20 in Algorithm 4 and the fact that $\mathbf{V}_\ell$ is orthogonal; and (c) follows from the fact that the rows of $\mathbf{q}^+_{k\ell}$ converge empirically to $Q^+_{k\ell}$ from hypothesis $\mathcal{H}^+_{k,\ell}$. Since $\mathbf{p}^0_\ell = \mathbf{V}_\ell \mathbf{q}^0$, we similarly obtain that $\mathbb{E}(P^{0\mathsf{T}}_\ell P^+_{k\ell}) = \mathbb{E}(Q^{0\mathsf{T}}_\ell Q^+_{k\ell}), \quad \mathbb{E}(P^{0\mathsf{T}}_\ell P^0_\ell) = \mathbb{E}(Q^{0\mathsf{T}}_\ell Q^0_\ell)$, from which we conclude

$$\text{Cov}(P^0_\ell, P^+_{k\ell}) = \text{Cov}(Q^0_\ell, Q^+_{k\ell}) =: \mathbf{K}^+_{k\ell}, \tag{A.35}$$

where the last step follows from the definition of $\mathbf{K}^+_{k\ell}$ in line 20 of Algorithm 5. Finally, we observe that for $0 \leqslant j \leqslant k$

$$\mathbb{E}(P^{+\mathsf{T}}_{k\ell} Q^-_{j,\ell+1}) \overset{(a)}{=} \beta^{0\mathsf{T}}_\ell \mathbb{E}(P^{0\mathsf{T}}_\ell Q^-_{j,\ell+1}) + \sum_{i=0}^{k-1} \beta^{+\mathsf{T}}_{i\ell} \mathbb{E}(P^{+\mathsf{T}}_{i\ell} Q^-_{j,\ell+1}) + \mathbb{E}(U^{\mathsf{T}}_{k\ell} Q^-_{j,\ell+1}) \overset{(b)}{=} \mathbf{0}, \tag{A.36}$$

where (a) follows from (A.34) and, in (b), we used the fact that $\mathbb{E}(P^{0\mathsf{T}}_\ell Q^-_{j,\ell+1}) = \mathbf{0}$ and $\mathbb{E}(P^{+\mathsf{T}}_{i\ell} Q^-_{j,\ell+1}) = \mathbf{0}$ since (A.15) is true for $i \leqslant k$–1 corresponding to $\mathcal{H}^-_{k,\ell+1}$ and $\mathbb{E}(U^{\mathsf{T}}_{k\ell} Q^-_{j,\ell+1}) = \mathbf{0}$

since $U_{k\ell}$ is independent of $\overline{\mathfrak{G}}^+_{k\ell}$, and $Q^-_{j,\ell+1}$ is $\overline{\mathfrak{G}}^+_{k\ell}$ measurable. Thus, with (A.35) and (A.36), we have proven all the correlations in (A.15) corresponding to $\mathcal{H}^+_{k,\ell+1}$.

Next, we prove the convergence of the parameter lists $\Upsilon^+_{k,\ell+1}$ to $\overline{\Upsilon}^+_{k,\ell+1}$. Since $\Upsilon^+_{k\ell} \to \overline{\Upsilon}^+_{k\ell}$ due to hypothesis $\mathcal{H}^+_{k\ell}$, and $\varphi^+_{k,\ell+1}(\cdot)$ is uniformly Lipschitz continuous, we have that $\lim_{N\to\infty} \mu^+_{k,\ell+1}$ from line 17 in Algorithm 4 converges almost surely as

$$\lim_{N\to\infty} \left\langle \boldsymbol{\varphi}^+_{k,\ell+1}(\mathbf{p}^0_\ell, \mathbf{p}^+_{k\ell}, \mathbf{q}^-_{k,\ell+1}, \mathbf{w}_{\ell+1}, \overline{\Upsilon}^+_{k\ell}) \right\rangle = \mathbb{E}\left[ \varphi^+_{k,\ell+1}(P^0_\ell, P^+_{k\ell}, Q^-_{k,\ell+1}, W_{\ell+1}, \overline{\Upsilon}^+_{k\ell}) \right] = \overline{\mu}^+_{k,\ell+1}, \tag{A.37}$$

where $\overline{\mu}^+_{k,\ell+1}$ is the value in line 17 in Algorithm 5. Since $T^+_{k,\ell+1}(\cdot)$ is continuous, we have that $\lambda^+_{k,\ell+1}$ in line 18 in Algorithm 4 converges as $\lim_{N\to\infty} \lambda^+_{k,\ell+1} = T^+_{k,\ell+1}(\overline{\mu}^+_{k,\ell+1}, \overline{\Upsilon}^+_{k\ell}) =: \overline{\lambda}^+_{k,\ell+1}$, from line 18 in Algorithm 5. Therefore, we have the limit

$$\lim_{N\to\infty} \Upsilon^+_{k,\ell+1} = \lim_{N\to\infty} (\Upsilon^+_{k,\ell}, \lambda^+_{k,\ell+1}) = (\overline{\Upsilon}^+_{k,\ell}, \overline{\lambda}^+_{k,\ell+1}) = \overline{\Upsilon}^+_{k,\ell+1}, \tag{A.38}$$

which proves the convergence of the parameter lists stated in $\mathcal{H}^+_{k,\ell+1}$. Finally, using (A.38), the empirical convergence of the matrix sequences $\mathbf{p}^0_\ell$, $\mathbf{p}^+_{k\ell}$ and $\mathbf{q}^-_{k,\ell+1}$ and the uniform Lipschitz continuity of the update function $f^+_{k,\ell+1}(\cdot)$ we obtain that $\mathbf{q}^+_{k,\ell+1}$ equals

$$\mathbf{f}^+_{k,\ell+1}(\mathbf{p}^0_\ell, \mathbf{p}^-_{k\ell}, \mathbf{q}^-_{k,\ell+1}, \mathbf{w}_{\ell+1}, \Upsilon^+_{k,\ell+1}) \overset{2}{\Rightarrow} f^+_{k,\ell+1}(P^0_\ell, P^-_{k\ell}, Q^-_{k,\ell+1}, W_{\ell+1}, \overline{\Upsilon}^+_{k,\ell+1}) =: Q^+_{k,\ell+1},$$

which proves the claim (A.16) for $\mathcal{H}^+_{k,\ell+1}$. This completes the proof. ∎

An overview of the iterates in Algorithm 4 is depicted in (TOP) and (MIDDLE) of Figure A.1. Theorem 4 shows that the rows of the iterates of Algorithm 4 converge empirically with $2^{\text{nd}}$ order moments to random variables defined in Algorithm 5. The random variables defined in Algo. 5 are depicted in Figure A.1 (BOTTOM).

**Algorithm 4** General Multi-Layer Matrix (Gen-ML-Mat) Recursion

**Require:** Initial matrix functions $\{\mathbf{f}_\ell^0\}$. Matrix update functions $\{\mathbf{f}_{k\ell}^\pm(\cdot)\}$. Parameter statistic functions $\{\boldsymbol{\varphi}_{k\ell}^\pm(\cdot)\}$. Parameter update functions $\{T_{k\ell}^\pm(\cdot)\}$. Orthogonal matrices $\{\mathbf{V}_\ell\}$. Perturbation variables $\{\mathbf{w}_\ell^\pm\}$. Initial matrices $\{\mathbf{q}_{0\ell}^-\}$. Initial parameter list $\Upsilon_{01}^-$.

// `Initial Pass`
$\mathbf{q}_0^0 = \mathbf{f}_0^0(\mathbf{w}_0), \quad \mathbf{p}_0^0 = \mathbf{V}_0\mathbf{q}_0^0$
**for** $\ell = 1, \ldots, L-1$ **do**
  $\mathbf{q}_\ell^0 = \mathbf{f}_\ell^0(\mathbf{p}_{\ell-1}^0, \mathbf{w}_\ell, \Upsilon_{01}^-)$
  $\mathbf{p}_\ell^0 = \mathbf{V}_\ell\mathbf{q}_\ell^0$
**end for**

**for** $k = 0, 1, \ldots$ **do**
  // `Forward Pass`
  $\lambda_{k0}^+ = T_{k0}^+(\mu_{k0}^+, \Upsilon_{0k}^-)$
  $\mu_{k0}^+ = \langle \boldsymbol{\varphi}_{k0}^+(\mathbf{q}_{k0}^-, \mathbf{w}_0, \Upsilon_{0k}^-) \rangle$
  $\Upsilon_{k0}^+ = (\Upsilon_{k1}^-, \lambda_{k0}^+)$
  $\mathbf{q}_{k0}^+ = \mathbf{f}_{k0}^+(\mathbf{q}_{k0}^-, \mathbf{w}_0, \Upsilon_{k0}^+)$
  $\mathbf{p}_{k0}^+ = \mathbf{V}_0\mathbf{q}_{k0}^+$
  **for** $\ell = 1, \ldots, L-1$ **do**
    $\lambda_{k\ell}^+ = T_{k\ell}^+(\mu_{k\ell}^+, \Upsilon_{k,\ell-1}^+)$
    $\mu_{k\ell}^+ = \langle \boldsymbol{\varphi}_{k\ell}^+(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k\ell}^-, \mathbf{w}_\ell, \Upsilon_{k,\ell-1}^+) \rangle$
    $\Upsilon_{k\ell}^+ = (\Upsilon_{k,\ell-1}^+, \lambda_{k\ell}^+)$
    $\mathbf{q}_{k\ell}^+ = \mathbf{f}_{k\ell}^+(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k\ell}^-, \mathbf{w}_\ell, \Upsilon_{k\ell}^+)$
    $\mathbf{p}_{k\ell}^+ = \mathbf{V}_\ell\mathbf{q}_{k\ell}^+$
  **end for**

  // `Backward Pass`
  $\lambda_{k+1,L}^- = T_{kL}^-(\mu_{kL}^-, \Upsilon_{k,L-1}^+)$
  $\mu_{kL}^- = \langle \boldsymbol{\varphi}_{kL}^-(\mathbf{p}_{k,L-1}^+, \mathbf{w}_L, \Upsilon_{k,L-1}^+) \rangle$
  $\Upsilon_{k+1,L}^- = (\Upsilon_{k,L-1}^+, \lambda_{k+1,L}^+)$
  $\mathbf{p}_{k+1,L-1}^- = \mathbf{f}_{kL}^-(\mathbf{p}_{L-1}^0, \mathbf{p}_{k,L-1}^+, \mathbf{w}_L, \Upsilon_{k+1,L}^-)$
  $\mathbf{q}_{k+1,L-1}^- = \mathbf{V}_{L-1}^\mathsf{T}\mathbf{p}_{k+1,L-1}$
  **for** $\ell = L-1, \ldots, 1$ **do**
    $\lambda_{k+1,\ell}^- = T_{k\ell}^-(\mu_{k\ell}^-, \Upsilon_{k+1,\ell+1}^-)$
    $\mu_{k\ell}^- = \langle \boldsymbol{\varphi}_{k\ell}^-(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k+1,\ell}^-, \mathbf{w}_\ell, \Upsilon_{k+1,\ell+1}^-) \rangle$
    $\Upsilon_{k+1,\ell}^- = (\Upsilon_{k+1,\ell+1}^-, \lambda_{k+1,\ell}^-)$
    $\mathbf{p}_{k+1,\ell-1}^- = \mathbf{f}_{k\ell}^-(\mathbf{p}_{\ell-1}^0, \mathbf{p}_{k,\ell-1}^+, \mathbf{q}_{k+1,\ell}^-, \mathbf{w}_\ell, \Upsilon_{k+1,\ell}^-)$
    $\mathbf{q}_{k+1,\ell-1}^- = \mathbf{V}_{\ell-1}^\mathsf{T}\mathbf{p}_{k+1,\ell-1}^-$
  **end for**
**end for**

**Algorithm 5** Gen-ML-Mat State Evolution (SE)

**Require:** Matrix update row-wise functions $f_\ell^0(\cdot)$ and $f_{k\ell}^\pm(\cdot)$, parameter statistic row-wise functions $\varphi_{k\ell}^\pm(\cdot)$, parameter update functions $T_{k\ell}^\pm(\cdot)$, initial parameter list limit: $\overline{\Upsilon}_{01}^-$, initial random variables $W_\ell$, $Q_{0\ell}^-$, $\ell = 0, \ldots, L-1$.

// `Initial pass`
$Q_0^0 = f_0^0(W_0, \overline{\Upsilon}_{01}^-), \quad P_0^0 \sim \mathcal{N}(0, \tau_0^0), \quad \tau_0^0 = \mathbb{E}(Q_0^0)^2$
**for** $\ell = 1, \ldots, L-1$ **do**
  $Q_\ell^0 = f_\ell^0(P_{\ell-1}^0, W_\ell, \overline{\Upsilon}_{01}^-)$
  $P_\ell^0 \sim \mathcal{N}(0, \tau_\ell^0), \quad \tau_\ell^0 = \mathrm{Cov}(Q_\ell^0)$
**end for**

**for** $k = 0, 1, \ldots$ **do**
  // `Forward Pass`
  $\overline{\lambda}_{k0}^+ = T_{k0}^+(\overline{\mu}_{k0}^+, \overline{\Upsilon}_{0k}^-)$
  $\overline{\mu}_{k0}^+ = \mathbb{E}(\varphi_{k0}^+(Q_{k0}^-, W_0, \overline{\Upsilon}_{0k}^-))$
  $\overline{\Upsilon}_{k0}^+ = (\overline{\Upsilon}_{k1}^-, \overline{\lambda}_{k0}^+)$
  $Q_{k0}^+ = f_{k0}^+(Q_{k0}^-, W_0, \overline{\Upsilon}_{k0}^+)$
  $(P_0^0, P_{k0}^+) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k0}^+), \quad \mathbf{K}_{k0}^+ = \mathrm{Cov}(Q_0^0, Q_{k0}^+)$
  **for** $\ell = 1, \ldots, L-1$ **do**
    $\overline{\lambda}_{k\ell}^+ = T_{k\ell}^+(\overline{\mu}_{k\ell}^+, \overline{\Upsilon}_{k,\ell-1}^+)$
    $\overline{\mu}_{k\ell}^+ = \mathbb{E}(\varphi_{k\ell}^+(P_{\ell-1}^0, P_{k,\ell-1}^+, Q_{k\ell}^-, W_\ell, \overline{\Upsilon}_{k,\ell-1}^+))$
    $\overline{\Upsilon}_{k\ell}^+ = (\overline{\Upsilon}_{k,\ell-1}^+, \overline{\lambda}_{k\ell}^+)$
    $Q_{k\ell}^+ = f_{k\ell}^+(P_{\ell-1}^0, P_{k,\ell-1}^+, Q_{k\ell}^-, W_\ell, \overline{\Upsilon}_{k\ell}^+)$
    $(P_\ell^0, P_{k\ell}^+) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{k\ell}^+), \quad \mathbf{K}_{k\ell}^+ = \mathrm{Cov}(Q_\ell^0, Q_{k\ell}^+)$
  **end for**

  // `Backward Pass`
  $\overline{\lambda}_{k+1,L}^- = T_{kL}^-(\overline{\mu}_{kL}^-, \overline{\Upsilon}_{k,L-1}^+)$
  $\overline{\mu}_{kL}^- = \mathbb{E}(\varphi_{kL}^-(P_{L-1}^0, P_{k,L-1}^+, W_L, \overline{\Upsilon}_{k,L-1}^+))$
  $\overline{\Upsilon}_{k+1,L}^- = (\overline{\Upsilon}_{k,L-1}^+, \overline{\lambda}_{k+1,L}^+)$
  $P_{k+1,L-1}^- = f_{kL}^-(P_{L-1}^0, P_{k,L-1}^+, W_L, \overline{\Upsilon}_{k+1,L}^-)$
  $Q_{k+1,L-1}^- \sim \mathcal{N}(0, \tau_{k+1,L-1}^-), \quad \tau_{k+1,L-1}^- = \mathrm{Cov}(P_{k+1,L-1}^-)$
  **for** $\ell = L-1, \ldots, 1$ **do**
    $\overline{\lambda}_{k+1,\ell}^- = T_{k\ell}^-(\overline{\mu}_{k\ell}^-, \overline{\Upsilon}_{k+1,\ell+1}^-)$
    $\overline{\mu}_{k\ell}^- = \mathbb{E}(\varphi_{k\ell}^-(P_{\ell-1}^0, P_{k,\ell-1}^+, Q_{k+1,\ell}^-, W_\ell, \overline{\Upsilon}_{k+1,\ell+1}^-))$
    $\overline{\Upsilon}_{k+1,\ell}^- = (\overline{\Upsilon}_{k+1,\ell+1}^-, \overline{\lambda}_{k+1,\ell}^-)$
    $P_{k+1,\ell-1}^- = f_{k\ell}^-(P_{\ell-1}^0, P_{k,\ell-1}^+, Q_{k+1,\ell}^-, W_\ell, \overline{\Upsilon}_{k+1,\ell}^-)$
    $Q_{k+1,\ell-1}^- \sim \mathcal{N}(0, \tau_{k+1,\ell-1}^-), \quad \tau_{k+1,\ell-1}^- = \mathrm{Cov}(P_{k+1,\ell-1}^-)$
  **end for**
**end for**

# Appendix B

# Appendix for Asymptotics of Ridge Regression in Convolutional Models

## B.1 Complex Normal Distribution

Complex normal is the distribution of a complex random variable whose imaginary and real parts are jointly Gaussian.

