2023

Topics in sparse Bayesian machine learning

BOSTON UNIVERSITY

GRADUATE SCHOOL OF ARTS AND SCIENCES

Dissertation

TOPICS IN SPARSE BAYESIAN MACHINE LEARNING

by

QIUYUN ZHU

B.S., Nanjing University, 2016
M.S., University of Wisconsin-Madison, 2017

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

2023

Approved by

First Reader

Yves Atchadé, Ph.D.
Professor of Mathematics and Statistics

Second Reader

Konstantinos Spiliopoulos, Ph.D.
Professor of Mathematics and Statistics

Third Reader

Uri Eden, Ph.D.
Professor of Mathematics and Statistics

Fourth Reader

Jonathan Huggins, Ph.D.
Assistant Professor of Mathematics and Statistics

Acknowledgments

I would like to first thank my advisor, Yves Atchadé. He encouraged me to pursue interesting and deep ideas, gave me absolute research freedom, and provided extremely valuable guidance in my research. I am very grateful for his generous guidance and support through my doctoral studies.

I thank Prof. Konstantinos Spiliopoulos, Prof. Uri Eden and Prof. Jonathan Huggins to serve on my thesis committee. Their constructive feedback helped me improve the thesis.

I am very fortunate to be a member of the department of mathematics and statistics, which is filled with intelligent, generous, and sincere colleagues. I enjoyed the inspiring conversations with many of them.

I am also thankful to my amazing boyfriend, Renbo Zhao. His encouragement and support during the rough time of my research and life helped me to gain confidence and move on. I dedicate this thesis to him.

Qiuyun Zhu

TOPICS IN SPARSE BAYESIAN MACHINE LEARNING

QIUYUN ZHU

Boston University, Graduate School of Arts and Sciences, 2023

Major Professor: Yves Atchadé, PhD
Professor of Mathematics and Statistics

ABSTRACT

This dissertation is devoted to addressing several challenging problems in machine learning via the Bayesian approach. These problems frequently arise in diverse fields, such as epidemiology, biomedicine, robust statistics and imaging science, and are usually high-dimensional and have certain sparsity assumptions. In this dissertation, we will focus on three important problems, which are *sparse canonical correlation analysis*, *minimum distance estimation* and *inverse problems*. For each problem, we will develop a new method from the Bayesian perspective to solve it effectively and efficiently, with statistical guarantees and numerical evidence.

Contents

List of Tables

List of Figures

List of Abbreviations

CCA	.....	Canonical Correlation Analysis
FNN	.....	Feedforward neural network
GAN	.....	Generative adversarial network
GDN	.....	Gradient descent Network
MALA	.....	Metropolis-adjusted Langevin Algorithm
MCMC	.....	Markov Chain Monte Carlo
MDE	.....	Minimum distance estimation
MEDE	.....	Minimum empirical distance estimator
NN	.....	Neural Network
PDF	.....	Probability Distribution Function
$\mathbb{R}^n$	.....	the $n-$dimensional Real space
SA-SGLD	.....	Sparse Asynchronous Stochastic Gradient Langevin Dynamics
SGLD	.....	Stochastic Gradient Langevin Dynamics

Chapter 1

Introduction

In the era of big data, where technological advancements have led to the collection and storage of vast amounts of information, traditional statistical methods face significant challenges when applied to high-dimensional datasets, where the number of parameters can greatly exceed the sample size, leading to issues of overfitting, model selection, and variable selection (Wainwright (2019)). The field of high-dimensional statistics addresses these challenges by developing novel methodologies capable of effectively analyzing complex datasets with a large number of variables or features.

In the frequentist paradigm, estimation methods incorporating sparsity constraints are proposed, such as LASSO (Tibshirani (1996)) and Elastic Net (Zou and Hastie (2005a)). Those methods are based on optimization algorithms to solve the problems. However, when the problem is complicated (usually non-convex or non-concave), optimization algorithms may fail to find the optimal solution.

Bayesian statistics, rooted in the principles of probability theory and inference, offers a powerful framework for tackling high-dimensional statistical problems. Unlike classical frequentist approaches, Bayesian methods treat unknown parameters as random variables and provide a probabilistic interpretation of uncertainty. In high-dimensional settings, one can use prior knowledge that contains sparsity information, such as spike-and-slab prior (George and McCulloch (1993)) and horseshoe prior (Carvalho et al. (2009)).

This thesis aims to explore the power and applicability of Bayesian statistics in

high-dimensional statistical analysis, where the likelihood of data is very complicated. By delving into the underlying principles and methodologies of Bayesian inference, we will investigate how these techniques can effectively address the challenges posed by high-dimensional datasets. Throughout this thesis, we will discuss both theoretical foundations and practical implementations of Bayesian methods in high-dimensional statistics. The thesis is organized as follows.

In the second chapter, we propose a minimax quasi-Bayesian estimator for sparse canonical correlation analysis (CCA). For a long time, CCA has been a popular statistical technique for exploring relationships between datasets. In recent years, the estimation of sparse canonical vectors has emerged as an important but challenging variant of the CCA problem, with widespread applications. Unfortunately, existing rate-optimal estimators for sparse canonical vectors have high computational cost. We propose a quasi-Bayesian estimation procedure that not only achieves the minimax estimation rate, but also is easy to compute by Markov Chain Monte Carlo (MCMC). The method builds on (Tan et al. (2018)) and uses a re-scaled Rayleigh quotient function as the quasi-log-likelihood. However, unlike (Tan et al. (2018)), we adopt a Bayesian framework that combines this quasi-log-likelihood with a spike-and-slab prior to regularize the inference and promote sparsity. We investigate the empirical behavior of the proposed method on both continuous and truncated data, and we demonstrate that it outperforms several state-of-the-art methods. As an application, we use the proposed methodology to maximally correlate clinical variables and proteomic data for better understanding the Covid-19 disease.

The third chapter is on the Bayesian perspective of the minimum distance estimation (MDE). MDE is a classical statistical method for fitting a parametric model to data, by seeking to minimize the distance between the distribution of statistical model and the (empirical) distribution of data. We focus on the probability distance (e.g.,

the integral probability metric) that allows re-formulating the MDE problem into a minimax form. By placing priors on the problem parameters, we propose a Bayesian formulation of this minimax problem, and show the existence and robustness of its solution. In addition, we propose a scalable algorithm to approximate the solution. We also investigate the performance of the algorithm for data that are generated from distributions with no analytic form.

In the fourth chapter, we consider inverse problems where the conditional distribution of the observation $\mathbf{y}$ given the latent variable of interest $\mathbf{x}$ (also known as the forward model) is known, and we have access to a data set in which multiple instances of $\mathbf{x}$ and $\mathbf{y}$ are both observed. In this context, algorithm unrolling has become a very popular approach for designing state-of-the-art deep neural network architectures that effectively exploits the forward model. We analyze the statistical complexity of gradient descent network (GDN), an algorithm unrolling architecture driven by proximal gradient descent. We show that the unrolling depth needed for the optimal statistical performance of GDN is of order $\log(n)/\log(\rho^{-1})$, where n is the sample size, and ρ is the convergence rate of the corresponding gradient descent algorithm. We also show that when the negative log-density of the latent variable $\mathbf{x}$ has a simple proximal operator, then GDN can solve the inverse problem at the parametric rate $1/\sqrt{n}$.

Chapter 2

Minimax quasi-Bayesian estimation in sparse canonical correlation analysis via a Rayleigh quotient function

2.1 Introduction

Canonical correlation analysis (CCA) is a statistical technique –dating back at least to Hotelling (1936) – that is used to maximally correlate multiple datasets for joint analysis. The technique has become a fundamental tool in biomedical research where technological advances have made it possible to observe fundamental biological phenomena from multiple viewpoints — the so-called multi-omic datasets (Witten and Tibshirani (2009); Mo et al. (2017); Rappoport and Shamir (2018)). Over the past two decades, limited sample size and growing dimensionality in these datasets, and the search for meaningful biological interpretations, have led to the development of sparse CCA (Wiesel et al. (2008); Witten and Tibshirani (2009); Parkhomenko et al. (2009); Waaijenborg and Zwinderman (2009); Hardoon and Shawe-Taylor (2011)), where a sparsity assumption is imposed on the canonical vectors.

Statistically optimal estimation of sparse CCA has been recently considered in the literature. (Gao et al. (2015)) derived the minimax rate of estimation of sparse CCA, and proposed a two-stage estimation procedure that achieves the rate. (Tan et al. (2018)) uses a generalized Rayleigh quotient approach to propose a two-stage estimator that also achieves the minimax rate. These two rate-optimal estimation

procedures share the same limitation, that is, high computational cost. Specifically, in both approaches, each iteration of the first-stage optimization problem has a computational cost of $O(p^3)$, where p is the joint number of variables in the datasets. Furthermore, the two-stage nature of these estimators can also be a problem in practice, since it can be hard to set the required stopping criterion of the first-stage solver that guarantees a good behavior of the final estimator.

We address these issues by proposing a conceptually simple, yet rate-optimal quasi-Bayesian estimator for sparse CCA. More specifically, building on (Tan et al. (2018)), we propose a quasi-Bayesian approach that employs a re-scaled version of the Rayleigh quotient function as the quasi-log-likelihood together with a spike and slab prior to obtain a quasi-posterior distribution. The method is agnostic to the covariance matrix estimators used in constructing the Rayleigh quotient function. For example, we observe in our experiments that both the sample covariance matrix estimator and the Kendall's-tau-based covariance matrix estimator (Yoon et al. (2018)) can be used to construct the Rayleigh quotient function, and these matrices are allowed to be singular. Although we do not pursue this here, one can straightforwardly extend our method to solve other generalized eigenvalue problems in the same spirit as (Tan et al. (2018)). In fact, at a high level, our method can be viewed as an improved version of simulated annealing (Kirkpatrick et al. (1983); Bertsimas and Tsitsiklis (1993)) for minimizing the Rayleigh quotient under a sparsity constraint. As such, it can be easily extended to tackle other similarly challenging non-convex statistical optimization problems with sparsity constraints.

We analyze the proposed estimator and derive its convergence rate (see Theorem 1). In the particular case where sample covariance matrices are used to estimate the Rayleigh quotient, we show that the estimator achieves the minimax rate for sparse CCA estimation, under some modest sample size conditions.

We propose a Markov Chain Monte Carlo algorithm based on simulated tempering to sample from the quasi-posterior distribution, and compute the estimator. At stationarity, the proposed algorithm has a per-iteration cost of $O(\bar{s}^2 p)$, where $\bar{s}$ is the underlying sparsity level of the posterior distribution. In all our numerical experiments, we have observed that $\bar{s}$ is of the same order as $s_\star$, namely the true sparsity level of the principal canonical vectors, leading to a very small percentage of false-positives. Furthermore, we show empirically that for sufficiently large sample size, the mixing time of the algorithm scales linearly in p. As a result, our estimator has a much lower computational cost than the Rifle estimator in (Tan et al. (2018)). We also compare our method with the popular mixedCCA estimator in (Yoon et al. (2018)). The results show that although our method is computationally slower than mixedCCA, it produces statistically better estimates. We note that the estimation rate of mixedCCA is currently unknown.

The chapter is organized as follows. In Section 2.2 we introduce our estimation procedure and derive its convergence rate. In Section 2.3 we detail a simulated tempering algorithm to sample from the resulting quasi-posterior distribution. In Section 2.4, we study the behavior of the proposed method on both continuous and truncated data, and compare it with other methods. In Section 2.5, we apply the method to a case study, where one aims to correlate clinical and proteomic data from Covid-19 patients, for a better understanding of the disease. Our analysis identifies that Alpha-1-acid glycoprotein 1 (AGP 1) plays an important role in the progression of Covid-19 into a severe illness.

2.2 Quasi-Bayesian sparse CCA using a Rayleigh quotient function

Let $(X, Y) \in \mathbb{R}^{p_x} \times \mathbb{R}^{p_y}$ be a pair of high-dimensional zero-mean random vectors with joint distribution f and covariance matrices $\Sigma_x \stackrel{\text{def}}{=} \mathbb{E}(XX^{\mathrm{T}})$, $\Sigma_y \stackrel{\text{def}}{=} \mathbb{E}(YY^{\mathrm{T}})$ and $\Sigma_{xy} \stackrel{\text{def}}{=} \mathbb{E}(XY^{\mathrm{T}})$. Let $(v_{x\star}, v_{y\star}) \in \mathbb{R}^{p_x} \times \mathbb{R}^{p_y}$ be a pair of principal canonical vectors of f, that is, the vector pair that solves the following optimization problem:

$$\max_{v_x \in \mathbb{R}^{p_x},\, v_y \in \mathbb{R}^{p_y}} \; v_x^T \Sigma_{xy} v_y \quad \text{s.t.} \quad v_x^{\mathrm{T}} \Sigma_x v_x = v_y^{\mathrm{T}} \Sigma_y v_y = 1. \tag{2.1}$$

Since we are only interested in the directions of $v_{x\star}^{\mathrm{T}}$ and $v_{y\star}^{\mathrm{T}}$, we set $\theta_\star \stackrel{\text{def}}{=} \frac{(v_{x\star}^{\mathrm{T}}, v_{y\star}^{\mathrm{T}})^{\mathrm{T}}}{\|(v_{x\star}^{\mathrm{T}}, v_{y\star}^{\mathrm{T}})^{\mathrm{T}}\|_2}$ (so that $\|\theta_\star\|_2 = 1$) to be our main parameter of interest. The parameter $\theta_\star$ is identifiable only up to a change of sign, and hence, we shall focus on the estimation of the related projector $\theta_\star \theta_\star^{\mathrm{T}}$. Let us define $p \stackrel{\text{def}}{=} p_x + p_y$, and the matrices

$$A \stackrel{\text{def}}{=} \begin{bmatrix} 0 & \Sigma_{xy} \\ \Sigma_{xy}^{\mathrm{T}} & 0 \end{bmatrix}, \qquad B \stackrel{\text{def}}{=} \begin{bmatrix} \Sigma_x & 0 \\ 0 & \Sigma_y \end{bmatrix} \quad \text{and} \quad \Sigma \stackrel{\text{def}}{=} A + B = \begin{bmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{xy}^{\mathrm{T}} & \Sigma_y \end{bmatrix}. \tag{2.2}$$

Using simple arguments, we notice that the problem in (2.1) is equivalent to the following generalized eigenvalue problem (GEP):

$$\max_{\theta = (v_x^{\mathrm{T}}, v_y^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^p} \; \theta^{\mathrm{T}} A \theta \quad \text{s.t.} \quad \theta^{\mathrm{T}} B \theta = 2. \tag{2.3}$$

Clearly, finding a solution of (2.3) is equivalent to finding a solution of

$$\max_{\theta = (v_x^{\mathrm{T}}, v_y^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^p} \mathsf{R}(\theta) \stackrel{\text{def}}{=} \frac{\theta^{\mathrm{T}} A \theta}{\theta^{\mathrm{T}} B \theta}, \tag{2.4}$$

where we convene that $0/0 = 0$. The objective function $\mathsf{R}(\cdot)$ in (2.4) is known as the *(generalized) Rayleigh quotient* of A and B. The reformulation in (2.4) suggests a way to estimate the sparse canonical vectors by directly targeting the Rayleigh quotient, and this idea was first proposed in (Tan et al. (2018)). Note that solving (2.4) requires

specifying matrices A and B, which are typically unknown in practice. Instead, given n i.i.d. samples $\mathbf{Z} \overset{\text{def}}{=} \{(X_i, Y_i)\}_{i=1}^n$ from f, one first constructs estimators of Σ_x, Σ_y and Σ_{xy}, denoted by $\hat{\Sigma}_x$, $\hat{\Sigma}_y$ and $\hat{\Sigma}_{xy}$, respectively, and then construct estimators of A and B (denoted by $\hat{A}$ and $\hat{B}$, respectively) from $\hat{\Sigma}_x$, $\hat{\Sigma}_y$, and $\hat{\Sigma}_{xy}$ in the same way as in (2.2). (In Section 2.4, we will provide some examples of constructing $\hat{\Sigma}_x$, $\hat{\Sigma}_y$, and $\hat{\Sigma}_{xy}$.) Based on $\hat{A}$ and $\hat{B}$, one then solves (2.4) with the Rayleigh quotient $\mathsf{R}(\cdot)$ replaced by its sample version $\mathsf{R}_n(\cdot; \mathbf{Z})$, which is defined as

$$\mathsf{R}_n(\theta; \mathbf{Z}) \overset{\text{def}}{=} \frac{\theta^{\mathrm{T}} \hat{A} \theta}{\theta^{\mathrm{T}} \hat{B} \theta}, \quad \forall\, \theta \in \mathbb{R}^p.$$

To guarantee that the Rayleigh quotient $\mathsf{R}_n(\cdot; \mathbf{Z})$ is well-defined, we maintain the following assumption throughout this work.

H 1. *For all $\theta \in \mathbb{R}^p$, $|\theta^{\mathrm{T}} \hat{A} \theta| \leq \theta^{\mathrm{T}} \hat{B} \theta$.*

H1 implies that $\theta^{\mathrm{T}} \hat{A} \theta = 0$ whenever $\theta^{\mathrm{T}} \hat{B} \theta = 0$, in which case we have $\mathsf{R}_n(\theta; \mathbf{Z}) = 0/0 = 0$. We note that H1 naturally holds when $\hat{\Sigma}_x$, $\hat{\Sigma}_y$, and $\hat{\Sigma}_{xy}$ are sample covariance matrices.

It is worth mentioning that in high-dimensional regimes where $p > n$, the constructed estimators $\hat{\Sigma}_x$ and $\hat{\Sigma}_y$ (e.g., sample covariance matrices) are usually singular, thereby making a direct maximization of R_n challenging. Similarly, other classical CCA algorithms based on eigen-decomposition of $\hat{B}^{-1}\hat{A}$, or the singular value decomposition of $\hat{\Sigma}_x^{-1/2} \hat{\Sigma}_{x,y} \hat{\Sigma}_y^{-1/2}$ (see e.g., (Mardia et al. (1979); Andrew et al. (2013))) are also difficult to use under these regimes. Furthermore, these classical methods do not yield sparse estimates of the canonical correlation vectors.

(Tan et al. (2018)) addressed these issues by maximizing $\mathsf{R}_n(\cdot; \mathbf{Z})$ under a sparsity constraint. The authors show that this maximization problem can be solved provided that a good initial value that is sufficiently close to global maxima is provided. However, finding such a good initial value is very costly. Furthermore, the Rayleigh

quotient typically admits several local maxima (as well as local minima and saddle points) that correspond to other canonical vectors, making direct maximization of R_n very challenging.

2.2.1 A Quasi-Bayesian approach

We propose a quasi-Bayesian framework that turns maximizing the Rayleigh quotient into a Bayesian procedure. More precisely, we propose using

$$\theta \mapsto \sigma_n \mathsf{R}_n(\theta; \mathbf{Z}) \tag{2.5}$$

as the quasi-log-likelihood, where $\sigma_n > 0$ is a scaling parameter. We combine this quasi-log-likelihood with a spike-and-slab prior distribution, which is a common choice for Bayesian sparse modeling (George and McCulloch (1997)). Specifically, given a variable selection parameter $\delta \in \Delta \stackrel{\text{def}}{=} \{0,1\}^p$, we let the conditional distribution of θ given δ be

$$\pi(\theta|\delta) = \prod_{j=1}^{p} \pi(\theta_j|\delta), \quad \text{where} \quad \theta_j|\delta \;=\; \theta_j|\delta_j \sim \begin{cases} \mathbf{N}(0, \rho_1^{-1}), & \text{if } \delta_j = 1 \\ \mathbf{N}(0, \rho_0^{-1}), & \text{if } \delta_j = 0 \end{cases}, \tag{2.6}$$

where $\rho_0 > \rho_1 > 0$ are precision parameters. Given some parameter $\mathsf{u} > 1$ and integer $s \geq 1$, the prior distribution of δ is taken as the independent product of Bernoulli distribution $\mathsf{Ber}(1/(1+p^{\mathsf{u}}))$ conditioned to stay in the set $\Delta_s \stackrel{\text{def}}{=} \{\delta \in \Delta : \; \|\delta\|_0 \leq s\}$. More specifically,

$$\pi(\delta) \propto \mathbf{1}_{\Delta_s}(\delta) \prod_{j=1}^{p} \left(\frac{1}{1+p^{\mathsf{u}}}\right)^{\delta_j} \left(\frac{p^{\mathsf{u}}}{1+p^{\mathsf{u}}}\right)^{1-\delta_j} \;\propto\; \mathbf{1}_{\Delta_s}(\delta) \left(\frac{1}{p^{\mathsf{u}}}\right)^{\|\delta\|_0}, \qquad \forall\, \delta \in \Delta. \tag{2.7}$$

If we combine the spike-and-slab prior with the quasi-log-likelihood in (2.5), we then obtain the quasi-posterior distribution

$$\Pi(\delta, \mathrm{d}\theta|\mathbf{Z}) \propto \mathbf{1}_{\Delta_s}(\delta) \left(\frac{1}{p^{\mathsf{u}}}\sqrt{\frac{\rho_1}{\rho_0}}\right)^{\|\delta\|_0} \exp\left(-\frac{\rho_1}{2}\|\theta_\delta\|_2^2 - \frac{\rho_0}{2}\|\theta - \theta_\delta\|_2^2 + \sigma_n \mathsf{R}_n(\theta_\delta; \mathbf{Z})\right) \mathrm{d}\theta, \tag{2.8}$$

where θ_δ is the component-wise product of θ and δ, $\|\cdot\|_2$ is the Euclidean norm. Note that in this posterior distribution, the parameter θ is typically dense. However, since δ is sparse, so is θ_δ. We note that the Rayleigh quotient R_n can take value $+\infty$ when its numerator is non-zero while its denominator is zero. If this happens over a set with non-zero Lebesgue measure, then (2.8) is not well-defined. However, if $(\hat{A}, \hat{B})$ satisfies H1 then $\mathsf{R}_n(\delta, \theta) < \infty$ for all δ, θ, in which case (2.8) is a well-defined probability measure.

The spike-and-slab prior shown in (2.6) and (2.7) is fairly standard, and goes back at least to (George and McCulloch (1997)). However the way it is combined with the pseudo-likelihood to yield (2.8) is nonstandard, and follows from (Atchadé and Bhattacharyya (2019)). The key feature of this approach is that the parameter θ enters the quasi-likelihood only through its sparsified form θ_δ (see (2.8)). This decouples the active components (namely those corresponding to $\delta_j = 1$) and the non-active components (namely those corresponding to $\delta_j = 0$), and is particularly attractive from the computational standpoint. The approach should be viewed as an approximation of the point-mass spike-and-slab prior (Mitchell and Beauchamp (1988)), using the pseudo-prior device in (Carlin and Chib (1995)).

Hyper-parameter tuning

The posterior distribution Π is very robust to the choice of ρ_1 and u, and we recommend choosing $\rho_1 \approx 1$ and $\mathsf{u} \in (1, 2]$ (as suggested by Atchadé and Bhattacharyya (2019)) for best performance. The parameter ρ_0 has no effect on the statistical re-

covery of the selected components of θ, but can adversely impact the MCMC mixing if its value is too large. We suggest setting $\rho_0 \sim n$, in order to match the posterior variance of the selected components that are actually zero (false-positives), and the posterior variance of the true-negatives.

The sparsity level s is an upper-bound on the true sparsity of the signal, which is typically unknown. We observe that if $\delta_1, \ldots, \delta_p \overset{i.i.d.}{\sim} \mathsf{Ber}(1/(1+p^{\mathsf{u}}))$, then by Chernoff's inequality (see e.g., (Vershynin, 2018, Theorem 2.3.1)), for any $s_0 \geq \exp(1)$, we have $\mathbb{P}(\|\delta\|_0 > s_0) \leq (1/p)^{(\mathsf{u}-1)s_0}$. This suggests simply choosing $s = p$ in (2.6), and the resulting prior distribution would still automatically concentrate on sets Δ_{s_0}, for s_0 small. We made this choice in all our numerical implementations. We found that the resulting posterior distribution is always automatically sparse, and learns the true sparsity of the signal. However, for the theoretical analysis of the method we will assume that a sparsity level s is given such that $n \geq c_0 s \log(p)$, for some absolute constant c_0[1]. We discuss the choice of σ_n below after the statement of Theorem 1.

2.2.2 Connection with simulated annealing

Our methodology can be viewed as a principled version of simulated annealing algorithm (Kirkpatrick et al. (1983); Bertsimas and Tsitsiklis (1993)) for computing the Rifle estimator of (Tan et al. (2018)). Given $s \geq 1$, let $\Theta_s \overset{\text{def}}{=} \{\theta \in \mathbb{R}^p : \ \|\theta\|_0 \leq s\}$. Let $\sigma_t > 0$ be given such that $\lim_{t\to\infty} \sigma_t = +\infty$, and define

$$\Pi_t(\mathrm{d}\theta) \propto e^{\sigma_t \mathsf{R}_n(\theta;\mathbf{Z})} \mathbf{1}_{\Theta_s}(\theta)\mathrm{d}\theta, \tag{2.9}$$

where $\mathrm{d}\theta$ denotes the extension of the Lebesgue measure to the set Θ_s. The maximization problem tackled by the authors of Rifle in (Tan et al. (2018)) is $\max_{\theta\in\Theta_s} \mathsf{R}_n(\theta;\mathbf{Z})$. A simulated annealing solution to this problem consists in simulating a non-homogeneous

[1]For the absolute constant in this thesis, we are referring to a constant that doesn't depend on parameters. For example, here we are saying that c_0 doesn't depend on parameters p, n, s.

Markov chain with sequence of transition kernels $\{\mathcal{M}_k,\ k \geq 1\}$, such that $\mathcal{M}_k$ has invariant distribution Π_{t_k}. As $\sigma_{t_k} \to \infty$, the distribution Π_{t_k} puts most of its probability mass around the global modes of R_n, and the resulting Markov chain behaves similarly (under appropriate conditions). There are several limitations to simulated annealing in this particular setting. First, the set Θ_s is a union of a large number of subsets with varying dimensions. Therefore, sampling from Π_{t_k} is actually non-trivial – even using MCMC. Second, the convergence of simulated annealing is known to be highly dependent on the choice of the sequence $\{\sigma_{t_k},\ k \geq 1\}$. Our approach circumvents the first issue by working with a relaxation of Θ_s, using the spike-and-slab prior. We circumvent the second issue by tuning $\sigma_{t_k} = \sigma_n$ (see details below), in such a way that the fluctuations in the resulting distribution Π_{t_k} matches the statistical uncertainty of the underlying CCA problem.