**Standard complex normal distribution.** A random variable $Z = X + \mathrm{i}Y$ where $X, Y \in \mathbb{R}$ has standard complex normal distribution represented by $\mathcal{CN}(0, 1)$ if

$$X, Y \sim \mathcal{N}(0, 1/2), \quad X \perp\!\!\!\perp Y.$$

**General complex Gaussian distribution.** A random vector $\mathbf{Z} = \mathbf{X} + \mathrm{i}\mathbf{Y}$ where $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^n$ has complex Gaussian distribution $\mathcal{CN}(\boldsymbol{\mu}, \boldsymbol{\Gamma}, \mathbf{C})$ if $\mathbf{X}$ and $\mathbf{Y}$ are jointly Gaussian with

$$\boldsymbol{\mu} = \mathbb{E}[\mathbf{Z}], \tag{B.1}$$

$$\boldsymbol{\Gamma} = \mathbb{E}[(\mathbf{Z} - \boldsymbol{\mu})(\mathbf{Z} - \boldsymbol{\mu})^{\mathsf{H}}], \tag{B.2}$$

$$\mathbf{C} = \mathbb{E}[(\mathbf{Z} - \boldsymbol{\mu})(\mathbf{Z} - \boldsymbol{\mu})^{\mathsf{T}}]. \tag{B.3}$$

The parameters $\boldsymbol{\mu}, \boldsymbol{\Gamma}$, and $\mathbf{C}$ are called mean vector, covariance matrix, and relation matrix respectively. Alternatively, if we define

$$\begin{aligned}
\mathbf{C}_{XX} &= \mathbb{E}[(\mathbf{X} - \boldsymbol{\mu}_X)(\mathbf{X} - \boldsymbol{\mu}_x)^{\mathsf{T}}], \quad \boldsymbol{\mu}_X = \mathbb{E}[\mathbf{X}], \\
\mathbf{C}_{YY} &= \mathbb{E}[(\mathbf{Y} - \boldsymbol{\mu}_Y)(\mathbf{Y} - \boldsymbol{\mu}_Y)^{\mathsf{T}}], \quad \boldsymbol{\mu}_X = \mathbb{E}[\mathbf{Y}], \\
\mathbf{C}_{XY} &= \mathbf{C}_{YX}^{\mathsf{T}} = \mathbb{E}[(\mathbf{X} - \boldsymbol{\mu}_X)(\mathbf{Y} - \boldsymbol{\mu}_Y)^{\mathsf{T}}],
\end{aligned}$$

then $\mathbf{X}, \mathbf{Y}$ are jointly Gaussian with distribution

$$(\mathbf{X}, \mathbf{Y}) \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_X \\ \boldsymbol{\mu}_Y \end{bmatrix}, \begin{bmatrix} \mathbf{C}_{XX} & \mathbf{C}_{XY} \\ \mathbf{C}_{YX} & \mathbf{C}YY \end{bmatrix}\right).$$

The matrices $\boldsymbol{\Gamma}$ and $\mathbf{C}$ are related to covariance matrices of $\mathbf{X}$ and $\mathbf{Y}$ through the following equations:

$$\begin{aligned}
\boldsymbol{\Gamma} &= \mathbf{C}_{XX} + \mathbf{C}_{YY} + \mathrm{i}(\mathbf{C}_{YX} - \mathbf{C}_{XY}), \\
\boldsymbol{\Gamma} &= \mathbf{C}_{XX} - \mathbf{C}_{YY} + \mathrm{i}(\mathbf{C}_{YX} + \mathbf{C}_{XY}).
\end{aligned}$$

## B.2 Empirical Convergence of Vector Sequences

Here we review some standard definitions that are widely used in the papers that use approximate message passing framework.

**Definition 9** (Pseudo Lipschitz Continuity)**.** A function $\mathbf{f}$ is called pseudo-Lipschitz continuous of order $p$ with constant $C$ if for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathrm{dom}(\mathbf{f})$

$$\|\mathbf{f}(\mathbf{x}_1) - \mathbf{f}(\mathbf{x}_2)\| \leqslant C \|\mathbf{x}_1 - \mathbf{x}_2\| (1 + \|\mathbf{x}_1\|^{p-1} + \|\mathbf{x}_2\|^{p-1}). \tag{B.4}$$

Note that for $p = 1$ this definition is a equivalent to the definition of the standard Lipschitz-continuity.

**Definition 10** (Uniform Lipschitz-continuity)**.** A function $\mathbf{f}$ on $\mathcal{X} \times \mathcal{W}$ is *uniformly Lipschitz-continuous* in $\mathbf{x}$ at $\bar{\boldsymbol{\omega}}$ if there exists constants $L_1, L_2 \geqslant 0$ and an open neighborhood $U$ of $\bar{\boldsymbol{\omega}}$

such that for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}, \boldsymbol{\omega} \in U$

$$\|\mathbf{f}(\mathbf{x}_1, \boldsymbol{\omega}) - \mathbf{f}(\mathbf{x}_2, \boldsymbol{\omega})\| \leqslant L_1 \|\mathbf{x}_1 - \mathbf{x}_2\|, \tag{B.5}$$

and for all $\mathbf{x} \in \mathcal{X}, \boldsymbol{\omega}_1, \boldsymbol{\omega}_2 \in \mathcal{W}$

$$\|\mathbf{f}(\mathbf{x}, \boldsymbol{\omega}_1) - \mathbf{f}(\mathbf{x}, \boldsymbol{\omega}_2)\| \leqslant L_2(1 + \|\mathbf{x}\|) \|\boldsymbol{\omega}_1 - \boldsymbol{\omega}_2\|. \tag{B.6}$$

**Definition 11** (Empirical convergence of sequences)**.** Consider a sequence of vectors $\mathbf{x}(N) = \{\mathbf{x}_n(N)\}_{n=1}^N$ with $\mathbf{x}_n(N) \in \mathbb{R}^d$, i.e. each $\mathbf{x}(N)$ is a block vector with a total of $Nd$ components. For a finite $p \geqslant 1$, we say that the vector sequence $\mathbf{x}(N)$ converges empirically with $p$th order moments if there exists a random variable $X \in \mathbb{R}^d$ such that

- $\mathbb{E}\|X\|_p^p < \infty$;
- for any $f : \mathbb{R}^d \to \mathbb{R}$ that is pseudo-Lipschitz of order $p$,

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(\mathbf{x}_n(N)) = \mathbb{E}[f(X)]. \tag{B.7}$$

With some abuse of notation, we represent this with

$$\lim_{N\to\infty} \mathbf{x}_n \overset{PL(p)}{=} X, \tag{B.8}$$

where we have omitted the dependence on $N$ to ease the notation. In this definition the sequence $\{\mathbf{x}(N)\}$ can be random or deterministic. If it is random we require the equality in (B.7) to hold almost surely. In particular, if the sequence $\{\mathbf{x}_n\}$ is i.i.d. with $\mathbf{x}_n \sim p_X(\cdot)$, with $\mathbb{E}\|X\|_p^p < \infty$, then $\{\mathbf{x}_n\}$ converges empirically to $X$ with $p$th order. The extension of this definition to sequence of matrices and higher order tensors is straightforward.

**Definition 12** (Convergence in distribution)**.** A sequence of random vectors $\mathbf{x}_n \in \mathbb{R}^d$ converges in distribution (also known as weak convergence) to $\mathbf{x}$ if for all bounded functions $f : \mathbb{R}^d \to \mathbb{R}$

$$\lim_{n\to\infty} \mathbb{E}f(\mathbf{x}_n) = \mathbb{E}f(\mathbf{x}). \tag{B.9}$$

PL(p) convergence is equivalent to convergence in distribution plus convergence of the $p$th moment [Bayati and Montanari, 2011a].

**Definition 13** (Wasserstein-$p$ distance)**.** Wasserstein-$p$ distance between two probability measures $\mu, \nu$ on Euclidean space $\mathbb{R}^d$ is

$$W_p(\mu,\nu) = \inf_{\gamma\in\Gamma} \left(\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\gamma} \|\mathbf{x}-\mathbf{y}\|_p^p\right)^{\frac{1}{p}}, \tag{B.10}$$

where $\Gamma$ is the set of all probability measures on the product space $\mathbb{R}^d \times \mathbb{R}^d$ with marginals $\mu$ and $\nu$.

PL(p) convergence is also equivalent to convergence the empirical measure of the sequence $\mathbf{x}_n$ to probability measure of $X$ in Wasserstein-$p$ distance [Villani, 2008].

## B.3 1D Convolution Operators in Matrix Form

In this section we derive the matrix form of 1D convolution operators to show how these operators look like in time domain. As we will see, convolution operators in time domain can be represented as a *doubly block circulant matrix*. Because of this structure, approximate message passing (AMP) (discussed in Appendix 2.3) cannot be directly used to obtain estimation error of ridge regression for convolutional inverse problem in time domain. This is due to the assumption in AMP that the measurement matrix has i.i.d. entries. If this assumption can be relaxed, we can analyze estimators other than ridge, and compute error metrics other than MSE. We hope to follow this direction in a future work.

First assume that in the convolutional model in (4.1), $n_x = n_y = 1$, i.e. the input and output both have one channel. Also for a matrix $\mathbf{Z} \in \mathbb{R}^{m\times n}$, let $\vec{(\mathbf{Z})} \in \mathbb{R}^{nm}$ represent the vector constructed by stacking $\mathbf{X}$ in a vector row by row. To simplify the notation, we zero pad the convolution kernel which in this case is a vector of size $k$, so that it will have size $T$ and we still use $\mathbf{K}$ to represent the zero-padded kernel to simplify the notation. In this case, the convolution operator $\mathbf{K} : \mathbf{X} \mapsto \mathbf{K} * \mathbf{X}$ can be represented as a circulant matrix

$\mathbf{C} : \mathrm{vec}(\mathbf{X}) \mapsto \mathbf{C}\,\mathrm{vec}(\mathbf{X})$

$$\mathbf{C} = \begin{bmatrix} K_1 & K_2 & K_3 & \dots & K_T \\ K_T & K_1 & K_2 & \dots & K_{T-1} \\ K_{T-1} & K_T & K_1 & \dots & K_{T-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ K_2 & K_3 & K_4 & \dots & K_1 \end{bmatrix} \tag{B.11}$$

When the number of input channels and output channels are $n_x$ and $n_y$ respectively, the convolution can be represented in matrix form as matrix with blocks of circulant matrices

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} & \dots & \mathbf{C}1, n_x \\ \mathbf{C}_{21} & \mathbf{C}_{22} & \dots & \mathbf{C}2, n_x \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{n_y,1} & \mathbf{C}_{n_y,2} & \dots & \mathbf{C}n_y, n_x \end{bmatrix}, \tag{B.12}$$

where each $\mathbf{C}_{ij}$ is a circulant matrix of the form (B.11) constructed from $\mathbf{K}_{ij*}$. Since the adjoint of a circulant matrix is also a circulant matrix, one can see that the adjoint of a 1D convolution (with stride 1) is also a convolution with respect to another kernel.

### B.3.1 AMP for ridge regression

In this section we show how AMP can be used to derive asymptotic error of ridge regression

$$\widehat{\mathbf{x}}_{\mathsf{ridge}} = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_2^2 . \tag{B.13}$$

The solution to this optimization problem is

$$\widehat{\mathbf{x}}_{\mathsf{ridge}} = (\mathbf{A}^\mathsf{T}\mathbf{A} + \lambda \mathbf{I})^{-1}\mathbf{A}^\mathsf{T}\mathbf{y}. \tag{B.14}$$

Next, consider the AMP recursion in (2.29) and (2.30) with a fixed denoiser $\boldsymbol{\eta}_t(\mathbf{x}) = \alpha\mathbf{x}$

$$\mathbf{x}^{t+1} = \alpha(\mathbf{A}^\mathsf{T}\mathbf{z}^t + \mathbf{x}^t), \tag{B.15}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t + \frac{\alpha}{\delta}\mathbf{z}^{t-1}. \tag{B.16}$$

The next lemma shows that this recursion solves the ridge regression for a specific regularization parameter $\lambda$.

**Lemma 19.** *The fixed point of AMP algorithm with $\boldsymbol{\eta}_t(\mathbf{x}) = \alpha\mathbf{x}$ is the solution of ridge regression with*

$$\lambda = \frac{(1-\alpha)(1-\alpha/\delta)}{\alpha}, \tag{B.17}$$

*where $\delta = n_y/n_x$.*

*Proof.* Let $\mathbf{x}^*$ and $\mathbf{y}^*$ denote the fixed points of the AMP recursion. Then we have

$$\mathbf{x}^* = \alpha(\mathbf{A}^\mathsf{T}\mathbf{z}^* + \mathbf{x}^*), \tag{B.18}$$

$$\mathbf{z}^* = \mathbf{y} - \mathbf{A}\mathbf{x}^* + \frac{\alpha}{\delta}\mathbf{z}^*. \tag{B.19}$$

Therefore,

$$\mathbf{z}^* = \frac{1}{1-\alpha/\delta}(\mathbf{y} - \mathbf{A}\mathbf{x}). \tag{B.20}$$

Plugging this back to Equation (B.18) we get

$$\mathbf{x}^* = \left(\mathbf{A}^\mathsf{T}\mathbf{A} + \frac{(1-\alpha)(1-\alpha/\delta)}{\alpha}\mathbf{I}\right)^{-1}\mathbf{A}^\mathsf{T}\mathbf{y}. \tag{B.21}$$

Comparing this to (B.14) proves the result. ∎

Given a $\lambda$, one can solve the quadratic equation (B.17) to find the $\alpha$ that satisfies the equation. This is a quadratic equation that has two solutions. As we show in Section B.3.2, so long as the regularization parameter $\lambda$ is non-negative, this quadratic equation always has two real and positive solutions. But only for the smaller solution the AMP recursions for solving ridge regression converges, and hence only the smaller one is valid.

Having found the $\alpha$ we can use the state evolution (2.33) to find its fixed point. For ridge regression, this can be done in closed form. The state evolution for ridge regression can be written as

$$\tau_{t+1}^2 = \sigma^2 + \frac{1}{\delta}((1-\alpha)^2\sigma_X^2 + \alpha^2\tau_t^2), \tag{B.22}$$

If we define the fixed point value $\tau := \lim_{t\to\infty} \tau_t$ we have that it should satisfy

$$\tau^2 = \sigma^2 + \frac{1}{\delta}((1-\alpha)^2\sigma_X^2 + \alpha^2\tau^2), \tag{B.23}$$

from which we obtain

$$\tau^2 = \frac{\sigma^2 + \frac{1}{\delta}(1-\alpha^2)\sigma_X^2}{1 - \frac{\alpha^2}{\delta}}. \tag{B.24}$$

The mean squared error then can be obtained as

$$\begin{aligned}\frac{1}{n_x}\left\|\widehat{\mathbf{x}}_{\mathsf{ridge}} - \mathbf{x}_0\right\|_2^2 &= \mathbb{E}\left[(\alpha(X_0 + \tau Z) - X_0)^2\right] \\ &= (\alpha-1)^2\mathbb{E}X_0^2 + \alpha^2\tau^2.\end{aligned}$$

### B.3.2 Convergence of AMP

As mentioned in the previous section, when we use AMP to find the solution of ridge regression, we first need to find an $\alpha$ that satisfies Equation (B.17). This is a quadratic equation that has two solutions. In theory, the solution of ridge regression with a given $\lambda$ is the fixed points of AMP iterations for both values of $\alpha$. However, we should also note that the results of AMP are only valid if the iterations converge to a fixed point. This is equivalent to stability of the dynamics corresponding to AMP recursion. We saw in Lemma 19 that a linear denoiser $\eta_t(\mathbf{x}) = \alpha\mathbf{x}$ can be used to solve for a ridge regression with regularization parameter $\lambda$. Recall that the AMP iterations for this denoiser are

$$\mathbf{x}^{t+1} = \alpha(\mathbf{A}^\mathsf{T}\mathbf{z}^t + \mathbf{x}^t) \tag{B.25}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t + \frac{\alpha}{\delta}\mathbf{z}^{t-1}. \tag{B.26}$$

Plugging Equation (B.26) in Equation (B.25) we get

$$\mathbf{x}^{t+1} = \alpha(\mathbf{I} - \mathbf{A}^\mathsf{T}\mathbf{A})\mathbf{x}^t + \frac{\alpha^2}{\delta}\mathbf{z}^{t-1} + \alpha\mathbf{A}^\mathsf{T}\mathbf{y}, \tag{B.27}$$

$$\mathbf{z}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t + \frac{\alpha}{\delta}\mathbf{z}^{t-1}. \tag{B.28}$$

These equations correspond to a linear time invariant system with state matrix

$$\mathcal{A} = \begin{bmatrix} \alpha(\mathbf{I} - \mathbf{A}^\mathsf{T}\mathbf{A}) & \frac{\alpha^2}{\delta}\mathbf{A}^\mathsf{T} \\ -\mathbf{A} & \frac{\alpha}{\delta}\mathbf{I} \end{bmatrix}. \tag{B.29}$$

The system is stable if and only if all the eigenvalues of $\mathcal{A}$ lie inside the unit disk. A simple row operation (which does not change the eigenvalues) shows that the eigenvalues of $\mathcal{A}$ are the same as eigenvalues of

$$\mathcal{A}' = \begin{bmatrix} \alpha\mathbf{I} & \mathbf{0} \\ -\mathbf{A} & \frac{\alpha}{\delta}\mathbf{I} \end{bmatrix}. \tag{B.30}$$

Therefore, the AMP recursions in (B.25) and (B.26) are stable, i.e. converge to the fixed points corresponding to the ridge regression if and only if

$$|\alpha| \leqslant 1, \quad |\frac{\alpha}{\delta}| \leqslant 1. \tag{B.31}$$

Since $\delta > 0$, this is equivalent to

$$|\alpha| \leqslant \min(1, \delta). \tag{B.32}$$

If regularization parameter $\lambda \geqslant 0$, solving the quadratic equation (B.17) for $\alpha$, it is not hard to show that it has two solutions $\alpha_1, \alpha_2$ that are always real and satisfy

$$0 < \alpha_1 \leqslant \min(1, \delta) \leqslant \max(1, \delta) \leqslant \alpha_2. \tag{B.33}$$

Comparing this to (B.32), we see that only $\alpha_1$ satisfies the stability condition. To summarize, (B.17) always has two real positive solutions, but only the smaller one satisfies the stability condition.

As a sanity check, we can also verify that if AMP iterations for ridge regression in (B.25) and (B.26) are stable, so is the state evolution recursion. The state evolution for ridge regression is given in (B.22). This is a scalar linear time invariant system that is stable if and only if

$$-1 \leqslant \frac{\alpha^2}{\delta} \leqslant 1. \tag{B.34}$$

Clearly, the stability conditions in (B.31) imply this inequality. Therefore, the stability of

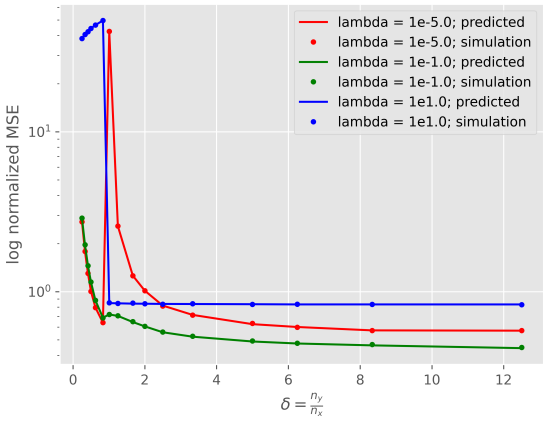



**Figure B.1:** Log of normalized error for the AR(1) features with the process noise $\mathcal{N}(0, s^2)$, with respect to $\delta = n_y/n_x$ for three different values of $\lambda$. The figure is essentially indistinguishable from Figure 4.2.

AMP recursions for ridge regression also implies the stability of the state evolution for ridge regression. As a result, the smaller value of $\alpha$ that satisfies (B.17) should be used to get the correct prediction of error.

### B.3.3 AMP for complex ridge regression

Approximate message passing can also be used when the signals in (2.26) are complex valued. So long as the sensing matrix has i.i.d. complex normal entries $\mathbf{A}_{ij} \sim \mathcal{CN}(0, \sigma_A^2/n_y)$ (see Appendix B.1 for a brief overview of complex normal distribution), i.e. the real and imaginary parts of each entry are i.i.d. Gaussian random variables with variance $\sigma_A^2/(2n_y)$ and independent of each other, the state evolution holds [Maleki et al., 2013]. Therefore, by changing all variables to complex variables, we can use AMP exactly as in Appendix B.3.1 and get the asymptotic error of complex ridge regression using the state evolution almost without any changes.

## B.4 Experiment with Gaussian AR(1) Process

As mentioned in the experiments, for an AR(1) process as in (4.37), the auto-correlation function derived in Equation (4.40) does not depend on the distribution of the noise $\xi_t$, but only its second moment. This is true in general for an AR($p$) process that evolves as a linear time-invariant (LTI) system driven with zero-mean i.i.d. noise. For such processes the auto-correlation only depends on the second order statistics of the noise as well parameters of the linear system. Therefore, we expect to get identical results in the limit if the any zero mean noise is driving the process so long as the variances match. In Figure 4.2, we showed the results for the case where the noise was a scaled Rademacher random variable. Figure B.1 shows the same results for the case where the noise is Gaussian with the matched variance. As expected, this plot is almost indistinguishable from Figure 4.2.

# Appendix C

# Appendix for Generalized Autoregressive Linear Models for Discrete High-dimensional Data

## C.1 Proofs of Lemmas in Sections 5.5 and 5.6

**Lemma 20.** *Under (A1)–(A3),* $\Theta_i^* \in \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\ \mathbb{E}\,\mathcal{L}_i(\boldsymbol{\beta})$.

*Proof.* This is a direct consequence of Lemma 10 and assumption (A3). Notice that from Lemma 9 we have

$$\mathcal{L}_i(\Theta_i^* + \Delta_i) \geqslant \mathcal{L}_i(\Theta^*) + \langle \nabla \mathcal{L}_i(\Theta_i^*), \Delta_i \rangle + \mathcal{E}(\Delta_i; \mathbb{X}).$$

Taking expectations on both sides, and applying lemma 10, we get

$$\mathbb{E}\mathcal{L}_i(\Theta_i^* + \Delta_i) \geqslant \mathbb{E}\mathcal{L}_i(\Theta^*) + \langle \mathbb{E}\nabla \mathcal{L}_i(\Theta_i^*), \Delta_i \rangle + C_{\mathscr{X}}^2 \|\Delta_i\|_F^2.$$

It follows from Assumption (A3) that $\mathbb{E}\nabla\mathcal{L}_i(\Theta^*) = 0$. Thus we get

$$\mathbb{E}\mathcal{L}_i(\Theta_i^* + \Delta_i) \geqslant \mathbb{E}\mathcal{L}_i(\Theta^*)$$

for all $\Delta_i \in \mathbb{R}^{N \times L}$, which proves the claim. ∎

| | |
|---|---|
| $n$ | number of samples |
| $N$ | number of variables |
| $p$ | number of lags |
| $L$ | number of filters |
| $[n]$ | $\{1, 2, \ldots, n\}$ |
| $i, j$ | index of variable $\in [N]$ |
| $k$ | index of lag $\in [p]$ |
| $\ell$ | index of filter $\in [L]$ |
| $t$ | index of sample $\in [n]$ |
| $\mathcal{X}_i$ | discrete subset of $\mathbb{R}$ |
| $\mathcal{X}_i^{\times p}$ | discrete subset of $\mathbb{R}^p$ |
| | $\{(a_1, a_2, \ldots, a_p) : a_k \in \mathcal{X}_i, \forall\, k \in [p]\}$ |
| $\mathcal{X}$ | discrete subset of $\mathbb{R}^N$ |
| | $\prod_{i\in[N]} \mathcal{X}_i$ |
| $\mathcal{X}^{\times p}$ | discrete subset of $\mathbb{R}^{N\times p}$ |
| | $\prod_{i\in[N]} \mathcal{X}_i^{\times p}$ |
| $x_i^t$ | (scalar) $\in \mathcal{X}_i$ |
| $\boldsymbol{x}^t$ | $(x_i^t)_{i=1}^N \in \mathcal{X}$ |
| $\mathbf{x}_j^{t-*}$ | $(x_j^{t-k})_{k=1}^p \in \mathcal{X}_i^{\times p}$ |
| $\boldsymbol{d}_\ell$ | filter, $\in \mathbb{R}^p$ |
| $\mathbf{X}^t$ | $p$-lag history at sample $t$, $\in \mathcal{X}^{\times p}$ |
| | $\begin{bmatrix}\boldsymbol{x}^t & \boldsymbol{x}^{t-1} & \ldots & \boldsymbol{x}^{t-p+1}\end{bmatrix}$ |
| $\mathbf{D}$ | dictionary, $\in \mathbb{R}^{p\times L}$ |
| | $\begin{bmatrix}\boldsymbol{d}_1 & \boldsymbol{d}_2 & \ldots & \boldsymbol{d}_L\end{bmatrix}$ |
| $\Theta_i$ | ($N \times L$ matrix) parameter for variable $i$ |
| $\mathcal{U}$ | subset of variables $\subseteq [N]$ |
| $\Theta_\mathcal{U}$ | ($\lvert\mathcal{U}\rvert \times N \times L$ tensor) $(\Theta_i)_{i\in\mathcal{U}}$ |
| $\Theta$ | ($N \times N \times L$ tensor) $\Theta_\mathcal{U}$ w/ $\mathcal{U} = [N]$ |

**Table C.1:** List of notations used in the Chapter 5.

### C.1.1 Choice of regularization hyperparameter

**Lemma 21.** *For any constant $c_1 > 2$,*

$$\|\nabla \mathcal{L}_\mathcal{U}(\Theta_\mathcal{U}^*)\|_{\infty,\infty,\infty} \leqslant BC_\mathcal{L}C_\mathbf{D}\sqrt{c_1 \log(|\mathcal{U}|NL)/n} \tag{C.1}$$

*with probability at least $1 - (|\mathcal{U}|NL)^{-c}$, where $c = c_1/2 - 1$.*

*Proof.* Fix $i, j \in [N]$ and $\ell \in [L]$. Then we have,

$$\frac{\partial \mathcal{L}_i(\Theta_i)}{\partial \Theta_{ij\ell}} = \frac{1}{n}\sum_{t=1}^n \mathcal{L}'_{it}(x_i^t, i{+}\Theta_i, \mathbf{X}^{t-1}\mathbf{D})\,(\mathbf{X}^{t-1}\mathbf{D})_{j\ell} = \frac{1}{n}\sum_{t=1}^n \mathcal{L}'_{it}(x_i^t, i{+}\Theta_i, \mathbf{X}^{t-1}\mathbf{D})\langle \mathbf{x}_j^{t-*}, \boldsymbol{d}_\ell\rangle$$

where $\mathcal{L}'_{it}(u,v) := \partial_v \mathcal{L}_{it}(u,v)$ It follows that

$$\frac{\partial \mathcal{L}_i(\Theta_i^*)}{\partial \Theta_{ij\ell}} = \frac{1}{n}\sum_{t=1}^{n} D_{ij\ell}^t \quad \text{where} \quad D_{ij\ell}^t := \mathcal{L}'_{it}(x_i^t, \langle \Theta_i^*, \mathbf{X}^{t-1}\mathbf{D}\rangle)\langle \boldsymbol{x}_j^{t-*}, \boldsymbol{d}_\ell\rangle.$$

Let $\mathcal{F}^{t-1} = \sigma(\boldsymbol{x}^{t-1}, \boldsymbol{x}^{t-2}, \dots)$ be the $\sigma$-field generated by the past observations of the process. From assumption (A3), we have $\mathbb{E}[\mathcal{L}'_{it}(x_i^t, \langle \Theta_i^*, \mathbf{X}^{t-1}\mathbf{D}\rangle) \mid \mathcal{F}^{t-1}] = 0$, hence

$$\mathbb{E}[D_{ij\ell}^t \mid \mathcal{F}^{t-1}] = 0.$$

That is, $\{D_{ij\ell}^t\}_t$ is a *martingale difference sequence*. Similarly, by assumption (A3), we get $\|\mathcal{L}'_{it}\|_\infty \leqslant C_{\mathcal{L}}$. If follows that $\{D_{ij\ell}^t\}_t$ is also bounded, i.e., $|D_{ij\ell}^t| \leqslant C_{\mathcal{L}} \cdot C_D$. By the Azuma–Hoeffding inequality for martingale differences [van de Geer, 2002],

$$\mathbb{P}\left(\Big|\frac{\partial \mathcal{L}(\Theta^*)}{\partial \Theta_{ij\ell}}\Big| > t\right) = \mathbb{P}\left(\Big|\frac{1}{n}\sum_{i=1}^{n} D_{ij\ell}^t\Big| > t\right) \leqslant 2\exp\Big(-\frac{nt^2}{2C_{\mathcal{L}}^2 C_D^2}\Big), \quad t > 0.$$

Writing $\|\nabla \sum_i \mathcal{L}(\Theta_i^*)\|_{\infty,\infty,\infty} = \sup_{ij\ell} |\frac{\partial \mathcal{L}_i(\Theta_i^*)}{\partial \Theta_{ij\ell}}|$, by the union bound we have,

$$\mathbb{P}\Big(\|\nabla \sum_{i\in\mathcal{U}} \mathcal{L}_i(\Theta_i^*)\|_{\infty,\infty,\infty} > t\Big) \leqslant 2|\mathcal{U}|NL \cdot \exp\Big(-\frac{nt^2}{2C_{\mathcal{L}}^2 \cdot C_D^2}\Big) \leqslant \delta, \quad t > 0.$$

Taking $t = C_{\mathcal{L}} \cdot C_D\sqrt{2\log(|\mathcal{U}|NL/\delta)/n}$ with $\delta = (|\mathcal{U}|NL)^{-c}$ establishes the result. ∎

### C.1.2 Quadratic lower bound on Remainder terms: Proof of Lemma 9

Fix $i \in \mathcal{U}$. Recall that the loss $\mathcal{L}_i$ can be written as

$$\mathcal{L}_i(\Theta_i) = \frac{1}{n}\sum_{t=1}^{n} \mathcal{L}_{i,t}(x_i^t, i + \Theta_i, \mathbf{X}^{t-1}\mathbf{D})$$

We have

$$\begin{aligned}\frac{\partial^2 \mathcal{L}_i(\Theta_i)}{\partial \Theta_{iab}\partial \Theta_{ik\ell}} &= \frac{1}{n}\sum_{t=1}^{n} \mathcal{L}''_{i,t}(i+\Theta_i, \mathbf{X}^{t-1}\mathbf{D})\,(\mathbf{X}^{t-1}\mathbf{D})_{ab}(\mathbf{X}^{t-1}\mathbf{D})_{k\ell} \\ &= \frac{1}{n}\sum_{t=1}^{n} \mathcal{L}''_{i,t}(i+\Theta_i, \mathbf{X}^{t-1}\mathbf{D})\langle \mathbf{x}_a^{t-*}, \boldsymbol{d}_b\rangle\langle \mathbf{x}_k^{t-*}, \boldsymbol{d}_\ell\rangle.\end{aligned}$$

Let $\nabla^2 \mathcal{L}_i(\Theta_i) \in \mathbb{R}^{(N\times L)\times(N\times L)}$ denote the Hessian matrix of $\mathcal{L}_i$, i.e.

$$\nabla^2 \mathcal{L}_i(\Theta_i) = \left[\frac{\partial^2 \mathcal{L}_i(\Theta_i)}{\partial \Theta_{iab}\partial \Theta_{ik\ell}}\right], \quad (a,b) \in [N]\times[L], (k,\ell) \in [N]\times[L],$$

and define the vector $\mathbf{h}^t := [\langle \mathbf{x}_a^{t-*}, \boldsymbol{d}_b \rangle] \in \mathbb{R}^{N \times L}$. Then we have

$$\nabla^2 \mathcal{L}_i(\Theta_i) = \frac{1}{n} \sum_{t=1}^{n} \mathcal{L}''_{i,t}(i{+}\Theta_i, \mathbf{X}^{t-1}\mathbf{D}) \mathbf{h}^t \mathbf{h}^{t\top}. \tag{C.2}$$

Hence, for all $\Theta_i, \boldsymbol{\beta} \in \mathbb{R}^{N \times L}$, the quadratic form of the Hessian of $\mathcal{L}_i$ satisfies

$$\begin{aligned} i{+}\boldsymbol{\beta} \nabla^2 \mathcal{L}_i(\Theta_i), \boldsymbol{\beta} &= \frac{1}{n} \sum_{t=1}^{n} \mathcal{L}''_{i,t}(i{+}\Theta_i, \mathbf{X}^{t-1}\mathbf{D}) \operatorname{vec}(\boldsymbol{\beta})^\top \mathbf{h}^t \mathbf{h}^{t\top} \operatorname{vec}(\boldsymbol{\beta}) \\ &= \frac{1}{n} \sum_{t=1}^{n} \mathcal{L}''_{i,t}(i{+}\Theta_i, \mathbf{X}^{t-1}\mathbf{D}) i{+}\boldsymbol{\beta}, \mathbf{X}^{t-1}\mathbf{D}^2 \\ &\overset{(i)}{\geqslant} \frac{\kappa_i}{n} \sum_{t=1}^{n} i{+}\boldsymbol{\beta}, \mathbf{X}^{t-1}\mathbf{D}^2 := \kappa_i \mathcal{E}(\boldsymbol{\beta}; \mathbb{X}), \end{aligned} \tag{C.3}$$

where $\operatorname{vec}(\beta)$ represents the vectorized form of the matrix $\boldsymbol{\beta}$ (in the same order as rows/columns of $\nabla^2 \mathcal{L}_i$), and inequality (i) follows from $\mathcal{L}''_{i,t}(x_i^t, \cdot) \geqslant \kappa_i > 0$, which holds by Assumption (A2).

Next, consider the function $f(t) := \mathcal{L}(\Theta_i^* + t\boldsymbol{\beta})$. By Taylor's Theorem we have

$$f(1) - f(0) - f'(0) = \frac{1}{2} f''(\xi), \quad \text{for some } \xi \in [0, 1].$$

Therefore, there exist a $\xi \in [0, 1]$ such that

$$R\mathcal{L}_i(\boldsymbol{\beta}; \Theta_i^*) = \frac{1}{2} i{+}\boldsymbol{\beta} \nabla^2 \mathcal{L}_i(\Theta_i^* + \xi\boldsymbol{\beta}), \boldsymbol{\beta} \geqslant \frac{\kappa_i}{2} \mathcal{E}(\boldsymbol{\beta}; \mathbb{X}),$$

where the last inequality follows from (C.3). This completes the proof. □

### C.1.3 Uniform lower bound on $\mathbb{E}\mathcal{E}(\beta; \mathbb{X})$: Proof of Lemma 10

Using the notation in (C.10) and (C.11), equation (C.12) implies

$$\mathbb{E}\mathcal{E}(\Delta; \mathbb{X}) = \mathbb{E} \left\| \mathsf{X}_{t*} \mathtt{S}(\boldsymbol{\beta}) \right\|_2^2 \quad \text{for all } t,$$

since by assumption the process is wide-sense stationary (i.e., the second moments of the distribution of $\mathsf{X}_{t*}$ is the same for all $t$). Recall the stacking operator $\mathtt{S}(\boldsymbol{\beta}) \in \mathbb{R}^{NL}$ defined in (C.11), and let $\mathbf{R} := \mathbb{E}\mathsf{X}_{t*}^\top \mathsf{X}_{t*} \in \mathbb{R}^{NL \times NL}$ be the population autocorrelation matrix, again independent of $t$ by stationarity. Then,

$$\mathbb{E}\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) = \mathbb{E} \left\| \mathsf{X}_{t*} \mathtt{S}(\boldsymbol{\beta}) \right\|_2^2 = \mathbb{E} \operatorname{tr} \left( \mathsf{X}_{t*}^\top \mathsf{X}_{t*} \mathtt{S}(\boldsymbol{\beta}) \mathtt{S}(\boldsymbol{\beta})^\top \right) = \operatorname{tr} \left( \mathbf{R} \mathtt{S}(\boldsymbol{\beta}) \mathtt{S}(\boldsymbol{\beta})^\top \right).$$

Since $\mathbf{R} - \lambda_{\min}(\mathbf{R})I \succeq 0$, we have that

$$\mathbb{E}\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) \;\geqslant\; \lambda_{\min}(\mathbf{R}) \left\| \mathtt{S}(\boldsymbol{\beta}) \right\|_2^2 . \tag{C.4}$$

We note that $\mathbf{R}$ is a block symmetric matrix with blocks $\mathbf{R}_{ij} := \mathbb{E}[\boldsymbol{x}^{t-i}(\boldsymbol{x}^{t-j})^\top] \in \mathbb{R}^{N \times N}$. We also note that due to the stationarity, $\mathbf{R}_{ij}$ only depends on $i-j$, so with some abuse of notation we write $\mathbf{R}_{ij} = \mathbf{R}_{i-j}$, i.e., $\mathbf{R}$ is block Toeplitz. Let $\mathbf{C}_{i-j}$ denote the centered autocorrelation matrix $\mathbb{E}[(\boldsymbol{x}^{t-i} - \mathbb{E}\boldsymbol{x}^t)(\boldsymbol{x}^{t-j} - \mathbb{E}\boldsymbol{x}^t)^\top]$, whereby $\mathbf{R}_{i-j} = \mathbf{C}_{i-j} + \mathbb{E}\boldsymbol{x}^t(\mathbb{E}\boldsymbol{x}^t)^\top$. Define $\mathbf{C}$ similarly as a block Toeplitz matrix with $\mathbf{C}_{ij} = \mathbf{C}_{i-j}$. Consequently $\lambda_{\min}(\mathbf{R}) \geqslant \lambda_{\min}(\mathbf{C})$.