2.2.3 Rate of convergence

Although the Rayleigh quotient $\mathsf{R}_n(\cdot; \mathbf{Z})$ may possess multiple local modes, we show in this section that most of the probability mass of the quasi-posterior distribution $\Pi(\cdot|\mathbf{Z})$ are located around $\{\pm\theta_\star\}$. For $M, N \in \mathbb{R}^{q\times q}$, we define

$$\langle M, N\rangle_{\mathsf{F}} \stackrel{\text{def}}{=} \mathsf{Tr}(M^{\mathrm{T}}N), \quad \|M\|_{\mathsf{F}} \stackrel{\text{def}}{=} \sqrt{\langle M, M\rangle_{\mathsf{F}}}, \quad \text{and} \quad \|M\|_{\mathsf{op}} \stackrel{\text{def}}{=} \sup_{u\in\mathbb{R}^q:\ \|u\|_2=1} \|Mu\|_2.$$

For $J \subseteq [1:q] \stackrel{\text{def}}{=} \{1, \ldots, q\}$, let $M_{J,J}$ denote the submatrix $(M_{ij})_{i,j\in J}$. Given $k \geq 1$, we let

$$\lambda_{\mathsf{min}}(M, k) \stackrel{\text{def}}{=} \min_{u\in\mathbb{R}^q:\ \|u\|_2=1, \|u\|_0\leq k} u^{\mathrm{T}}Mu, \quad \text{and} \quad \lambda_{\mathsf{max}}(M, k) \stackrel{\text{def}}{=} \max_{u\in\mathbb{R}^q:\ \|u\|_2=1, \|u\|_0\leq k} u^{\mathrm{T}}Mu.$$

Given an integer $\alpha \geq 1$, we set

$$\lambda^{(\alpha)}_{\mathsf{max}}(M, s) \stackrel{\text{def}}{=} \max_{J\subseteq[1:q]:\ \|J\|_0=s} \ \max_{\substack{A\in\mathbb{R}^{s\times s}:\ \|A\|_{\mathsf{F}}=1 \\ \mathsf{Rank}(A)\leq\alpha}} |\langle M_{J,J}, A\rangle|\,.$$

We note that $\lambda^{(1)}_{\mathsf{max}}(M,k) = \lambda_{\mathsf{max}}(M,k)$. We first make the following basic assumption without which the sparse CCA problem would not be well defined.

H 2. *$(X^{\mathrm{T}}, Y^{\mathrm{T}}) \sim f$ with positive definite covariance matrices Σ_x, Σ_y, and Σ, and a principal canonical vector pair $\theta_\star = (v_{x\star}^{\mathrm{T}}, v_{y\star}^{\mathrm{T}})^{\mathrm{T}}$, ($\|\theta_\star\|_2 = 1$) with density level[2] $s_\star \stackrel{\mathrm{def}}{=} \|\theta_\star\|_0$. Furthermore, the difference between the largest and second largest eigenvalue of $S \stackrel{\mathrm{def}}{=} B^{-1/2}\Sigma B^{-1/2}$ (denoted by gap), is positive.*

Our main assumption on the data generation process is the following.

H3. $\mathbf{Z} \stackrel{\mathrm{def}}{=} \{(X_1^{\mathrm{T}}, Y_1^{\mathrm{T}})^{\mathrm{T}}, \ldots, (X_n^{\mathrm{T}}, Y_n^{\mathrm{T}})^{\mathrm{T}}\}$ *is a sequence of n i.i.d. random variables from f, and we choose the integer $s \geq s_\star$ such that:*

1. *for some absolute constants $0 < \underline{\kappa} \leq \bar{\kappa}$,*

$$\min\left(\lambda_{\mathit{min}}(\hat{\Sigma}_x, s+s_\star),\ \lambda_{\mathit{min}}(\hat{\Sigma}_y, s+s_\star),\ \lambda_{\mathit{min}}(\hat{\Sigma}, s+s_\star)\right) \geq \underline{\kappa},$$

$$\max\left(\lambda_{\mathit{max}}(\hat{\Sigma}_x, s+s_\star),\ \lambda_{\mathit{max}}(\hat{\Sigma}_y, s+s_\star), \lambda_{\mathit{max}}(\hat{\Sigma}, s+s_\star)\right) \leq \bar{\kappa}.$$

2. *for some constant r_1 (depending possibly on n, p),*

$$\max\left(\lambda^{(2)}_{\mathit{max}}(\hat{\Sigma}_x - \Sigma_x, s+s_\star),\ \lambda^{(2)}_{\mathit{max}}(\hat{\Sigma}_y - \Sigma_y, s+s_\star),\ \lambda^{(2)}_{\mathit{max}}(\hat{\Sigma} - \Sigma, s+s_\star)\right) \leq \mathsf{r}_1.$$

Theorem 1. *Assume H1-H3, and choose $\mathsf{u} > 1$ such that $p^{\mathsf{u}-1} > 2$, and $p^{\mathsf{u}} \geq s_\star \exp(1)$. Set*

$$\epsilon \stackrel{\mathrm{def}}{=} \frac{\mathsf{r}_1}{\mathsf{gap}}. \tag{2.10}$$

There exists some absolute constant C_0 that depends only on $\underline{\kappa}$ and $\bar{\kappa}$, such that the following holds. For all $M > C_0$ such that

$$\frac{M^2}{8\mathit{gap}}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \sigma_n \mathsf{r}_1^2 \geq (\mathsf{s}_\star + 1)(\mathsf{u}+1)\log\left(\mathsf{p} \vee (\mathsf{c}_0\sigma_{\mathsf{n}})\right), \tag{2.11}$$

where c_0 is some absolute constant that depends only on $\underline{\kappa}$ and $\bar{\kappa}$, we have

$$\Pi\left((\delta,\theta):\ \left\|\frac{\theta_\delta\theta_\delta^{\mathrm{T}}}{\|\theta_\delta\|_2^2} - \theta_\star\theta_\star^{\mathrm{T}}\right\|_{\mathsf{F}} > M\epsilon|\mathbf{Z}\right) \leq 2e^{-\frac{M^2}{8\mathit{gap}}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2\sigma_n\mathsf{r}_1^2}.$$

[2]Throughout this work, the density level of a vector refers to the proportion of its non-zero elements.

Proof. See Section A.1 in supplementary material. □

Since the term $2e^{-\frac{M^2}{8\mathsf{gap}}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2\sigma_n\mathsf{r}_1^2}$ is much smaller than $p^{-1-s_\star}$ (under (2.11)), the main conclusion of the theorem is that the posterior $\Pi(\cdot|\mathbf{Z})$ contracts around $\theta_\star\theta_\star^{\mathrm{T}}$ at the rate at least $M\epsilon$. Furthermore, setting

$$\widehat{\mathcal{P}} \stackrel{\text{def}}{=} \int_{\Delta_s\times\mathbb{R}^p} \frac{\theta_\delta\theta_\delta^{\mathrm{T}}}{\|\theta_\delta\|_2^2}\Pi(\mathrm{d}\delta,\mathrm{d}\theta|\mathbf{Z}),$$

the result implies that $\widehat{\mathcal{P}}$ (as a frequentist estimator) estimates $\theta_\star\theta_\star^{\mathrm{T}}$ at the rate $M\epsilon$. Indeed, we have

$$\left\|\widehat{\mathcal{P}} - \theta_\star\theta_\star^{\mathrm{T}}\right\|_{\mathsf{F}} \leq \int_{\Delta_s\times\mathbb{R}^p} \left\|\frac{\theta_\delta\theta_\delta^{\mathrm{T}}}{\|\theta_\delta\|_2^2} - \theta_\star\theta_\star^{\mathrm{T}}\right\|_{\mathsf{F}} \Pi\left(\mathrm{d}\delta,\mathrm{d}\theta|\mathbf{Z}\right) \leq M\epsilon + 2e^{-\frac{M^2}{8\mathsf{gap}}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 n\mathsf{r}_1^2}. \quad (2.12)$$

We note that Theorem 1 applies only to a given dataset $\mathbf{Z}$ that satisfies H3. In the particular case where $\hat{\Sigma}_x, \hat{\Sigma}_y$ and $\hat{\Sigma}$ are covariances matrices, we show in Proposition 2 below that if $\mathbf{Z} \stackrel{\text{def}}{=} \{(X_1^{\mathrm{T}}, Y_1^{\mathrm{T}})^{\mathrm{T}}, \ldots, (X_n^{\mathrm{T}}, Y_n^{\mathrm{T}})^{\mathrm{T}}\}$ is a sequence of n i.i.d. random variables from a sub-Gaussian distribution f, then H3 holds with high probability. Furthermore $\mathsf{r}_1 = \mathsf{C}_0\sqrt{(\mathsf{s}+\mathsf{s}_\star)\log(\mathsf{p})/\mathsf{n}}$. In that case the condition in (2.11) becomes

$$\frac{M^2}{8\mathsf{gap}}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 C_0^2\left(\frac{\sigma_n}{n}\right)(s+s_\star)\log(p) \geq (s_\star+1)(\mathsf{u}+1)\log\left(p\vee(c_0\sigma_n)\right),$$

which is easily satisfied when the scaling parameter σ_n satisfies $n = O(\sigma_n)$, as $n\to\infty$. In this case the convergence rate of $\widehat{\mathcal{P}}$ towards $\theta_\star\theta_\star^{\mathrm{T}}$ is

$$\epsilon = \frac{1}{\mathsf{gap}}\sqrt{\frac{(s+s_\star)\log(p)}{n}}, \quad (2.13)$$

which achieves the minimax rate of the CCA problem, as derived in (Gao et al. (2015)), by taking s as some constant multiple of $s_\star$. Further increasing σ_n has no impact on this rate, but of course, makes the MCMC computation more challenging. This suggests the choice $\sigma_n \propto n$.

Remark 2.2.1. *The discussion so far has focused on estimating the projector $\theta_\star\theta_\star^{\mathrm{T}}$. If the vector $\theta_\star$ itself is needed, we are able to construct an estimator of $\theta_\star$ from the projector estimator $\widehat{\mathcal{P}}$. Specifically, let $v_1(\widehat{\mathcal{P}})$ denote the leading eigenvector of $\widehat{\mathcal{P}}$, then from the Davis-Kahan theorem (see e.g., (Vershynin, 2018, Theorem 4.5.5)), we have*

$$\min\Big(\|v_1(\widehat{\mathcal{P}})-\theta_\star\|_2,\ \|v_1(\widehat{\mathcal{P}})+\theta_\star\|_2\Big)\le 2^{3/2}\|\widehat{\mathcal{P}}-\theta_\star\theta_\star^{\mathrm{T}}\|\le 2^{3/2}\|\widehat{\mathcal{P}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}},\quad (2.14)$$

and $\|\widehat{\mathcal{P}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}$ can be bounded as in (2.12)

On Assumption H3

It is well-known that Assumption H3-(1) holds true in the particular case of covariance matrices of sub-Gaussian random vectors, provided that the sample size satisfies $n\ge c_0(s+s_\star)\log(p)$, for some absolute constant c_0. See for instance Raskutti et al. (2010) Theorem 1, or Gao et al. (2017) Lemma 6.5 for the Gaussian case, and Rudelson and Zhou (2013) Theorem 3.2 for more general sub-Gaussian distributions. Under roughly the same sample size conditions, H3-(2) is also known to hold as we show below.

Proposition 2. *Suppose that $Z_i\overset{\mathrm{def}}{=}(X_i^{\mathrm{T}},Y_i^{\mathrm{T}})^{\mathrm{T}}$ are i.i.d. random vectors from a mean-zero sub-Gaussian distribution f, with sub-Gaussian norm $K\overset{\mathrm{def}}{=}\sup\{\|\langle Z,u\rangle\|_{\psi_2},\ u\in\mathbb{R}^p,\ \|u\|_2=1\}$, where $\|\cdot\|_{\psi_2}$ refers to the sub-Gaussian norm of a random variable. Let $\hat\Sigma_x=n^{-1}\sum_{i=1}^n X_iX_i^{\mathrm{T}}$, $\hat\Sigma_y=n^{-1}\sum_{i=1}^n Y_iY_i^{\mathrm{T}}$, and $\hat\Sigma=n^{-1}\sum_{i=1}^n Z_iZ_i^{\mathrm{T}}$. There exist absolute constants $c_0,C>1$, such that for all $1\le s\le p$, and all $n\ge 4c_0s\log(p)$,*

$$\max\Big(\lambda_{\mathsf{max}}^{(\alpha)}(\hat\Sigma_x-\Sigma_x,s),\ \lambda_{\mathsf{max}}^{(\alpha)}(\hat\Sigma_y-\Sigma_y,s),\ \lambda_{\mathsf{max}}^{(\alpha)}(\hat\Sigma-\Sigma,s)\Big)$$
$$\le CK^2\lambda_{\mathsf{max}}(\Sigma,s)\sqrt{\frac{c_0\alpha s\log(p)}{n}},$$

with probability $1-2p^{-(c_0-1)s}$.

Proof. See Section A.2 in supplementary material. □

2.3 Computation using Markov Chain Monte Carlo

As shown in Section 2.2.3, by re-scaling (annealing) the Rayleigh quotient function, we have created a posterior distribution $\Pi(\cdot|\mathbf{Z})$ that puts most of its probability mass around its global mode (located near $\{\pm\theta_\star\}$). However, the annealing also significant decreases the accessibility of the global starting from the local modes. To effectively deal with this configuration, we propose a Markov Chain Monte Carlo sampling strategy based on simulated tempering (Geyer and Thompson (1995); Liu (2008)). Given K temperatures $1 = t_1 < t_2 < \ldots < t_K$, and K positive weights $c_1, \ldots, c_K$, we introduce an extended distribution on $\mathsf{X} \stackrel{\text{def}}{=} \Delta \times \mathbb{R}^p \times \{1, \ldots, K\}$, which is

$$\bar{\Pi}(\delta, \mathrm{d}\theta, k|\mathbf{Z}) \propto \frac{1}{c_k} \exp\left(\frac{\mathsf{a}}{t_k}\|\delta\|_0 - \frac{\rho_1}{2t_k}\|\theta_\delta\|_2^2 - \frac{\rho_0}{2t_k}\|\theta - \theta_\delta\|_2^2 + \frac{\sigma_n}{t_k}\mathsf{R}_n(\theta_\delta; \mathbf{Z})\right) \mathrm{d}\theta. \tag{2.15}$$

We recover the distribution (2.8) as the conditional distribution of (δ, θ) given $k = 1$ in (2.15). To sample from (2.15), we use a simulated tempering Metropolis-Hastings-within-Gibbs strategy that is described in the supplementary material (Section A.3). The algorithm is very fast and scales well with the dimension p, and iteration k of the algorithm has computational cost $O(p\|\delta^{(k)}\|_0^2)$.

Algorithm 2 generates a Markov chain $\{X^{(t)},\ t \geq 0\}$, where $X^{(t)} = (\delta^{(t)}, \theta^{(t)}, k^{(t)}) \in \mathsf{X}$ that is phi-irreducible aperiodic with invariant distribution given by (2.15). The pairs $(\delta^{(t)}, \theta^{(t)})$ at times t where $\{k^{(t)} = 1\}$ then give the desired approximate samples from $\Pi(\cdot|\mathbf{Z})$ We investigate the mixing of the algorithm below.

2.3.1 Mixing times

At least empirically, simulated tempering is well-known to improve mixing when dealing with multimodal distributions (Geyer and Thompson (1995); Liu (2008)). However, rigorous results are far less well-established. Using a Markov kernel decom-

position approach, (Woodard et al. (2009)) gives a lower bound on the spectral gap of simulated tempering in terms of the spectral gaps of the component kernels and the so-called projection kernel. However, applying their result to a specific problem remains non-trivial. Furthermore, their lower bound decays exponentially fast in the number of components in the partition, which clearly limits its relevance in our setting. Using a similar Markov kernel decomposition technique, (Ge et al. (2018)) has a more explicit upper bound on the mixing time of simulated tempering. However their result applies to a different algorithm than the one considered here, and they consider a specific form of the target distribution that does not include (2.15).

Given the lack of theoretical mixing time analysis of simulated tempering, we take a more empirical approach based on the unbiased Markov Chain Monte Carlo framework in (Biswas et al. (2019); Jacob et al. (2017)). Let $\{X^{(t)},\ t \geq 0\}$ be the Markov chain generated by our simulated tempering algorithm, where $X^{(t)} = (\delta^{(t)}, \theta^{(t)}, k^{(t)}) \in \mathsf{X}$. Let P denote its transition kernel (which is described in Section A.4 in the supplementary material). Following Jacob et al. (2017), we construct a coupling $\check{P}$ of P with itself: that is, a transition kernel on $\mathsf{X} \times \mathsf{X}$ such that $\check{P}((x,y), A \times \mathsf{X}) = P(x, A)$, $\check{P}((x,y), \mathsf{X} \times B) = P(y, B)$, for all $x, y \in \mathsf{X}$, and all measurable sets A, B. Furthermore, $\check{P}((x,x), \mathsf{X} \times \mathsf{X}) = 1$ for all $x \in \mathsf{X}$. The construction of the Markov kernel $\check{P}$ is described in Section A.4 in the supplementary material.

The crucial point of the method is that the coupling $\check{P}$ is not a theoretical construction: we actually implement $\check{P}$ to simulate a bivariate Markov chain $\{(X_t, Y_{t-L}),\ t \geq L\}$ for some lag $L \geq 1$, such that both $\{X_t,\ t \geq 0\}$ and $\{Y_t,\ t \geq 0\}$ are the same copy of our simulated tempering sampler. Specifically, given the lag $L \geq 1$, we first draw $X^{(0)} \sim \Pi^{(0)}$ and $Y^{(0)} \sim \Pi^{(0)}$, where $\Pi^{(0)}$ is the initial distribution as given in the initialization step of Algorithm 2. Next, for $1 \leq t \leq L$, we draw

$X_t|(X_0, Y_0, X_1, X_2, \ldots, X_{t-1}) \sim P(X_{t-1}, \cdot)$. Then for $t > L$, we draw

$$(X_t, Y_{t-L})|\left\{(X_{t-1}, Y_{t-L-1}), \ldots, (X_L, Y_0), X_{L-1}, \ldots, X_0\right\} \sim \check{P}\left((X_{t-1}, Y_{t-L-1}), \cdot\right).$$

In other words, at each time $t > L$ we attempt to couple the two chains while maintaining the correct marginals. We define $\tau^{(L)} \stackrel{\text{def}}{=} \inf\{t \geq L : X_t = Y_{t-L}\}$, and have the following:

Proposition 3. *Let $\{X^{(t)},\ t \geq 0\}$ be the Markov chain generated by the simulated tempering algorithm, and let $\bar{\Pi}^{(t)}$ denote the distribution of $X^{(t)}$. For all $t \geq 0$, we have*

$$\|\bar{\Pi}^{(t)} - \bar{\Pi}\|_{\text{tv}} \leq \mathbb{E}\left[\max\left(0, \left\lceil \frac{\tau^{(L)} - L - t}{L} \right\rceil\right)\right]. \tag{2.16}$$

Proof. See Section A.4.1. □

This inequality implies that by simulating multiple copies of the bivariate chain, and approximating the expectation in (2.16) by Monte Carlo, we can actually estimate the mixing time of our algorithm. This gives us the possibility to investigate empirically the mixing time of our sampler with some theoretical guarantees.

2.4 Numerical studies

We perform extensive numerical tests on our proposed approach. First we investigate the mixing time of the simulated tempering algorithm in connection to the CCA computational barrier conjectured by (Gao et al. (2017)). And we take a closer look at the mixing times as the dimension p increases. Then we perform a simulation study that compares our approach to the frequentist methods Rifle in Tan et al. (2018) and mixedCCA in Yoon et al. (2018). We investigate the behavior of these methods in two settings: (i) continuous datasets, where we use sample covariance matrix estimator and (ii) mixed datasets, where we use Kendall's-tau-based estimator

as proposed in Yoon et al. (2018). The Python codes for our method is available from `https://github.com/rachelwho/Sparse-CCA`.

2.4.1 Simulated data generation

We simulate the datasets using the following model from Tan et al. (2018). Specifically, we let $p_x = p_y = p/2$, and consider two $(p/2)$-dimensional random vectors X and Y with joint distribution $(X, Y) \sim \mathbf{N}(0, \Sigma)$. Here we let

$$\Sigma = \begin{pmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{xy}^{\mathrm{T}} & \Sigma_y \end{pmatrix} \quad \text{and} \quad \Sigma_{xy} = \frac{\lambda_1 \Sigma_x v_{x\star} v_{y\star}^{\mathrm{T}} \Sigma_y}{\sqrt{v_{x\star}^{\mathrm{T}} \Sigma_x v_{x\star}} \sqrt{v_{y\star}^{\mathrm{T}} \Sigma_y v_{y\star}}},$$

where $0 < \lambda_1 < 1$ is the largest generalized eigenvalue, and $v_{x\star}$ and $v_{y\star}$ are the principal canonical vectors. The structures of Σ_x and Σ_y vary across different experimental settings, and will be described in the subsequent sections. Clearly, $(v_{x\star}, v_{y\star})$ is the maximizer of the Rayleigh quotient in (2.4), and λ_1 is the maximum value. Then we generate n samples $\mathbf{Z} = \{(x^{(i)}, y^{(i)})\}_{i=1}^n$ from $\mathbf{N}(0, \Sigma)$.

2.4.2 Empirical studies of our algorithm

On sparse CCA computational barrier

It was conjectured by (Gao et al. (2017)) that it is not possible to solve the sparse CCA problem in polynomial time at the statistical rate ϵ obtained in (2.13), in the data regime $n = o(s_\star^2 \log(p))$. The authors made a compelling argument for this conjecture by showing that any such estimator for the sparse CCA can be used to solve the planted clique problem in a regime where it is widely believed to be computationally intractable. Since our estimator achieves the rate ϵ under the weaker condition $n \geq C_0 s_\star \log(p)$, we have the opportunity to test empirically this conjecture.

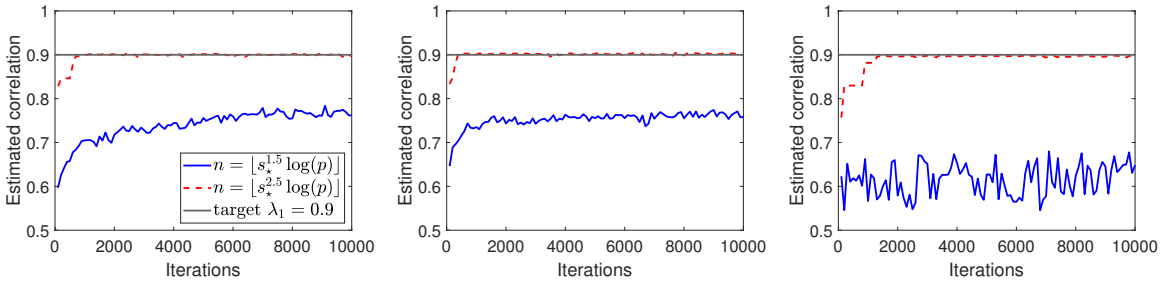
In our simulation, we let $\Sigma_x \in \mathbb{R}^{(p/2)\times(p/2)}$ and $\Sigma_y \in \mathbb{R}^{(p/2)\times(p/2)}$ share the same structure, namely, a block diagonal matrix with five blocks, each of dimension $p/10 \times p/10$, where the (j, j')-th element of each block takes value $0.8^{|j-j'|}$. We let $\lambda_1 =$

$0.9, (v_{x\star})_j = (v_{y\star})_j = 1/\sqrt{3}$ for $j \in \{1, 6, 11\}$, and $(v_{x\star})_j = (v_{y\star})_j = 0$ otherwise. Therefore, the true density level $s_\star = 6$. For each $p \in \{500, 2000, 5000\}$, we generate data from the model described in Section 2.4.1 with two values of the sample size n, namely $\lceil s_\star^{1.5} \log(p) \rceil$ and $\lceil s_\star^{2.5} \log(p) \rceil$. We use the sample covariance matrices as estimators of Σ_x, Σ_y and Σ_{xy}, and set the scaling parameter $\sigma_n = 2n$ to construct the extended posterior distribution $\bar{\Pi}$ in (2.15). We sample from $\bar{\Pi}$ using Algorithm 3, with the set of temperatures $\{1, 1/0.9, 1/0.8, 1/0.7\}$. Since in this particular data model, the largest value of the (population) Rayleigh quotient is $\lambda_1 = 0.9$, proximity of the sample Rayleigh quotient $\mathsf{R}_n(\cdot; \mathbf{Z})$ to λ_1 along the MCMC iterations is a good empirical measure of mixing.

We run each MCMC sampler for $N = 10,000$ iterations, repeated 30 times (each time with a newly generated dataset). At each iteration time, we average the values of $\mathsf{R}_n(\cdot; \mathbf{Z})$ across the 30 repetitions. Fig. 2·1 shows the plot of the averaged sample Rayleigh quotient along iterations. The difference in behavior is striking. We clearly see that for all values of p, the sample Rayleigh quotient $\mathsf{R}_n(\cdot; \mathbf{Z})$ corresponding to $n = \lceil s_\star^{2.5} \log(p) \rceil$ quickly converges to the population Rayleigh quotient $\lambda_1 = 0.9$, whereas the one corresponding to $n = \lceil s_\star^{1.5} \log(p) \rceil$ fails to converge even after 10,000 iterations. This suggests that the condition $n \geq C_0 s_\star^2 \log(p)$ is indeed needed for the simulated tempering sampler to mix well, which appears to confirm the conjecture by (Gao et al. (2017)).

Empirical mixing time of Algorithm 2

We investigate more carefully the mixing time of Algorithm 2 as a function of the dimension p, using the coupled chain approach of (Biswas et al. (2019); Jacob et al. (2017)) as described in Section 2.3.1. We focus on a data-rich setting where the sample size $n = p/2$. Now, let us describe the implementation details. We let Σ_x, Σ_y,



(a) $p = 500$ (b) $p = 2000$ (c) $p = 5000$

Figure 2·1: Estimated canonical correlation along the MCMC iterations, averaged over 30 data repetitions for different values of dimension p and sample size n.

$v_{x\star}$ and $v_{y\star}$ all have the same structures as in Section 2.4.2 and set $\lambda_1 = 0.9$. We generate datasets from the model in Section 2.4.1 for each $p \in \{100, 200, \ldots, 5000\}$, with sample size $n = p/2$. The extended posterior distribution $\bar{\Pi}$ in (2.15) is constructed in the same way as in Section 2.4.2, except with the set of temperatures $\{1, 1/0.9, 1/0.8, 1/0.7, 1/0.6\}$. We set the lag $L = p$ and the maximum iterations $N = 10p + 1000$. For each value of p, we repeat the simulation 50 times to estimate the distribution of the meeting time $\tau^{(L)}$ of the chain. More precisely, using $\varepsilon = 0.1$, we estimate the mixing time of the chain as the first iteration t for which the Monte Carlo estimate of the right hand side of (2.16) is less than ε. Fig. 2·2 below shows the plot of the mean of meeting times and the estimated mixing times as functions of p. The results suggest that Algorithm 3 has a mixing time that scales roughly linearly in the dimension p.