Let $\mathscr{X}(\omega) \in \mathbb{C}^{N \times N}$ be the power spectrum matrix of the process as in assumption (A1) so that

$$\mathbf{C}_\ell := \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathscr{X}(\omega)\, e^{j\omega\ell} d\omega, \tag{C.5}$$

Also, recall from assumption (A1) that

$$C_{\mathscr{X}}^2 := \min_{\omega \in [-\pi, \pi)} \lambda_{\min}(\mathscr{X}(\omega)) > 0. \tag{C.6}$$

It is well-known that $\lambda_{\min}(\mathbf{C}) \geqslant C_{\mathscr{X}}^2$. See for example [Basu et al., 2015, Proposition 2.3] or [Gray et al., 2006, Lemma 4.1]. For completeness, we prove this assertion below. This together with equation (C.4) and $\left\| \mathtt{S}(\boldsymbol{\beta}) \right\|_2^2 = \|\boldsymbol{\beta}\|_F^2$ proves Lemma 10. □

**Proof of $\lambda_{\min}(\mathbf{C}) \geqslant C_{\mathscr{X}}^2$**

Fix $\mathbf{u}^\top = \begin{bmatrix} u_0^\top & u_1^\top & \dots & u_{p-1}^\top \end{bmatrix}$, where $u_i \in \mathbb{R}^N$ and set $G(\omega) = \frac{1}{\sqrt{2\pi}} \sum_{r=0}^{p-1} u_r e^{-jr\omega}$. Then, $\mathbf{u}^\top \mathbf{C} \mathbf{u}$ equals,

$$\sum_{r,s=0}^{p-1} u_r^\top \mathbf{C}_{r-s} u_s = \sum_{r,s=0}^{p-1} u_r^\top \Big[ \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathscr{X}(\omega) e^{j(r-s)\omega} d\omega \Big] u_s = \int_{-\pi}^{\pi} G^{\mathsf{H}}(\omega) \mathscr{X}(\omega) G(\omega) d\omega. \tag{C.7}$$

Since $\mathscr{X}(\omega)$ is a Hermitian matrix, $G^{\mathsf{H}}(\omega)\mathscr{X}(\omega)G(\omega)$ is always a real matrix. Moreover, we have that

$$G^{\mathsf{H}}(\omega)\mathscr{X}(\omega)G(\omega) \geqslant \lambda_{\min}(\mathscr{X}(\omega))\, G^{\mathsf{H}}(\omega)G(\omega) \geqslant C_{\mathscr{X}}^2\, G^{\mathsf{H}}(\omega)G(\omega)$$

hence

$$\mathbf{u}^\top \mathbf{C}\mathbf{u} \geqslant C_{\mathscr{X}}^2 \int_{-\pi}^{\pi} G^{\mathsf{H}}(\omega) G(\omega) d\omega = C_{\mathscr{X}}^2 \sum_{r,s=0}^{p-1} u_r^\top (\delta_{r-s} I_N) u_s = C_{\mathscr{X}}^2 \left\|\mathbf{u}\right\|_2^2,$$

by Parseval's theorem. (Alternatively, reverse the operation in (C.7) with $\mathscr{X}(\omega) = 1 \cdot I_N$ and recall that the inverse of a flat spectrum is the delta function). Here, $\delta_x = 1\{x = 0\}$. Taking the minimum over $\left\|\mathbf{u}\right\|_2 = 1$ completes the proof. □

### C.1.4 Proof of Lemma 13

We start by stating a general result that for sets $A, B, \{A_i\}_{i=1}^N, \{B_i\}_{i=1}^N$ from a $\sigma$-algebra such that (i) $\bigcap_i A_i \subseteq A \subseteq B$, and (ii) $B_i \subseteq A_i$ for all $i$, then

$$\mathbb{P}(B) \geqslant \mathbb{P}(A) \geqslant \mathbb{P}\Big(\bigcap_i A_i\Big) \geqslant 1 - \sum_{i=1}^N \mathbb{P}(A_i^c) \geqslant 1 - N \max_i \mathbb{P}(A_i^c) \geqslant 1 - N \max_i \mathbb{P}(B_i^c). \quad \text{(C.8)}$$

The first two inequalities follows from (i), the third inequality is the union bound to $\mathbb{P}(\bigcap_i A_i) = 1 - \mathbb{P}(\cup_i A_i^c)$. The last inequality follows from (ii).

Recall that $Y_i > a_i X_i$, and consider the set definitions $B_i = \{X_i > b_i - d_i\}$, $A_i = \{a_i X_i > (\min_i a_i) b_i - (\max_i a_i) d_i\}$, $A = \{\sum_i a_i X_i > (\min_i a_i) \sum_i b_i - (\max_i a_i) \sum_i d_i\}$ and $B = \{\sum_i Y_i > (\min_i a_i) \sum_i b_i - (\max_i a_i) \sum_i d_i\}$ which satisfy the above inclusion for $a_i, b_i, d_i > 0$. The lemma follows immediately from (C.8). □

## C.2 Uniform law for $\mathcal{E}(\beta; \mathbb{X})$: Proof of Lemma 12

For the current proof, we have fixed $i \in [N]$. We also use the notation $\left\|\boldsymbol{\beta}\right\|_q := \left\|\boldsymbol{\beta}\right\|_{q,q}$ for the $\ell_q$ norm of a matrix $\boldsymbol{\beta} \in \mathbb{R}^{N \times L}$. Note that $\left\|\boldsymbol{\beta}\right\|_2 = \left\|\boldsymbol{\beta}\right\|_F$. We also use the following notation.

$$\begin{aligned}
&\mathbb{B}_1(r) := \{\boldsymbol{\beta} \in \mathbb{R}^{N \times L} : \left\|\boldsymbol{\beta}\right\|_1 \leqslant r\}, \quad \partial\mathbb{B}_2(r) := \{\boldsymbol{\beta} \in \mathbb{R}^{N \times L} : \left\|\boldsymbol{\beta}\right\|_2 = r\}, \\
&\mathbb{B}_p^d(u) := \{D \in \mathbb{R}^d : \left\|D\right\|_p \leqslant u\}. \\
&\omega_i := \omega_{s_i}(\Theta_i^*) = \min_{\boldsymbol{\beta} \in \mathbb{R}^{N \times L}} \{\left\|\boldsymbol{\beta} - \Theta_i^*\right\|_1 \mid \left\|\boldsymbol{\beta}\right\|_0 \leqslant s_i\}. \qquad \text{(C.9)} \\
&\mathbb{C}_i^* := \mathbb{C}(S_i^*, \Theta_i^*) = \big\{\boldsymbol{\beta} \in \mathbb{R}^{N \times L} : \left\|\boldsymbol{\beta}_{S_i^{*c}}\right\|_1 \leqslant 3 \left\|\boldsymbol{\beta}_{S_i^*}\right\|_1 + 4\omega_i\big\}.
\end{aligned}$$

where $S_i^*$ is the support of the best $\ell_1$ approximator of $\Theta_i^*$ that has cardinality $s_i$, i.e., the support of the optimal solution to (C.9). One can then show that $\|\Theta^*_{S_i^{*c}}\|_1 = \omega_i$.

We want to show the following inequality,

$$\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) \geqslant \frac{1}{4} C_{\mathscr{X}}^2 \|\boldsymbol{\beta}\|_F^2 - \tau_i^2, \qquad \forall\, \boldsymbol{\beta} \in \mathbb{C}_i^*.$$

We show this inequality by breaking $\mathbb{C}^*$ into the sets

$$\{\mathbb{C}_i^* \cap \partial\mathbb{B}_F(r_1)\} \;\cup\; \{\mathbb{C}_i^* \cap (\mathbb{B}_F(r_1))^c\} \;\cup\; \{\mathbb{C}_i^* \cap \mathbb{B}_F(\omega_i^2/\sqrt{s_i})\}.$$

For the first two sets of these, the inequality can be shown without any tolerance ($\tau_i^2 = 0$). We need to allow for some tolerance $\tau_i^2 = \omega_i^2/s_i$ when $\omega_i > 0$.

### Fixed $\ell_2$ norm

Consider the set $\mathbb{C}_i^* \cap \partial\mathbb{B}_2(r_1)$, where $r_1^2 = (\omega_i^2)/s_i + \mathbf{1}_{\{\omega_i = 0\}}$.

Note that for any $\boldsymbol{\beta} \in \mathbb{C}_i^*$, we have $\boldsymbol{\beta} = \boldsymbol{\beta}_{S_i^*} + \boldsymbol{\beta}_{S_i^{*c}}$, and hence

$$\|\boldsymbol{\beta}\|_1 = \|\boldsymbol{\beta}_{S_i^*}\|_1 + \|\boldsymbol{\beta}_{S_i^{*c}}\|_1 \leqslant 4\left\|\boldsymbol{\beta}_{S_i^*}\right\|_1 + 4\left\|\Theta_{S_i^{*c}}\right\|_1 \leqslant 4\big(\sqrt{s}\,\|\boldsymbol{\beta}\|_F + \omega_i\big) \qquad \forall\, \boldsymbol{\beta} \in \mathbb{C}_i^*$$

using $\left\|\boldsymbol{\beta}_{S_i^*}\right\|_1 \leqslant \sqrt{s_i}\left\|\boldsymbol{\beta}_{S_i^*}\right\|_F$ and $\left\|\Theta_{S_i^{*c}}\right\|_1 \leqslant \omega_i$. It follows that for any $r_1 > 0$,

$$\mathbb{C}_i^* \cap \partial\mathbb{B}_F(r_1) \;\subseteq\; \mathbb{B}_1\big(r_2\big), \qquad \text{where} \quad r_2 := 4\big(r_1\sqrt{s_i} + \omega_i\big)$$

Next we consider covering $\mathbb{C}_i^* \cap \partial\mathbb{B}_F(r_1)$ by finding a minimum $\varepsilon$-cover of $\mathbb{B}_1(r_2)$. For a metric space $(T, \rho)$, let $\mathcal{N}$ be a minimum $\varepsilon$-cover of $T$ in $\rho$, i.e., the smallest set $\mathcal{N}$ which satisfies

$$\forall\, \boldsymbol{\beta} \in T, \quad \exists\, \boldsymbol{\beta}' \in \mathcal{N}, \qquad \text{such that} \qquad \rho(\boldsymbol{\beta}, \boldsymbol{\beta}') \leqslant \varepsilon.$$

The quantity $\mathscr{N}(\varepsilon, T, \rho) := \log|\mathcal{N}|$ for a minimum $\varepsilon$-cover $\mathcal{N}$ is called the metric entropy. The following is an adaptation of a result of [Raskutti et al., 2011, Lemma 3, case $q = 1, p = 2$]:

**Lemma 22.** *Let $\mathsf{X} \in \mathbb{R}^{n\times d}$ be a matrix with column normalization $\|\mathsf{X}_{*j}\|_2 \leqslant \sqrt{n}$ for all $j$. Consider the following (pseudo) metric in the space $\mathbb{R}^d$, $\rho(D_1, D_2) := \frac{1}{\sqrt{n}}\|\mathsf{X}(D_1 - D_2)\|_2$ on $\mathbb{R}^d$. Then, for a sufficiently small constant $C_1 > 0$, the metric entropy of $\mathbb{B}_1(u)$ in $\rho$ is*

*bounded as*

$$\mathscr{N}\big(\varepsilon, \mathbb{B}_1^d(u), \rho\big) \leqslant \widetilde{C}_2 \frac{u^2}{\varepsilon^2} \log(d), \quad \forall \varepsilon \leqslant \widetilde{C}_1 u.$$

Now, consider a design matrix $\mathsf{X} \in \mathbb{R}^{n \times NL}$ defined as,

$$\mathsf{X}_{t*} := \left[(\mathbf{X}^{t-1}\boldsymbol{d}_1)^\top \ (\mathbf{X}^{t-1}\boldsymbol{d}_2)^\top \dots (\mathbf{X}^{t-1}\boldsymbol{d}_L)^\top\right] \in \mathbb{R}^{1 \times NL}, \quad t = 1, 2, \dots n \tag{C.10}$$

Note that $\mathsf{X}$ satisfies the column normalization property $\|\mathsf{X}_{*j}\|_2 \leqslant C_{\mathbb{X}}\sqrt{n}$ for all $\ell$ since $\mathsf{X}_{tj} \in [-C_{\mathbb{X}}, C_{\mathbb{X}}]$ for all $t \in [n]$ and $j \in [NL]$. Fix $\varepsilon \in (0, 2\widetilde{C}_1 r_2/r_1)$ for sufficiently small $\widetilde{C}_1 > 0$. It follows that there exists an $(r_1\varepsilon/2)$-cover, denoted by $\mathcal{N}_i''$, of $\mathbb{B}_1^{NL}(r_2)$ in the metric defined in Lemma 22 with cardinality bounded as

$$\log|\mathcal{N}_i''| \ \leqslant \widetilde{C}_2 \ \frac{r_2^2}{r_1^2\varepsilon^2} \log(NL).$$

Define a *stacking operator* $\mathtt{S} : \mathbb{R}^{N \times L} \to \mathbb{R}^{NL}$ that flattens a matrix into a vector columnwise:

$$\mathtt{S}(\boldsymbol{\beta}) := \begin{bmatrix} \boldsymbol{\beta}_{*1} \\ \vdots \\ \boldsymbol{\beta}_{*L} \end{bmatrix} \in \mathbb{R}^{NL}. \tag{C.11}$$

Also denote for a set $A$ denote by $\mathtt{S}(A) = \{\mathtt{S}(a) \mid a \in A\}$. Then we have

$$\mathtt{S}(\mathbb{C}_i^* \cap \partial\mathbb{B}_F(r_1)) \ \subseteq \ \mathtt{S}(\mathbb{B}_1\big(r_2\big)) = \mathbb{B}_1^{NL}(r_2).$$

Define a (pseudo) metric on the matrix space $\mathbb{R}^{N \times L}$ as $\overline{\rho}(\boldsymbol{\beta}, \boldsymbol{\beta}') := \rho(\mathtt{S}(\boldsymbol{\beta}), \mathtt{S}(\boldsymbol{\beta}'))$. Since $\mathtt{S}$ is a bijection, it follows that there is an exterior $(r_1\varepsilon/2)$-covering of $\mathbb{C}_i^* \cap \partial\mathbb{B}_F(r_1)$ in metric $\overline{\rho}$ with the same cardinality as $\mathcal{N}_i''$; call it $\mathcal{N}_i'$. (Here, the exterior covering means that the elements need not belong the set they cover. Elements of $\mathcal{N}_i'$ are matrices in $\mathbb{B}_1(r_2)$ but not necessarily in $\mathbb{C}_i^* \cap \partial\mathbb{B}_F(r_1)$.)

We can pass from $\mathcal{N}_i'$ to an $(r_1\varepsilon)$-cover of $\mathbb{C}_i^* \cap \partial\mathbb{B}_F(r_1)$, denoted by $\mathcal{N}_i$ such that $|\mathcal{N}_i| \leqslant |\mathcal{N}_i'|$ (see Exercise 4.2.9 in [Vershynin, 2018, p.75]). In particular, we have $\mathcal{N}_i \subseteq \mathbb{C}_i^* \cap \mathbb{B}_F(r_1)$.

Using the following equality which is proved in Appendix C.2.1,

$$\mathcal{E}(\Delta; \mathbb{X}) = \frac{1}{n} \left\|\mathsf{X}\,\mathtt{S}(\boldsymbol{\beta})\right\|_2^2, \tag{C.12}$$

by the triangle inequality $\big||a|-|b|\big| \leqslant |a-b|$, we get,

$$|\sqrt{\mathcal{E}(\boldsymbol{\beta};\mathbb{X})}-\sqrt{\mathcal{E}(\boldsymbol{\beta}';\mathbb{X})}| \leqslant \overline{\rho}(\boldsymbol{\beta},\boldsymbol{\beta}'), \qquad \boldsymbol{\beta},\boldsymbol{\beta}' \in \mathbb{R}^{N\times L}$$

for any two matrices $\boldsymbol{\beta}$ and $\boldsymbol{\beta}'$. Using $(a-b)^2 \geqslant \frac{1}{2}a^2 - b^2$, with $b = \mathcal{E}(\boldsymbol{\beta};\mathbb{X})$, and $a = \mathcal{E}(\boldsymbol{\beta}';\mathbb{X})$ we have

$$\mathcal{E}(\boldsymbol{\beta};\mathbb{X}) \geqslant \frac{1}{2}\mathcal{E}(\boldsymbol{\beta}';\mathbb{X}) - \overline{\rho}^2(\boldsymbol{\beta},\boldsymbol{\beta}').$$

If follows that

$$\inf_{\boldsymbol{\beta}\in\mathbb{C}_i^*\cap\partial\mathbb{B}_F(r_1)} \mathcal{E}(\boldsymbol{\beta};\mathbb{X}) \;\geqslant\; \tfrac{1}{2}\inf_{\boldsymbol{\beta}\in\mathcal{N}_i} \mathcal{E}(\boldsymbol{\beta};\mathbb{X}) - (r_1\varepsilon)^2$$

By Lemma 11 and the union bound, with probability at least $1-|\mathcal{N}_i|\exp(\frac{-nt^2}{G})$ , we have

$$\mathcal{E}(\boldsymbol{\beta};\mathbb{X}) - \mathbb{E}\mathcal{E}(\boldsymbol{\beta};\mathbb{X}) \geqslant -t\,\|\boldsymbol{\beta}\|_{1,1}^2\,, \quad \forall\,\boldsymbol{\beta}\in\mathcal{N}_i.$$

Since $\mathcal{N}_i \subseteq \mathbb{C}_i^*\cap\mathbb{B}_F(r_1)$, for any $\boldsymbol{\beta}\in\mathcal{N}_i$ we have $\|\boldsymbol{\beta}\|_{1,1}^2 \leqslant s_i\|\boldsymbol{\beta}\|_F^2$ and $\|\boldsymbol{\beta}\|_F = r_1$. It follows that with the same probability $1-|\mathcal{N}_i|\exp(\frac{-nt^2}{G})$,

$$\mathcal{E}(\boldsymbol{\beta};\mathbb{X}) \;\geqslant\; \mathbb{E}\mathcal{E}(\boldsymbol{\beta};\mathbb{X}) - t\,sr_1^2 \;\geqslant\; (C_{\mathscr{X}}^2 - ts)\,r_1^2, \quad \forall\boldsymbol{\beta}\in\mathcal{N}_i$$

where we have used Lemma 10 in the second inequality. It follows that with the same probability

$$\inf_{\boldsymbol{\beta}\in\mathbb{C}^*\cap\partial\mathbb{B}_F(r_1)} \mathcal{E}(\boldsymbol{\beta};\mathbb{X}) \;\geqslant\; \Big(\tfrac{1}{2}C_{\mathscr{X}}^2 - \tfrac{1}{2}ts - \varepsilon^2\Big)\,r_1^2. \tag{C.13}$$

Taking $r_1 = (\omega_i + \mathbf{1}_{\{\omega_i=0\}})/\sqrt{s_i}$, we can balance the two terms in $r_2$. We obtain

$$4\sqrt{s_i} \;\leqslant r_2/r_1 \;\leqslant 8\sqrt{s_i}.$$

The constraint on $\varepsilon$ is $\varepsilon \leqslant 2\widetilde{C}_1(r_2/r_1)$. It is enough to require $\varepsilon \leqslant 8\widetilde{C}_1\sqrt{s_i}$. Taking $\varepsilon^2 = \frac{1}{8}C_{\mathscr{X}}^2$ and assuming that $s_i \geqslant \frac{C_{\mathscr{X}}^2}{512\widetilde{C}_1^2} =: C_{\mathscr{X}}^2/C_1$ satisfies the constraint. Also, taking $t = \frac{1}{4}C_{\mathscr{X}}^2/s_i$, we obtain

$$\mathbb{P}\Big(\inf_{\boldsymbol{\beta}\in\mathbb{C}_i^*\cap\partial\mathbb{B}_F(r_1)} \mathcal{E}(\boldsymbol{\beta};\mathbb{X}) \;\geqslant\; \big(\frac{1}{4}C_{\mathscr{X}}^2\big)\,r_1^2\Big) \geqslant 1-\exp\Big(\log|\mathcal{N}_i| - C_{\mathscr{X}}^4\frac{n}{16s_i^2G}\Big) =: P_i \tag{C.14}$$

Noting that

$$\log|\mathcal{N}_i| \leqslant \widetilde{C}_2\big(8\sqrt{s_i}\big)^2\Big(\frac{8}{C_{\mathscr{X}}^2}\Big)\log(NL),$$

the probability is further bounded as

$$1 - P_1 \;\leqslant\; \exp\Big(\frac{C_2}{C_{\mathscr{X}}^2} s_i \log(NL) - C_{\mathscr{X}}^4 \frac{n}{16 s_i^2 G}\Big),$$

where $C_2 := 512\widetilde{C}_2$. Thus, we have established RSC with high probability for matrices in $\mathbb{C}_i^* \cap \partial\mathbb{B}_F(r_1)$ with curvature $\kappa = \frac{1}{4}C_{\mathscr{X}}^2$ and tolerance $\tau^2 = 0$, as shown in equation (C.14).