Remark 2.4.1. *As far as we know, the existing literature on simulated tempering gives only general guidelines on choosing the temperatures (Geyer and Thompson (1995); Atchadé et al. (2011)). The implementation of these guidelines for choosing the temperature scaling remains non-trivial, and may require further adaptive MCMC methods (Miasojedow et al. (2013)). In our case, the Rayleigh quotient responds well to temperature tuning, and in particular does not require high temperatures to mix well. As a result, we have chosen not to further explore these more complicated scaling.*

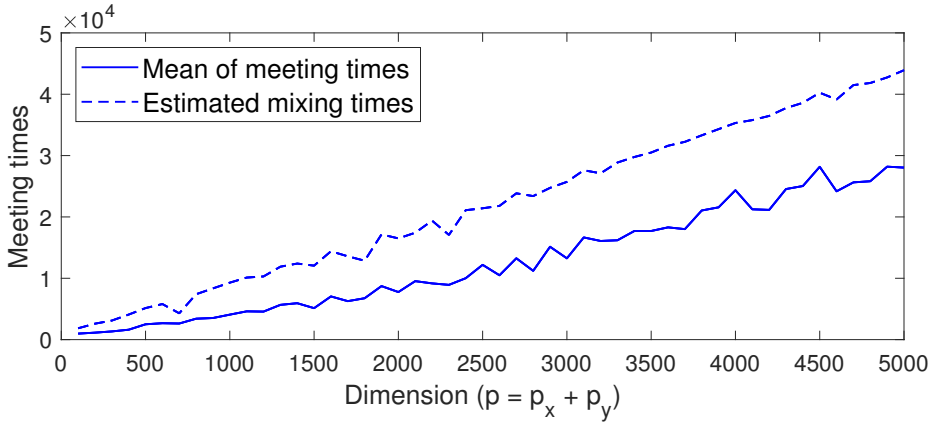


Figure 2·2: The mean of meeting times versus the estimated mixing times. The estimated mixing times are with respect to the total variation distance 0.1.

2.4.3 Comparison with other methods

We compare our method to two other methods, namely Rifle in Tan et al. (2018) and mixedCCA in Yoon et al. (2018). We investigate the behaviors of these methods in two settings. In the first setting, we use continuous datasets and compare our method with both Rifle and mixedCCA. In the second setting, we use mixed datasets and compare our method with mixedCCA (since Rifle is only designed for the continuous datasets).

Description of Rifle and mixedCCA

Before presenting our experimental results, let us briefly describe the other two methods, namely Rifle and mixedCCA. Rifle is a two-stage algorithm, where in the first stage, it (approximately) solves a convex relaxation of the problem in (2.1) in matrix variable. Next, the leading singular vectors of the output matrix are refined in the second stage using gradient ascent on the Rayleigh quotient $\mathsf{R}_n(\cdot; \mathbf{Z})$, and then truncated such that only the m entries with the largest absolute values are kept (and the

remaining entries are set to zero). Here m is a user-specified parameter that indicates the desired sparsity level of the estimated principle canonical vectors (v_x, v_y) – similar to s above. Note that since the first stage involves solving a matrix optimization problem, its computational time is typically much higher than that of the second stage. In a different approach, mixedCCA proposes a novel and robust estimator $\hat{\Sigma}$ for the covariance matrix Σ, namely the Kendall's-tau-based estimator, and estimates the canonical vectors (v_x, v_y) by solving the following convex problem:

$$\max_{v_x, v_y} v_x^{\mathrm{T}} \hat{\Sigma}_{xy} v_y - \lambda_1 \|v_x\|_1 - \lambda_2 \|v_y\|_1, \quad \text{s.t. } v_x^{\mathrm{T}} \hat{\Sigma}_x v_x \leq 1, \;\; v_y^{\mathrm{T}} \hat{\Sigma}_y v_y \leq 1, \tag{2.17}$$

where λ_1 and λ_2 are positive regularization parameters that need to be selected.

Continuous datasets

We randomly generate 100 continuous datasets using the model in Section 2.4.1, with the covariance matrices Σ_x and Σ_y constructed in a similar way to Yoon et al. (2018). Specifically, we set the sample size $n = 200$ and the dimension $p = 500$, and let Σ_x and Σ_y have the same structure, namely a block-diagonal matrix with five blocks of dimensions $\{d_1, ..., d_5\}$, respectively, and the (j, j')-th element of each block takes value $0.7^{|j-j'|}$. We set $\{d_1, ..., d_5\} = \{25, 50, 83, 50, 42\}$ for Σ_x and $\{d_1, ..., d_5\} = \{83, 50, 62, 31, 24\}$ for Σ_y. In addition, we let $v_{x\star}$ and $v_{y\star}$ have the same structures as in Section 2.4.2 (so that the true sparsity $s_\star = 6$), and set $\lambda_1 = 0.8$. In constructing the Rayleigh quotient $\mathsf{R}_n(\cdot; \mathbf{Z})$, we use the sample covariance matrices as estimators of Σ_x, Σ_y and Σ_{xy}.

In Algorithm 3, we let the set of temperatures be $\{1, 1/0.9, 1/0.8, 1/0.7, 1/0.6\}$, and only record the iterations corresponding to temperature 1. For comparison, we use the implementation of Rifle in the R package rifle, and set the parameter $m = 2s_\star = 12$, namely, m is twice as large as the true density level $s_\star$. Also, we

use the implementation of mixedcca in the R package mixedCCA, where λ_1 and λ_2 are selected using two criteria, namely BIC1 and BIC2. For this reason, we shall call the resulting algorithms mixedCCA-BIC1 and mixedCCA-BIC2, respectively. All the other parameters in Rifle and mixedCCA are set to the default values in the R packages. Both our algorithm and mixedCCA use the starting point found in the R package of mixedCCA. The outputs of all the algorithms are normalized to have unit Euclidean norms. We measure the quality of the estimated principle canonical vectors v_x and v_y by the squared-l_2 distance to the true canonical vectors $v_{x\star}$ and $v_{y\star}$, respectively. Specifically, we have

$$\mathsf{error}(v_x) \stackrel{\text{def}}{=} \min\left(\|v_x - v_{x\star}\|_2^2,\ \|v_x + v_{x\star}\|_2^2\right), \tag{2.18}$$

and $\mathsf{error}(v_y)$ is defined similarly. We also measure the quality of variable selection by the estimated v_x and v_y through computing the true-positive rate (TPR) and true-negative rate (TNR). For v_x, its TPR and TNR are defined as

$$\mathsf{TPR}(v_x) \stackrel{\text{def}}{=} \frac{|\{j : (v_x)_j \neq 0, (v_{x\star})_j \neq 0\}|}{|\{j : (v_{x\star})_j \neq 0\}|} \text{ and } \mathsf{TNR}(v_x) \stackrel{\text{def}}{=} \frac{|\{j : (v_x)_j = 0, (v_{x\star})_j = 0\}|}{|\{j : (v_{x\star})_j = 0\}|}, \tag{2.19}$$

respectively. For v_y, $\mathsf{TPR}(v_y)$ and $\mathsf{TNR}(v_y)$ are defined similarly. In the following, we will measure the quality of the estimated v_x and v_y using all three metrics above, i.e., error, TPR and TNR.

For each algorithm, we plot the quality of the estimated v_x and v_y averaged across the 100 datasets generated above, and the results are shown in Figure 2·3. Note that for Rifle, we only plot its second stage, which has a better starting point as compared to the other two algorithms, since it makes use of the output computed in its first stage. Our algorithm (Simulated tempering), not only outperforms Rifle in terms of the quality of estimated v_x and v_y (across all the three metrics), but

also enjoys much shorter running time. Indeed, our algorithm takes an average of 33 seconds (amounting to 200 iterations) to converge, whereas Rifle takes 280 seconds to complete its first stage, and about 4 seconds for running 1,000 iterations at its second stage. mixedCCA has better computational cost than ours — it takes about 7.2 seconds (amounting to 4 iterations) to converge. However, the quality of estimated v_x and v_y from our algorithm is better, and the advantage is especially significant in terms of error and TPR.

Mixed datasets

We randomly generate 100 mixed datasets in a similar way as in Section 2.4.3, except with an additional truncation step on the random vector Y. Specifically, we set the sample size $n = 150$ and the dimension $p = 200$, and let Σ_x and Σ_y each have five diagonal blocks of dimensions $\{d_1, ..., d_5\}$, respectively, and the (j, j')-th element of each block takes value $0.7^{|j-j'|}$. We set $\{d_1, ..., d_5\} = \{10, 20, 33, 20, 17\}$ for Σ_x and $\{d_1, ..., d_5\} = \{33, 20, 25, 12, 10\}$ for Σ_y. In addition, we let $v_{x\star}$ and $v_{y\star}$ have the same structures as in Section 2.4.2 (so that the true density level $s_\star = 6$), and set $\lambda_1 = 0.8$. Let $\mathsf{truc}(\cdot; C)$ be the (elementwise) truncation operator at level $C > 0$, such that given any vector y, $\mathsf{truc}(y; C)_j = y_j$ if $y_j > C$ and $\mathsf{truc}(y; C)_j = C$ otherwise. Then, for each dataset, we generate n samples from $(X, \mathsf{truc}(Y; C))$, where $(X, Y) \sim \mathbf{N}(0, \Sigma)$. Regarding the estimators of Σ_x, Σ_y, and Σ_{xy}, we will use Kendall's-tau-based estimators as in Yoon et al. (2018), which are stabler than sample covariance matrices for truncated data.

We run Algorithm 3 for $N = 10,000$ iterations with the set of temperatures $\{1, 1/0.9,$

$1/0.8, 1/0.7, 1/0.6\}$, and we use the output of last iteration as the estimator. We compare our algorithm with both mixedCCA-BIC1 and mixedCCA-BIC2 in the same fashion as in Section 2.4.3. The averaged results of these algorithms (as measured

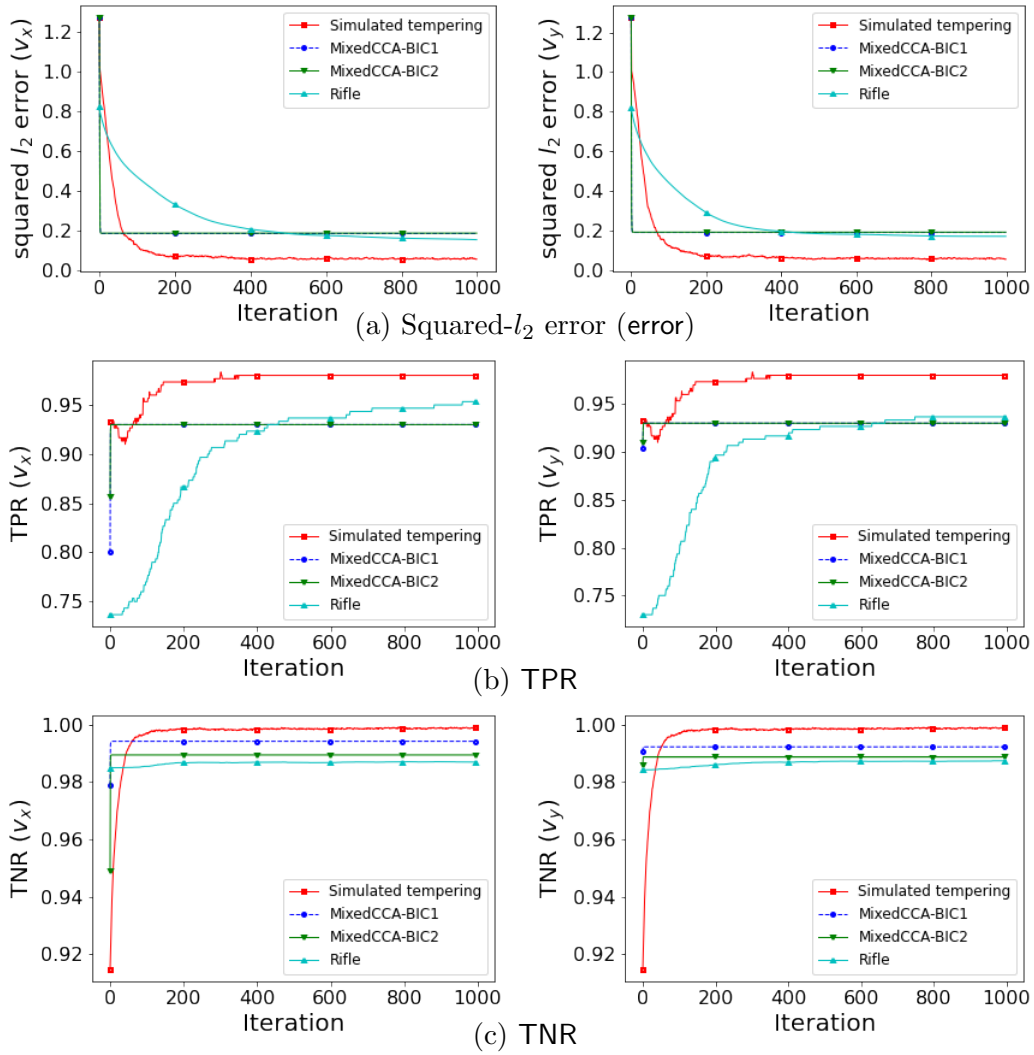


Figure 2·3: Comparison of the quality of estimated v_x and v_y by all the algorithms in terms of (a) squared-l_2 error (error), (b) TPR and (c) TNR. The results are averaged over 100 continuous datasets. The performances of mixedCCA-BIC1 and mixedCCA-BIC2 are indistinguishable on plots (a) and (b).

error	v_x			v_y		
C (Truncation level)	-2	-1	0	-2	-1	0
Simulated tempering	0.35 (0.00)	0.39 (0.01)	0.70 (0.01)	0.41 (0.60)	0.45 (0.65)	0.78 (0.76)
MixedCCA-BIC1	0.64 (0.74)	0.57 (0.70)	0.86 (0.79)	0.66 (0.73)	0.63 (0.74)	0.97 (0.74)
MixedCCA-BIC2	0.48 (0.67)	0.48 (0.66)	0.73 (0.72)	0.48 (0.64)	0.52 (0.67)	0.83 (0.71)

(a) error

TPR	v_x			v_y		
C (Truncation level)	-2	-1	0	-2	-1	0
Simulated tempering	0.85 (0.30)	0.83 (0.33)	0.67 (0.39)	0.85 (0.30)	0.83 (0.33)	0.67 (0.39)
MixedCCA-BIC1	0.71 (0.41)	0.74 (0.39)	0.60 (0.43)	0.71 (0.41)	0.74 (0.39)	0.60 (0.43)
MixedCCA-BIC2	0.82 (0.37)	0.82 (0.30)	0.75 (0.38)	0.83 (0.35)	0.81 (0.36)	0.71 (0.39)

(b) TPR

TNR	v_x			v_y		
C (Truncation level)	-2	-1	0	-2	-1	0
Simulated tempering	0.99 (0.01)	0.99 (0.02)	0.98 (0.02)	0.99 (0.01)	0.99 (0.02)	0.98 (0.02)
MixedCCA-BIC1	0.98 (0.01)	0.98 (0.01)	0.98 (0.02)	0.98 (0.01)	0.98 (0.01)	0.98 (0.02)
MixedCCA-BIC2	0.97 (0.03)	0.97 (0.03)	0.93 (0.09)	0.97 (0.02)	0.97 (0.03)	0.93 (0.09)

(c) TNR

Table 2.1: Mean (and standard deviation) of error, TPR and TNR of our method and mixedCCA for different values of truncation level C.

by error, TPR and TNR) over the 100 mixed datasets generated above are shown in Table 2.1. We can see from Table 2.1 that our approach outperforms the other two methods in terms of both error and TNR, and the advantage is particularly significant in terms of error. In terms of TPR, our algorithm is much superior to mixedCCA-BIC1. In addition, its performance is comparable to that of mixedCCA-BIC2 — specifically, it has slightly better performance when the truncation level $C = -2$ or -1 and slightly worse performance when $C = 0$.

2.5 Principal canonical correlation of clinical and proteomic data in Covid-19 patients

Covid-19 is an infectious disease that is rapidly sweeping through the world. The disease is caused by a severe acute respiratory syndrome coronavirus (SARS-CoV-2). There is currently an intense global effort to better understand the virus and find cures and vaccines. We use our methodology to re-analysis a data set produced by

Shen et al. (2020) that aims to identify biomarkers for early detection of severely ill Covid-19 patients[3]. To that end, the study enrolled 86 patients (some non-Covid-19 patients, and among the Covid-19 patients, some that developed mild symptoms, and some that became severely ill). The exact protocol for recruiting these patients is unclear. For each patient they measured three (3) physical characteristics (sex, age, and body mass index), twelve (12) clinical variables as routinely measured from blood samples (white blood cells count, lymphocytes count, C-reactive protein, etc...). Furthermore, the serum of each patient is analyzed by liquid mass spectrometry-based proteomics to quantify their proteome and metabolome. In Shen et al. (2020), the data is used to build a statistical model to predict whether or not a Covid-19 patient will progress to a severe state of illness. The dataset of Shen et al. (2020) is freely available from the journal website.

We use canonical correlation analysis to re-analyze the data. A common working assumption is that SARS-CoV-2 induces patterns of molecular changes that can be detected in the sera of patients. Canonical correlation analysis may help identify these patterns. To do this we focus on the proteomic data, and we estimate the principal sparse canonical correlation between the physical and clinical variables on one hand and the proteomic variables on the other. See for instance Rousu et al. (2013) for a similar analysis on tuberculosis and malaria.

We pre-process the data by removing all the proteins for which 50% or more values are missing, leading to a total of $p_y = 513$ proteins, and $p_x = 16$ clinical and physical variables. The sample size $n = 86$. Liquid mass spectrometry-based proteomics typically produces a large quantity of missing values (Karpievitch et al. (2010); O'Brien et al. (2018)). We make the assumption here that the missing values are driven mainly by detection limit truncation (Karpievitch et al. (2010)). We apply

[3]For reasons that are still poorly understood, about 80% of patients infected by SARS-CoV-2 experience mild to no symptoms, whereas in about 20% of the cases, patients become severely ill.

both our algorithm and mixedCCA to this problem, with the same parameter setting as in the simulation test on the mixed datasets (cf. Section 2.4.3). We run our algorithm for $N = 10,000$ iterations. Since we do not know the true canonical pair, we will focus on the estimated canonical correlation to measure the performance of two algorithms. In terms of the estimated canonical correlation, both our algorithm and mixedCCA takes less than 1 second to converge.

Our estimate of the principal canonical vectors of first dataset ($v_{x\star}$) has only one selected component (corresponding to C-reactive protein – CRP) with estimated inclusion probability of $\Pi(\delta_j = 1|\mathbf{Z}) = 0.99$. All other physical and clinical variables have inclusion probabilities smaller than 0.1. We found also that the principal canonical vectors of the proteomic data is also driven by a single protein (P02763, also known as Alpha-1-acid glycoprotein 1 or AGP 1), with estimated inclusion probability of $\Pi(\delta_j = 1|\mathbf{Z}) = 0.89$. All other proteins have inclusion probability smaller than 0.1. Fig. 2·4 shows the traceplot of the estimated canonical correlation $\hat{\rho}$ between the two data set, as well as the boxplot and autocorrelation function of the MCMC output (after burning in 3/4 of iterations) of the coefficients of CRP and AGP 1 in the quasi-posterior distribution. The fast decay of the autocorrelation functions show a good mixing of the MCMC sampler.

MixedCCA also selects CRP for the clinical dataset and AGP 1 for the proteomic dataset, but both BIC1 and BIC2 criterion select many other variables. mixedCCA-BIC1 also selects glucose for clinical dataset and 3 other variables for the proteomic dataset, with estimated canonical correlation 0.90. mixedCCA-BIC1 selects 8 other variables for clinical dataset and 3 additional variables for the proteomic dataset with estimated canonical correlation 0.93. Although the estimated canonical correlation of mixedCCA is larger than the estimated canonical correlation (0.80) in our algorithm, the highly sparse nature of the estimated canonical vectors estimated

from our method is striking.

Several studies have observed the predictive power of C-reative protein (CRP) in the progression of Covid-19 into a severe illness (see for instance Sahu et al. (2020) for a meta-analysis). This suggests that the correlation detected in our analysis between the two datasets is indeed driven by the progression of Covid-19 into a severe illness. Therefore, our analysis suggests that protein AGP 1 may also be playing an important role in the progression of Covid-19 into a severe illness. In Fig. 2·5, we present the boxplot of CRP and AGP 1 by group of patients. We can see that severe covid patients will have higher value of CRP and AGP 1, compared to non-covid and non-severe patients. We learn from Uniprot[4], that AGP 1 functions as transport protein in the blood stream, and appears to function in modulating the activity of the immune system during the acute-phase reaction. Furthermore, AGP 1 appears on the list of differentially expressed proteins in the sera of severely ill Covid-19 patients designed by Shen et al. (2020), and also appeared in the literature as playing a role in the immune system's response to malaria (Friedman (1983)).

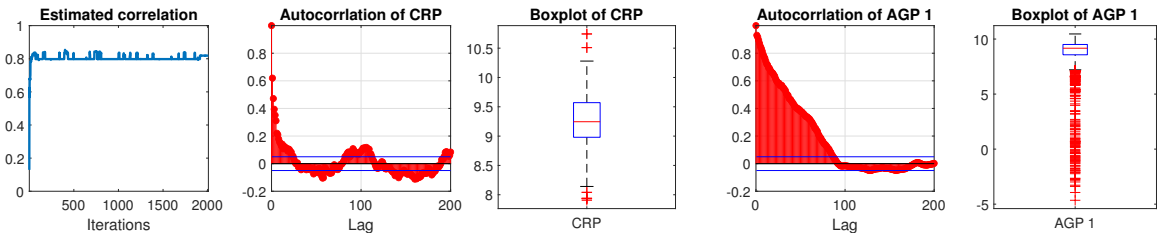
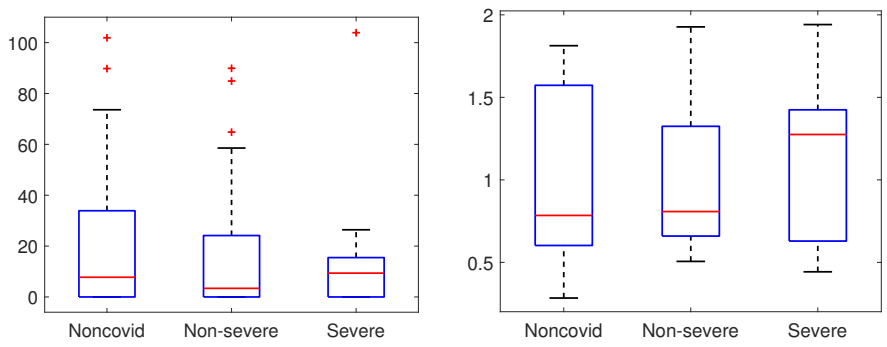


Figure 2·4: From left to right: The first plot is the trace plot of estimated canonical correlation; The second and third plot is the autocorrelation and boxplot of the coefficient of CRP from MCMC output; The fourth and fifth plot is the autocorrelation and boxplot of the coefficient of AGP 1 from MCMC output.

[4]https://www.uniprot.org



(a) CRP (b) AGP

Figure 2·5: Boxplots of (a) CRP and (b) AGP 1 by group of patients.

2.6 Conclusion

In this work, we have developed a minimax optimal estimation procedure for sparse canonical correlation analysis using a quasi-Bayesian framework. Our method can be further extended to capture more than one canonical vector, either by deflation, or by reformulating the problem as a higher dimensional canonical correlation analysis estimation problem as in Tan et al. (2018). Furthermore, one can straightforwardly extend our method to solve other generalized eigenvalue problems that arise in other statistical problems, as for instance in Fisher discriminant analysis. At a higher level, the method developed in this work can be viewed as a more statistical implementation of simulated annealing for optimization under sparsity constraints. As such, it can be applied more widely to solve non-convex optimization problems with sparsity constraints.

Chapter 3

A statistical perspective on algorithm unrolling models for inverse problems

3.1 Introduction

Inverse problems are common problems in science and engineering where one seeks information on a latent variable of interest, given some related observation. We consider an inverse problem with a latent quantity of interest $\mathbf{x} \in \mathbb{R}^{d_x}$ that is related to the observed variable $\mathbf{y} \in \mathbb{R}^{d_y}$ through the so-called forward statistical model

$$\mathbf{y} \mid \mathbf{x} \sim e^{-f(\mathbf{y}|\mathbf{x})} \mathrm{d}\mathbf{y}, \tag{3.1}$$

for some function $f(\cdot|\mathbf{x}) : \mathbb{R}^{d_y} \to \mathbb{R}$. Throughout this chapter, unless otherwise stated, all model densities are defined with respect to the corresponding Lebesgue measure. Although the function f is unknown in general, we focus in this work on inverse problems for which the forward model is well-understood and f is known. This is the case with many inverse problems in imaging. An important special case in the applications is the Gaussian linear model corresponding (up to an additive constant that we ignore) to

$$f(\mathbf{y}|\mathbf{x}) = \frac{1}{2v^2} \|\mathbf{y} - A\mathbf{x}\|_2^2, \tag{3.2}$$

with known parameters $v > 0$, and $A \in \mathbb{R}^{d_y \times d_x}$. When the inverse problem is ill-posed, prior knowledge is fundamental for good recovery of $\mathbf{x}$. For example in the linear

regression model (3.2), it is well-known that without any additional assumption, the minimax optimal rate in the estimation of $\mathbf{x}$ is of order $\sqrt{d_x/d_y}$. However this rate can be improved if $\mathbf{x}$ is known to possess some additional features such as smoothness or sparsity. A Bayesian perspective is particularly simple. If μ_0 denotes a prior distribution that encodes the prior information available on $\mathbf{x}$, then $\mathbf{x}$ is inferred using its posterior distribution

$$\pi_{\mu_0}(\mathrm{d}\mathbf{x}|\mathbf{y}) \propto \mu_0(\mathrm{d}\mathbf{x})e^{-f(\mathbf{y}|\mathbf{x})}. \tag{3.3}$$

Inverse problems have a long history in statistics and applied mathematics, and the posterior distribution in (3.3) as well as related penalized estimators are the backbone of rigorous inference Bissantz et al. (2007); Stuart (2010); Knapik et al. (2011); Blanchard and Mücke (2018); Rastogi et al. (2020). When valid information are available on $\mathbf{x}$ and appropriately encoded in μ_0, the posterior distribution π_{μ_0} can enjoy better statistical properties than say, the minimizer of $\mathbf{x} \mapsto f(\mathbf{y}|\mathbf{x})$. However, finding such good prior distributions is often very challenging in many applications.

3.1.1 Learning to solve inverse problems

In a growing number of settings, particular in image restoration tasks, researchers have access to datasets in which the latent variable $\mathbf{x}$ and the related observation $\mathbf{y}$ are both observed. Indeed such datasets can often be simulated in settings where f is known. Hence, suppose that we have a dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i),\ 1 \le i \le n\}$ of i.i.d. samples, such that for $1 \le i \le n$,

$$\mathbf{x}_i \sim \mu, \quad \text{and} \quad \mathbf{y}_i \mid \mathbf{x}_i \sim e^{-f(\mathbf{y}|\mathbf{x}_i)}\mathrm{d}\mathbf{y}, \text{ and where } \mu(\mathrm{d}\mathbf{x}) = \frac{1}{c_\mu}e^{-\mathcal{R}(\mathbf{x})}\mathrm{d}\mathbf{x}, \tag{3.4}$$

for some function $\mathcal{R} : \mathbb{R}^{d_x} \to \mathbb{R}$, and a normalizing constant c_μ. Hence under (3.4), μ is the marginal distribution of the latent variables. The conditional distribution of

$\mathbf{x}_i$ given $\mathbf{y}_i$ is then given by

$$\pi(\mathrm{d}\mathbf{x}|\mathbf{y}_i) \propto \exp\left(-\mathcal{R}(\mathbf{x}) - f(\mathbf{y}_i|\mathbf{x})\right) \mathrm{d}\mathbf{x},$$

and its modal value is given by the function $g: \mathbb{R}^{d_y} \to \mathbb{R}^{d_x}$ with

$$g(\mathbf{y}) \stackrel{\text{def}}{=} \operatorname*{\mathsf{Argmin}}_{\mathbf{x}\in\mathbb{R}^{d_x}} \left[f(\mathbf{y}|\mathbf{x}) + \mathcal{R}(\mathbf{x})\right]. \tag{3.5}$$

We will assume below that $g(\mathbf{y})$ is uniquely defined. We stress again that the distribution μ in (3.4) is not a prior distribution of $\mathbf{x}$ as selected by the researcher, but the actual marginal distribution of $\mathbf{x}$ unknown to the researcher. Hence g and $\pi(\cdot|\mathbf{y})$ and typically unknown. In fact, one of the key challenges in inverse problems is building a prior distribution μ_0 that is as close as possible to μ so that the resulting posterior distribution as given in (3.3) approximates well the corresponding conditional distribution.

In keeping with the assumption that $\mathbf{x}$ possesses additional structures, in many inverse problems the support of the marginal distribution μ lays in a much smaller (but unknown) subspace of $\mathbb{R}^{d_x}$. As a result of such marginal distribution concentration, it is often the case that the conditional distribution of $\mathbf{x}$ given $\mathbf{y}$ is also tightly concentrated around $g(\mathbf{y})$, in the sense that

$$\mathbf{x}_i = g(\mathbf{y}_i) + \boldsymbol{\xi}_i, \text{ where } \mathbb{E}(\boldsymbol{\xi}_i \mid \mathbf{y}_i) \approx 0, \ 1 \le i \le n. \tag{3.6}$$

The representation (3.6) makes clear that in such settings where we have an informative (but unknown) marginal distribution, and given a dataset $\mathcal{D}$, one can learn the function g by regressing $\mathbf{x}$ on $\mathbf{y}$. In other words, we can learn to solve directly the inverse problem by regression using the dataset $\mathcal{D}$. The approach has become popular in computational imaging (Burger et al. (2012); Xie et al. (2012); Lucas et al. (2018); Yang et al. (2016); Ravishankar et al. (2017); Aggarwal et al. (2017); Chun

and Fessler (2018); Zhang et al. (2017); Liu et al. (2019); Li et al. (2020)). A remarkable contribution of this literature is a number of specific deep neural network architectures generally called algorithm unrolling networks that leverage the structure of the forward model (Gregor and LeCun (2010); Sreter and Giryes (2018); Sulam et al. (2020); Tolooshams et al. (2020)), see also the reviews (Ongie et al. (2020); Shlezinger et al. (2021); Monga et al. (2021)).

However a fundamental question that has not been addressed in the literature so far is how well one can estimate the function g using these unrolling-based deep neural network architectures.

3.1.2 Main contributions

To address this problem, and assuming that the data generating process (3.6) holds, we consider the nonparametric regression model

$$\mathbf{x}_i = g_W(\mathbf{y}_i) + \boldsymbol{\epsilon}_i, \quad i = 1, \dots, n, \tag{3.7}$$

with regression errors $\boldsymbol{\epsilon}_i \overset{i.i.d.}{\sim} \mathbf{N}(0, \sigma^2 I_{d_x})$, for some positive variance parameter σ^2 taken as known for simplicity, and for a function class $\{g_W,\ W \in \mathcal{W}\}$, where $g_W : \mathbb{R}^{d_y} \to \mathbb{R}^{d_x}$ is a gradient descent neural network (GDN) function obtained by unrolling D' times a parametrized proximal gradient descent algorithm for solving (3.5). We give precise definition below. The architecture thus makes explicit use of the forward map f. We develop a sparse Bayesian framework for estimating (3.7) using a spike-and-slab prior distribution on the parameter W, and we analyze the statistical performance of the resulting estimator of g. We focus on the setting where the function $\mathcal{R}$ is convex, but not necessarily differentiable. Under some additional regularity conditions, and ignoring logarithmic terms, we show that GDN for

estimating g achieves the statistical error rate

$$C_1(D') \times n^{-\frac{1}{2+d_x}},$$

for some constant $C_1(D')$ that depends on the unrolling depth D', but also on the input dimensions d_x, d_y. Keeping dimensions and the number of unrolling D' fixed, the result implies for example that when $d_y \geq d_x$, the GDN architecture, by making explicit use of the forward model, achieves a better rate than the minimax rate $C_2 n^{-\frac{1}{2+d_y}}$ for estimating $g : \mathbb{R}^{d_y} \to \mathbb{R}^{d_x}$ viewed as a Lipschitz function. We note however that these constants $C_1(D'), C_2$ can depend poorly[1] on the dimensions d_x, d_y.

The convergence rate of the estimator can be faster than the aforementioned rate. Indeed, we also show that when the proximal map of $\mathcal{R}$ is simple and can be well-approximated by a simple neural network function, then the GDN architecture achieves a faster statistical rate. For instance, if μ is as in (3.17) below, a common assumption in image restoration tasks, then ignoring log terms, our result shows that GDN achieves the parametric rate $C_3 \times D'/\sqrt{n}$, for some dimension-dependent constant C_3.

One of the practical challenges in building a GDN is the lack of theoretical guidelines in the choice of the depth D'. An offshoot of our theoretical analysis is the derivation that the best performance of GDN is achieved by scaling the network depth as $D' \sim \log(n)/\log(\varrho_n^{-1})$, where ϱ_n is the convergence rate of the proximal gradient algorithm for solving (3.5). Our results also seem to suggest that the convergence rates of GDN unrolled at depth D' deteriorates as D' increases, which would imply an overfitting phenomenon of the model as D' increases. Although we do not have a matching lower bound theory to confirm this overfitting phenomenon, we have

[1]This potentially poor dependence on the input dimensions is not specific to our work, and is rooted in the current state of knowledge in deep learning approximation theory (see e.g. Yarotsky (2017))

performed extensive numerical experiments that all show an overfitting of the model as D' increases.

3.1.3 Related work

Most of the existing theoretical results on algorithm unrolling have studied the approximation capability of the resulting function class in the linear case. For instance Chen et al. (2018) studied the capability of the GDN function class to recover directly the signal $\mathbf{x}$ in the linear model (3.2). Gilton et al. (2020) proposed a novel unrolling architecture based on the Neumann series identity, and studied its approximation capability in the noiseless version of the linear model (3.2). To the best of our knowledge, our work is the first to analyze the statistical properties of algorithm unrolling in a way that accounts for both its approximation capability *and* its complexity.

Several prior works have also considered the statistical complexity of other deep learning models using a similar nonparametric regression setting where regularization is explicitly introduced to control model complexity Barron and Klusowski (2018); Schmidt-Hieber (2020); Taheri et al. (2021); Ee et al. (2020). Our framework is closer to Polson and Ročková (2018) and employs a Bayesian approach. However none of these results can be directly applied to algorithm unrolling architectures. Another unique feature of our framework that is worth emphasizing is that it produces posterior distributions that are both computationally and mathematically tractable.

Finally we contrast our nonparametric regression approach with the two-step approach proposed for instance by Chang et al. (2017), where the proximity operator of $\mathcal{R}$ is first estimated from the dataset $\mathcal{D}$, and g is then estimated by solving (3.5) using the estimated proximal operator obtained from the first step. The strategy seems statistically sub-optimal because the estimation of the proximal operator requires estimating the density of μ in general, which is a fundamentally more difficult statistical problem (Samworth (2018)). However it is conceivable that an adaptive

density estimation method may exist that achieve better rates for densities with simple proximal maps, thus matching the approach developed here. More research is needed on this topic.

3.1.4 Outline of this chapter

The remainder of the chapter is organized as follows. The main results are described in Section 3.2. The results are obtained using a more general Bayesian posterior contraction result of independent interest that we described in Section 3.3. Some supporting numerical illustrations are presented in Section 3.4. All the proofs are postponed to Appendix B.4.

3.1.5 Notations

We define the sub-Gaussian norm of a probability measure ν on $\mathbb{R}^d$ with expected value m as the smallest constant c for which the following holds

$$\int_{\mathbb{R}^d} e^{\langle u, z-m\rangle} \nu(\mathrm{d}z) \leq e^{\frac{c^2\|u\|_2^2}{2}}, \quad \text{for all } u \in \mathbb{R}^d.$$

If Z is a random variable with distribution ν, we write $\|Z\|_{\psi_2}$ to denote the sub-Gaussian norm of ν. We note that this definition applies also to conditional densities, and we write $\|Z|X\|_{\psi_2}$ to denote the sub-Gaussian norm of the conditional distribution of Z given X.

Throughout the chapter the notation $a \lesssim b$ means that $a \leq cb$, for some constant c that does not depend on the sample size n.

Vectorization

Let $\{h_W,\ W \in \mathcal{W}\}$ denote a generic deep neural network class of function where $h_W :\ \mathbb{R}^{p_0} \to \mathbb{R}^{p_D}$, with parameter $W = (W_D, \ldots, W_1) \in \mathcal{W} \stackrel{\text{def}}{=} \mathbb{R}^{p_D \times p_{D-1}} \times \cdots \times \mathbb{R}^{p_1 \times p_0}$. By vectorization, we will view W as the Euclidean space $\mathbb{R}^q$ (where $q \stackrel{\text{def}}{=} \sum_{\ell=1}^{D} p_\ell p_{\ell-1}$),

and we will use a generic notation $\|\cdot\|_2$ to denote its Euclidean norm. Similarly, we will write $\|W\|_0$ (resp. $\|W\|_\infty$) to denote the number of non-zeros components of W (resp. the largest absolute value of the components of W). For any $1 \leq \ell \leq D$, we will similarly view W_ℓ as a vector element of $\mathbb{R}^{p_\ell p_{\ell-1}}$, and define similarly $\|W_\ell\|_2$, $\|W_\ell\|_0$ and $\|W_\ell\|_\infty$. Hence, in what follows, for a matrix M, $\|M\|_2$ will always denote the Frobenius norm of M, not its spectral norm. We will write the spectral norm as $\|\cdot\|_{\mathsf{op}}$.

3.2 Learning to solve inverse problems

Summarizing the introductory discussion on the data generating process, we make the following assumption.

H 4. *We have a data set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i),\ 1 \leq i \leq n\}$ of i.i.d. samples generated according to (3.4) such that for $1 \leq i \leq n$,*

$$\mathbf{x}_i = g(\mathbf{y}_i) + \boldsymbol{\xi}_i, \quad \textit{where} \quad \mathbb{E}(\boldsymbol{\xi}_i \mid \mathbf{y}_i) = 0,$$

for some independent error terms $(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_n)$. Furthermore we assume that each $\boldsymbol{\xi}_i$ is a conditionally sub-Gaussian random vector given $\mathbf{y}_i$, with a non-random sub-Gaussian norm $\sigma_i < \infty$.

Remark 4. Assumption H4 formalizes the discussion in the introduction on the concentration of the conditional distribution of $\mathbf{x}_i$ given $\mathbf{y}_i$. As expanded upon in the introduction, this assumption is conceptually justified in seetings where the latent variable $\mathbf{x}$ has additional structures, and the marginal distribution of $\mathbf{x}$ is concentrated on a low-dimensional subset of $\mathbb{R}^{d_x}$. Checking H4 is similar to establishing a Bernstein-von Mises theorem for the conditional distribution of $\mathbf{x}_i$ given $\mathbf{y}_i$ (Vaart (1998); Bickel and Kleijn (2012); Nickl (2017); Nickl and Sohl (2019)).

Let ς_i denote the conditional sub-Gaussian norm of $\|\boldsymbol{\xi}_i\|_2$ given $\mathbf{y}_i$. The conditional sub-Gaussian assumption on $\boldsymbol{\xi}_i$ imposed in Assumption 4 implies that $\varsigma_i < \infty$ (see

e.g. Theorem 3.1.1 of Vershynin (2018)). Throughout we set

$$\bar{\sigma} \stackrel{\text{def}}{=} \max_{1 \le i \le n} \sigma_i, \quad \text{and} \quad \bar{\varsigma} \stackrel{\text{def}}{=} \max_{1 \le i \le n} \varsigma_i.$$

3.2.1 Gradient descent networks

We consider the nonparametric regression (3.7), where $\{g_W,\ W \in \mathcal{W}\}$ is a gradient descent network (GDN) function class that we now define. First we introduce a generic feed-forward deep neural network function $H_W :\ \mathbb{R}^{d_x} \to \mathbb{R}^{d_x}$. Let $D > 0$ be the depth of the network. Let $(p_D, \ldots, p_0)$ be a sequence of integers representing the sizes of the layers of the network, with $p_0 = d_x$, and $p_D = d_x$. For $1 \le \ell \le D$, let $\mathsf{a}_\ell :\ \mathbb{R}^{p_\ell} \to \mathbb{R}^{p_\ell}$ be activation functions that we assume Lipschitz: for all $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^{p_\ell}$,

$$\mathsf{a}_\ell(\mathbf{0}) = \mathbf{0}, \quad \text{and} \quad \|\mathsf{a}_\ell(\mathbf{z}_1) - \mathsf{a}_\ell(\mathbf{z}_2)\|_2 \le \|\mathbf{z}_1 - \mathbf{z}_2\|_2. \tag{3.8}$$

For $B \in \mathbb{R}^{p_\ell \times p_{\ell-1}}$, we set

$$\Psi_B^{(\ell)}(\mathbf{z}) \stackrel{\text{def}}{=} \mathsf{a}_\ell(B\mathbf{z}), \quad \mathbf{z} \in \mathbb{R}^{p_{\ell-1}}. \tag{3.9}$$

With parameter $W = (W_D, \ldots, W_1)$, where $W_\ell \in \mathbb{R}^{p_\ell \times p_{\ell-1}}$, we consider the function $H_W : \mathbb{R}^{d_x} \to \mathbb{R}^{d_x}$ defined as

$$H_W(\mathbf{x}) = \Psi_{W_D}^{(D)} \circ \cdots \circ \Psi_{W_1}^{(1)}(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^{d_x}, \tag{3.10}$$

where $f \circ g$ is the composition of f with g.

Remark 5. Feed-forward deep neural network models are usually written with additional bias terms (that is, by defining $\Psi_B^{(\ell)}(\mathbf{z})$ as $\mathsf{a}_\ell(B\mathbf{z}+\mathbf{b})$). However our formulation incurs no loss of generality, since these bias parameters can always be subsumed into the matrix B, by appropriately enlarging B and adding an intercept to the input.

We use the function H_W to approximate the proximal map of $\gamma\mathcal{R}$ defined for some

$\gamma > 0$ as

$$\mathrm{Prox}^{\gamma\mathcal{R}}(\mathbf{x}) \stackrel{\text{def}}{=} \underset{\mathbf{u}\in\mathbb{R}^{d_x}}{\mathsf{Argmin}} \left[\gamma\mathcal{R}(\mathbf{u}) + \frac{1}{2}\|\mathbf{u}-\mathbf{x}\|_2^2\right].$$

Given a step-size $\gamma > 0$, $W \in \mathcal{W}$, and $\mathbf{y} \in \mathbb{R}^{d_y}$ we thus define the function $F_{\mathbf{y},W} : \mathbb{R}^{d_x} \to \mathbb{R}^{d_x}$ by

$$F_{\mathbf{y},W}(\mathbf{x}) \stackrel{\text{def}}{=} H_W\left(\mathbf{x} - \gamma\nabla_{\mathbf{x}} f(\mathbf{y}|\mathbf{x})\right).$$

Given $D' \geq 1$ (the unrolling depth of the network), we consider the function g_W defined as

$$g_W(\mathbf{y}) \stackrel{\text{def}}{=} \underbrace{F_{\mathbf{y},W} \circ \cdots \circ F_{\mathbf{y},W}}_{D' \text{ times}}(\mathbf{x}^{(0)}), \tag{3.11}$$

for some initial value $\mathbf{x}^{(0)} \in \mathbb{R}^{d_x}$. We note that in addition to W, the function g_W depends also on the step-size γ, the unrolling depth D', and the initial value $\mathbf{x}^{(0)}$. The network architecture in (3.11) is the so-called (proximal) gradient descent network (GDN), and belong to the class of so-called algorithm unrolling (or unfolding) deep learning models, where a statistical model is built by iterating an optimization algorithm. Many variations have been proposed in the literature based on various other optimization schemes (we refer the reader to the references in the introduction).

For $\mathbf{x} \in \mathbb{R}^{d_x}$, and $\mathbf{y} \in \mathbb{R}^{d_y}$, we set

$$F_{\mathbf{y}}(\mathbf{x}) \stackrel{\text{def}}{=} \mathrm{Prox}^{\gamma\mathcal{R}}\left(\mathbf{x} - \gamma\nabla_{\mathbf{x}} f(\mathbf{y}|\mathbf{x})\right).$$

Looking at the definition of (3.11), it is clear that under appropriate convexity assumptions and for well-selected step size γ, the convergence of $F_{\mathbf{y}}^j(\mathbf{x})$ toward $g(\mathbf{y})$ is guaranteed, where h^j denotes the composition of h, j times. Therefore, if for some W, $H_W \approx \mathrm{Prox}^{\gamma\mathcal{R}}$, then we can expect $g_W \approx g$ for D' sufficiently large, by standard convex optimization theory. As a result, the function class $\{g_W,\ W \in \mathcal{W}\}$ typically has good skills in approximating g. We impose next the necessary assumptions for the intuition above to hold.

H 5. *1. The function $\mathcal{R}: \mathbb{R}^{d_x} \to \mathbb{R}$ is convex, and there exists M such that for all $\mathbf{y} \in \mathbb{R}^{d_y}$, the function $\mathbf{x} \mapsto f(\mathbf{y}|\mathbf{x})$ is convex, differentiable, and with a M-Lipschitz gradient. Furthermore, the step-size γ satisfies $0 < \gamma \leq M^{-1}$, and for all $\mathbf{y} \in \mathbb{R}^{d_y}$, $g(\mathbf{y})$ is uniquely defined.*

2. There exist $R_0 < \infty$, $\varrho_n \in [0, 1)$ such that for all $\mathbf{y} \in \mathsf{Y}$, and all $k \geq 1$,

$$\max_{1 \leq i \leq n} \|F^k_{\mathbf{y}_i}(\mathbf{x}^{(0)}) - g(\mathbf{y}_i)\| \leq R_0 \varrho_n^k.$$

Remark 6. Assumption H5-(1) is a standard set up if one wishes to approximate the function g by proximal gradient descent. Assumption H5-(2) is stronger and imposes a linear convergence rate. It is well-known that this holds for strongly convex problems. In some special cases, H5-(2) can be shown to follow from H5-(1) when $\mathbf{x}^{(0)}$ is close enough to the solution and the algorithm satisfies a local linear convergence. For instance it is known that such local linear convergence of proximal gradient descent holds for the lasso problem (Tao et al. (2016)). Note that in order for the proximal gradient algorithms to have computational guarantees on the (Euclidean) distance to the optimal solution of some convex problem (similar to the one in H5-(2)), the problem typically needs to possess certain (local or global) growth property around the optimal solution (see e.g., Necoara et al. (2019)), and under such a property, the proximal gradient algorithms converge linearly — this explains why we impose H5-(2) in terms of linear rate rather than sub-linear rate.

We also impose the following assumption that models the approximation of the proximal map $\text{Prox}^{\gamma\mathcal{R}}$.

H 6. *There exist $\beta_1, \beta_2 \geq 0$, such that for all $\epsilon \in (0, 1)$, and all $R \geq 1$, we can construct a feed-forward deep neural network H_W, as in (3.10), with depth $1 \leq D \leq D_0 \log(\sqrt{d_x}/\epsilon)$, maximum layer size no larger than $N_0 \left(\sqrt{d_x}/\epsilon\right)^{\beta_1}$, maximum parameter absolute value $\|W\|_\infty$ no larger than 1, and maximum sparsity $\|W\|_0$ no larger than $s_0 \left(\sqrt{d_x}/\epsilon\right)^{\beta_2}$, for constants D_0, N_0, s_0 that do not depend on ϵ such that*

$$\sup_{\mathbf{x}:\, \|\mathbf{x}\|_2 \leq R} \left\| H_W(\mathbf{x}) - \text{Prox}^{\gamma\mathcal{R}}(\mathbf{x}) \right\|_2 \leq \epsilon.$$

Remark 7. Since $\mathbf{x} \mapsto \text{Prox}^{\gamma\mathcal{R}}(\mathbf{x})$ is a Lipschitz map, we can always invoke classical deep learning approximation theory for smooth functions (see e.g. Schmidt-Hieber

(2020); DeVore et al. (2021)) to conclude that Assumption 6 holds with $\beta_1 = \beta_2 = d_x$. However better approximation is possible if $\text{Prox}^{\gamma\mathcal{R}}$ is a simple map.

3.2.2 Bayesian inference using spike-and-slab priors

We consider the problem of fitting model (3.7), where $\{g_W,\ W \in \mathcal{W}\}$ is the GDN function class constructed in (3.11). The parameter space is $\mathsf{W} \stackrel{\text{def}}{=} \mathbb{R}^{\mathsf{p_D} \times \mathsf{p_{D-1}}} \times \cdots \times \mathbb{R}^{\mathsf{p_1} \times \mathsf{p_0}}$. As indicated at the end of the introduction, at times we shall view W as the Euclidean space $\mathbb{R}^{\mathsf{q}}$, where

$$\mathsf{q} \stackrel{\text{def}}{=} \sum_{\ell=1}^{D} (p_\ell \times p_{\ell-1}).$$

Our initial motivation in this work comes from inverse problems in remote sensing. It was therefore important for us to analyze a statistical procedure that can be implemented in practice. An important shortcoming of the current statistical theory of deep learning models under sparsity constraints (Barron and Klusowski (2018); Schmidt-Hieber (2020); Taheri et al. (2021); Ee et al. (2020)) is the lack of computational tractability of the resulting estimators. To address this issue we propose to fit the model $\{g_W,\ W \in \mathcal{W}\}$ in a Bayesian framework using a spike and slab priors (Atchadé and Bhattacharyya (2019)). To that end, we introduce a sparsity structure parameter $\Lambda = (\Lambda_D, \ldots, \Lambda_1) \in \mathcal{S} \stackrel{\text{def}}{=} \{0,1\}^{p_D \times p_{D-1}} \times \cdots \times \{0,1\}^{p_1 \times p_0}$, with a prior distribution given by

$$\Pi_0(\Lambda) \propto \left(\frac{1}{q}\right)^{(\mathsf{u}+1)\|\Lambda\|_0}, \quad \Lambda \in \mathcal{S}, \tag{3.12}$$

for some parameter $\mathsf{u} \geq 1$. This prior corresponds to the assumption that the entries of Λ are independent Bernoulli random variables $\mathbf{Ber}((1+q^{\mathsf{u}+1})^{-1})$. Given Λ we assume that the entries of W are conditionally independent with joint density

$$\Pi_0(W|\Lambda) \quad = \quad \prod_{\ell=1}^{D} \prod_{(i,k):\ \Lambda_{\ell,i,k}=1} \frac{\rho_1}{2} e^{-\rho_1 |W_{\ell,ik}|} \prod_{(i,k):\ \Lambda_{\ell,k}=0} \sqrt{\frac{\rho_0}{2}} e^{-\frac{\rho_0}{2} W_{\ell,i,k}^2}, \tag{3.13}$$

for some parameters $0 < \rho_1 < \rho_0$. Using this prior distribution and the regression model (3.7), we consider the posterior distribution on $\Theta \stackrel{\text{def}}{=} \mathcal{S} \times \mathsf{W}$ with density given by

$$\Pi(\Lambda, W \mid \mathcal{D}) \propto \Pi_0(\Lambda, W) \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n} \|\mathbf{x}_i - g_{W\odot\Lambda}(\mathbf{y}_i)\|_2^2\right), \tag{3.14}$$

where $W \odot \Lambda$ denotes the component-wise product of W and Λ, whose element is $W_{\ell,i,k} \cdot \Lambda_{\ell,i,k}$. To use this posterior distribution we draw sample $(\Lambda, W) \sim \Pi(\cdot|\mathcal{D})$, and use $g_{\Lambda\odot W}$ as inversion map. Since Λ is typically sparse under Π, $g_{\Lambda\odot W}$ is a sparse GDN. For $h : \mathbb{R}^{d_y} \to \mathbb{R}^{d_x}$, we set

$$\|h\|_n \stackrel{\text{def}}{=} \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|h(\mathbf{y}_i)\|_2^2}.$$

Our goal is to derive a bound on $\|g_{\Lambda\odot W} - g\|_n$, when $(\Lambda, W) \sim \Pi(\cdot|\mathcal{D})$.

Theorem 8. *Assume H4-H6. Consider the nonparametric regression (3.7) for estimating g, where the function class $\{g_W,\ W \in \mathcal{W}\}$ is as defined in (3.11), and the regression variance parameter σ satisfies $\sigma \geq \bar{\sigma}$. Then we can construct a function class $\{H_W,\ W \in \mathcal{W}\}$, such that at unrolling depth D' that satisfies*

$$D' \gtrsim \frac{\log(n)}{-\log(\varrho_n)},$$

the posterior distribution $\Pi(\cdot|\mathcal{D})$ in (3.14) satisfies

$$\Pi\left(\|g_{\Lambda\odot W} - g\|_n > M\bar{\sigma}\frac{(D')^{1+\frac{\beta_2}{2}}}{n^{\frac{1}{2+\beta_2}}} \mid \mathcal{D}\right) \leq \frac{12}{q}, \tag{3.15}$$

with probability at least $1 - e^{-c_1 n} - \frac{c_1}{q}$, for some absolute constant c_1, and a constant $M \lesssim \log(q)\log(n)^{3/2}$.

Proof. See Section B.4.2. □

We make several remarks here.

(a) In contrast to common practice where D' is often chosen on an ad-hoc manner,

Theorem 8 recommends carefully scaling the depth parameter D' as

$$D' \sim -\log(n)/\log(\varrho_n),$$

for optimal performance. The result thus suggests that GDN can easily overfit as D' increases, a behavior that we observed in the simulations below.