Note that when $\omega_i = 0$ (i.e., the case of hard sparsity), $\mathbb{C}_i^*$ is a cone hence the above extends immediately to all $\boldsymbol{\beta} \in \mathbb{C}_i^*$, since $\mathcal{E}(c\boldsymbol{\beta}; \mathbb{X}) = c^2\mathcal{E}(\boldsymbol{\beta}; \mathbb{X})$ for all $c > 0$, thus completing the proof. Let us assume $\omega_i > 0$ in the rest of the proof.

**Extending to the complement of the $\ell_2$ norm ball**

For $\omega_i > 0$, since $\mathbb{C}_i^*$ is not a cone, we cannot use a scale-invariance argument to extend to general matrices. However, we have the following:

**Lemma 23.** *Assume that RSC holds for $\mathcal{E}$ in the sense of $\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) \geqslant \kappa\|\boldsymbol{\beta}\|_F^2$, for all $\boldsymbol{\beta} \in \mathbb{C}_i^* \cap \partial\mathbb{B}_F(r)$. Then, RSC holds in the same sense for all $\boldsymbol{\beta} \in \mathbb{C}_i^* \cap \{\boldsymbol{\beta} : \|\boldsymbol{\beta}\|_F \geqslant r\}$.*

We skip the proof since it can be verified without much difficulty. The lemma establishes the RSC of the previous step for all of $\mathbb{C}_i^* \cap \{\boldsymbol{\beta} : \|\boldsymbol{\beta}\|_F \geqslant r_1\}$. The proof is straightforward and follows from the observation that $\mathcal{E}(\cdot\,; \mathbb{X})$ satisfies $\mathcal{E}(c\boldsymbol{\beta}; \mathbb{X}) = c^2\mathcal{E}(\boldsymbol{\beta}; \mathbb{X})$, for $c \geqslant 1$.

**Extending to small radii**

It remains to extend the result to $\boldsymbol{\beta} \in \mathbb{C}^* \cap \{\boldsymbol{\beta} : \|\boldsymbol{\beta}\|_F < r_1\}$. In this case, we simply take $\tau^2 := r_1^2 = \omega_i^2/s_i$ (since $\omega_i > 0$ by assumption) so that

$$\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) \geqslant 0 \geqslant \|\boldsymbol{\beta}\|_F^2 - \tau^2$$

so that the RSC holds with curvature = 1 and tolerance $\tau^2$. Putting the pieces together, we have the RSC for all $\boldsymbol{\beta} \in \mathbb{C}_i$ with the probability given in Step 1, curvature $\kappa_i = \min\{\frac{1}{4}C_{\mathscr{X}}^2, 1\}$ and tolerance $\tau_i^2 = \omega_i^2/s_i$. This concludes the proof. □

### C.2.1 Proof of equality (C.12)

The right hand side is

$$\begin{aligned}\tfrac{1}{n}\sum_{t=1}^{n}(\mathsf{X}_{t*}\mathbb{S}(\boldsymbol{\beta}))^2 = \tfrac{1}{n}\sum_{t=1}^{n}\sum_{\ell=1}^{L}(\mathbf{X}^{t-1}\boldsymbol{d}_\ell)^\top\boldsymbol{\beta}_{*\ell} = \tfrac{1}{n}\sum_{t=1}^{n}\sum_{\ell=1}^{L}\boldsymbol{\beta}_{*\ell}^\top(\mathbf{X}^{t-1}\boldsymbol{d}_\ell) = \tfrac{1}{n}\mathrm{Tr}(\boldsymbol{\beta}^\top\mathbf{X}^{t-1}\mathbf{D}) \\ = \tfrac{1}{n}\sum_{t=1}^{n}\langle\boldsymbol{\beta},\mathbf{X}^{t-1}\mathbf{D}\rangle\end{aligned}$$

This proves the claim. □

## C.3 Intermediate lemmas mentioned in Section 5.7: Contraction in $p$-Markov chains.

In this section, we prove the following two main lemmas used in Section 5.7.

**Lemma 24.** *The map $\mathbb{X}\mapsto\mathcal{E}(\boldsymbol{\beta};\mathbb{X})$ is Lipschitz with respect to the Hamming distance on $\mathcal{X}^{\times(n+p-1)}$, with Lipschitz constant at most $(4B^2C_{\mathbf{D}}^2/n)\,\|\boldsymbol{\beta}\|_{1,1}^2$.*

A process over a countable space $\mathcal{X}$ is referred to as a *$p$-Markov chain* if for some finite $p$,

$$\mathbb{P}(\boldsymbol{x}^t = z|\{\boldsymbol{x}^{t-k}\}_{k\in\mathbb{N}_+}) = \mathbb{P}(\boldsymbol{x}^t = z|\{\boldsymbol{x}^{t-k}\}_{k=1}^{p}), \tag{C.15}$$

for all $z\in\mathcal{X}$, for all $t\in\mathbb{Z}$. To keep the exposition simple, we assume that $\mathbb{P}$ above does not depend on $t$, i.e., the process is homogeneous.

**Lemma 25.** *For a $p$-Markov process over $\mathcal{X}$, with equivalent kernel $\mathcal{K}\in\mathbb{R}^{|\mathcal{X}|^p\times|\mathcal{X}|^p}$ given by* (C.24) *with $r=p$, the mixing coefficients defined in* (5.34) *are bounded as*

$$\eta_{k\ell} \;\leqslant\; \tau_1(\Theta^*)^{1+\lfloor(\ell-k-1)/p)\rfloor},\quad \ell\;\geqslant k. \tag{C.16}$$

*In particular, for any $\tau\in[\tau_1(\Theta^*),1)$*

$$\|\!|\mathbf{m}H|\!\|_\infty^2 := \Big(\max_{k\in[n]}\sum_{\ell\geqslant k}\eta_{k\ell}\Big)^2 \;\leqslant\; 2+\frac{2p^2}{(\tau^{-1}-1)^2}. \tag{C.17}$$

### C.3.1 Proof of Lemma 24

It is enough to consider two sequences $\{\boldsymbol{x}^t\}$ and $\{\boldsymbol{y}^t\}$ that differ in a single time step, say at time point $r$, so that the state vectors can be written as $\mathbb{X} = (\boldsymbol{x}^{-p+1}, \boldsymbol{x}^{-p+2}, \ldots, \boldsymbol{x}^r, \ldots, \boldsymbol{x}^{n-1})$ and $\mathbb{Y} = (\boldsymbol{x}^{-p+1}, \boldsymbol{x}^{-p+2}, \ldots, \boldsymbol{y}^r, \ldots, \boldsymbol{x}^{n-1})$, where $r$ will be fixed. The general case follows, via triangle inequality, since any $\widetilde{\mathbb{Y}}$ can be reached from $\mathbb{X}$ by a sequence $\mathbb{X} =: \mathbb{X}_{(0)}, \mathbb{X}_{(1)}, \ldots, \mathbb{X}_{(h)} := \widetilde{\mathbb{Y}}$ such that $\mathbb{X}_{(i)}$ and $\mathbb{X}_{(i-1)}$ are Hamming distance 1 apart, for $i = 1, 2, \ldots h$, where $h$ is the hamming distance of $\mathbb{X}$ and $\widetilde{\mathbb{Y}}$ in $\mathcal{X}^{n+p-1}$.

Let $\mathbf{X}^{t-1}$ and $\mathbf{Y}^{t-1}$ be defined based on $\mathbb{X}$ and $\mathbb{Y}$ as before, i.e., the corresponding $p$-lag history at time $t-1$. Note that $\mathbf{X}^{t-1}$ and $\mathbf{Y}^{t-1}$ are different only for $t$ such that $t \in \{r+1, \ldots, r+p\}$, and for such $t$, we have via Hölder's inequality:

$$|i{+}\boldsymbol{\beta}, \mathbf{X}^{t-1}\mathbf{D} - \mathbf{Y}^{t-1}\mathbf{D}| \leqslant 2B \left\|(\boldsymbol{\beta}\mathbf{D}^\top)_{*,t-r}\right\|_1 \quad \text{and} \quad |i{+}\boldsymbol{\beta}, \mathbf{X}^{t-1}\mathbf{D} + \mathbf{Y}^{t-1}\mathbf{D}| \leqslant 2B \left\|\boldsymbol{\beta}\mathbf{D}^\top\right\|_{1,1}.$$

where $\boldsymbol{M}_{*,i}$ is the $i^{\text{th}}$ column of a matrix $\boldsymbol{M}$. Note the inner products above are over matrices in $\mathbb{R}^{N\times L}$. In the above inequality we have also used the fact that for any $\boldsymbol{M} \in \mathbb{R}^{N\times p}$, we have $\langle \boldsymbol{\beta}, \boldsymbol{M}\mathbf{D}\rangle = \langle \boldsymbol{\beta}\mathbf{D}^\top, \boldsymbol{M}\rangle$ where the second inner product is over $\mathbb{R}^{N\times p}$. Combining the above inequalities we obtain

$$|\mathcal{E}(\boldsymbol{\beta}; \mathbb{X}) - \mathcal{E}(\boldsymbol{\beta}; \mathbb{Y})| = \frac{1}{n}\Big| \sum_{t=r+1}^{r+p} \left[i{+}\boldsymbol{\beta}, \mathbf{X}^{t-1}\mathbf{D}^2 - i{+}\boldsymbol{\beta}, \mathbf{Y}^{t-1}\mathbf{D}^2\right]\Big| \leqslant \sum_{t=r+1}^{r+p} |i{+}\boldsymbol{\beta}, (\mathbf{X}^{t-1} - \mathbf{Y}^{t-1})\mathbf{D}||i{+}\boldsymbol{\beta}, (\mathbf{X}^{t-1}$$

$$\leqslant \frac{4B^2}{n} \sum_{t=r+1}^{r+p} \left\|(\boldsymbol{\beta}\mathbf{D}^\top)_{*,t-r}\right\|_1 \left\|\boldsymbol{\beta}\mathbf{D}^\top\right\|_{1,1} = \frac{4B^2}{n} \left\|\boldsymbol{\beta}\mathbf{D}^\top\right\|_{1,1}^2$$

Finally, $\left\|\boldsymbol{\beta}\mathbf{D}^\top\right\|_{1,1} = \left\|\mathbf{D}\boldsymbol{\beta}^\top\right\|_{1,1} = \sum_{\ell=1}^{L} \left\|\mathbf{D}(\boldsymbol{\beta}^\top)_{*,\ell}\right\|_1 \leqslant C_{\mathbf{D}} \left\|\boldsymbol{\beta}_{\ell,*}\right\|_1 = C_{\mathbf{D}} \left\|\boldsymbol{\beta}\right\|_{1,1}$, where we have used the fact that $C_{\mathbf{D}}$ is the $1 \to 1$ operator norm of the matrix $\mathbf{D}$, i.e., $C_{\mathbf{D}} = \max_{\boldsymbol{u}\neq\mathbf{0}} \frac{\|\mathbf{D}\boldsymbol{u}\|_1}{\|\boldsymbol{u}\|_1}$. This proves the claim. □

### C.3.2 Bounding $\|\mathbf{H}\|_\infty$ using $p$-Markov contraction

We start by recalling a well-known contraction quantity, the *Dobrushin ergodicity coefficient*, and relating it to the mixing coefficients of $p$-Markov processes.

**Dobrushin ergodicity coefficient**

For a Markov chain (or 1-Markov process) over a discrete space $\mathcal{X}$, let $P = (P_{ij}) \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{X}|}$ be its transition kernel. The kernel is a nonnegative stochastic matrix, i.e., each row is a probability distribution. Thus, $P \geqslant 0$ and $P\mathbf{1} = \mathbf{1}$ where $\mathbf{1} \in \mathbb{R}^{|\mathcal{X}|}$ is the all-ones vector. Let

$$\mathcal{H}_1 := \big\{u \in \mathbb{R}^{|\mathcal{X}|} \mid \mathbf{1}^\top u = 0\big\}. \tag{C.18}$$

This subspace is invariant to every Markov kernels $P \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{X}|}$, i.e., for all $u \in \mathcal{H}_1$, we have $u^\top P \in \mathcal{H}_1$. The *Dobrushin ergodicity coefficient* of $P$ is defined as

$$\tau_1(P) := \sup_{u\in\mathcal{H}_1} \frac{\|u^\top P\|_1}{\|u\|_1}. \tag{C.19}$$

It follows from the invariance of $\mathcal{H}_1$ to $P$ that

$$\|u^\top P^\ell\|_1 \leqslant \tau_1(P)^\ell \|u\|_1 \quad \forall u \in \mathcal{H}_1. \tag{C.20}$$

The following alternative characterization is well-known [Rhodius, 1997] (cf. Appendix C.3 for a proof):

**Lemma 26.** *The Dobrushin ergodicity coefficient of $P$ satisfies*

$$\tau_1(P) = \tfrac{1}{2} \sup_{x,y\in\mathcal{X}} \|(e_x - e_y)^T P\|_1 \tag{C.21}$$

*where $e_x$ is the $x$-th basis vector of $\mathbb{R}^{\mathcal{X}}$.*

*Proof.* Optimization problem in (C.19) is scale invariant, hence,

$$\tau_1(P) = \sup_{u\,\in\,\mathcal{H}_1(1)} \|u^\top P\|_1, \tag{C.22}$$

where $\mathcal{H}_1(1) = \{u \in \mathcal{H}_1 \mid \|u\|_1 \leqslant 1\}$. We will show that the set $\mathcal{H}_1(1) = C := \mathrm{conv}(\{\frac{1}{2}(e_x - e_y)\})$. Using this, (C.22) is a maximization of a convex function $\|u^\top P\|_1$ over a polytope with extreme points $\frac{1}{2}(e_x - e_y), x, y \in \mathcal{X}$. It follows that the maximum occurs, at least, at an extreme point, which gives the desired result. The inequality in the statement of the lemma follows since the total-variation is bounded by 1.

The rest of the proof establishes $\mathcal{H}_1(1) = C$. The inclusion $C \subseteq \mathcal{H}_1(1)$ can be verified easily by checking the membership of extreme points of $C$ in $\mathcal{H}_1(1)$, since $\mathcal{H}_1(1)$ is a convex

set. We now prove the nontrivial direction $\mathcal{H}_1(1) \subseteq C$.

Let the ambient space be $\mathbb{R}^m$, $\Delta_m$ the probability simplex in $\mathbb{R}^m$, and $\partial\,\mathbb{B}_1 := \{u \in \mathbb{R}^m : \|u\|_1 = 1\}$ the boundary of $\ell_1$ ball. We have $C = \frac{1}{2}\Delta_m + \frac{1}{2}(-\Delta_m)$, which is a Minkowski sum. This follows since taking the Minkowski sum and taking the convex hull commute [Krein and Smulian, 1940, Theorem 3]. Hence, it suffices to show that for any vector $u \in \mathcal{H}_1(1)$, there exists a pair of probability vectors $\pi_1, \pi_2 \in \Delta_m$ such that $u = \frac{1}{2}(\pi_1 - \pi_2)$. Since $0 \in C$, and $\mathcal{H}_1(1) = \mathrm{conv}(0, \partial\,\mathbb{B}_1 \cap \mathcal{H}_1)$, it is enough to consider $u \in \partial\,\mathbb{B}_1 \cap \mathcal{H}_1$.

Let $u \in \partial\,\mathbb{B}_1 \cap \mathcal{H}_1$, and let $u_+$ and $u_-$ be the positive and negative parts of $u$, that is, $(u_+)_i = \max(u_i, 0)$ and $(u_-)_i = -\min(u_i, 0)$. Taking $\pi_1 = 2u_+$ and $\pi_2 = 2u_-$, we have $u = \frac{1}{2}(\pi_1 - \pi_2)$. Also, due to $u \in \partial\,\mathbb{B}_1$, $1 = \|u\|_1 = \frac{1}{2}\|\pi_1\|_1 + \frac{1}{2}\|\pi_2\|_1$ whereas due to $u \in \mathcal{H}_1$, $0 = \mathbf{1}^\top u = \frac{1}{2}\|\pi\|_1 - \|\pi_2\|_1$. It follows that $\|\pi_1\|_1 = \|\pi_2\|_1 = 1$, that is, $\pi_1, \pi_2 \in \Delta_m$. This concludes the proof. ∎

Recall that $\|\pi_1 - \pi_2\|_{\mathrm{TV}}$ denotes the *total variation* distance between probability distributions $\pi_1$ and $\pi_2$. For discrete distributions we have, $\|\pi_1 - \pi_2\|_{\mathrm{TV}} = \frac{1}{2}\|\pi_1 - \pi_2\|_1 \leqslant 1$, with equality if and only if $\pi_1$ and $\pi_2$ are orthogonal, i.e., have completely mismatched supports. Consequently, for any stochastic matrix $P$, we have $\tau_1(P) \leqslant 1$. Furthermore, the inequality is strict if and only if no two rows of $P$ are orthogonal. Markov kernels with $\tau_1(\cdot) < 1$ are said to be *scrambling*. A sufficient condition for $\tau_1(P) < 1$ is $P$ having at least one column with all entries positive.