(b) We can invoke classical deep learning approximation theory to conclude that Assumption 6 holds with $\beta_1 = \beta_2 = d_x$. In that case, up to log-terms, we deduce that GDN achieves the convergence rate

$$C_1 n^{-\frac{1}{2+d_x}}.$$

It is interesting to note that the function g of interest is at best Lipschitz under the conditions of Theorem 8[2]. Therefore the minimax rate in the estimation of g is given by

$$C_2 n^{-\frac{1}{2+d_y}}.$$

Hence, Theorem 8 implies that in inverse problems where d_y is larger than d_x, the unrolling framework by exploiting the forward model, achieves a better convergence rate than the minimax rate of estimating g from the data $\mathcal{D}$ in a nonparametric regression. However, it is important to nuance this conclusion. Firstly, the constants C_1, C_2 in the rates posted above depend on d_x and d_y in ways that are poorly understood. Another limitation of current minimax rates is the fact that deep learning models can often adapt to additional properties of the function of interest and converge much faster than the theoretical minimax rate. For instance Schmidt-Hieber (2020) shows that FNN models achieves faster rate in the estimation of compositional functions. We give a similar example below.

[2]Indeed, it can be easily shown that if for all $\mathbf{y} \in \mathsf{Y}$, $\mathbf{x} \mapsto f(\mathbf{y}|\mathbf{x})$ is strongly convex with strong convexity parameter $\underline{\kappa}$, and $\mathbf{x} \mapsto \nabla_{\mathbf{x}} f(\mathbf{y}|\mathbf{x})$ is $\bar{\kappa}$ Lipschitz then $\mathbf{y} \mapsto g(\mathbf{y})$ is L_g-Lipschitz with $L_g \leq 2\bar{\kappa}/\underline{\kappa}$

(c) The use of the empirical norm $\|u\|_n = \sqrt{\sum_{i=1}^n u(\mathbf{y}_i)^2}$ instead of the L^2 population norm of $\mathbf{y}$ in (3.15) is a fairly common practice in nonparametric estimation, that does not fundamentally change the resulting contraction rate. More technically, working in the L^2 norm amounts to the additional control of the term

$$\sup_{W \in \widetilde{W}^{(j)}} \left| n^{-1} \sum_{i=1}^n (g_W(\mathbf{y}_i) - g(\mathbf{y}_i))^2 - \|g_W - g\|_2^2 \right|, \tag{3.16}$$

in Lemma D.5. Because the sup in (3.16) is taken over well behaved sets $\widetilde{W}^{(j)}$, this uniform deviation can be controlled using standard tools as in , but would require additional assumptions on the marginal distribution of $\mathbf{y}$ that we wish to avoid making.

Application to sparse marginal distributions

We give another application of Theorem 8 where the posterior predictive function obtained from the GDN achieves the parametric rate. When dealing with images, several authors such as Beck and Teboulle (2010); Dong et al. (2011) have argued that natural image data are often sparse after linear transformation (such as a difference operators, or wavelet transforms), and suggested modeling the marginal distribution μ as

$$\mu(\mathrm{d}\mathbf{x}) = \frac{1}{c_\mu} e^{-\mathcal{R}_0(B\mathbf{x})} \mathrm{d}\mathbf{x}, \tag{3.17}$$

for some simple sparsity inducing function $\mathcal{R}_0$, and a non-singular matrix $B \in \mathbb{R}^{d_x \times d_x}$. In other words, $\mathcal{R}(\mathbf{x}) = \mathcal{R}_0(B\mathbf{x})$. A common choice is $\mathcal{R}_0(\mathbf{x}) = \lambda\|\mathbf{x}\|_1$, for some parameter $\lambda > 0$. If $\mathrm{Prox}^{\gamma\mathcal{R}_0}$ denote the proximal operator of $\mathcal{R}_0$, then by proximal calculus (see e.g. Lemma 2.8 of Combettes and Wajs (2005)), we have

$$\mathrm{Prox}^{\gamma\mathcal{R}}(\mathbf{x}) = B^{-1} \mathrm{Prox}^{\gamma\mathcal{R}_0}(B\mathbf{x}). \tag{3.18}$$

For example, given $\lambda_1 > 0, \lambda_2 \geq 0$, suppose that $\mathcal{R}_0$ is the elastic-net regularization prior of Zou and Hastie (2005b) given by

$$\mathcal{R}_0(\mathbf{x}) = \lambda_1 \|\mathbf{x}\|_1 + \frac{\lambda_2}{2}\|\mathbf{x}\|_2^2. \tag{3.19}$$

Then the proximal of $\gamma\mathcal{R}_0$ is $\text{Prox}^{\gamma\mathcal{R}_0}(\mathbf{x}) = (\mathsf{s}_\gamma(x_1), \cdots \mathsf{s}_\gamma(x_{d_x}))^\intercal$, where

$$\mathsf{s}_\gamma(x) = \mathsf{ReLU}\left(\frac{x - \gamma\lambda_1}{1 + \gamma\lambda_2}\right) - \mathsf{ReLU}\left(\frac{-x - \gamma\lambda_1}{1 + \gamma\lambda_2}\right),$$

and where $\mathsf{ReLU}(t) \stackrel{\text{def}}{=} \max(t, 0)$. Therefore, $\text{Prox}^{\gamma\mathcal{R}_0}(\mathbf{x})$ can be represented exactly using a 2-layer relu-neural network with layer sizes $(d_x, 2d_x, d_x)$, and $\text{Prox}^{\gamma\mathcal{R}}(\mathbf{x})$ can be represented exactly using a 4-layer ReLU-neural network with layer sizes $(d_x, d_x, 2d_x, d_x, d_x)$. Hence, H6 holds with depth $D = 4$, $\beta_1 = \beta_2 = 0$. Furthermore, since $\mathcal{R}$ is strongly convex, if we focus on the linear regression model and take the forward model as in (3.2), then H5 holds. Hence Theorem 8 yields the following.

Corollary 9. *Suppose that H4 holds where μ is as in (3.17), with $\mathcal{R}_0$ as in (3.19), and f is as in (3.2). Suppose also that $\sigma \geq \bar{\sigma}$. Then we can construct a deep learning function class $\{H_W,\ W \in \mathcal{W}\}$, with depth $D = 4$, such that at unrolling depth $D' \gtrsim -\log(n)/\log(\varrho_n)$ the posterior distribution $\Pi(\cdot|\mathcal{D})$ in (3.14) satisfies*

$$\Pi\left(\|g_{\Lambda\odot W} - g\|_n \geq \frac{M\bar{\sigma}D'}{\sqrt{n}}|\ \mathcal{D}\right) \leq \frac{12}{q},$$

with probability at least $1 - \frac{c_1}{q} - e^{-c_1 n}$, for some absolute constant c_1, where M depends on some log terms that we ignore.

3.3 A general posterior contraction result

Theorem 8 is derived as special cases of a more general result that we establish in this section. We consider again the regression model (3.7), where $\{g_W,\ W \in \mathcal{W}\}$ is some arbitrary deep neural network function class. We assume that the parameter space is $\mathsf{W} \stackrel{\text{def}}{=} \mathbb{R}^{\mathsf{p}_D \times \mathsf{p}_{D-1}} \times \cdots \times \mathbb{R}^{\mathsf{p}_1 \times \mathsf{p}_0}$, for some depth $D \geq 1$, and layer dimensions

$p_0, p_1, \ldots, p_D \geq 1$. As indicated at the end of the introduction, at times we shall view W as the Euclidean space $\mathbb{R}^q$, with Euclidean norm denoted $\|\cdot\|_2$, where

$$q \stackrel{\text{def}}{=} \sum_{\ell=1}^{D} (p_\ell \times p_{\ell-1}).$$

We make the following local Lipschitz assumption on the function class.

H 7. *For all $0 < \eta < \infty$, there exists $L(\eta) \geq 1$ such that for all $W, W' \in \mathcal{W}$ that satisfy $\max(\|W\|_2, \|W'\|_2) \leq \eta$, and for all $\mathbf{y} \in \mathsf{Y}$, we have*

$$\|g_W(\mathbf{y}) - g_{W'}(\mathbf{y})\|_2 \leq L(\eta)\|W - W'\|_2. \tag{3.20}$$

The constant $L(\eta)$ is a local Lipschitz constant of the function $W \mapsto g_W(\mathbf{y})$. Controlling appropriately these local Lipschitz constants is a major theoretical challenges in dealing with deep neural networks.

Theorem 10. *Suppose that the dataset $\mathcal{D}$ is generated as in H4, and consider the nonparametric regression (3.7) for some function class $\{g_W,\ W \in \mathcal{W}\}$ that satisfies H7. Suppose that the regression variance parameter σ satisfies $\sigma \geq \bar{\sigma}$. Let $\varpi_\star \geq 0$, $s_\star \geq 1$, $\beta_\star \geq 1$ be such that*

$$\min\ \{\|g_W - g\|_\infty,\ W \in \mathcal{W}\ \ s.t.\ \ \|W\|_0 \leq s_\star,\ \ \|W\|_\infty \leq \beta_\star\} \leq \varpi_\star,$$

and set $\mathsf{L}_\star \stackrel{\text{def}}{=} L(2s_\star^{1/2}\beta_\star)$, where the function L is as in H7. Suppose also that $q \geq n \geq 4$, and the prior parameter ρ_1 satisfies $\max(2, \beta_\star) \leq \rho_1^{-1} \leq \sqrt{n}\min(\sigma, \sqrt{n})/12$. Define

$$s \stackrel{\text{def}}{=} \left(1 + \frac{\log(n\mathsf{L}_\star)}{\mathsf{u}\log(q)}\right) s_\star + \frac{4n\varpi_\star^2}{\sigma^2 \mathsf{u}\log(q)},\quad \textit{and}\ \ \mathsf{r} \stackrel{\text{def}}{=} \bar{\sigma}\sqrt{\frac{\mathsf{s}\log(\mathsf{q}\mathsf{L}_\mathsf{s})}{\mathsf{n}}},$$

where

$$\mathsf{L}_s \stackrel{\text{def}}{=} L(s^{1/2}b_s),\quad \textit{with}\quad b_s \stackrel{\text{def}}{=} \rho_1^{-1}(\mathsf{u}+1)s\log(q).$$

We can find a constant $M^2 \geq \mathsf{u}\max((\sigma/\bar{\sigma})^2, 1)$, and absolute constants c_0, c_1 such that

$$\Pi\left(\|g_{\Lambda\odot W} - g\|_n > M\mathsf{r}\ |\mathcal{D}\right) \leq \frac{12}{q}, \tag{3.21}$$

with probability at least $1 - e^{-c_0 n} - c_1/q$.

Proof. See Section B.4.1. □

Remark 11. Theorem 10 applies well beyond the GDN of interest in this work. For any function class $\{g_W,\ W \in \mathcal{W}\}$ trained under the proposed sparse spike-and-slab prior, one can read off the posterior contraction rate of $\Pi(\cdot|\mathcal{D})$ from Theorem 10. The rate is driven by the local Lipschitz constant $L(\eta)$ of the function class, and the relationship between $(s_\star, \beta_\star)$ and $\varpi_\star$, which captures the approximation capability of the function class.

3.3.1 Sketch of the proof of Theorem 10

We start with a high-level description of the proof. Several approaches have been developed in the literature to study the contraction of posterior distributions. Here we follow an approach due to Shen and Wasserman (2001). The merit of their approach is that it makes a direct connection between the contraction properties of the posterior distribution and the properties of the corresponding log-likelihood empirical process.

Let $f,\ \{f_\theta,\ \theta \in \Theta\}$ be a family of densities on a measurable space $\mathcal{Z}$ equipped with a reference sigma-finite measure that we write as $\mathrm{d}z$. All densities considered on the sample space $\mathcal{Z}$ are defined with respect to $\mathrm{d}z$. The parameter space Θ is some arbitrary measurable space. Let π be a prior probability measure on Θ. We consider the posterior distribution of θ given by

$$\Pi(A|z) = \frac{\int_A f_\theta(z)\pi(\mathrm{d}\theta)}{\int_\Theta f_\theta(z)\pi(\mathrm{d}\theta)},\ \ A \text{ meas.},\ \ z \in \mathcal{Z}.$$

The next lemma is a generalization of Shen and Wasserman (2001), and summarizes the main arguments used in the proof of Theorem 10.

Lemma 12. *Let S, B and $\{\Xi_k,\ k \geq 1\}$ be measurable subsets of Θ, such that $S \cap B^c \subseteq \cup_{k\geq 1}\Xi_k$. Let $\beta > 0$, $\rho \geq 0$ and $\{r_j,\ j \geq 1\}$ a sequence of positive numbers. If $\mathcal{E}$ is a*

subset of $\mathcal{Z}$ *such that*

$$\mathcal{E} \subseteq \Bigg\{ z \in \mathcal{Z} : \int_\Theta \frac{f_\theta(z)}{f(z)} \pi(\mathrm{d}\theta) \geq e^{-\beta}, \quad \int_{S^c} \frac{f_\theta(z)}{f(z)} \pi(\mathrm{d}\theta) \leq \rho \\ \text{and} \quad \sup_{\theta \in \Xi_j} \left[\log f_\theta(z) - \log f(z) \right] \leq -r_j \quad \text{for all} \quad j \geq 1 \Bigg\}, \quad (3.22)$$

then

$$\Pi(B^c | Z) \leq e^\beta \left(\rho + \sum_{j \geq 1} e^{-r_j} \right), \quad (3.23)$$

with probability at least $\mathbb{P}(Z \in \mathcal{E})$.

Proof. Using the lower bound on the normalizing constant provided by the event (3.22), for $z \in \mathcal{E}$, we have

$$\Pi(B^c|z) = \frac{\int_{B^c} \frac{f_\theta(z)}{f(z)} \pi(\mathrm{d}\theta)}{\int_\Theta \frac{f_\theta(z)}{f(z)} \pi(\mathrm{d}\theta)} \leq e^\beta \left(\int_{S^c} \frac{f_\theta(z)}{f(z)} \pi(\mathrm{d}\theta) + \int_{S \cap B^c} \frac{f_\theta(z)}{f(z)} \pi(\mathrm{d}\theta) \right) \\ \leq e^\beta \left(\rho + \int_{S \cap B^c} \frac{f_\theta(z)}{f(z)} \pi(\mathrm{d}\theta) \right).$$

Furthermore, for $z \in \mathcal{E}$, the last integral in the last display satisfies

$$\int_{S \cap B^c} \frac{f_\theta(z)}{f(z)} \pi(\mathrm{d}\theta) \leq \sum_{j \geq 1} \int_{\Xi_j} \exp\left(\log f_\theta(z) - \log f(z) \right) \pi(\mathrm{d}\theta) \leq \sum_{j \geq 1} e^{-r_j} \pi(\Xi_j).$$

Equation (3.23) follows by collecting the terms. □

Remark 13. From the lemma we are left with the problem of finding $\rho, \beta, \{r_j,\ j \geq 1\}$ such that the right hand size of the last display is small and $\mathbb{P}(Z \notin \mathcal{E})$ is small.

3.4 Numerical illustration

We illustrate our theoretical results with a toy example, a simulated and a real data deblurring problem. For all examples we draw samples from the posterior distribution (3.14) using the Sparse Asynchronous Stochastic Gradient Langevin Dynamics

(SA-SGLD) sampler of Atchadé and Wang (2021), an approximate MCMC sampler designed for posterior distributions of the form (3.14), that employs asynchronicity for fast sampling. The algorithm is an approximate Metropolis-Hastings-within-Gibbs that alternates between an update of W using the Stochastic Gradient Langevin Dynamics of Welling and Teh (2011) while keeping Λ fixed, followed by an update of Λ with W fixed. The update of Λ is performed using a variation of the asynchronous Gibbs sampler of De Sa et al. (2016). For more details on the approximate correctness of the sampler we refer the reader to Atchadé and Wang (2021). Computationally the SA-SGLD sampler is implemented at the cost of 2 back-propagation through the GDN per MCMC iteration.

3.4.1 Learning the Elastic Net regression map

In this section, we evaluate the performance of GDN with a toy problem: learning to solve a linear regression model with the elastic-net regularization.

Data generation: We generate a dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i),\ 1 \leq i \leq n\}$ where $\mathbf{x}_i \overset{i.i.d.}{\sim} \mu$, and $\mathbf{y}_i|\mathbf{x}_i \sim \mathbf{N}(A\mathbf{x}_i, v^2\mathbf{I}_{d_y})$. We choose $\mu(\mathrm{d}\mathbf{x}) \propto e^{-\mathcal{R}_0(B\mathbf{x})}\mathrm{d}\mathbf{x}$ as in (3.17), where $\mathcal{R}_0$ is the elastic net density as in (3.19). We consider two scenarios: $B = \mathbf{I}_{d_x}$ (where the $\mathbf{x}_i$'s are drawn directly from the elastic net distribution), and $B \neq \mathbf{I}_{d_x}$ (where the $\mathbf{x}_i = B^{-1}\mathbf{z}_i$, and the $\mathbf{z}_i$'s are drawn from the elastic net distribution). In both simulations, we have $n = 200$, $d_x = d_y = 100$, and $\lambda_1 = 1$, $\lambda_2 = 1$. We refer the reader to Appendix B.1 for further details on the data generation.

Model architecture. When $B = \mathbf{I}_{d_x}$ we specify H_W as a FNN with depth $D = 2$, and layer $(p_0, p_1, p_2) = (d_x, 2d_x, d_x)$. In the second simulation where $B \neq \mathbf{I}_{d_x}$, we specify H_W as a FNN with depth $D = 4$, and layer $(p_0, p_1, p_2, p_3, p_4) = (d_x, d_x, 2d_x, d_x, d_x)$, where the first and last layers only contain weight matrices but no bias term. In order to prevent overfitting, we specify the models to only learn the parameters that connect the nodes between layers in each dimension of $\mathbf{x}_i$, which

reduces the number of model parameters in the first simulation to be $q = 7 * d_x = 700$ and second simulation to be $q = 100^2 + 7 * d_x + 100^2 = 20700$. We consider several values of the unrolling depth D': $D' = 1$ (GDN1), $D' = 5$ (GDN2), $D' = 10$ (GDN3), and $D' = 20$ (GDN4) for comparison. More details on the Bayesian prior and the MCMC parameters are provided in Section B.1.

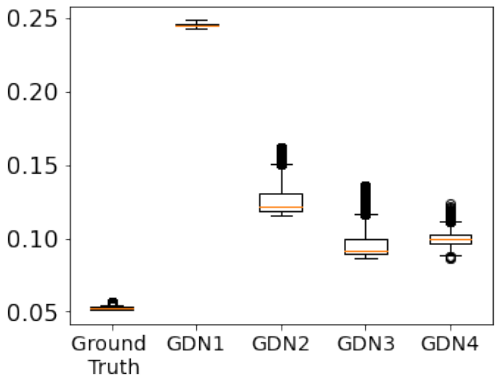
Evaluation procedure: For the comparison, we generate 1000 test samples and evaluate the prediction errors of the resulting estimator $g_{\Lambda \odot W}$, where $(\Lambda, W) \sim \Pi(\cdot|\mathcal{D})$. The performance of GDN are compared to the performance of g, which in this problem is easily calculated by proximal gradient descent. We do this comparison by computing the error

$$e(\Lambda, W) = \frac{1}{1000} \sum_{i=1}^{1000} \|g_{\Lambda \odot W}(\mathbf{y}_i) - g(\mathbf{y}_i)\|_2,$$

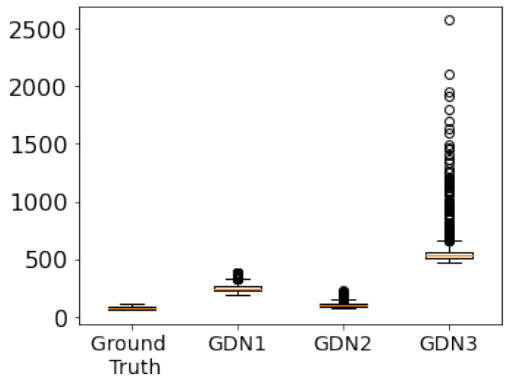
where the average is taking over the test sample. The boxplots in Figure 3·1 show the distributions of the errors obtained by taking 500 samples of (Λ, W) along the MCMC sampler. In this toy example, $D' = 10$ (GDN3) in simulation 1 and $D' = 5$ (GDN2) in simulation 2 yield the best results and GDN deteriorates slightly as the unrolling depth increases, as predicted by our theory.

3.4.2 Illustration with a simulated data deblurring problem

Image deblurring is a common inverse problem in computational imaging. Here for illustration, one can easily construct a data set $\mathcal{D}$ by blurring real images. In the first experiment we generate $n = 500$ images of size 16×16 (see Appendix B.2 for more details), that we then blurred using a Gaussian blurring convolution kernel with variance 3 without restriction on the kernel range. We fit the nonparametric regression model (3.7) with a GDN where H_W in (3.10) is taken as 3-layer relu-convolutional neural network. The total number of parameters is $q = 18,881$. We evaluate the



(a) Distributions of prediction errors on test samples: GDN1 (D'=1), GDN2 (D'=5) and GDN3 (D'=10), GDN4 (D'=20).



(b) Test loss distributions between different depth True elastic net function g, GDN1 (D'=1), GDN2 (D'=5) and GDN3 (D'=10).

Figure 3·1

GDN model at unrolling depth $D' = 2$ (GDN1), $D' = 4$ (GDN2), $D' = 12$ (GDN3) and $D' = 24$ (GDN4), and we do a comparison with two feedforward convolutional neural network (FNN) that do not make use of the forward problem. The first FNN has the same architecture as H_W (FNN1), while the second is a 6-layer FNN with total number of parameter $q = 136,641$ (FNN2). We refer to the Appendix B.2 for more details of the architectures of the neural networks.

We generate 500 test samples to evaluate the prediction errors of the six models. The boxplots in Figure 3·2 show the distribution of the mean square error of the last 2000 iterations of each model. Figure 3·3 shows an example of reconstruction from FNN1, FNN2, GDN1, and GDN3. We observe that GDN outperforms FNN1, and can achieve similar performance as FNN2 when the unrolling depth is appropriately selected (not too small, nor too large). The experiment again confirms our theoretical results.

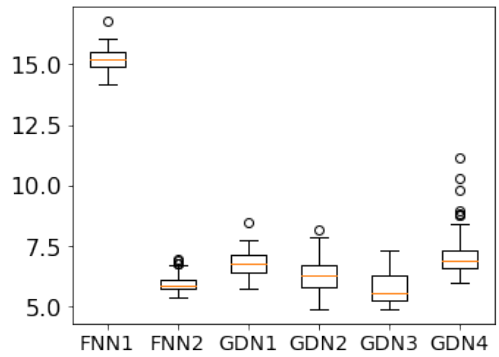


Figure 3·2: Test loss comparison between FNN1 (3 conv), FNN2 (3 conv + 1 cfc + 3 deconv), GDN1 ($D' = 2$), GDN2 ($D' = 4$), GDN3 ($D' = 12$) and GDN4 ($D' = 24$)

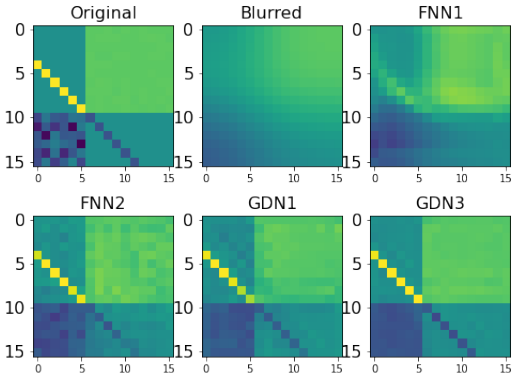


Figure 3·3: Reconstruction result from FNN1 (upper right), FNN2 (lower left), GDN1 $D' = 2$ (lower middle) and GDN3 $D' = 12$ (lower right)

3.4.3 Illustration with CelebA dataset

We extend the comparison to the deblurring of CelebA images Liu et al. (2015). We randomly select $20,000$ images from the celebA dataset that we resize to 64×64. The blurring is performed as above. More details are given in Section B.3. We fit the nonparametric regression model (3.7) with a GDN where H_W in (3.10) is taken as 3-layer relu-convolutional neural network (more details in Section B.3). The depth of the GDN is either $D' = 4$ (GDN1), $D' = 12$ (GDN2), or $D' = 24$ (GDN3). The

total number of parameters in the same in all three cases and equal to $q = 267,777$. We compare this model with a feedforward architecture that doe not make use of the forward model with 3 convolutional layers followed by 4 corresponding deconvolutional layers (more details in Section B.3). The total number of parameter of the FNN is $q = 2,271,297$. The rest of the implementation is similar to Section B.2.

Figure 3·4 shows the distributions of the test error from 500 drawn from the MCMC sampler after burn-in. We see again that at appropriate depth GDN matches FNN. However we see a decrease in performance at deeper depth $D' = 24$, which again suggests overfitting. Figure 3·5 shows three examples of reconstructed images.

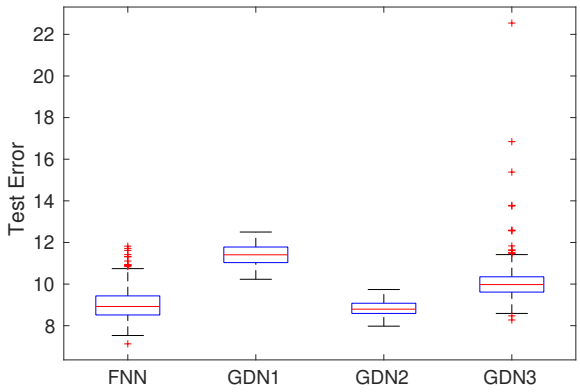


Figure 3·4: Test loss comparison between FNN (3 conv + 4 deconv), GDN1 (D'=4), GDN2 (D'=12) and GDN3 (D'=24)

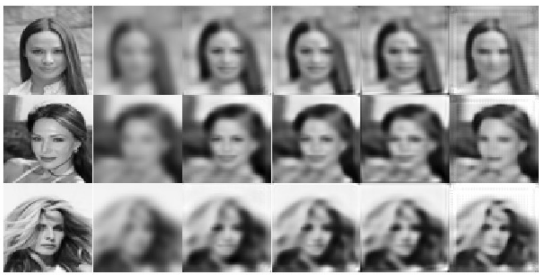


Figure 3·5: Some random examples of reconstructions. From left to right: CelebA image, blurred image, GDN1, GDN2, GDN3, and FNN

3.5 Concluding remarks

There is a need for a deeper theoretical understanding of algorithm unrolling models for inverse problems. We have shown in this work that for convex inverse problems and under a concentration of measure assumption, GDN can recover the inverse map at optimal rate, provided that the unrolling depth is appropriately tuned. These findings are confirmed in our numerical example. Our results also suggest that algorithm unrolling models are prone to overfitting as the unrolling depth D' increases. The theoretical results are obtained as special cases of a more general posterior contraction result for Bayesian deep learning.

Without the concentration of measure assumption in H4, one would need to estimate the entire conditional distribution, not just its mean. Several algorithm unrolling architectures have been proposed recently for conditional distribution estimation in inverse problems (Ardizzone et al. (2019)). Extending our work to these conditional density models is an important direction for future research.

Chapter 4

Minimum Distance Estimation from the view of Game Theory

4.1 Introduction

Many machine learning problems, at a fundamental level, can be cast as the following problem: we wish to estimate a probability distribution $\mu_\star$, by minimizing certain divergence $D(\cdot, \cdot)$ between the empirical distribution of the data and the parametric distribution family $\{\mu_\theta : \theta \in \Theta\}$. This is called the minimum distance estimation (MDE) problem. xA popular choice of divergence D is the Kullback-Leibler divergence. In that case, the problem will be formulated as maximum likelihood estimation. However, one drawback of using the KL-divergence is that the model density has to exist.

Generative Adversarial Networks (GANs) in Goodfellow et al. (2014) solve this problem by sampling from the model distribution and then minimizing the distance between two empirical distributions. Different types of divergence are proposed to stabilize the training process of GANs, including Wasserstein GAN in Arjovsky et al. (2017) and f-GAN in Nowozin et al. (2016). However, there is no theoretical guarantees for the existence of the equilibrium. Arora et al. (2017) proves the existence of an approximate equilibrium by generalizing GANs using neural net distance.

In this chapter, we borrow the idea of GANs by minimizing the distance of two empirical distributions. However, instead of specifying the divergence D, we gen-

eralize the problem over a class of divergences satisfying certain conditions, which re-formulates the minimum distance estimation into minimax problems. By allowing both players to play mixed strategies, we proof the existence our estimator in a new approach. A well known approach for proving the existence of mixed strategy is using Sion's minimax theorem. However, it requires the compactness assumption on action spaces for both players. Our proof circumvents these conditions by applying Schauder fixed-point theorem. We also show our estimator is an approximate mixed strategy equilibrium, by showing the rate of convergence of our estimator.

The chapter is organized as follows. In Section 4.2, we formulate the problem and our estimation distribution. In Section 4.3, we show the statistical properties of our estimator. In Section 4.4, we propose an algorithm to realize our estimator. In Section 4.5, we show the performance our algorithm by numerical experiments.

4.2 Problem formulation

We wish to estimate a probability distribution $\mu_\star$ supported on a set $\mathcal{X}$ that lies in a space $\mathbb{R}^{p_x}$, by minimizing certain divergence $D(\cdot,\cdot)$ between $\mu_\star$ and the parametric distribution family $\{\mu_\theta : \theta \in \Theta\}$, which can be written as

$$\min_{\theta\in\Theta} D(\mu_\star, \mu_\theta).$$

If $D(\cdot,\cdot)$ falls under certain class such as integral probability metrics (IPMs) (Müller (1997)), whose specific instances include the Wasserstein distance and the maximum mean discrepancy (MMD), we can rewrite $D(\cdot,\cdot)$ into the following variational form: for all probability distributions μ, μ' supported on $\mathcal{X}$, we have

$$D(\mu,\mu') = \sup_{l\in\mathcal{L}} \int_{\mathcal{X}} \int_{\mathcal{X}} l(x,y)\ \mu(\mathrm{d}x)\mu'(\mathrm{d}y),$$

where the function class $\mathcal{L}$ depends on $D(\cdot,\cdot)$. For example, for Wasserstein distance, $\mathcal{L}$ can be the class of Lipschitz functions, and for total variation, $\mathcal{L}$ can be all bounded functions. For standard GAN (Goodfellow et al. (2014)), we can let $\mathcal{L} = \{\log D_0(x) + \log(1 - D_0(y))\}$.

In this work, for computational tractability, we would like to maximize over a parameterized function class $\{l_\phi : \phi \in \Phi\} \subset \mathcal{L}$, which approximates the maximization over $\mathcal{L}$. Then we have $D(\mu, \mu') = \max_{\phi\in\Phi} \int_{\mathcal{X}} l_\phi(x, y)\ \mu(\mathrm{d}x)\mu'(\mathrm{d}y)$ and the MDE problem in (4.2) can be written in the following minimax form:

$$\min_{\theta\in\Theta} \max_{\phi\in\Phi} \left\{ \ell_\star(\theta, \phi) \stackrel{\text{def}}{=} \int_{\mathcal{X}} \int_{\mathcal{X}} l_\phi(x, y)\ \mu_\star(\mathrm{d}x)\mu_\theta(\mathrm{d}y) \right\}.$$

And if the supremum in (4.2) can be attained at some $\phi \in \Phi$ for all probability distributions μ, μ', then solving (4.2) is same as solving (4.2).

Since the definition of $\ell_\star(\theta, \phi)$ involves multi-dimensional integrals with respect to distributions $\mu_\star$ and μ_θ, it typically has no closed form. However, if we can draw samples from $\mu_\star$ and μ_θ, then we can define the empirical version of $\ell_\star(\theta, \phi)$ as

$$\ell_{n,m}(\theta, \phi) \stackrel{\text{def}}{=} \int_{\mathcal{X}} \int_{\mathcal{X}} l_\phi(x, y)\ \mu_n(\mathrm{d}x)\mu_{\theta,m}(\mathrm{d}y), \quad \mu_n \stackrel{\text{def}}{=} \tfrac{1}{n}\textstyle\sum_{i=1}^n \delta_{X_i},\ \ \mu_{\theta,m} \stackrel{\text{def}}{=} \tfrac{1}{m}\textstyle\sum_{i=1}^m \delta_{V_i},$$

where $\{X_i\}_{i=1}^n$ and $\{V_i\}_{i=1}^m$ are i.i.d. samples drawn from $\mu_\star$ and μ_θ, respectively. Based on $\ell_{n,m}(\theta, \phi)$, we can solve the following empirical version of the minimax problem in (4.2):

$$\min_{\theta\in\Theta} \max_{\phi\in\Phi} \ell_{m,n}(\theta, \phi). \tag{4.1}$$

The formulation in (4.2) and (4.1) is also known as two-player zero-sum game with reward function $\ell_\star$ and $\ell_{m,n}$ respectively . The solution (saddle point) of minimax problems is the same as Nash equilibrium of two-player zero-sum game. Note that generative adversarial networks (GANs) in Goodfellow et al. (2014) is an application for such problem, where it is minimizing the distance between the generator $\mu_{\theta,m}$,

that is generating fake samples via trained neural network with input as Gaussian noise, and the discriminator l_ϕ, that is assessing the distribution distance between $\mu_{\theta,m}$ and μ_n.

4.2.1 A Bayesian Perspective of MDE

In most cases, the minimax problem is usually non-convex (in θ) and non-concave (in ϕ). The solutions of such minimax problems may not exist. Hence we allow the players θ and ϕ to play a mixed strategy in Θ and Φ respectively. Naturally we can extend the minimax problem in (4.1) into the Bayesian framework.

As in all Bayesian analysis we start with the prior density. Let Θ be a subset of $\mathbb{R}^{p_g}$ equipped with the Euclidean norm and Lebesgue measure $\mathrm{d}\theta$. Let π_g be the prior density (with respect to $\mathrm{d}\theta$) on Θ. Similarly, let Φ be a subset of $\mathbb{R}^{p_d}$ equipped with the Euclidean norm and Lebesgue measure $\mathrm{d}\phi$. The only work that we are aware of on Bayesian GAN is Saatchi and Wilson (2017) which proceeded as follows. The authors set up a Gibbs sampler that iteratively sample θ from $\Pi(\theta|\phi, \mathbf{X}) \propto \pi_g(\theta)e^{\ell_g(\theta,\phi)}$, and then sample ϕ from $\Pi(\phi|\theta, \mathbf{X}) \propto \pi_d(\phi)e^{\ell_d(\theta,\phi)}$. The issue with this approach is that, put together these two conditional distributions do not in general define a valid joint distribution (Arnold et al. (2001)). This conceptual shortcoming of (Saatchi and Wilson (2017)) was also noted by Christian Robert in a comment posted on his blog[1].

Here we propose a formal definition of Bayesian GAN that doesn't have this issue, using ideas from fictitious game theory (Perkins and Leslie (2014)).

Let $\mathcal{P}(\Theta)$ be the space of all probability measures on Θ that are absolutely continuous with respect to π_g. We convexify the target function $\ell_\star$ by considering the

[1] https://xianblog.wordpress.com/2018/06/26/bayesian-gan-gan-style/

map $\bar{\ell}_\star : \mathcal{P}(\Theta) \times \mathcal{P}(\Phi) \to \mathbb{R}$ defined as

$$\bar{\ell}_\star(\mu, \nu) \stackrel{\text{def}}{=} \int \ell_\star(\theta, \phi)\mu(\mathrm{d}\theta)\nu(\mathrm{d}\phi).$$

Similarly we convexify $\ell_{n,m}$ to be $\bar{\ell}_{n,m}$. Now we focus on the following minimax problems:

$$\min_{\nu_g \in \mathcal{P}(\Theta)} \max_{\nu_d \in \mathcal{P}(\Phi)} \bar{\ell}_\star(\nu_g, \nu_d) \tag{4.2}$$

A solution to this problem is called a mixed strategy Nash equilibrium.

Similar as in the frequentist setting, $\ell_\star(\theta, \phi)$ is intractable. However, if we can draw samples from $\mu_\star$ and μ_θ, we introduce the following estimator. We define the map $M : \mathcal{P}(\Theta) \times \mathcal{P}(\Phi) \to \mathcal{P}(\Theta) \times \mathcal{P}(\Phi)$ as follows. If the joint distribution $\Pi \in \mathcal{P}(\Theta) \times \mathcal{P}(\Phi)$ has θ-marginal Π_g and ϕ-marginal Π_d, then $M(\Pi)(\mathrm{d}\theta, \mathrm{d}\phi)$ is defined as $M_g(\Pi)(\mathrm{d}\theta)M_d(\Pi)(\mathrm{d}\phi)$, where M_g and M_d satisfy that

$$M_g(\Pi)(\mathrm{d}\theta) \propto \pi_g(\theta) \exp\left(\int_\Phi -\sigma_n \ell_{n,m}(\theta, v)\Pi_d(\mathrm{d}v)\right)\mathrm{d}\theta,$$
$$M_d(\Pi)(\mathrm{d}\phi) \propto \pi_d(\phi) \exp\left(\int_\Theta \sigma_n \ell_{n,m}(u, \phi)\Pi_g(\mathrm{d}u)\right)\mathrm{d}\phi, \tag{4.3}$$

and $\sigma_n > 0$ is a scaling parameter.

Definition 4.2.1. *We define the minimum empirical distance estimator (MEDE) posterior distribution of (θ, ϕ) as any fixed points of the map M.*

We define a penalized version of the empirical reward function

$$L_{n,m}(\mu, \nu) \stackrel{\text{def}}{=} \sigma_n \bar{\ell}_{n,m}(\mu, \nu) + \mathsf{KL}\left(\mu|\pi_g\right) - \mathsf{KL}\left(\nu|\pi_d\right), \quad \mu \in \mathcal{P}(\Theta),\ \nu \in \mathcal{P}(\Phi), \tag{4.4}$$

we will have the following property:

Proposition 4.2.1. *An element $\Pi = (\Pi_g, \Pi_d)$ is a mixed strategy Nash equilibrium with reward function $L_{n,m}(\mu, \nu)$ if and only if Π is a MEDE posterior distribution, i.e. a fixed point of the map M.*

Proof. By definition $\Pi = (\Pi_g, \Pi_d)$ is a mixed strategy Nash equilibrium, means that

$$\Pi_g = \underset{\mu \in \mathcal{P}(\Theta)}{\mathsf{Argmin}}\, L_{n,m}(\mu, \Pi_d) = \underset{\mu \in \mathcal{P}(\Theta)}{\mathsf{Argmin}}\, \sigma_n \bar{\ell}_{n,m}(\mu, \Pi_d) + \mathsf{KL}\left(\mu | \pi_g\right) = M_g(\Pi),$$

and similarly for Π_d. □

The above proposition says that the MEDE posterior distribution is a mixed strategy Nash equilibrium for a two-player zero-sum game with penalized reward function. In the following section, we will show the statistical properties (existence, rate of convergence) for our MEDE posterior distribution.

4.3 Statistical Properties

If we assume that $\Theta \subseteq \mathbb{R}^{p_g}$ is compact and equipped with its Lebesgue measure. Let $\mathcal{M}(\Theta)$ be the space of all finite signed measures on Θ. Then $\mathcal{P}(\Theta)$ is a compact and closed subset of $\mathcal{M}(\Theta)$ under the weak* topology by Banach-Alaoglu. Hence, by Sion's minimax theorem, if $\Theta \subseteq \mathbb{R}^{p_g}$ is compact, the solution of problem (4.2) exists, as well as our MEDE posterior distribution of the map M. However, in Section 4.3.1, we show the existence of our MEDE posterior distribution without the compactness assumptions on Θ, but instead with some mild conditions on $\ell_{n,m}$ and prior distributions, using Schauder's fixed-point theorem. In Section 4.3.2, we will introduce a metric to evaluate the performance of an estimator. And we will use this metric to show the rate of convergence for our estimator with on additional boundedness assumption on Φ.

4.3.1 Existence

The assumptions needed for proving existence of our estimator is as follows.

H 4.3.1. *We assume $\mathcal{X}$ is compact. We also assume that the reward function $\ell_{n,m}$ is bounded by 1 and L-Lipschitz w.r.t. ϕ and θ without loss of generality.*

H 4.3.2. *We put an assumption on the prior distributions $\pi_g(\theta)$ and $\pi_d(\phi)$ that there exist $a \geq \sigma_n L$ and $b \geq 0$ such that $\|\nabla_\theta \log \pi_g(\theta)\|_2 \leq a + b\|\theta\|_2$ and $\|\nabla_\phi \log \pi_d(\phi)\|_2 \leq a + b\|\phi\|_2$. And we also assume $\int_\Theta (1 + \|\theta\|_2) \exp(b\|\theta\|_2)|\pi_g(\theta)\, d\theta < \infty$ and $\int_\Phi (1 + \|\phi\|_2) \exp(b\|\phi\|_2)|\pi_d(\phi)\, d\phi < \infty$.*

Remark 4.3.1. *In H.4.3.2, we assume that $\|\nabla_\theta \log \pi_g(\theta)\|_2 \leq a + b\|\theta\|_2$. This condition is more general than the Lipschitz condition on $\nabla_\theta \log \pi_g(\theta)$. For example, when $\nabla_\theta \log \pi_g(\theta) = \sqrt{\theta}, \theta \in \mathbb{R}$, it doesn't satisfy the Lipschitz condition, but it satisfies our assumption.*

Lemma 14. *Under H.4.3.1, M is Lipschitz continuous with respect to total variation distance.*

Lemma 15. *Under H.4.3.1 and H.4.3.2, $M(\mathcal{P}(\Theta) \times \mathcal{P}(\Phi))$ is totally bounded.*

Theorem 16. *Under H.4.3.1 and H.4.3.2, there exists at least one MEDE posterior distribution of map M.*

Proof. We apply Schauder fixed-point theorem to the map M. Clearly, $\mathcal{P}(\Theta) \times \mathcal{P}(\Phi)$ is a closed convex subset of the Banach space equipped with TV norm. It suffices to show that M is continuous and the $M(\mathcal{P}(\Theta) \times \mathcal{P}(\Phi))$ is totally bounded. We can obtain the result from Lemma 14 and 15.

□

4.3.2 Rate of Convergence

In this section, we introduce a metric (duality gap function) to evaluate the performance of MEDE posterior distribution. We show that the MEDE is an approximate mixed strategy Nash equilibrium under this metric.

Let $g : \mathcal{X} \times \mathsf{Y} \to \mathbb{R}$. $(x_\star, y_\star)$ is a Nash equilibrium for g (a zero-sum game with reward $-g$ for x and g for y) if for all $x \in \mathcal{X}$, $y \in \mathsf{Y}$, it holds

$$g(x_\star, y) \leq g(x_\star, y_\star) \leq g(x, y_\star).$$

Define

$$\mathsf{Gap}(x, y) = \max_v g(x, v) - \min_u g(u, y).$$

Proposition 17. *$(x_\star, y_\star)$ is a Nash equilibrium for g if and only if $\mathsf{Gap}(x_\star, y_\star) = 0$. In which case we have*

$$g(x_\star, y_\star) = \min_x \max_y g(x, y) = \max_y \min_x g(x, y). \tag{4.5}$$

Furthermore, if $\mathcal{X}$ and Y are compact sets and g is continuous in both of its arguments, then any point $(x_\star, y_\star)$ that satisfies (4.5) is a Nash equilibrium.

According to the above proposition, we may make use of the Gap function to measure the distance to Nash equilibrium. To leverage this metric in our case, we focus on $\mathsf{Gap} : \mathcal{P}(\Theta) \times \mathcal{P}(\Phi) \to \mathbb{R}^+$ with respect to our target function $\bar{\ell}_\star$ and defined as below:

$$\mathsf{Gap}(\mu, \nu) = \max_{\pi_2 \in \mathcal{P}(\Phi)} \bar{\ell}_\star(\mu, \pi_2) - \min_{\pi_1 \in \mathcal{P}(\Theta)} \bar{\ell}_\star(\pi_1, \nu).$$

By this function, we introduce approximate mixed strategy Nash equilibrium as follows. We prove that our estimator is an approximate mixed strategy, and show the rate of convergence of our estimator under this metric.

Definition 4.3.1 ((ϵ, B)-approximate mixed strategy Nash equilibrium)**.** *A probability measure Π with θ and ϕ marginals Π_g and Π_d is an (ϵ, B)-approximate mixed strategy Nash equilibrium with reward function $\bar{\ell}_\star$ if it satisfies the following condition:*

$$\widetilde{\mathsf{Gap}}_B(\Pi_g, \Pi_d) \stackrel{\text{def}}{=} \max_{\pi_2 \in \mathcal{P}(\Phi): \mathsf{KL}(\pi_2 | \pi_d) \le B} \bar{\ell}_\star(\Pi_g, \pi_2) - \min_{\pi_1 \in \mathcal{P}(\Theta): \mathsf{KL}(\pi_1 | \pi_g) \le B} \bar{\ell}_\star(\pi_1, \Pi_d) < \epsilon.$$

H 4.3.3. *We assume Φ is bounded. We say that for any $\phi \in \Phi$, $\|\phi\|_2 \le R$.*

Theorem 18. *Without loss of generality, we assume that $n = m$. Under H.4.3.3 and the assumption that the MEDE posterior of map M exists, the MEDE posterior distribution of (θ, ϕ) given X is an $(C\sqrt{n}^{-1}(\sqrt{p_2}+t), C\sigma_n\sqrt{n}^{-1}(\sqrt{p_2}+t))$-approximate mixed strategy Nash equilibrium for some constant C, with probability at least $1 - 2\exp(-t^2)$, for $t > 0$.*

The above theorem gives the rate of convergence of our MEDE posteior distribution. It also gives a suggestion on how to choose σ_n. If $\sigma_n \gg \sqrt{n}$, the constrained *Gap* function will become unconstrained when $n \to \infty$.

4.3.3 Sparsified version

In this section, we consider high-dimensional version of GAN. Here introduce the sparsified version on Φ. We introduce parameter δ to denote the $\|\phi\|_0$ and impose prior distribution also on δ. Thus we define the prior distribution on $\phi_\delta = \langle \phi, \delta \rangle$ as $\pi_d(\phi_\delta) = \sum_{\{\phi:\phi_\delta = \langle \phi, \delta \rangle\}} \pi_0(\delta)\pi_1(\phi)$. For convenience, we denote the product prior as $\bar{\pi}(\theta, \phi_\delta) = \pi_g(\theta)\pi_d(\phi_\delta) = \sum_{\{\phi:\phi_\delta = \langle \phi, \delta \rangle\}} \pi_g(\theta)\pi_0(\delta)\pi_1(\phi)$.

We denote $\Phi_\Delta = \{\phi_\delta : \phi \in \Phi, \delta \in \Delta\}$. Let $\mathcal{P}(\Theta \times \Phi_\Delta)$ denote the space of all probability measure on $\Theta \times \Phi_\Delta$ that are absolutely continuous with respect to $\bar{\pi}$, equipped with the total variation metric:

$$\|\Pi_1 - \Pi_2\|_{TV} = \int_{\Theta \times \Phi_\Delta} |f_1(\theta, \phi_\delta) - f_2(\theta, \phi_\delta)| \bar{\pi}(\mathrm{d}\theta \mathrm{d}\phi_\delta). \tag{4.6}$$

We introduce the map $M : \mathcal{P}(\Theta \times \Phi_\Delta) \to \mathcal{P}(\Theta \times \Phi_\Delta)$ as follows. If $\Pi \in \mathcal{P}(\Theta \times \Phi_\Delta)$ with θ-marginal Π_g and ϕ_δ-marginal Π_d, then $M(\Pi)(\mathrm{d}\theta, \mathrm{d}\phi_\delta) = M_g(\Pi)(\mathrm{d}\theta) M_d(\Pi)(\mathrm{d}\phi_\delta)$, where

$$M_g(\Pi)(\mathrm{d}\theta) \propto \pi_g(\theta) \exp\left(\int_{\Phi_\Delta} -\sigma_n \ell_{n,m}(\theta, v) \Pi_d(\mathrm{d}v) \right) \mathrm{d}\theta, \tag{4.7}$$

$$M_d(\Pi)(\mathrm{d}\phi_\delta) \propto \pi_d(\phi_\delta) \exp\left(\int_{\Theta} \sigma_n \ell_{n,m}(u, \phi_\delta) \Pi_g(\mathrm{d}u) \right) \mathrm{d}\phi_\delta. \tag{4.8}$$

Similarly, we put exact the same hypothesis on π_d for ϕ_δ instead of ϕ. Then we will have the same theorem for existence.

Theorem 19. *Under H.4.3.1 and H.4.3.2, there exists at least one MEDE posterior distribution.*

Next, we will introduce theorem on Nash equilibrium. We want to reduce the bound, thus we introduce the following sparsity assumption on Φ_Δ,

H 4.3.4. *We assume that $\pi_0(\delta)$ choose δ such that $\|\delta\|_0 \leq s$.*

Theorem 20. *Without loss of generality, we assume that $n = m$. Under H.4.3.1, H.4.3.2, H.4.3.3 and H.4.3.4, the MEDE posterior distribution of (θ, ϕ_δ) given X is an $(C\sqrt{n}^{-1}(s\sqrt{\log p_2} + t), C\sigma_n\sqrt{n}^{-1}(s\sqrt{\log p_2} + t))$-approximate mixed strategy Nash equilibrium for some constant C, with probability at least $1 - 2\exp(-t^2)$, for $t > 0$.*

4.4 Algorithm

To find MEDE posterior distribution, e.g. the fixed points of the map M or the mixed strategy Nash Equilibrium with the reward function $L_{n,m}(\mu, \nu)$, can be a very hard problem. Given a posterior distribution $(\Pi_g(\cdot|\mathbf{Z}), \Pi_d(\cdot|\mathbf{Z}))$, a harder question is how to sample from it. As apparent in (4.3), the difficulty is the recursive nature of the map M, and the intractability of the integrals involved.

Now we introduce a new approximation for the map M in (4.3) by unrolling $M_d(\Pi)$. Let SGLD_θ^L consists in unrolling L iterations of SGLD on the density

$$\phi \mapsto \pi_d(\phi) \exp\left(\sigma_n \ell_{n,m}(\theta, \phi)\right), \tag{4.9}$$

starting from initial distribution ν_0. Here we are assuming $\ell_{n,m}(\theta, \phi)$ is close to $\ell_{n,m}(\Pi_g^\star, \phi)$ for $\theta \sim \Pi_g^\star$, where $\Pi^\star = (\Pi_g^\star, \Pi_d^\star)$ is our MEDE estimator. In that case, we are saying that $M_d(\Pi^\star) = \Pi_d^\star$ can be approximated by SGLD_θ^L. Now the approximated posterior distribution $M_g(\Pi^\star) = \Pi_g^\star$ for θ can be expressed as

$$\tilde{\Pi}_g(\mathrm{d}\theta) \propto \pi_g(\theta) \exp\bigg(- \sigma_n \ell_{n,m}(\theta, \mathsf{SGLD}_\theta^L)\bigg)\mathrm{d}\theta. \tag{4.10}$$

4.5 Numerical Analysis

4.5.1 log-Normal random variables

In this section, we would like to estimate the parameter of the distribution of the sum of log-Normal random variables, which doesn't have analytical formula for probability

Algorithm 1 Unrolling

At the k-th iteration,

1. Sample n_0 initial points ϕ^0 from ν_0 and run L iterations of SGLD starting from ϕ^0 with invariant distribution (4.9), denoted by ϕ^L. Use ϕ^L to approximate SGLD_θ^L.

2. Sample $\theta_k \sim \tilde{\Pi}_g$ by SGLD starting from θ_{k-1}, with invariant distribution (4.10) by substituting SGLD_θ^L using ϕ^L.

density function. Let $\{X_i\}_{i=1}^L \sim N(\theta, 1)$, and $Y = \sum_{i=1}^L \exp(x_i)$. Then Y is a sum of log-normal random variable. We generate n samples $\{Y_i\}_{i=1}^n$ from this distribution with $\theta_\star = 1$. We would like to estimate θ using our algorithm with Wasserstein-1 metric. In that case, we can use a neural network l_ϕ with weights ϕ lying in a compact sapce Φ to approximate the maximization over Lipschitz function class as in Arjovsky et al. (2017). For the experiment, we let $L = 5, n = 5000$.

To form the posterior distribution, we set $\sigma_n = 2n$. To generate ϕ^L, we set $n_0 = 10$, $L = 60$ and the step size of SGLD to be 2×10^{-6}. We use a neural network with 2 hidden layers, whose first layer is size 5 and second layer is size 3, and tanh as activation function. To generate θ_k, we use SGLD with step size 10^{-5}. Below is trace plot for θ when running our algorithm for 100 iterations.

4.6 Conclusion

In this chapter, we propose a new framework on minimum distance estimation from the view of game theory. We have shown the statistical properties of our estimator including the existence and rate of convergence. However, the optimization error of the algorithm remains unsolved. We will explore it in the future work.