### The $p$-step chain

A $p$-Markov process can be equivalently represented by a Markov kernel $\mathcal{K} \in [0,1]^{|\mathcal{X}|^p \times |\mathcal{X}|^p}$ that gives transition probabilities for consecutive blocks of size $p$. For any $t \in \mathbb{Z}$,

$$\mathcal{K}_{\boldsymbol{ij}} = \mathbb{P}\Big((\boldsymbol{x}^{t+1-k})_{k=1}^{p} = \boldsymbol{j} \,\Big|\, (\boldsymbol{x}^{t-k})_{k=1}^{p} = \boldsymbol{i}\Big), \tag{C.23}$$

for all $\boldsymbol{i}, \boldsymbol{j} \in \mathcal{X}^{\times p}$. Kernel matrix $\mathcal{K}$ is constrained since $\mathcal{K}_{\boldsymbol{ij}}$ can be nonzero only if $(j_2, j_3, \ldots, j_p) = (i_1, i_2, \ldots, i_{p-1})$. The $r$-step chain associated with $\mathcal{K}$ has kernel $\mathcal{K}^r$. In general, for all

$\boldsymbol{i}, \boldsymbol{j} \in \mathcal{X}^{\times p}$ and for $r \geqslant 1$

$$(\mathcal{K}^r)_{\boldsymbol{ij}} = \mathbb{P}\Big((\boldsymbol{x}^{t+r-k})_{k=1}^{p} = \boldsymbol{j} \;\Big|\; (\boldsymbol{x}^{t-k})_{k=1}^{p} = \boldsymbol{i}\Big). \tag{C.24}$$

Similarly, $(\mathcal{K}^r)_{\boldsymbol{ij}}$ can be nonzero only if $(j_{r+1}, j_{r+2}, \dots j_p) = (i_1, i_2, \dots, i_{p-r})$, for $r < p$. However, no such constraint applies for $r \geqslant p$. Moreover, one can verify that for $r < p$, a pair of rows $(\mathcal{K}^r)_{\boldsymbol{i}*}$ and $(\mathcal{K}^r)_{\boldsymbol{i'}*}$ are always orthogonal for $\boldsymbol{i}, \boldsymbol{i'} \in \mathcal{X}^{\times p}$ such that $i_1 \neq i'_1$. Consequently, $\tau_1(\mathcal{K}^r) = 1$ for all $r < p$.

Fortunately for $r = p$, one can show that $\tau_1(\mathcal{K}^p) < 1$, under the mild assumption that

$$\mathbb{P}\big(\boldsymbol{x}^t = z | (\boldsymbol{x}^{t-k})_{k=1}^{p} = \boldsymbol{j}\big) > 0 \;\text{ for all } z \in \mathcal{X} \text{ and } \boldsymbol{j} \in \mathcal{X}^{\times p},$$

since this implies that $\mathcal{K}^p$ is a positive matrix and hence *scrambling*. Note that the above condition always holds for the process defined in (5.3).

### C.3.3 Proof of Lemma 25

Recall the notation $\tau_1(\Theta^*)$ defined in equation (5.11), whereby $\tau_1(\mathcal{K}^p) = \tau_1(\Theta^*)$ by definition. The following lemma provides an upper bound for $\|\mathbf{m}H\|_\infty$ as a function of $\tau_1(\Theta^*)$.

**Proof of** (C.16)

Recall that $\mathbb{X}_{-p+1}^{n} := \{\boldsymbol{x}^n, \boldsymbol{x}^{n-1}, \dots, \boldsymbol{x}^{-p+1}\}$ together make $n$ steps of the $p$-Markov process. Fix $k \geqslant 1$ and take $w \in \mathcal{X}$, $\boldsymbol{y} \in \mathcal{X}^{p-1}$, and $\mathbf{z} \in \mathcal{X}^{k-1}$. We use the shorthand $\mathbb{X}_{-p+1}^{k} = w\boldsymbol{y}\mathbf{z}$, to denote $\boldsymbol{x}^k = w, \mathbb{X}_{k-p+1}^{k-1} = \boldsymbol{y}$ and $\mathbb{X}_{-p+1}^{k-p} = \mathbf{z}$ and define the law

$$\mathcal{L}_k^{(\ell \to n)}(w\boldsymbol{y}\mathbf{z}) := \mathbb{P}\big(\mathbb{X}_\ell^n = \cdot \mid \mathbb{X}_{-p+1}^{k} = w\boldsymbol{y}\mathbf{z}\big) = \mathbb{P}\big(\mathbb{X}_\ell^n = \cdot \mid \mathbb{X}_{k-p+1}^{k} = w\boldsymbol{y}\big) =: \mathcal{L}_k^{(\ell \to n)}(w\boldsymbol{y})$$

using the $p$-Markov property, showing that $\mathcal{L}_k^{(\ell \to n)}(w\boldsymbol{y}\mathbf{z})$ does not depend on $\mathbf{z}$. Thus, we also write $\mathcal{L}_k^{(\ell \to n)}(w\boldsymbol{y})$ for $\mathcal{L}_k^{(\ell \to n)}(w\boldsymbol{y}\mathbf{z})$.

**Case 1.** Assuming $\ell + p \leqslant n$, we have

$$\begin{aligned}
&\mathbb{P}\big(X_\ell^n = x_\ell^n \mid X_{k-p+1}^k = w\boldsymbol{y}\big) \\
&\quad = \mathbb{P}\big(\mathbb{X}_{\ell+p}^n = x_{\ell+p}^n \mid \mathbb{X}_\ell^{\ell+p-1} = x_\ell^{\ell+p-1}\big) \cdot \mathbb{P}\big(\mathbb{X}_\ell^{\ell+p-1} = x_\ell^{\ell+p-1} \mid \mathbb{X}_{k-p+1}^k = w\boldsymbol{y}\big) \\
&\quad = \phi\big(x_{\ell+p}^n \mid x_\ell^{\ell+p-1}\big) \cdot \psi_{w\boldsymbol{y}}(x_\ell^{\ell+p-1})
\end{aligned}$$

where we have defined $\phi(u \mid v) := \mathbb{P}\big(\mathbb{X}_{\ell+p}^n = u \mid \mathbb{X}_\ell^{\ell+p-1} = v\big)$ and

$$\psi_{w\boldsymbol{y}}(\beta) := \mathbb{P}\big(\mathbb{X}_\ell^{\ell+p-1} = \beta \mid \mathbb{X}_{k-p+1}^k = w\boldsymbol{y}\big)$$

We note that $\psi_{w\boldsymbol{y}}(\cdot)$ is the $w\boldsymbol{y}$-th row of $\mathcal{K}^{\ell+p-k-1}$ which follows by comparing the definition of $\psi_{w\boldsymbol{y}}$ with (C.24) applied with $t = k+1$ and $r = \ell + p - k - 1$. Letting $e_i$ denote the $i^{\text{th}}$ row of identity in $\mathbb{R}^{|\mathcal{X}|^p \times |\mathcal{X}|^p}$, we have

$$\psi_{w\boldsymbol{y}} = e_{w\boldsymbol{y}}^\top \, \mathcal{K}^{\ell+p-k-1}.$$

Now, we have

$$\begin{aligned}
2\|\mathcal{L}_{w\boldsymbol{y}\boldsymbol{z}}^{(\ell\to n)} - \mathcal{L}_{w'\boldsymbol{y}\boldsymbol{z}}^{(\ell\to n)}\|_{\text{TV}} &= \sum_{x_\ell^n} \big|\mathbb{P}\big(\mathbb{X}_\ell^n = x_\ell^n \mid \mathbb{X}_{k-p+1}^k = w\boldsymbol{y}\big) - \mathbb{P}\big(\mathbb{X}_\ell^n = x_\ell^n \mid \mathbb{X}_{k-p+1}^k = w'\boldsymbol{y}\big)\big| \\
&= \sum_{x_\ell^{\ell+p-1}} \sum_{x_{\ell+p}^n} \phi\big(x_{\ell+p}^n \mid x_\ell^{\ell+p-1}\big)\big|\psi_{w\boldsymbol{y}}(x_\ell^{\ell+p-1}) - \psi_{w'\boldsymbol{y}}(x_\ell^{\ell+p-1})\big| \\
&= \sum_{x_\ell^{\ell+p-1}} \big|\psi_{w\boldsymbol{y}}(x_\ell^{\ell+p-1}) - \psi_{w'\boldsymbol{y}}(x_\ell^{\ell+p-1})\big| \\
&= \|\psi_{w\boldsymbol{y}} - \psi_{w'\boldsymbol{y}}\|_1 = 2\|\mathcal{L}_{w\boldsymbol{y}}^{(\ell\to\ell+p-1)} - \mathcal{L}_{w'\boldsymbol{y}}^{(\ell\to\ell-p+1)}\|_{\text{TV}}. \qquad \text{(C.25)}
\end{aligned}$$

Thus, we have

$$\begin{aligned}
\eta_{k\ell} &= \sup_{w,w',\boldsymbol{y},\boldsymbol{z}} \|\mathcal{L}_k^{(\ell\to n)}(w\boldsymbol{y}\boldsymbol{z}) - \mathcal{L}_k^{(\ell\to n)}(w'\boldsymbol{y}\boldsymbol{z})\|_{\text{TV}} \\
&= \tfrac{1}{2} \sup_{w,w',y} \|\psi_{w\boldsymbol{y}} - \psi_{w'\boldsymbol{y}}\|_1 = \tfrac{1}{2} \sup_{w,w',y} \|(e_{w\boldsymbol{y}} - e_{w'\boldsymbol{y}})^\top \mathcal{K}^{\ell+p-1-k}\|_1.
\end{aligned}$$

Let $m = \ell - k - 1$. Writing $m = p\lfloor m/p \rfloor + (m \bmod p)$ and using $\frac{1}{2}(e_{w\boldsymbol{y}} - e_{w'\boldsymbol{y}}) \in \mathcal{H}_1$ (see

Definition (C.18)), we get

$$\eta_{k\ell} \leqslant \sup_{v\in\mathcal{H}_1} \|v^\top \mathcal{K}^{p+p\lfloor m/p\rfloor+(m \bmod p)}\|_1 \overset{\text{(a)}}{\leqslant} \sup_{v\in\mathcal{H}_1} \tau_1\left(\mathcal{K}^{(m \bmod p)}\right) \|v^\top \mathcal{K}^{p+p\lfloor m/p\rfloor}\|_1$$
$$\overset{\text{(b)}}{\leqslant} \sup_{v\in\mathcal{H}_1} \|v^\top (\mathcal{K}^p)^{1+\lfloor m/p\rfloor}\|_1 \;\leqslant\; \tau_1 (\mathcal{K}^p)^{1+\lfloor m/p\rfloor}, \tag{C.26}$$

where (a) follows from (C.20) applied for $u^\top = v^\top \mathcal{K}^{p+p\lfloor m/p\rfloor}$ which also belongs to $\mathcal{H}_1$, while (b) follows from the inequality in Lemma 26 and the last inequality follows from inequality (C.20) applied for $u = v$. This is the desired result which holds for $\ell + p \leqslant n$.

**Case 2.** When $\ell + p > n$, the reduction in (C.25) is unnecessary, i.e., there are fewer than $p$ variables between $\ell$ and $n$. We cannot write the difference of the two underlying laws in terms of rows of $\mathcal{K}^r$ for some integer $r$. But, we can augment and consider $\mathcal{L}_k^{(\ell\to n+u)}(w\boldsymbol{yz})$ where $u = \ell + p - n$ and then get $\mathcal{L}_k^{(\ell\to n)}(w\boldsymbol{yz})$ by marginalization. We have for any $w, w' \in \mathcal{X}$,

$$\|\mathcal{L}_k^{(\ell\to n)}(w\boldsymbol{yz}) - \mathcal{L}_k^{(\ell\to n)}(w'\boldsymbol{yz})\|_{\text{TV}} \;\leqslant\; \|\mathcal{L}_k^{(\ell\to n+u)}(w\boldsymbol{yz}) - \mathcal{L}_k^{(\ell\to n+u)}(w'\boldsymbol{yz})\|_{\text{TV}}$$

since marginalization does not increase the total variation distance. This follows from the triangle inequality: Assuming $p(\cdot,\cdot)$ and $q(\cdot,\cdot)$ to be some probability mass functions,

$$\sum_x \Big|p(x) - q(x)\Big| = \sum_x \Big|\sum_y p(x,y) - \sum_y q(x,y)\Big| \leqslant \sum_x \sum_y |p(x,y) - q(x,y)|.$$

Since $\ell + p = n + u$, the proof in this case reduces to that of Case 1. The proof of (C.16) is complete. □

**Proof of** (C.17)

It is enough to prove the inequality for $\tau = \tau_1(\mathcal{K}^p)$. Then, the result follows since $\frac{1}{(\frac{1}{x}-1)^2}$ is increasing on $[\tau_1(\mathcal{K}^p), 1)$. For this $\tau$, we have for any fixed $k$ (recalling $\eta_{kk} = 1$),

$$\sum_{\ell\geqslant k} \eta_{k\ell} \;\leqslant\; 1 + \sum_{\ell>k} \tau^{1+\lfloor(\ell-k-1)/p\rfloor} \leqslant 1 + \sum_{m\geqslant 1} \sum_{\ell=(m-1)p+k+1}^{mp+k} \tau^m = 1 + \frac{p\tau}{1-\tau}.$$

It follows that

$$\|\mathbf{m}H\|_\infty^2 := \Big(\max_k \sum_{\ell\geqslant k} \eta_{k\ell}\Big)^2 \leqslant \Big(1 + \frac{p\tau}{1-\tau}\Big)^2 \leqslant 2 + 2\frac{p^2\tau^2}{(1-\tau)^2}$$

which is the desired result. □

## C.4 Proofs of other Technical Lemmas

### C.4.1 Proof of Lemmas 7 and 8

We start by defining some notation. Recall that for $z \in \mathcal{X}^{\times p}$,

$$\mathbb{P}_{\mathbf{z}} := \mathbb{P}(\mathbb{X}_t^{t+p-1} = \cdot \mid \mathbb{X}_{t-p}^{t-1}) = \mathbb{P}(\mathbb{X}_1^p = \cdot \mid \mathbb{X}_{1-p}^0),$$

using the invariance of the conditional distribution to time shifts. We also write $p_{\mathbf{z}}(\cdot)$ for the probability mass function of $\mathbb{P}_{\mathbf{z}}$, i.e.,

$$p_{\mathbf{z}}(\boldsymbol{a}) := \mathbb{P}(\mathbb{X}_t^{t+p-1} = \boldsymbol{a} \mid \mathbb{X}_{t-p}^{t-1} = z) = \mathbb{P}(\mathbb{X}_1^p = \boldsymbol{a} \mid \mathbb{X}_{1-p}^0 = \mathbf{z}), \qquad \forall \boldsymbol{a} \in \mathcal{X}^{\times p}.$$

We also let $q(\xi \mid \boldsymbol{a}) := \mathbb{P}\big(\boldsymbol{x}^t = \xi \mid \mathbb{X}_{t-p}^{t-1} = \boldsymbol{a}\big)$ for $\xi \in \mathcal{X}$, $\boldsymbol{a} \in \mathcal{X}^{\times p}$, and define

$$d_K(\boldsymbol{a}; \boldsymbol{a}') := D_{\mathrm{KL}}\Big(q(\cdot \mid \boldsymbol{a}) \parallel q(\cdot \mid \boldsymbol{a}')\Big), \qquad \boldsymbol{a}, \boldsymbol{a}' \in \mathcal{X}^{\times p},$$

where $D_{\mathrm{KL}}$ denotes the KL divergence. The following lemma gives a decomposition for the KL divergence between two samples of a $p$-Markov process. Lemmas 27, 28 and 29 are proved later in Appendix C.4.

**Lemma 27.** *Assume that the process is $p$-Markov in the sense of* (C.15)*. Then,*

$$D_{\mathrm{KL}}(\mathbb{P}_{\mathbf{z}} \,\|\, \mathbb{P}_{\boldsymbol{y}}) = \sum_{t=1}^{p} \mathbb{E}_{\mathbf{z}}\left[ d_K\Big((\mathbb{X}_1^{t-1}, \mathbf{z}_{t-p}^0)\ ;\ (\mathbb{X}_1^{t-1}, \boldsymbol{y}_{t-p}^0)\Big)\right].$$

Here, $\mathbb{E}_{\mathbf{z}}$ denotes the expectation assuming that $\mathbb{X}_t^{t-1}$ is distributed as $\mathbb{P}_{\mathbf{z}}$. The notation $(\mathbb{X}_1^{t-1}, \mathbf{z}_{t-p}^0) \in \mathcal{X}^{\times p}$ denotes an $N \times p$ matrix with columns in $\mathcal{X}$, partitioned across columns into $N \times (t-1)$ matrix $\mathbb{X}_1^{t-1}$ and $N \times (p-t+1)$ matrix $\mathbf{z}_{t-p}^0$.

We also note the following bounds on the KL divergences between Bernoulli random variables and Poisson random variables to be used in proving Lemmas 7 and 8 respectively.