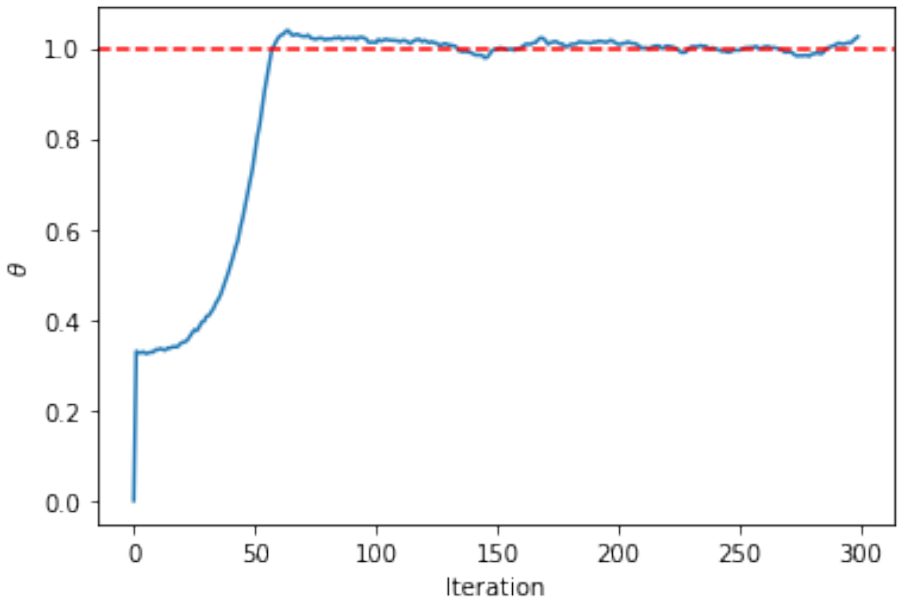


Figure 4·1: θ_k sampled from posterior distribution (4.10)

4.7 Proofs

4.7.1 Proof of Existence of MEDE posterior distribution

Lemma 21 (Schauder's fixed point theorem (Schauder (1930))). *Let A be an nonempty closed convex subset of a Banach space and assume there exists a compact map $T : A \to A$ (i.e. T is continuous and maps bounded subsets to relatively compact sets). Then T has fixed points.*

Lemma 22 (Kolmogorov-Riesz (Hanche-Olsen and Holden (2010))). *A subset $\mathcal{F}$ of $L^1(\mathbb{R}^n)$ is totally bounded if, and only if,*

1. *$\mathcal{F}$ is bounded,*
2. *for every $\epsilon > 0$ there is some R such that, for every $f \in \mathcal{F}$,*

$$\int_{\|x\|_\infty > R} |f(x)|\,dx < \epsilon,$$

3. *for every $\epsilon > 0$ there is some $\rho > 0$ so that, for every $f \in \mathcal{F}$, and $y \in \mathbb{R}^n$ with*

$\|y\|_\infty < \rho$,

$$\int_{\mathbb{R}^n} |f(x+y) - f(x)| dx < \epsilon. \tag{4.11}$$

Proof of Lemma 14

Proof. Since $M(\Pi)(\mathrm{d}\theta, \mathrm{d}\phi) = M_g(\Pi)(\mathrm{d}\theta)M_d(\Pi)(\mathrm{d}\phi)$, it suffices to show that $M_g(\Pi)(\mathrm{d}\theta)$ is Lipschitz continuous w.r.t. TV distance.

For any $\Pi^1, \Pi^2 \in \mathcal{P}(\Theta) \times \mathcal{P}(\Phi)$, their marginal distribution Π^1_d, Π^2_d on Θ should be absolute continuous with respect to $\pi_d(\theta)$.

Note

$$\begin{aligned}
\Delta \overset{\text{def}}{=} & \|M_g(\Pi^1) - M_g(\Pi^2)\|_{TV} && (4.12)\\
= & \int_\Theta \left| \pi_g(\theta) \exp\left(\int_\Phi -\sigma_n \ell_{n,m}(\theta, v) \Pi^1_d(\mathrm{d}v) \right) / z_1 \right. && (4.13)\\
& \left. - \pi_g(\theta) \exp\left(\int_\Phi -\sigma_n \ell_{n,m}(\theta, v) \Pi^2_d(\mathrm{d}v) \right) / z_2 \right| \mathrm{d}\theta, && (4.14)
\end{aligned}$$

where z_1, z_2 is the corresponding normalizing constant.
Define $\eta_t(\theta) = \pi_g(\theta) \exp\left(t \int_\Phi -\sigma_n \ell_{n,m}(\theta, v) \Pi^1_d(\mathrm{d}v) + (1-t) \int_\Phi -\sigma_n \ell_{n,m}(\theta, v) \Pi^2_d(\mathrm{d}v)\right) / z_t$ for $0 \leq t \leq 1$. Here z_t is the corresponding normalizing constant, i.e. $z_t = \int_\Theta \pi_g(\theta) \exp\left(t \int_\Phi -\sigma_n \ell_{n,m}(\theta, v) \Pi^1_d(\mathrm{d}v) + (1-t) \int_\Phi -\sigma_n \ell_{n,m}(\theta, v) \Pi^2_d(\mathrm{d}v) \right) \mathrm{d}\theta$.

Thus

$$\begin{aligned}
\frac{\mathrm{d} \log z_t}{\mathrm{d}t} = & \frac{\mathrm{d}z_t}{\mathrm{d}t} / z_t && (4.15)\\
= & \int_\Theta \left[\int_\Phi -\sigma_n \ell_{n,m}(\theta, v) \left(\Pi^1_d(\mathrm{d}v) - \Pi^2_d(\mathrm{d}v) \right) \right] \eta_t(\theta) \mathrm{d}\theta && (4.16)\\
\leq & \sigma_n \|\ell_{n,m}(\theta, v)\|_\infty \|\Pi^1_d - \Pi^2_d\|_{TV} && (4.17)
\end{aligned}$$

The second equality holds by dominated convergence theorem, which assures the

exchangeability of integral and derivative. This is true because

$$\begin{aligned}
&\Big|\big[\int_\Phi -\sigma_n \ell_{n,m}(\theta,v)\big(\Pi_d^1(\mathrm{d}v)-\Pi_d^2(\mathrm{d}v)\big)\big]\\
&\exp\big(t\int_\Phi -\sigma_n\ell_{n,m}(\theta,v)\Pi_d^1(\mathrm{d}v)+(1-t)\int_\Phi -\sigma_n\ell_{n,m}(\theta,v)\Pi_d^2(\mathrm{d}v)\big)\Big|\\
\leq&\Big|\big[\int_\Phi -\sigma_n \ell_{n,m}(\theta,v)\big(\Pi_d^1(\mathrm{d}v)-\Pi_d^2(\mathrm{d}v)\big)\big]\\
&\exp\big(\int_\Phi l-\sigma_n\ell_{n,m}(\theta,v)\Pi_d^1(\mathrm{d}v)\big)+\exp\big(\int_\Phi -\sigma_n\ell_{n,m}(\theta,v)\Pi_d^2(\mathrm{d}v)\big)\Big|
\end{aligned}$$

by Jensen's Inequality.

Then

$$\begin{aligned}
\Delta =& \int_\Theta |\eta_1(\theta)-\eta_0(\theta)|\mathrm{d}\theta && (4.18)\\
=& \int_\Theta |\int_0^1 \frac{\mathrm{d}}{\mathrm{d}t}\eta_t(\theta)\mathrm{d}t|\mathrm{d}\theta && (4.19)\\
=& \int_\Theta \Big|\int_0^1 \Big[\frac{\mathrm{d}}{\mathrm{d}t}\log\eta_t(\theta)\Big]\eta_t(\theta)\mathrm{d}t\Big|\mathrm{d}\theta && (4.20)\\
=& \int_\Theta \Big|\int_0^1 \Big[\int_\Phi -\sigma_n\ell_{n,m}(\theta,v)\big(\Pi_d^1(\mathrm{d}v)-\Pi_d^2(\mathrm{d}v)\big)-\frac{\mathrm{d}\log z_t}{\mathrm{d}t}\Big]\eta_t(\theta)\mathrm{d}t\Big|\mathrm{d}\theta && (4.21)\\
\leq& \int_\Theta\int_0^1 \Big[\Big|\int_\Phi -\sigma_n\ell_{n,m}(\theta,v)\big(\Pi_d^1(\mathrm{d}v)-\Pi_d^2(\mathrm{d}v)\big)\Big|+\Big|\frac{\mathrm{d}\log z_t}{\mathrm{d}t}\Big|\Big]\eta_t(\theta)\mathrm{d}t\mathrm{d}\theta && (4.22)\\
\leq& \int_\Theta\int_0^1 \sigma_n\Big[\|\ell_{n,m}(\theta,v)\|_\infty\|\Pi_d^1-\Pi_d^2\|_{TV}+\|-\sigma_n\ell_{n,m}(\theta,v)\|_\infty\|\Pi_d^1-\Pi_d^2\|_{TV}\Big]\\
&\qquad\qquad \eta_t(\theta)\mathrm{d}t\mathrm{d}\theta && (4.23)\\
=& 2\sigma_n\|\ell_{n,m}(\theta,v)\|_\infty\|\Pi_d^1-\Pi_d^2\|_{TV}\int_0^1\int_\Theta \eta_t(\theta)\mathrm{d}\theta\mathrm{d}t && (4.24)\\
=& 2\sigma_n\|\ell_{n,m}(\theta,v)\|_\infty\|\Pi_d^1-\Pi_d^2\|_{TV} && (4.25)
\end{aligned}$$

The last second equality is due to Tonelli's theorem. Thus M_g is Lipschitz continuous since $\ell_{n,m}$ is bounded by 1 according to H 4.3.1.

Proof is similar for M_d function. Together, we can show that M is Lipschitz continuous w.r.t. TV distance.

□

Appendix A

Supplemental Materials to Chapter 2

A.1 Proof of Theorem 1

Throughout the proof c_0 denotes a generic absolute constant that depends only on $\underline{\kappa}$ and $\bar{\kappa}$ in H3, but whose actual value or expression may change during the text. We recall that $\Delta_s \stackrel{\text{def}}{=} \{\delta \in \Delta : \ \|\delta\|_0 \leq s\}$, and for $\delta \in \Delta_s$, we let

$$\mathsf{B}_\delta \stackrel{\text{def}}{=} \left\{ \theta \in \mathbb{R}^p : \ \left\| \frac{\theta_\delta \theta_\delta^{\mathrm{T}}}{\|\theta_\delta\|_2^2} - \theta_\star \theta_\star^{\mathrm{T}} \right\|_{\mathsf{F}} \leq M\epsilon \right\},$$

and B_δ^c its complement in $\mathbb{R}^p$. We then set

$$\mathsf{B} \stackrel{\text{def}}{=} \bigcup_{\delta \in \Delta_s} \{\delta\} \times \mathsf{B}_\delta, \quad \text{and} \quad \mathsf{A} \stackrel{\text{def}}{=} \bigcup_{\delta \in \Delta_s} \{\delta\} \times \mathsf{B}_\delta^c.$$

Clearly we have $\Delta_s \times \mathbb{R}^p = \mathsf{B} \cup \mathsf{A}$. Hence our objective is to establish that $\Pi(\mathsf{A}|\mathbf{Z})$ is small. From the definition of $\Pi(\cdot|\mathbf{Z})$, for any measurable subset C of $\Delta_s \times \mathbb{R}^p$, by integrating out the non-selected component $\theta - \theta_\delta$, we have

$$\Pi(C|\mathbf{Z}) \tag{A.1}$$

$$= \frac{\sum_{\delta \in \Delta_s} e^{\mathsf{a}\|\delta\|_0} \int_{\mathbb{R}^p} \mathbf{1}_C(\delta, \theta)\,(\theta) \exp\left(-\frac{\rho_1}{2}\|\theta_\delta\|_2^2 - \frac{\rho_0}{2}\|\theta - \theta_\delta\|_2^2 + \sigma_n \mathsf{R}_n(\theta_\delta; \mathbf{Z})\right) \mathrm{d}\theta}{\sum_{\delta \in \Delta_s} e^{\mathsf{a}\|\delta\|_0} \int_{\mathbb{R}^p} \exp\left(-\frac{\rho_1}{2}\|\theta_\delta\|_2^2 - \frac{\rho_0}{2}\|\theta - \theta_\delta\|_2^2 + \sigma_n \mathsf{R}_n(\theta_\delta; \mathbf{Z})\right) \mathrm{d}\theta} \tag{A.2}$$

$$= \frac{\sum_{\delta \in \Delta_s} \left(\frac{1}{p^{\mathsf{u}}} \sqrt{\frac{\rho_1}{2\pi}}\right)^{\|\delta\|_0} \int_{\mathbb{R}^{\|\delta\|_0}} \mathbf{1}_C(\delta, (u,0)_\delta) \exp\left(-\frac{\rho_1}{2}\|u\|_2^2 + \sigma_n \bar{\mathsf{R}}_n((u,0)_\delta; \mathbf{Z})\right) \mathrm{d}u}{\sum_{\delta \in \Delta_s} \left(\frac{1}{p^{\mathsf{u}}} \sqrt{\frac{\rho_1}{2\pi}}\right)^{\|\delta\|_0} \int_{\mathbb{R}^{\|\delta\|_0}} \exp\left(-\frac{\rho_1}{2}\|u\|_2^2 + \sigma_n \bar{\mathsf{R}}_n((u,0)_\delta; \mathbf{Z})\right) \mathrm{d}u}, \tag{A.3}$$

where

$$\bar{\mathsf{R}}_n(\theta;\mathbf{Z}) \stackrel{\text{def}}{=} \mathsf{R}_n(\theta;\mathbf{Z}) - \mathsf{R}_n(\theta_\star;\mathbf{Z}).$$

Here $\mathbf{1}_C(\delta,\theta)$ denotes an indicator function of set C.

We show in Lemma 23 that the denominator on the right hand side of (A.1) is bound from below by

$$\varpi \stackrel{\text{def}}{=} e^{-(s_\star+1)(\mathsf{u}+1)\log(p\vee(c_0\sigma_n))},$$

for some absolute constant c_0. This implies that

$$\Pi(\mathsf{A}_1|\mathbf{Z}) \le \frac{1}{\varpi} \times \sum_{\delta\in\Delta_s}\left(\frac{1}{p^{\mathsf{u}}}\sqrt{\frac{\rho_1}{2\pi}}\right)^{\|\delta\|_0}\int_{\mathbb{R}^{\|\delta\|_0}} \mathbf{1}_{\mathsf{B}_\delta^c}((u,0)_\delta)\exp\left(-\frac{\rho_1}{2}\|u\|_2^2+\sigma_n\bar{\mathsf{R}}_n((u,0)_\delta;\mathbf{Z})\right)\mathrm{d}u. \tag{A.4}$$

We show in Lemma 26 that any $\theta\in\mathbb{R}^p$, such that $\|\theta\|_0\le s$,

$$\mathsf{R}_n(\theta;\mathbf{Z}) - \mathsf{R}_n(\theta_\star;\mathbf{Z}) \le -\frac{\mathsf{gap}}{2}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2\|\theta\theta^{\mathrm{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}^2 + c_0\mathsf{r}_1\|\theta\theta^{\mathrm{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}},$$

for some absolute constant c_0 that depends only on $\underline{\kappa}$ and $\bar{\kappa}$. Therefore, for $\frac{4c_0\mathsf{r}_1}{\mathsf{gap}}\left(\frac{\bar{\kappa}}{\underline{\kappa}}\right)^2 \le \|\theta\theta^{\mathrm{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}$, we have

$$\mathsf{R}_n(\theta;\mathbf{Z}) - \mathsf{R}_n(\theta_\star;\mathbf{Z}) \le -\frac{\mathsf{gap}}{4}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2\|\theta\theta^{\mathrm{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}^2.$$

Therefore, for $M \geq 4c_0 \left(\frac{\bar{\kappa}}{\underline{\kappa}}\right)^2$, (A.4) becomes

$$\begin{aligned}
&\Pi(\mathsf{A}_1|\mathbf{Z}) \\
&\leq \frac{1}{\varpi} e^{-\frac{M^2 \mathsf{gap}}{4}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \sigma_n \epsilon^2} \sum_{\delta \in \Delta_s} \left(\frac{1}{p^{\mathsf{u}}}\sqrt{\frac{\rho_1}{2\pi}}\right)^{\|\delta\|_0} \int_{\mathbb{R}^{\|\delta\|_0}} \mathbf{1}_{\mathsf{B}_\delta}((u,0)_\delta) \exp\left(-\frac{\rho_1}{2}\|u\|_2^2\right) \mathrm{d}u \\
&\leq \frac{1}{\varpi} e^{-\frac{M^2 \mathsf{gap}}{4}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \sigma_n \epsilon^2} \sum_{\delta \in \Delta_s} \left(\frac{1}{p^{\mathsf{u}}}\sqrt{\frac{\rho_1}{2\pi}}\right)^{\|\delta\|_0} \left(2\pi\rho_1^{-1}\right)^{\|\delta\|_0/2} \\
&\leq 2e^{(s_\star+1)(\mathsf{u}+1)\log(p \vee (c_0\sigma_n))} e^{-\frac{M^2 \mathsf{gap}}{4}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \sigma_n \epsilon^2} \leq 2e^{-\frac{M^2}{8\mathsf{gap}}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \sigma_n \mathsf{r}_1^2},
\end{aligned}$$

under the sample size condition (2.11), where the third inequality follows from the assumptions $\mathsf{u} > 1$, and $p^{\mathsf{u}-1} > 2$. This proves the theorem.

□

We derive here a lower bound on the normalizing constant of the quasi-posterior distribution.

Lemma 23. *Under Assumption H3, and assuming that $p^{\mathsf{u}} \geq e^1 s_\star$, we have*

$$\begin{aligned}
\sum_{\delta \in \Delta_s} \left(\frac{1}{p^{\mathsf{u}}}\sqrt{\frac{\rho_1}{2\pi}}\right)^{\|\delta\|_0} \int_{\mathbb{R}^{\|\delta\|_0}} \exp\left(-\frac{\rho_1}{2}\|u\|_2^2 + \sigma_n \bar{\mathsf{R}}_n((u,0)_\delta; \mathbf{Z})\right) \mathrm{d}u \\
\geq e^{-(s_\star+1)(\mathsf{u}+1)\log(p \vee (c_0\sigma_n))}, \quad \text{(A.5)}
\end{aligned}$$

for some absolute constant c_0.

Proof. Clearly, the left hand side of (A.5) is bounded from below by

$$\left(\frac{1}{p^{\mathsf{u}}}\sqrt{\frac{\rho_1}{2\pi}}\right)^{s_\star} \int_{\mathbb{R}^{s_\star}} \exp\left(-\frac{\rho_1}{2}\|u\|_2^2 + \sigma_n \bar{\mathsf{R}}_n((u,0)_{\delta_\star}; \mathbf{Z})\right) \mathrm{d}u.$$

For any $\theta \in \mathbb{R}^p$ that has the same support as $\theta_\star$, we have

$$\begin{aligned}
\bar{\mathsf{R}}_n(\theta; \mathbf{Z}) &= \frac{\theta^{\mathrm{T}} \hat{A} \theta}{\theta^{\mathrm{T}} \hat{B} \theta} - \frac{\theta_\star^{\mathrm{T}} \hat{A} \theta_\star}{\theta_\star^{\mathrm{T}} \hat{B} \theta_\star} \\
&= \frac{\theta^{\mathrm{T}} \hat{\Sigma} \theta}{\theta^{\mathrm{T}} \hat{B} \theta} - \frac{\theta_\star^{\mathrm{T}} \hat{\Sigma} \theta_\star}{\theta_\star^{\mathrm{T}} \hat{B} \theta_\star} \\
&= \frac{\theta^{\mathrm{T}} \hat{\Sigma} \theta \left(\theta_\star^{\mathrm{T}} \hat{B} \theta_\star - \theta^{\mathrm{T}} \hat{B} \theta \right)}{(\theta_\star^{\mathrm{T}} \hat{B} \theta_\star)(\theta^{\mathrm{T}} \hat{B} \theta)} + \frac{1}{\theta_\star^{\mathrm{T}} \hat{B} \theta_\star} \left(\theta^{\mathrm{T}} \hat{\Sigma} \theta - \theta_\star^{\mathrm{T}} \hat{\Sigma} \theta_\star \right).
\end{aligned}$$

Since $\mathsf{R}_n(\cdot; \mathbf{Z})$ is invariant to rescaling, we can assume without any loss of generality that $\|\theta\|_2 = \|\theta_\star\|_2 = 1$. Therefore for $\mathbf{Z}$ satisfying H3-(1),we have from Lemma 25

$$\left| \bar{\mathsf{R}}_n(\theta; \mathbf{Z}) \right| \leq \left(2 \left(\frac{\bar{\kappa}}{\underline{\kappa}} \right)^2 + 2 \left(\frac{\bar{\kappa}}{\underline{\kappa}} \right) \right) \|\theta\theta^{\mathrm{T}} - \theta_\star \theta_\star^{\mathrm{T}}\|_{\mathsf{F}} \leq 4 \left(\frac{\bar{\kappa}}{\underline{\kappa}} \right)^2 \|\theta\theta^{\mathrm{T}} - \theta_\star \theta_\star^{\mathrm{T}}\|_{\mathsf{F}}. \quad \text{(A.6)}$$

It follows from the above observations that for $\mathbf{Z}$ satisfying H3 the left hand side of (A.5) is bounded from below by

$$\begin{aligned}
\left(\frac{1}{p^{\mathsf{u}}} \sqrt{\frac{\rho_1}{2\pi}} \right)^{s_\star} \int_{\mathbb{R}^{s_\star}} \exp\left(-\frac{\rho_1}{2} \|u\|_2^2 - \frac{C}{2} \sigma_n \|uu^{\mathrm{T}} - \theta_\star \theta_\star^{\mathrm{T}}\|_{\mathsf{F}} \right) \mathrm{d}u & \\
\geq \left(\frac{1}{p^{\mathsf{u}}} \sqrt{\frac{\rho_1}{2\pi}} \right)^{s_\star} e^{-C\eta^2 \sigma_n} \int_{\mathcal{S}_0} e^{-\frac{\rho_1}{2}\|u\|_2^2} \mathrm{d}u, &
\end{aligned}$$

where $C = 8(\bar{\kappa}/\underline{\kappa})^2$, $\eta \in (0,1)$ and $\mathcal{S}_0 \stackrel{\text{def}}{=} \{u \in \mathbb{R}^{s_\star} : \ \|uu^{\mathrm{T}} - \theta_\star \theta_\star^{\mathrm{T}}\|_{\mathsf{F}} \leq 2\eta^2\}$. Note that the integral $\int_{\mathcal{S}_0} e^{-\frac{\rho_1}{2}\|u\|_2^2} \mathrm{d}u$ is invariant to change of variables by orthogonal matrices. Hence in that integral we can replace $\theta_\star$ by the unit vector $e = (0, \ldots, 0, 1) \in \mathbb{R}^{s_\star}$. Using this and switching to polar coordinates, we write the integral as

$$\int_{\mathcal{S}_0} e^{-\frac{\rho_1}{2}\|u\|_2^2} \mathrm{d}u = \int_0^{+\infty} e^{-\frac{\rho_1}{2} r^2} r^{s_\star - 1} \mathrm{d}r \times \nu\left(\theta \in \mathcal{S}^{s_\star - 1} : \ |\sin(\theta)| \leq \eta \right),$$

where ν is the surface measure on the unit sphere $\mathcal{S}^{s_\star - 1} = \{u \in \mathbb{R}^{s_\star} : \ \|u\|_2 = 1\}$, and $\sin(\theta)$ is the sine of the angle between θ and e. The measure $\nu\left(\theta \in \mathcal{S}^{s_\star - 1} : \ |\sin(\theta)| \leq \eta \right)$ is equal to twice the spherical cap around the pole e defined by η. We use the formula

of the spherical cap from (Li (2011)) to write

$$\nu\left(\theta \in \mathcal{S}^{s_\star-1} : \ |\sin(\theta)| \leq \eta\right) = \frac{4\pi^{\frac{s_\star-1}{2}}}{\Gamma\left(\frac{s_\star-1}{2}\right)} \int_0^{\arcsin(\eta)} \sin^{s_\star-2}(\theta)\mathrm{d}\theta$$
$$= \frac{4\pi^{\frac{s_\star-1}{2}}}{\Gamma\left(\frac{s_\star-1}{2}\right)} \int_0^{\eta} \frac{x^{s_\star-2}}{\sqrt{1-x^2}}\mathrm{d}x \geq \frac{4\pi^{\frac{s_\star-1}{2}}}{\Gamma\left(\frac{s_\star-1}{2}\right)} \frac{\eta^{s_\star-1}}{s_\star-1}.$$

Whereas,

$$\int_0^{+\infty} e^{-\frac{\rho_1}{2}r^2} r^{s_\star-1}\mathrm{d}r = \frac{1}{2}\left(\frac{2}{\rho_1}\right)^{\frac{s_\star}{2}} \Gamma\left(\frac{s_\star}{2}\right).$$

It follows that

$$\int_{\mathcal{S}_0} e^{-\frac{\rho_1}{2}\|u\|_2^2}\mathrm{d}u \geq \frac{2}{s_\star\sqrt{\pi}}\left(\frac{2\pi}{\rho_1}\right)^{\frac{s_\star}{2}} \eta^{s_\star-1}.$$

We conclude that for $\mathbf{Z}$ satisfying H3, and any $\eta \in (0,1)$, the left hand side of (A.5) is bounded from below by

$$\frac{2}{\sqrt{\pi}s_\star}\left(\frac{1}{p^{\mathsf{u}}}\right)^{s_\star} e^{-(s_\star-1)\log(1/\eta)} e^{-C\eta^2\sigma_n} \geq e^{-(\mathsf{u}+1)(1+s_\star)\log(p\vee C\sigma_n)},$$

by taking $\eta = 1/\sqrt{C\sigma_n}$, and assuming that $p^{\mathsf{u}} \geq e^1 s_\star$. This concludes the proof. □

We make use of the following version of the Davis-Kahan $\sin\Theta$ theorem taken from Vu and Lei (2013) Lemma 4.2.

Lemma 24. *Let A be a $p \times p$ symmetric semipositive definite matrix and suppose that its eigenvalues satisfies $\lambda_1(A) > \lambda_2(A) \geq \ldots \geq \lambda_p(A)$. If a unit vector u is an eigenvector of A associated to the largest eigenvalue $\lambda_1(A)$, for all $v \in \mathbb{R}^p$, $\|v\|_2 = 1$ it holds*

$$\langle A, uu' - vv'\rangle \geq \frac{1}{2}\left(\lambda_1(A) - \lambda_2(A)\right)\|uu' - vv'\|_F^2.$$

We will need the following technical result.

Lemma 25. *For any unit vectors u, v and square matrix B with matching dimensions, we have*

$$|\langle B, uu^{\mathrm{T}} - vv^{\mathrm{T}}\rangle| \leq 2\|B\|_{op}\|uu^{\mathrm{T}} - vv^{\mathrm{T}}\|_F, \tag{A.7}$$

Proof. Indeed, we have

$$|\langle B, uu^{\mathrm{T}} - vv^{\mathrm{T}}\rangle| = |(u-v)^{\mathrm{T}}Bu + v^{\mathrm{T}}B(u-v)| \leq 2\|B\|_{\mathsf{op}}\|u-v\|_2.$$

Similarly, we have $|\langle B, uu^{\mathrm{T}} - vv^{\mathrm{T}}\rangle| \leq 2\|B\|_{\mathsf{op}}\|u+v\|_2$. Hence

$$|\langle B, uu^{\mathrm{T}} - vv^{\mathrm{T}}\rangle| \leq 2\|B\|_{\mathsf{op}} \min\left(\|u-v\|_2, \|u+v\|_2\right).$$

The result follows by noting that

$$\|uu^{\mathrm{T}} - vv^{\mathrm{T}}\|_{\mathsf{F}} \geq \min\left(\|u-v\|_2, \|u+v\|_2\right). \tag{A.8}$$

To see this, note that $\|uu^{\mathrm{T}} - vv^{\mathrm{T}}\|_{\mathsf{F}} = \|u-v\|_2\|u+v\|_2/\sqrt{2} = \|u-v\|_2\sqrt{2 - \|u-v\|_2^2/2}$. Hence, if $\|u-v\|_2^2 \leq 2$, then we have $\|uu^{\mathrm{T}} - vv^{\mathrm{T}}\|_{\mathsf{F}} \geq \|u-v\|_2$. But if $\|u-v\|_2^2 > 2$ then $\|uu^{\mathrm{T}} - vv^{\mathrm{T}}\|_{\mathsf{F}} > \|u+v\|_2$. Hence the result. □

The next result describes the behavior of the Rayleigh quotient function that yields the posterior contraction result.