**Lemma 28.** *Let $U \sim \mathrm{Ber}(p)$, and $V \sim \mathrm{Ber}(q)$ for $p, q \in [\varepsilon, 1-\varepsilon]$ for some $\varepsilon \in (0, \frac{1}{2})$. Then,*

$$D_{\mathrm{KL}}(U \| V) = p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q} \ \leqslant\ \frac{3}{4\varepsilon(1-\varepsilon)}(p-q)^2.$$

**Lemma 29.** *Let $U = \min\{M, \mathrm{Poisson}(p)\}$, and $V = \min\{M, \mathrm{Poisson}(q)\}$ for $p, q > \varepsilon > 0$ for*

*some $\varepsilon$. Then,*

$$D_{\mathrm{KL}}(U\|V) \leqslant p\log\frac{p}{q} + (q-p) \;\leqslant\; \frac{1}{q}(p-q)^2 \leqslant \frac{1}{\varepsilon}(p-q)^2$$

*Proof of Lemma 7.* Continuing with the proof of Lemma 7, recall that $\mathcal{S} = \{0,1\}^N$, and

$$\boldsymbol{x}^t \mid \mathbb{X}_{t-p}^{t-1} \;\sim\; \prod_{i=1}^{N} \mathrm{Ber}\left(f_i\left(\langle \Theta_i, \mathbb{X}_{t-p}^{t-1}\mathbf{D}\rangle\right)\right).$$

Let $\alpha_i^t = \langle \Theta_i^*, (\mathbb{X}_1^{t-1}, \mathbf{z}_{t-p}^0)\mathbf{D}\rangle$ and $\beta_i^t = \langle \Theta_i, (\mathbb{X}_1^{t-1}, \boldsymbol{y}_{t-p}^0)\mathbf{D}\rangle$. Then using the decomposability of the KL divergence for product measures,

$$\begin{aligned} d_K\big((\mathbb{X}_1^{t-1}, \mathbf{z}_{t-p}^0) \,\|\, (\mathbb{X}_1^{t-1}, \boldsymbol{y}_{t-p}^0)\big) &= \sum_{i=1}^{N} D_{\mathrm{KL}}\left(\mathrm{Ber}\big(f_i(\alpha_i^t)\big) \,\|\, \mathrm{Ber}\big(f_i(\beta_i^t)\big)\right), \\ &\leqslant \frac{3}{4\varepsilon(1-\varepsilon)} \sum_{i=1}^{N} \big[f_i(\alpha_i^t) - f_i(\beta_i^t)\big]^2. \end{aligned}$$

By the Lipschitz assumption, $[f_i(\alpha_i^t) - f_i(\beta_i^t)]^2 \leqslant L_i^2\big(\alpha_i^t - \beta_i^t\big)^2$. Using $\varepsilon < 1/2$, it follows that

$$D_{\mathrm{KL}}(\mathbb{P}_{\mathbf{z}} \,\|\, \mathbb{P}_{\boldsymbol{y}}) \leqslant \frac{3}{2\varepsilon} \sum_{i=1}^{N} L_i^2 \sum_{t=1}^{p} \mathbb{E}_{\mathbf{z}}\big(\alpha_i^t - \beta_i^t\big)^2.$$

Let $d_{m\ell} = (\boldsymbol{d}_\ell)_m$ be the $(m,\ell)$th entry of $\mathbf{D}$. Let $z_j^{t-m}, m = t, \ldots, p$ denote entries on the $j$th row of $\mathbf{z}_{t-p}^0$ and similarly for $\boldsymbol{y}_{t-p}^0$. We have

$$\alpha_i^t - \beta_i^t = \langle \Theta_i^*, (\mathbf{0}_{N\times(t-1)}, \mathbf{z}_{t-p}^0 - \boldsymbol{y}_{t-p}^0)\mathbf{D}\rangle = \sum_{j\ell} \Theta_{ij\ell}^* \sum_{m=t}^{p} (z_j^{t-m} - y_j^{t-m}) d_{m\ell},$$

where $\mathbf{0}_{N\times(t-1)}$ is the $N \times (t-1)$ zero matrix. Assuming that $\mathcal{X}_i \subset [-B_i, B_i]$, we have

$$\begin{aligned} |\alpha_i^t - \beta_i^t| &\leqslant \sum_{j\ell} |\Theta_{ij\ell}| \sum_{m=t}^{p} \big(|z_j^{t-m}| + |y_j^{t-m}|\big)|d_{m\ell}| \\ &\leqslant 2B \sum_{j\ell} |\Theta_{ij\ell}| \sum_{m=t}^{p} |d_{m\ell}|. \end{aligned}$$

Putting the pieces together finishes the proof. ∎

*Proof of Lemma 8.* The proof of Lemma 8, proceeds almost identically to that of 7. In this case however $\mathcal{S} = \mathbb{N}^N$, and

$$\boldsymbol{x}^t \mid \mathbb{X}_{t-p}^{t-1} \;\sim\; \prod_{i=1}^{N} \mathrm{Poisson}\left(f_i\left(\langle \Theta_i^*, \mathbb{X}_{t-p}^{t-1}\mathbf{D}\rangle\right)\right).$$

Let $\alpha_i^t = \langle \Theta_i^*, (\mathbb{X}_1^{t-1}, \mathbf{z}_{t-p}^0)\mathbf{D}\rangle$ and $\beta_i^t = \langle \Theta_i^*, (\mathbb{X}_1^{t-1}, \boldsymbol{y}_{t-p}^0)\mathbf{D}\rangle$. Then using the decomposability

of the KL divergence for product measures,

$$d_K\big((\mathbb{X}_1^{t-1}, \mathbf{z}_{t-p}^0) \parallel (\mathbb{X}_1^{t-1}, \boldsymbol{y}_{t-p}^0)\big) = \sum_{i=1}^N D_{\mathrm{KL}}\left(\mathrm{Poisson}\big(f_i(\alpha_i^t)\big) \parallel \mathrm{Poisson}\big(f_i(\beta_i^t)\big)\right),$$
$$\leqslant \frac{1}{\varepsilon}\sum_{i=1}^N \big[f_i(\alpha_i^t) - f_i(\beta_i^t)\big]^2 \;\leqslant\; \frac{1}{\varepsilon}\sum_{i=1}^N L_i^2\big(\alpha_i^t - \beta_i^t\big)^2,$$

where the first inequality is using Lemma 29 and the second by the Lipschitz assumption on $f_i$. The rest follows identically as in the proof of Lemma 7. ∎

### C.4.2 Proof of Lemma 27

Recall the notation $X_1^p = (x_p, \ldots, x_1)$. Similarly, let $a = (a_p, \ldots, a_1) \in \mathcal{X}^{\times p}$ so that $X_1^p = a$ is the same as $X_u = a_u$ for all $u \in [p]$. We also write $a_1^{t-1} = (a_{t-1}, \ldots, a_1)$ and so on for elements of $\mathcal{X}^{\times p}$. For any $a, z \in \mathcal{X}^{\times p}$, we have

$$\begin{aligned} p_z(a) &= \mathbb{P}(X_1^p = a \mid X_{1-p}^0 = z) \\ &= \prod_{t=1}^p \mathbb{P}(x_t = a_t \mid X_1^{t-1} = a_1^{t-1},\, X_{t-p}^0 = z_{t-p}^0) \\ &= \prod_{t=1}^p \mathbb{P}\big(x_t = a_t \;\big|\; X_{t-p}^{t-1} = (a_1^{t-1}, z_{t-p}^0)\big) = \prod_{t=1}^p q(a_t \mid (a_1^{t-1}, z_{t-p}^0)) \end{aligned}$$

where the second line is by the Markov property. Replacing $a$ with a random variable $X_1^p \in \mathcal{X}^{\times p}$,

$$p_z(X_1^p) = \prod_{t=1}^p q\big(x_t \mid (X_1^{t-1}, z_{t-p}^0)\big).$$

Letting $\mathbb{E}_z$ denote the expectation assuming $X_1^p \sim \mathbb{P}_z$, we have

$$\begin{aligned} D_{\mathrm{KL}}(\mathbb{P}_z \,\|\, \mathbb{P}_y) &= \mathbb{E}_z \log \frac{p_z(X_1^p)}{p_y(X_1^p)} \\ &= \sum_{t=1}^p \mathbb{E}_z \log \frac{q\big(x_t \mid (X_1^{t-1}, z_{t-p}^0)\big)}{q\big(x_t \mid (X_1^{t-1}, y_{t-p}^0)\big)} \\ &= \sum_{t=1}^p \mathbb{E}_z \mathbb{E}_z\Big[\log \frac{q\big(x_t \mid (X_1^{t-1}, z_{t-p}^0)\big)}{q\big(x_t \mid (X_1^{t-1}, y_{t-p}^0)\big)} \;\Big|\; X_1^{t-1}\Big] \\ &= \sum_{t=1}^p \mathbb{E}_z\, d_K\big((X_1^{t-1}, z_{t-p}^0) \parallel (X_1^{t-1}, y_{t-p}^0)\big) \end{aligned}$$

where the last line follows by noting that under $X_1^p \sim \mathbb{P}_z$, further conditioning on $X_1^{t-1}$ is equiv-

alent to conditioning on $X_1^{t-1}$ and $X_{t-p}^0 = z_{t-p}^0$, i.e., $x_t$ will have distribution $q(\cdot \mid (X_1^{t-1}, z_{t-p}^0))$ under this conditioning. □

### C.4.3 Proof of Lemma 28

It is enough to prove for the case $q \geqslant p$ (the other case follows by applying the proven case to $1-p$ and $1-q$). The second claim follows from the decomposition of the KL divergence for product distributions. Let $\delta := \varepsilon(1-\varepsilon)$. Fix $p$ and consider the function

$$f(q) = p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q} - \frac{1}{4\delta}(p-q)^2,$$

over $q \in [p, 1-\varepsilon]$. We have

$$f'(q) = (q-p)\Big(\frac{1}{q(1-q)} - \frac{1}{2\delta}\Big).$$

We have $f(q) = f(p) + f'(\widetilde{q})(q-p)$ for some $\widetilde{q} \in [p,q]$. Note that $f(p) = 0$ and

$$f'(\widetilde{q}) \leqslant (\widetilde{q}-p)\Big(\frac{1}{\delta} - \frac{1}{2\delta}\Big) \leqslant \frac{1}{2\delta}(q-p)$$

using the fact that $(\widetilde{q}(1-\widetilde{q}))^{-1} \in [4, \delta^{-1}]$. Thus, we have $f(q) \leqslant (q-p)^2/(2\delta)$. □

### C.4.4 Proof of Lemma 29

The KL divergence between two Poisson distributions with parameters $p$ and $q$ is given by

$$p \log \frac{p}{q} + (q-p) - \frac{(q-p)^2}{q} = p(\log \frac{p}{q} + 1 - \frac{p}{q})$$

We show that the truncation only reduces the KL divergence using Jensen's inequality for the convex function $g(u,v) = u \log(u,v)$. Let $p_i := e^{-p}\frac{p^i}{i!}$ and $q_i := e^{-q}\frac{q^i}{i!}$. Next, observe that the KL divergence for the truncated version is

$$\sum_{i<M} p_i \log \frac{p_i}{q_i} + \sum_{i \geqslant M} p_i \log \frac{\sum_{i \geqslant M} p_i}{\sum_{i \geqslant M} q_i}$$

Applying the Jensen's inequality to second term, we get that the quantity above is at most

$$\sum_{i<M} p_i \log \frac{p_i}{q_i} + \sum_{i \geqslant M} p_i \log \frac{p_i}{q_i}$$

which is the KL divergence between Poisson($p$) and Poisson($q$). Finally, observe that for $p, q > 0$

$$p \log \frac{p}{q} + (q - p) - \frac{(q-p)^2}{q} = p(\log \frac{p}{q} + 1 - \frac{p}{q}) \leqslant 0$$

where we use the inequality $\log x \leqslant x - 1$. □

# Bibliography

[Abadi et al., 2015] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

[Ahelegbey et al., 2016] Ahelegbey, D. F., Billio, M., and Casarin, R. (2016). Sparse graphical vector autoregression: A bayesian approach. *Annals of Economics and Statistics/Annales d'Économie et de Statistique*, (123/124):333–361.

[Alemohammad et al., 2020] Alemohammad, S., Wang, Z., Balestriero, R., and Baraniuk, R. (2020). The recurrent neural tangent kernel. *arXiv preprint arXiv:2006.10246*.

[Allen-Zhu et al., 2018] Allen-Zhu, Z., Li, Y., and Song, Z. (2018). A convergence theory for deep learning via over-parameterization. *arXiv preprint arXiv:1811.03962*.

[Anthony and Bartlett, 2009] Anthony, M. and Bartlett, P. L. (2009). *Neural network learning: Theoretical foundations*. cambridge university press.

[Aubin et al., 2018] Aubin, B., Maillard, A., Krzakala, F., Macris, N., Zdeborová, L., et al. (2018). The committee machine: Computational to statistical gaps in learning a two-layers neural network. In *Advances in Neural Information Processing Systems*, pages 3223–3234.

[Barbier et al., 2019] Barbier, J., Krzakala, F., Macris, N., Miolane, L., and Zdeborová, L. (2019). Optimal errors and phase transitions in high-dimensional generalized linear models. *Proc. Nat. Acad. Sci.*, 116(12):5451–5460.

[Basu et al., 2015] Basu, S., Michailidis, G., et al. (2015). Regularized estimation in sparse high-dimensional time series models. *The Annals of Statistics*, 43(4):1535–1567.

[Bayati and Montanari, 2011a] Bayati, M. and Montanari, A. (2011a). The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information Theory*, 57(2):764–785.

[Bayati and Montanari, 2011b] Bayati, M. and Montanari, A. (2011b). The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Trans. Inform. Theory*, 57(2):764–785.

[Belkin et al., 2019a] Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019a). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854.

[Belkin et al., 2019b] Belkin, M., Hsu, D., and Xu, J. (2019b). Two models of double descent for weak features. *arXiv preprint arXiv:1903.07571*.

[Belkin et al., 2018] Belkin, M., Ma, S., and Mandal, S. (2018). To understand deep learning we need to understand kernel learning. In *International Conference on Machine Learning*, pages 541–549. PMLR.

[Bertsekas, 2011] Bertsekas, D. P. (2011). Incremental gradient, subgradient, and proximal methods for convex optimization: A survey. *Optimization for Machine Learning*, 2010(1-38):3.

[Bickel et al., 2009] Bickel, P. J., Ritov, Y., and Tsybakov, A. B. (2009). Simultaneous analysis of lasso and dantzig selector. *The Annals of statistics*, 37(4):1705–1732.

[Bora et al., 2017] Bora, A., Jalal, A., Price, E., and Dimakis, A. G. (2017). Compressed sensing using generative models. *Proc. ICML*.

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

[Byrne et al., 2019] Byrne, E., Chatalic, A., Gribonval, R., and Schniter, P. (2019). Sketched clustering via hybrid approximate message passing. *IEEE Transactions on Signal Processing*, 67(17):4556–4569.

[Cai et al., 2016] Cai, T. T., Ren, Z., Zhou, H. H., et al. (2016). Estimating structured high-dimensional covariance and precision matrices: Optimal rates and adaptive estimation. *Electronic Journal of Statistics*, 10(1):1–59.

[Cakmak et al., 2014] Cakmak, B., Winther, O., and Fleury, B. H. (2014). S-AMP: Approximate message passing for general matrix ensembles. In *Proc. IEEE ITW*.

[Candes et al., 2006] Candes, E. J., Romberg, J. K., and Tao, T. (2006). Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(8):1207–1223.

[Candes and Tao, 2005] Candes, E. J. and Tao, T. (2005). Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215.

[Candes and Tao, 2006] Candes, E. J. and Tao, T. (2006). Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE transactions on information theory*, 52(12):5406–5425.

[Chen and Donoho, 1994] Chen, S. and Donoho, D. (1994). Basis pursuit. In *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 41–44. IEEE.

[Cheng et al., 2018] Cheng, X., Chatterji, N. S., Abbasi-Yadkori, Y., Bartlett, P. L., and Jordan, M. I. (2018). Sharp convergence rates for langevin dynamics in the nonconvex setting. *arXiv preprint arXiv:1805.01648*.

[Chung et al., 2012] Chung, K.-M., Lam, H., Liu, Z., and Mitzenmacher, M. (2012). Chernoff-hoeffding bounds for markov chains: Generalized and simplified. *arXiv preprint arXiv:1201.0559*.

[Cotter et al., 2005] Cotter, S. F., Rao, B. D., Engan, K., and Kreutz-Delgado, K. (2005). Sparse solutions to linear inverse problems with multiple measurement vectors. *IEEE Transactions on Signal Processing*, 53(7):2477–2488.

[Csiszar and Körner, 2011] Csiszar, I. and Körner, J. (2011). *Information theory: coding theorems for discrete memoryless systems*. Cambridge University Press.

[De Mol et al., 2008] De Mol, C., Giannone, D., and Reichlin, L. (2008). Forecasting using a large number of predictors: Is bayesian shrinkage a valid alternative to principal components? *Journal of Econometrics*, 146(2):318–328.

[DeJong and Whiteman, 1991] DeJong, D. N. and Whiteman, C. H. (1991). The temporal stability of dividends and stock prices: Evidence from the likelihood function. *The American Economic Review*, pages 600–617.

[Deng et al., 2019] Deng, Z., Kammoun, A., and Thrampoulidis, C. (2019). A model of double descent for high-dimensional binary linear classification. *arXiv preprint arXiv:1911.05822*.

[Dicker et al., 2016] Dicker, L. H. et al. (2016). Ridge regression and asymptotic minimax estimation over spheres of growing dimension. *Bernoulli*, 22(1):1–37.

[Dobriban et al., 2018] Dobriban, E., Wager, S., et al. (2018). High-dimensional asymptotics of prediction: Ridge regression and classification. *The Annals of Statistics*, 46(1):247–279.

[Donoho, 2006] Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306.

[Donoho et al., 2009] Donoho, D. L., Maleki, A., and Montanari, A. (2009). Message-passing algorithms for compressed sensing. *Proc. Nat. Acad. Sci.*, 106(45):18914–18919.

[Donoho et al., 2010a] Donoho, D. L., Maleki, A., and Montanari, A. (2010a). Message passing algorithms for compressed sensing. In *Proc. Inform. Theory Workshop*, pages 1–5.

[Donoho et al., 2010b] Donoho, D. L., Maleki, A., and Montanari, A. (2010b). Message passing algorithms for compressed sensing: I. motivation and construction. In *2010 IEEE information theory workshop on information theory (ITW 2010, Cairo)*, pages 1–5. IEEE.

[Du et al., 2018a] Du, S. S., Lee, J. D., Li, H., Wang, L., and Zhai, X. (2018a). Gradient descent finds global minima of deep neural networks. *arXiv preprint arXiv:1811.03804*.

[Du et al., 2018b] Du, S. S., Zhai, X., Poczos, B., and Singh, A. (2018b). Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*.

[Eldar and Kutyniok, 2012] Eldar, Y. C. and Kutyniok, G. (2012). *Compressed sensing: theory and applications*. Cambridge university press.

[Emami et al., 2020] Emami, M., Sahraee-Ardakan, M., Pandit, P., Rangan, S., and Fletcher, A. K. (2020). Generalization error of generalized linear models in high dimensions. *arXiv preprint arXiv:2005.00180*.

[Fan et al., 2018] Fan, J., Jiang, B., and Sun, Q. (2018). Hoeffding's lemma for markov chains and its applications to statistical learning. *arXiv preprint arXiv:1802.00211*.

[Fletcher et al., 2018] Fletcher, A. K., Rangan, S., and Schniter, P. (2018). Inference in deep networks in high dimensions. *Proc. IEEE Int. Symp. Information Theory*.

[Fletcher et al., 2016] Fletcher, A. K., Sahraee-Ardakan, M., Rangan, S., and Schniter, P. (2016). Expectation consistent approximate inference: Generalizations and convergence. In *Proc. IEEE Int. Symp. Information Theory*, pages 190–194.

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

[Gabrié et al., 2018] Gabrié, M., Manoel, A., Luneau, C., Barbier, J., Macris, N., Krzakala, F., and Zdeborová, L. (2018). Entropy and mutual information in models of deep neural networks. In *Proc. NIPS*.

[Gray et al., 2006] Gray, R. M. et al. (2006). Toeplitz and circulant matrices: A review. *Foundations and Trends® in Communications and Information Theory*, 2(3):155–239.

[Hall et al., 2018] Hall, E. C., Raskutti, G., and Willett, R. M. (2018). Learning high-dimensional generalized linear autoregressive models. *IEEE Transactions on Information Theory*, 65(4):2401–2422.

[Hand and Voroninski, 2017] Hand, P. and Voroninski, V. (2017). Global guarantees for enforcing deep generative priors by empirical risk. *arXiv:1705.07576*.

[Hastie et al., 2019] Hastie, T., Montanari, A., Rosset, S., and Tibshirani, R. J. (2019). Surprises in high-dimensional ridgeless least squares interpolation. *arXiv preprint arXiv:1903.08560*.

[He et al., 2017] He, H., Wen, C.-K., and Jin, S. (2017). Generalized expectation consistent signal recovery for nonlinear measurements. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 2333–2337. IEEE.

[Huang et al., 2018] Huang, Y., Cheng, Y., Chen, D., Lee, H., Ngiam, J., Le, Q. V., and Chen, Z. (2018). Gpipe: Efficient training of giant neural networks using pipeline parallelism. *CoRR*, abs/1811.06965.

[Jacob et al., 2009] Jacob, L., Obozinski, G., and Vert, J.-P. (2009). Group lasso with overlap and graph lasso. In *Proceedings of the 26th annual international conference on machine learning*, pages 433–440.

[Jacot et al., 2018] Jacot, A., Gabriel, F., and Hongler, C. (2018). Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pages 8571–8580.

[Kabashima, 2003] Kabashima, Y. (2003). A cdma multiuser detection algorithm on the basis of belief propagation. *Journal of Physics A: Mathematical and General*, 36(43):11111.

[Kabkab et al., 2018] Kabkab, M., Samangouei, P., and Chellappa, R. (2018). Task-aware compressed sensing with generative adversarial networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[Kalimeris et al., 2019] Kalimeris, D., Kaplun, G., Nakkiran, P., Edelman, B., Yang, T., Barak, B., and Zhang, H. (2019). Sgd on neural networks learns functions of increasing complexity. In *Advances in Neural Information Processing Systems*, pages 3496–3506.

[Katselis et al., 2018] Katselis, D., Beck, C., and Srikant, R. (2018). Mixing times and structural inference for bernoulli autoregressive processes. *IEEE Transactions on Network Science and Engineering*.

[Kazemipour, 2018] Kazemipour, A. (2018). Compressed sensing beyond the iid and static domains: Theory, algorithms and applications. *Ph.D. dissertation. arXiv:1806.11194*.

[Kazemipour et al., 2017] Kazemipour, A., Wu, M., and Babadi, B. (2017). Robust estimation of self-exciting generalized linear models with application to neuronal modeling. *IEEE Transactions on Signal Processing*, 65(14):3733–3748.

[Keriven et al., 2017a] Keriven, N., Bourrier, A., Gribonval, R., and Pérez, P. (2017a). Sketching for large-scale learning of mixture models. *Information and Inference: A Journal of the IMA*, 7(3):447–508.

[Keriven et al., 2017b] Keriven, N., Tremblay, N., Traonmilin, Y., and Gribonval, R. (2017b). Compressive k-means. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6369–6373. IEEE.

[Kim et al., 2006] Kim, Y., Kim, J., and Kim, Y. (2006). Blockwise sparse regression. *Statistica Sinica*, pages 375–390.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Kontorovich, 2012] Kontorovich, A. (2012). Obtaining measure concentration from markov contraction. *Markov Processes and Related Fields*, 18:613–638.

[Kontorovich et al., 2008] Kontorovich, L. A., Ramanan, K., et al. (2008). Concentration inequalities for dependent random variables via the martingale method. *The Annals of Probability*, 36(6):2126–2158.

[Krein and Smulian, 1940] Krein, M. and Smulian, V. (1940). On regularly convex sets in the space conjugate to a banach space. *Annals of Mathematics*, pages 556–583.

[Li et al., 2020] Li, M., Soltanolkotabi, M., and Oymak, S. (2020). Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 4313–4324. PMLR.

[Li and Liang, 2018] Li, Y. and Liang, Y. (2018). Learning overparameterized neural networks via stochastic gradient descent on structured data. In *Advances in Neural Information Processing Systems*, pages 8157–8166.

[Li et al., 2019] Li, Z., Wang, R., Yu, D., Du, S. S., Hu, W., Salakhutdinov, R., and Arora, S. (2019). Enhanced convolutional neural tangent kernels. *arXiv preprint arXiv:1911.00809*.

[Liang et al., 2009] Liang, D., Ying, L., and Liang, F. (2009). Parallel MRI Acceleration Using M-FOCUSS. In *Proc. International Conference on Bioinformatics and Biomedical Engineering*, pages 1–4. IEEE.

[Liang et al., 2020] Liang, T., Rakhlin, A., et al. (2020). Just interpolate: Kernel "ridgeless" regression can generalize. *Annals of Statistics*, 48(3):1329–1347.

[Liao et al., 2020] Liao, Z., Couillet, R., and Mahoney, M. W. (2020). A random matrix analysis of random fourier features: beyond the gaussian kernel, a precise phase transition, and the corresponding double descent. *arXiv preprint arXiv:2006.05013*.

[Liu, 1994] Liu, J. S. (1994). Siegel's formula via stein's identities. *Statistics & Probability Letters*, 21(3):247–251.

[Lütkepohl, 2005] Lütkepohl, H. (2005). *New introduction to multiple time series analysis*. Springer Science & Business Media.

[Ma and Ping, 2017] Ma, J. and Ping, L. (2017). Orthogonal AMP. *IEEE Access*, 5:2020–2033.

[Maleki et al., 2013] Maleki, A., Anitori, L., Yang, Z., and Baraniuk, R. G. (2013). Asymptotic analysis of complex lasso via complex approximate message passing (camp). *IEEE Transactions on Information Theory*, 59(7):4290–4308.

[Manoel et al., 2017] Manoel, A., Krzakala, F., Mézard, M., and Zdeborová, L. (2017). Multi-layer generalized linear estimation. In *Proc. IEEE Int. Symp. Information Theory*, pages 2098–2102.

[Manoel et al., 2018] Manoel, A., Krzakala, F., Varoquaux, G., Thirion, B., and Zdeborová, L. (2018). Approximate message-passing for convex optimization with non-separable penalties. *arXiv preprint arXiv:1809.06304*.

[Mark et al., 2017] Mark, B., Raskutti, G., and Willett, R. (2017). Network estimation via poisson autoregressive models. In *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2017 IEEE 7th International Workshop on*, pages 1–5. IEEE.

[Mark et al., 2018] Mark, B., Raskutti, G., and Willett, R. (2018). Network estimation from point process data. *IEEE Transactions on Information Theory*, 65(5):2953–2975.

[Marton et al., 1996] Marton, K. et al. (1996). Bounding $\overline{d}$-distance by informational divergence: A method to prove measure concentration. *The Annals of Probability*, 24(2):857–866.

[McMurry et al., 2015] McMurry, T. L., Politis, D. N., et al. (2015). High-dimensional autocovariance matrices and optimal linear prediction. *Electronic Journal of Statistics*, 9(1):753–788.

[McNally et al., 1999] McNally, J. G., Karpova, T., Cooper, J., and Conchello, J. A. (1999). Three-dimensional imaging by deconvolution microscopy. *Methods*, 19(3):373–385.

[Mei and Moura, 2017] Mei, J. and Moura, J. M. (2017). Signal processing on graphs: Causal modeling of unstructured data. *IEEE Trans. Signal Processing*, 65(8):2077–2092.

[Mei and Montanari, 2019] Mei, S. and Montanari, A. (2019). The generalization error of random features regression: Precise asymptotics and double descent curve. *arXiv preprint arXiv:1908.05355*.

[Mei et al., 2018] Mei, S., Montanari, A., and Nguyen, P.-M. (2018). A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671.

[Minka, 2001] Minka, T. P. (2001). Expectation propagation for approximate bayesian inference. In *Proc. UAI*, pages 362–369.

[Mixon and Villar, 2018] Mixon, D. G. and Villar, S. (2018). Sunlayer: Stable denoising with generative networks. *arXiv preprint arXiv:1803.09319*.

[Montanari et al., 2012] Montanari, A., Eldar, Y., and Kutyniok, G. (2012). Graphical models concepts in compressed sensing.

[Montanari et al., 2019] Montanari, A., Ruan, F., Sohn, Y., and Yan, J. (2019). The generalization error of max-margin linear classifiers: High-dimensional asymptotics in the overparametrized regime. *arXiv preprint arXiv:1911.01544*.

[Mueller, 1985] Mueller, C. S. (1985). Source pulse enhancement by deconvolution of an empirical green's function. *Geophysical Research Letters*, 12(1):33–36.

[Nakkiran et al., 2019] Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. (2019). Deep double descent: Where bigger models and more data hurt. *arXiv preprint arXiv:1912.02292*.

[Nakkiran et al., 2021] Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. (2021). Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003.

[Negahban et al., 2012] Negahban, S. N., Ravikumar, P., Wainwright, M. J., and Yu, B. (2012). A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science*, pages 538–557.

[Obozinski et al., 2006] Obozinski, G., Taskar, B., and Jordan, M. (2006). Multi-task feature selection. *Statistics Department, UC Berkeley, Tech. Rep*, 2(2.2).

[Okatan et al., 2005] Okatan, M., Wilson, M. A., and Brown, E. N. (2005). Analyzing functional connectivity using a network likelihood model of ensemble neural spiking activity. *Neural computation*, 17(9):1927–1961.

[Opper and Winther, 2005] Opper, M. and Winther, O. (2005). Expectation consistent approximate inference. *J. Machine Learning Res.*, 6:2177–2204.

[Oymak and Soltanolkotabi, 2019] Oymak, S. and Soltanolkotabi, M. (2019). Overparameterized nonlinear learning: Gradient descent takes the shortest path? In *International Conference on Machine Learning*, pages 4951–4960.

[Pandit et al., 2019] Pandit, P., Sahraee, M., Rangan, S., and Fletcher, A. K. (2019). Asymptotics of MAP inference in deep networks. In *Proc. IEEE Int. Symp. Information Theory*, pages 842–846.

[Pandit et al., 2019a] Pandit, P., Sahraee-Ardakan, M., Amini, A., Rangan, S., and Fletcher, A. K. (2019a). Sparse multivariate bernoulli processes in high dimensions. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 457–466.

[Pandit et al., 2019b] Pandit, P., Sahraee-Ardakan, M., Rangan, S., Schniter, P., and Fletcher, A. K. (2019b). Inference with deep generative priors in high dimensions.

[Pandit et al., 2020] Pandit, P., Sahraee-Ardakan, M., Rangan, S., Schniter, P., and Fletcher, A. K. (2020). Inference with deep generative priors in high dimensions. *IEEE Journal on Selected Areas in Information Theory*.

[Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

[Peligrad et al., 2010] Peligrad, M., Wu, W. B., et al. (2010). Central limit theorem for fourier transforms of stationary processes. *The Annals of Probability*, 38(5):2009–2022.

[Raginsky et al., 2012] Raginsky, M., Willett, R. M., Horn, C., Silva, J., and Marcia, R. F. (2012). Sequential anomaly detection in the presence of noise and limited feedback. *IEEE Transactions on Information Theory*, 58(8):5544–5562.

[Rakhlin et al., 2015] Rakhlin, A., Sridharan, K., and Tewari, A. (2015). Sequential complexities and uniform martingale laws of large numbers. *Probability Theory and Related Fields*, 161(1-2):111–153.

[Rangan et al., 2019a] Rangan, S., Schniter, P., and Fletcher, A. K. (2019a). Vector approximate message passing. *IEEE Transactions on Information Theory*, 65(10):6664–6684.

[Rangan et al., 2019b] Rangan, S., Schniter, P., and Fletcher, A. K. (2019b). Vector approximate message passing. *IEEE Trans. Information Theory*, 65(10):6664–6684.

[Raskutti et al., 2011] Raskutti, G., Wainwright, M. J., and Yu, B. (2011). Minimax rates of estimation for high-dimensional linear regression over $\ell_q$-balls. *IEEE transactions on information theory*, 57(10):6976–6994.

[Raskutti et al., 2019] Raskutti, G., Yuan, M., Chen, H., et al. (2019). Convex regularization for high-dimensional multiresponse tensor regression. *The Annals of Statistics*, 47(3):1554–1584.

[Reeves, 2017] Reeves, G. (2017). Additivity of information in multilayer networks via additive Gaussian noise transforms. In *Proc. Allerton Conf. Comm. Control & Comput.*, pages 1064–1070.

[Rhodius, 1997] Rhodius, A. (1997). On the maximum of ergodicity coefficients, the dobrushin ergodicity coefficient, and products of stochastic matrices. *Linear algebra and its applications*, 253(1-3):141–154.

[Rudelson et al., 2013] Rudelson, M., Vershynin, R., et al. (2013). Hanson-wright inequality and sub-gaussian concentration. *Electronic Communications in Probability*, 18.

[Sahraee-Ardakan et al., 2020] Sahraee-Ardakan, M., Pandit, P., Amini, A., Rangan, S., and Fletcher, A. K. (2020). *Multivariate Autoregressive Generalized Linear Model regression in PyTorch*. https://github.com/mojtabasah/AR_process.

[Samson et al., 2000] Samson, P.-M. et al. (2000). Concentration of measure inequalities for markov chains and Φ-mixing processes. *The Annals of Probability*, 28(1):416–461.

[Shah and Hegde, 2018] Shah, V. and Hegde, C. (2018). Solving linear inverse problems using GAN priors: An algorithm with provable guarantees. In *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, pages 4609–4613.

[Smith and Brown, 2003] Smith, A. C. and Brown, E. N. (2003). Estimating a state-space model from point process observations. *Neural computation*, 15(5):965–991.

[Soltanolkotabi et al., 2018] Soltanolkotabi, M., Javanmard, A., and Lee, J. D. (2018). Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *IEEE Transactions on Information Theory*, 65(2):742–769.

[Starck et al., 2002] Starck, J.-L., Pantin, E., and Murtagh, F. (2002). Deconvolution in astronomy: A review. *Publications of the Astronomical Society of the Pacific*, 114(800):1051.

[Takeuchi, 2017] Takeuchi, K. (2017). Rigorous dynamics of expectation-propagation-based signal recovery from unitarily invariant measurements. In *Proc. IEEE Int. Symp. Information Theory*, pages 501–505.

[Themelis and Patrinos, 2020] Themelis, A. and Patrinos, P. (2020). Douglas–rachford splitting and admm for nonconvex optimization: Tight convergence results. *SIAM Journal on Optimization*, 30(1):149–181.

[Tibshirani, 1996] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

[Timmermann, 1996] Timmermann, A. (1996). Excess volatility and predictability of stock prices in autoregressive dividend models with learning. *The Review of Economic Studies*, 63(4):523–557.

[Treitel and Lines, 1982] Treitel, S. and Lines, L. (1982). Linear inverse theory and deconvolution. *Geophysics*, 47(8):1153–1159.

[Tresp, 2000] Tresp, V. (2000). A bayesian committee machine. *Neural computation*, 12(11):2719–2741.

[Tripathi et al., 2018] Tripathi, S., Lipton, Z. C., and Nguyen, T. Q. (2018). Correction by projection: Denoising images with generative adversarial networks. *arXiv preprint arXiv:1803.04477*.

[Tzagkarakis et al., 2010] Tzagkarakis, G., Milioris, D., and Tsakalides, P. (2010). Multiple-measurement Bayesian compressed sensing using GSM priors for DOA estimation. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2610–2613. IEEE.

[Ulyanov et al., 2018] Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2018). Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454.

[van de Geer, 2002] van de Geer, S. A. (2002). On hoeffding's inequality for dependent random variables. In *Empirical process techniques for dependent data*, pages 161–169. Springer.

[Vershynin, 2018] Vershynin, R. (2018). *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge University Press.

[Villani, 2008] Villani, C. (2008). *Optimal transport: old and new*, volume 338. Springer Science & Business Media.

[Wainwright, 2019] Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press.

[Weber and Pillow, 2017] Weber, A. I. and Pillow, J. W. (2017). Capturing the dynamical repertoire of single neurons with generalized linear models. *Neural computation*, 29(12):3260–3289.

[Welling and Teh, 2011] Welling, M. and Teh, Y. W. (2011). Bayesian learning via stochastic gradient Langevin dynamics. In *Proc. 28th Int. Conf. Machine Learning*, pages 681–688.

[Wright et al., 2009] Wright, S. J., Nowak, R. D., and Figueiredo, M. A. (2009). Sparse reconstruction by separable approximation. *IEEE Transactions on signal processing*, 57(7):2479–2493.

[Yang, 2020] Yang, G. (2020). Tensor programs ii: Neural tangent kernel for any architecture. *arXiv preprint arXiv:2006.14548*.

[Yeh et al., 2016] Yeh, R., Chen, C., Lim, T. Y., Hasegawa-Johnson, M., and Do, M. N. (2016). Semantic image inpainting with perceptual and contextual losses. *arXiv:1607.07539*.

[Yi et al., 2014] Yi, X., Caramanis, C., and Sanghavi, S. (2014). Alternating minimization for mixed linear regression. In *International Conference on Machine Learning*, pages 613–621.

[Yuan and Lin, 2006] Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.

[Zhang et al., 2016] Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*.

[Zhang et al., 2008] Zhang, C.-H., Huang, J., et al. (2008). The sparsity and bias of the lasso selection in high-dimensional linear regression. *The Annals of Statistics*, 36(4):1567–1594.

[Zhou and Raskutti, 2018] Zhou, H. H. and Raskutti, G. (2018). Non-parametric sparse additive auto-regressive network models. *IEEE Transactions on Information Theory*, 65(3):1473–1492.

[Ziniel and Schniter, 2012] Ziniel, J. and Schniter, P. (2012). Efficient high-dimensional inference in the multiple measurement vector problem. *IEEE Transactions on Signal Processing*, 61(2):340–354.

[Zou et al., 2020] Zou, D., Cao, Y., Zhou, D., and Gu, Q. (2020). Gradient descent optimizes over-parameterized deep relu networks. *Machine Learning*, 109(3):467–492.