Lemma 26. *Assume H3. For any $\theta \in \mathbb{R}^p$ such that $\|\theta\|_0 \leq s$, we have*

$$R_n(\theta; \mathbf{Z}) - R_n(\theta_\star; \mathbf{Z}) \leq -\frac{\mathsf{gap}}{2}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \|\theta\theta^{\mathrm{T}} - \theta_\star\theta_\star^{\mathrm{T}}\|_F^2 + c_0 \mathsf{r}_1 \|\theta\theta^{\mathrm{T}} - \theta_\star\theta_\star^{\mathrm{T}}\|_F. \tag{A.9}$$

Proof. Fix $\theta \in \mathbb{R}^p$ such that $\|\theta\|_0 \leq s$. Since the Rayleigh quotient is invariant under rescaling we can assume without any loss of generality that $\|\theta\|_2 = 1$. We have

$$\begin{aligned}\bar{\mathsf{R}}_n(\theta; \mathbf{Z}) = \mathsf{R}_n(\theta; \mathbf{Z}) - \mathsf{R}_n(\theta_\star; \mathbf{Z}) &= \frac{\theta^{\mathrm{T}}\hat{\Sigma}\theta}{\theta^{\mathrm{T}}\hat{B}\theta} - \frac{\theta_\star^{\mathrm{T}}\hat{\Sigma}\theta_\star}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star} = \frac{\theta^{\mathrm{T}}\Sigma\theta}{\theta^{\mathrm{T}}B\theta} - \frac{\theta_\star^{\mathrm{T}}\Sigma\theta_\star}{\theta_\star^{\mathrm{T}}B\theta_\star} \\ &+ \left\langle \hat{\Sigma} - \Sigma, \frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta} - \frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star} \right\rangle + \left\langle \hat{\Sigma}, \left[\frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}\hat{B}\theta} - \frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}\right] - \left[\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star} - \frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right] \right\rangle.\end{aligned} \tag{A.10}$$

Set $S \stackrel{\text{def}}{=} B^{-1/2}\Sigma B^{-1/2}$, $w = B^{1/2}\theta/\|B^{1/2}\theta\|_2$, $w_\star = B^{1/2}\theta_\star/\|B^{1/2}\theta_\star\|_2$, and note that $w_\star$ is an eigenvector of S associated to the largest eigenvalue of S. Hence by the curvature lemma (Lemma 24) we have

$$\frac{\theta^{\mathrm{T}}\Sigma\theta}{\theta^{\mathrm{T}}B\theta} - \frac{\theta_\star^{\mathrm{T}}\Sigma\theta_\star}{\theta_\star^{\mathrm{T}}B\theta_\star} = \langle S, ww^{\mathrm{T}} - w_\star w_\star^{\mathrm{T}}\rangle \leq -\frac{\mathsf{gap}}{2}\|ww^{\mathrm{T}} - w_\star w_\star^{\mathrm{T}}\|_{\mathsf{F}}^2.$$

Let $\mathsf{I} \subseteq \{1, \ldots, p\}$ be the joint support of θ and $\theta_\star$ (hence $\|\mathsf{I}\|_0 \leq s + s_\star$). Then we

can express

$$\|ww^{\mathrm{T}} - w_\star w_\star^{\mathrm{T}}\|_{\mathsf{F}} = \left\| (B_{\mathsf{I},\mathsf{I}})^{1/2} \left(\frac{\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}}}{\theta_{\mathsf{I}}^{\mathrm{T}}(B_{\mathsf{I},I})\theta_{\mathsf{I}}} - \frac{\theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}}{\theta_{\star\mathsf{I}}^{\mathrm{T}}(B_{\mathsf{I},\mathsf{I}})\theta_{\star\mathsf{I}}} \right) (B_{\mathsf{I},\mathsf{I}})^{1/2} \right\|_{\mathsf{F}}.$$

We recall that for any square matrix A and invertible matrix B,

$$\|A\|_{\mathsf{F}} = \|B^{-1/2}B^{1/2}AB^{1/2}B^{-1/2}\|_{\mathsf{F}} \leq \|B^{-1/2}\|_{\mathsf{op}}^2 \|B^{1/2}AB^{1/2}\|_{\mathsf{F}},$$

where $\|M\|_{\mathsf{op}}$ denotes the operator norm of M. With these observations in mind, we get

$$\begin{aligned}\|ww^{\mathrm{T}} - w_\star w_\star^{\mathrm{T}}\|_{\mathsf{F}} &\geq \frac{1}{\|(B_{\mathsf{I},\mathsf{I}})^{-1/2}\|_{\mathsf{op}}^2} \left\| \frac{\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}}}{\theta_{\mathsf{I}}^{\mathrm{T}}(B_{\mathsf{I},I})\theta_{\mathsf{I}}} - \frac{\theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}}{\theta_{\star\mathsf{I}}^{\mathrm{T}}(B_{\mathsf{I},\mathsf{I}})\theta_{\star\mathsf{I}}} \right\|_{\mathsf{F}} \\ &\geq \underline{\kappa} \left\| \frac{\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}}}{\theta_{\mathsf{I}}^{\mathrm{T}}(B_{\mathsf{I},\mathsf{I}})\theta_{\mathsf{I}}} - \frac{\theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}}{\theta_{\star\mathsf{I}}^{\mathrm{T}}(B_{\mathsf{I},\mathsf{I}})\theta_{\star\mathsf{I}}} \right\|_{\mathsf{F}}.\end{aligned}$$

We note also that for any unit vectors u, v and symmetric invertible matrix B with matching dimension,

$$\left\| \frac{uu^{\mathrm{T}}}{u^{\mathrm{T}}Bu} - \frac{vv^{\mathrm{T}}}{v^{\mathrm{T}}Bv} \right\|_{\mathsf{F}}^2 = \frac{(u^{\mathrm{T}}Bu - v^{\mathrm{T}}Bv)^2}{(u^{\mathrm{T}}Bu)^2(v^{\mathrm{T}}Bv)^2} + \frac{\|uu^{\mathrm{T}} - vv^{\mathrm{T}}\|_{\mathsf{F}}^2}{(u^{\mathrm{T}}Bu)(v^{\mathrm{T}}Bv)} \geq \frac{\|uu^{\mathrm{T}} - vv^{\mathrm{T}}\|_{\mathsf{F}}^2}{(u^{\mathrm{T}}Bu)(v^{\mathrm{T}}Bv)}. \quad \text{(A.11)}$$

Hence, under H3,

$$\|ww^{\mathrm{T}} - w_\star w_\star^{\mathrm{T}}\|_{\mathsf{F}}^2 \geq \left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \|\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}} - \theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}\|_{\mathsf{F}}^2 = \left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \|\theta\theta^{\mathrm{T}} - \theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}^2.$$

In conclusion we have

$$\frac{\theta^{\mathrm{T}}\Sigma\theta}{\theta^{\mathrm{T}}B\theta} - \frac{\theta_\star^{\mathrm{T}}\Sigma\theta_\star}{\theta_\star^{\mathrm{T}}B\theta_\star} \leq -\frac{\mathsf{gap}}{2}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2 \|\theta\theta^{\mathrm{T}} - \theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}^2. \quad \text{(A.12)}$$

The second term from (A.10) can be written as

$$\begin{aligned}
&\left|\left\langle \hat{\Sigma}-\Sigma, \frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\rangle\right| \\
&\qquad = \left|\left\langle (\hat{\Sigma})_{\mathsf{I},\mathsf{I}}-\Sigma_{\mathsf{I},\mathsf{I}}, \frac{\frac{\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}}{\left\|\frac{\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\|_{\mathsf{F}}}\right\rangle\right| \left\|\frac{\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\|_{\mathsf{F}} \\
&\qquad \leq \max_{M\in\mathbb{R}^{\mathsf{I}\times\mathsf{I}}:\ \|M\|_{\mathsf{F}}=1,\ \mathsf{Rank}(M)\leq 2} \left|\left\langle (\hat{\Sigma})_{\mathsf{I},\mathsf{I}}-\Sigma_{\mathsf{I},\mathsf{I}}, M\right\rangle\right| \times \left\|\frac{\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\|_{\mathsf{F}}.
\end{aligned}$$

And we note from (A.11) and Lemma 25 that for $\mathbf{Z}$ satisfying H3,

$$\begin{aligned}
\left\|\frac{\theta_{\mathsf{I}}\theta_{\mathsf{I}}^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_{\star\mathsf{I}}\theta_{\star\mathsf{I}}^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\|_{\mathsf{F}}^2 &\leq \frac{1}{\underline{\kappa}^4}\left\langle B_{\mathsf{II}}, \theta_{\mathsf{I}}\theta_{\mathsf{I}}^T-\theta_{\star,\mathsf{I}}\theta_{\star,\mathsf{I}}^{\mathrm{T}}\right\rangle^2+\frac{1}{\underline{\kappa}^2}\left\|\theta_{\mathsf{I}}\theta_{\mathsf{I}}^T-\theta_{\star,\mathsf{I}}\theta_{\star,\mathsf{I}}^{\mathrm{T}}\right\|_{\mathsf{F}}^2 \\
&\leq \frac{1}{\underline{\kappa}^2}\left(1+2\left(\frac{\bar{\kappa}}{\underline{\kappa}}\right)^2\right)\|\theta\theta^T-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}^2. \quad \text{(A.13)}
\end{aligned}$$

Therefore for $\mathbf{Z}$ satisfying H3,

$$\left|\left\langle \hat{\Sigma}-\Sigma, \frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\rangle\right| \leq c_0\mathsf{r}_1\|\theta\theta^{\mathsf{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}. \quad \text{(A.14)}$$

We process the last term in (A.10) as follows.

$$\begin{aligned}
&\left\langle \hat{\Sigma}, \left[\frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}\hat{B}\theta}-\frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}\right]-\left[\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star}-\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right]\right\rangle \\
&\qquad = \frac{\theta^{\mathrm{T}}\hat{\Sigma}\theta}{\theta^{\mathrm{T}}\hat{B}\theta}\left\langle B-\hat{B}, \frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}\right\rangle-\frac{\theta_\star^{\mathrm{T}}\hat{\Sigma}\theta_\star}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star}\left\langle B-\hat{B}, \frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\rangle \\
&\quad = \left(\frac{\theta^{\mathrm{T}}\hat{\Sigma}\theta}{\theta^{\mathrm{T}}\hat{B}\theta}-\frac{\theta_\star^{\mathrm{T}}\hat{\Sigma}\theta_\star}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star}\right)\left\langle B-\hat{B}, \frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}\right\rangle+\frac{\theta_\star^{\mathrm{T}}\hat{\Sigma}\theta_\star}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star}\left\langle B-\hat{B}, \frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\rangle. \quad \text{(A.15)}
\end{aligned}$$

Hence for $\mathbf{Z}$ satisfying H3, the first term in the last display can be bounded, similar

to (A.6), as

$$\left|\left(\frac{\theta^{\mathrm{T}}\hat{\Sigma}\theta}{\theta^{\mathrm{T}}\hat{B}\theta}-\frac{\theta_\star^{\mathrm{T}}\hat{\Sigma}\theta_\star}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star}\right)\left\langle B-\hat{B},\frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}\right\rangle\right| \\ \leq \frac{1}{\underline{\kappa}}\lambda_{\mathsf{max}}(\hat{B}-B,s)\left[4\left(\frac{\bar{\kappa}}{\underline{\kappa}}\right)^2\|\theta\theta^{\mathrm{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}\right] \\ \leq \frac{4}{\underline{\kappa}}\lambda_{\mathsf{max}}(\hat{B}-B,s)\left(\frac{\bar{\kappa}}{\underline{\kappa}}\right)^2\|\theta\theta^{\mathrm{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}.$$

The rightmost of (A.15) is similar to (A.14):

$$\left|\frac{\theta_\star^{\mathrm{T}}\hat{\Sigma}\theta_\star}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star}\left\langle B-\hat{B},\frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}-\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right\rangle\right| \leq c_0\mathsf{r}_1\|\theta\theta^{\mathsf{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}.$$

In conclusion the last term in (A.10) is bounded from above by

$$\left|\left\langle\hat{\Sigma},\left[\frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}\hat{B}\theta}-\frac{\theta\theta^{\mathrm{T}}}{\theta^{\mathrm{T}}B\theta}\right]-\left[\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}\hat{B}\theta_\star}-\frac{\theta_\star\theta_\star^{\mathrm{T}}}{\theta_\star^{\mathrm{T}}B\theta_\star}\right]\right\rangle\right| \leq c_0\mathsf{r}_1\|\theta\theta^{\mathsf{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}. \tag{A.16}$$

We conclude from (A.12-A.16) that for $\mathbf{Z}$ satisfying H3

$$\mathsf{R}_n(\theta;\mathbf{Z})-\mathsf{R}_n(\theta_\star;\mathbf{Z}) \leq -\frac{\mathsf{gap}}{2}\left(\frac{\underline{\kappa}}{\bar{\kappa}}\right)^2\|\theta\theta^{\mathrm{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}^2+c_0\mathsf{r}_1\|\theta\theta^{\mathsf{T}}-\theta_\star\theta_\star^{\mathrm{T}}\|_{\mathsf{F}}.$$

This ends the proof. □

A.2 Proof of Proposition 2

We present the details of this claim for $\hat{\Sigma}$, the argument being similar for the other two covariance matrices. For any $J\subset[1:p]$ of size s, we have

$$\|\hat{\Sigma}_{J,J}-\Sigma_{J,J}\|_{\mathsf{op}} = \|\Sigma_{J,J}^{1/2}\left(\Sigma_{J,J}^{-1/2}\hat{\Sigma}_{J,J}\Sigma_{J,J}^{-1/2}-I_s\right)\Sigma_{J,J}^{1/2}\|_{\mathsf{op}} \\ \leq \|\Sigma_{J,J}^{1/2}\|_{\mathsf{op}}^2\times\|\frac{1}{n}\sum_{i=1}^n U_{iJ}U_{iJ}^{\mathrm{T}}-I_s\|_{\mathsf{op}},$$

where $U_{iJ} \stackrel{\text{def}}{=} \Sigma_{J,J}^{-1/2} Z_{iJ}$, where $Z_{iJ} = (Z_{ij})_{j \in J}$, is mean zero and isotropic. By Theorem 4.6.1 (Equation 4.22) of (Vershynin (2018)), provided that $n \geq 4c_0 s \log(p)$ for some absolute constant $c_0 > 1$, we have

$$\|\hat{\Sigma}_{J,J} - \Sigma_{J,J}\|_{\mathsf{op}} \leq CK^2 \|\Sigma_{J,J}^{1/2}\|_{\mathsf{op}}^2 \sqrt{\frac{c_0 s \log(p)}{n}},$$

with probability at least $1 - 2p^{-c_0 s}$. Therefore, for any matrix $A \in \mathbb{R}^{s \times s}$, with $\|A\|_{\mathsf{F}} = 1$, and $\mathsf{Rank}(A) \leq \alpha$, using the singular value decomposition of A, we have

$$\max_{\substack{A \in \mathbb{R}^{s \times s}: \, \|A\|_{\mathsf{F}} = 1 \\ \mathsf{Rank}(A) \leq \alpha}} \left| \left\langle \hat{\Sigma}_{J,J} - \Sigma_{J,J}, A \right\rangle \right| \leq \sqrt{\alpha} \|\hat{\Sigma}_{J,J} - \Sigma_{J,J}\|_{\mathsf{op}} \leq CK^2 \lambda_{\mathsf{max}}(\Sigma, s) \sqrt{\frac{c_0 \alpha s \log(p)}{n}}.$$

Since the number of subsets of $[1:p]$ of size s is smaller than p^s, we conclude with a union bound argument that

$$\lambda_{\mathsf{max}}^{(\alpha)}(\hat{\Sigma} - \Sigma, s) \leq CK^2 \lambda_{\mathsf{max}}(\Sigma, s) \sqrt{\frac{c_0 \alpha s \log(p)}{n}},$$

with probability $1 - 2p^{-(c_0 - 1)s}$.

A.3 MCMC sampling

We sample from the simulated tempering distribution (2.15) using a Metropolis-within-Gibbs strategy. We describe here one iteration of the algorithm, and its transition kernel. Given (δ, θ, k), we perform a three-step update. First, given k and δ, we update θ. We let $[\theta]_\delta$ to denote the δ-selected component of θ listed in their original order: $[\theta]_\delta \stackrel{\text{def}}{=} (\theta_j : j \in \{1 \leq k \leq p : \delta_k = 1\})$, and $[\theta]_{\delta^c} \stackrel{\text{def}}{=} (\theta_j : j \in \{1 \leq k \leq p : \delta_k = 0\})$. We employ the fact that the selected components $[\theta]_\delta$ and the un-selected components $[\theta]_{\delta^c}$ of θ are independent conditional on k and δ to update θ. In addition, given k and δ, the components of $[\theta]_{\delta^c}$ are i.i.d. $\mathbf{N}(0, t_k \rho_0^{-1})$ and the

Algorithm 2 Simulated tempering for sparse canonical correlation analysis

Model Input: Matrices $\hat{A}, \hat{B}$, prior parameters $\rho_0, \rho_1, \mathsf{u}$.
MCMC Input: Number of iterations N, batch size J, temperatures $1 = t_1 < \ldots < t_K$, weights $\{c_1, \ldots, c_K\}$, and step-sizes $\{\eta_1, \ldots, \eta_K\}$.
Initialization: Set the temperature index $k^{(0)} = 1$. Draw $\delta_j^{(0)} \overset{i.i.d.}{\sim} \mathbf{Ber}(0.5)$ for $j = 1, \ldots, p$, and independently draw $\theta^{(0)} \sim \mathbf{N}(0, I_p)$.
for $t = 0$ to $N - 1$, given $(\delta^{(t)}, \theta^{(t)}, k^{(t)}) = (\delta, \theta, k)$ **do**

1. **Update** θ: Draw the components of $[\bar{\theta}]_{\bar{\delta}^c}$independently from $\mathbf{N}(0, \rho_0^{-1} t_k)$. Draw $[\bar{\theta}]_{\bar{\delta}} \sim P_{k,\delta}([\theta]_\delta, \cdot)$, where $P_{k,\delta}$ denotes the transition kernel of the MALA with step-size η_k and invariant distribution given by (A.17).

2. **Update** δ: Uniformly randomly select a subset J from $\{1, \ldots, p\}$ of size J without replacement, and draw $\bar{\delta} \sim Q_{k,\bar{\theta}}^{(\mathsf{J})}(\delta, \cdot)$, where the transition kernel described in (A.20).

3. **Update** k: Draw $\bar{k} \sim T_{\bar{\delta},\bar{\theta}}(k, \cdot)$, where $T_{\delta,\theta}$ is the transition kernel of the Metropolis-Hastings on $\{1, \ldots, K\}$ with invariant distribution given by (A.21) and random walk proposal that has reflection at the boundaries.

4. **New MCMC state**: Set $(\delta^{(t+1)}, \theta^{(t+1)}, k^{(t+1)}) = (\bar{\delta}, \bar{\theta}, \bar{k})$.

end for
Output: $\{(\delta^{(t)}, \theta^{(t)}, k^{(t)}) : \ 0 \leq t \leq N \quad \text{s.t.} \quad k^{(t)} = 1\}$

distribution of $[\theta]_\delta$ has density on $\mathbb{R}^{\|\delta\|_0}$ proportional to

$$u \mapsto \exp\left(-\frac{\rho_1}{2t_k}\|u\|_2^2 + \frac{\sigma_n}{t_k}\mathsf{R}_n((u,0)_\delta; \mathbf{Z})\right), \tag{A.17}$$

where the notation $(u, 0)_\delta$ denotes the vector in $\mathbb{R}^p$ such that $[(u, 0)_\delta]_\delta = u$. Hence we update $[\theta]_\delta$ using a Metropolis adjusted Langevin algorithm (MALA) on $\mathbb{R}^{\|\delta\|_0}$ with target distribution (A.17), and step-size η_k (we use different step-sizes for different temperature levels). Let $M_{\delta,k}$ denote the resulting transition kernel on $\mathbb{R}^{\|\delta\|_0}$. For more details on the MALA, see e.g., Roberts and Tweedie (1996). For convenience, we write $P_{k,\delta}$ to denote the Markov kernel on $\mathbb{R}^p$ corresponding to the update of θ

just described. Specifically,

$$P_{\delta,k}(\theta, \mathrm{d}\theta') = M_{\delta,k}([\theta]_\delta, \mathrm{d}[\theta']_\delta) \prod_{j:\ \delta_j=0} \mathbf{N}(0, \rho_0^{-1})(\mathrm{d}\theta'_j),$$

where $\mathbf{N}(\mu, \sigma^2)(\mathrm{d}x)$ is a short for the Gaussian measure on $\mathbb{R}$ with mean μ and variance σ^2.

Secondly, we update δ by applying a Gibbs sampler to the conditional distribution of δ given k and θ. Note that the conditional distribution of δ_j given k, θ and δ_{-j}, where $\delta_{-j} \stackrel{\text{def}}{=} (\delta_1, \ldots, \delta_{j-1}, \delta_{j+1}, \ldots, \delta_p)$, is the Bernoulli distribution $\mathbf{Ber}(q_j)$, with probability of success given by

$$q_j \stackrel{\text{def}}{=} \left\{1 + \exp\left(-\frac{\mathsf{a}}{t_k} + \frac{1}{2t_k}(\rho_1 - \rho_0)\theta_j^2\right) \exp\left(\frac{\sigma_n}{t_k}\mathsf{R}_n(\theta_{\delta^{(j,0)}}; \mathbf{Z}) - \frac{\sigma_n}{t_k}\mathsf{R}_n(\theta_{\delta^{(j,1)}}; \mathbf{Z})\right)\right\}^{-1}, \tag{A.18}$$

where

$$\delta_i^{(j,0)} \stackrel{\text{def}}{=} \begin{cases} 0 & i = j \\ \delta_i & i \neq j \end{cases}, \quad \delta_i^{(j,1)} \stackrel{\text{def}}{=} \begin{cases} 1 & i = j \\ \delta_i & i \neq j \end{cases}. \tag{A.19}$$

Given k, θ and j, let $Q_{k,\theta}^{(j)}$ denote the transition kernel on Δ which, given δ, leaves δ_i unchanged for all $i \neq j$, and draws $\delta_j \sim \mathbf{Ber}(q_j)$. We update δ as follows: randomly draw a subset $\mathsf{J} = \{\mathsf{J}_1, \ldots, \mathsf{J}_J\}$ of size J from $\{1, \ldots, p\}$, and update δ using the transition kernel on Δ given by

$$Q_{k,\theta}^{(\mathsf{J})} \stackrel{\text{def}}{=} Q_{k,\theta}^{(\mathsf{J}_1)} Q_{k,\theta}^{(\mathsf{J}_2)} \cdots Q_{k,\theta}^{(\mathsf{J}_J)}. \tag{A.20}$$

The resulting overall kernel on Δ is

$$\bar{Q}_{k,\theta} = \sum_{\mathsf{J}:\ |\mathsf{J}|=J} \binom{p}{J}^{-1} Q_{k,\theta}^{(\mathsf{J})}.$$

Thirdly, given δ and θ, we update k using a standard Metropolis-Hastings algo-

rithm with a random walk proposal that has reflection at the boundaries. Specifically, at k we propose with equal probability either $k-1$ or $k+1$, except at 1, where we only propose 2, and at K, where we only propose $K-1$. We write $T_{\delta,\theta}$ to denote the transition kernel on $\{1,\ldots,K\}$ of this Metropolis-Hastings algorithm with invariant distribution

$$i \mapsto \frac{1}{c_i} \exp\left\{ \frac{\mathsf{a}}{t_i}\|\delta\|_0 - \frac{\rho_1}{2t_i}\|\theta_\delta\|_2^2 - \frac{\rho_0}{2t_i}\|\theta-\theta_\delta\|_2^2 + \frac{\sigma_n}{t_i}\mathsf{R}_n(\theta_\delta;\mathbf{Z}) \right\}. \tag{A.21}$$

Lastly, we collect samples by retaining the values of (δ,θ) at iterations at which $k=1$. In stationarity these samples have distribution (2.8).

A.3.1 Parameter choices and adaptive tuning

Throughout the simulation, we specify the parameters of the prior distribution in the following way. We let $\rho_1 = \frac{1}{2}$, and $\rho_0 = n/10$, where n is the sample size, and we set $\mathsf{u} = 1.5$.

Algorithm 2 also depends on the user-defined parameters J, K, $(t_1,\ldots,t_K)$, $(c_1,\ldots,c_K)$, and $(\eta_1,\ldots,\eta_K)$. The parameter J (the Gibbs sampling batch size) does not greatly impact performance, and setting $J = 100$ works well in most settings. Efficient selection and tuning of temperatures in simulated tempering has received some attention (Geyer and Thompson (1995); Atchadé et al. (2011)), and despite some progress (Miasojedow et al. (2013)), to the best of our knowledge, there is no practical and scalable algorithm to do so. In our implementation we use variations of the geometric scaling. We refer the reader to Section 2.4 for specific choices.

We tune the step-sizes $\eta = (\eta_1,\ldots,\eta_K)$ of MALA and the weights $(c_1,\ldots,c_K)$ of simulated tempering using adaptive MCMC methods , see e.g., Andrieu and Thoms (2008). To tune η_k, we follow the algorithm proposed in Atchadé and Rosenthal (2005), with a targeted acceptance probability of 30%. For simulated tempering to

visit all temperature levels frequently, the weights $(c_1, \ldots, c_K)$ need to be adequately tuned. We refer the reader to Geyer and Thompson (1995) for an extensive discussion of the issue. This problem can be efficiently solved using the Wang-Landau algorithm for simulated tempering as developed in Atchadé and Liu (2010). We follow this approach here. The fully adaptive MCMC sampler is presented in Algorithm 3.

A.4 Coupled Markov chains for mixing time estimation

We describe here the specific coupled Markov chain employed to estimate the mixing time plots presented in Section 2.4.2. We refer the reader to Biswas et al. (2019) and Jacob et al. (2017) for more details on the construction of such coupled kernels. We modify Algorithm 2 to construct the coupled kernel $\check{P}$. It suffices here to describe one iteration of the coupled chain. At some iteration $t \geq 1$, suppose that $(\delta^{(1,L+t)}, \theta^{(1,L+t)}, k^{(1,L+t)}) = (\delta^{(1)}, \theta^{(1)}, k^{(1)})$ and $(\delta^{(2,t)}, \theta^{(2,t)}, k^{(2,t)}) = (\delta^{(2)}, \theta^{(2)}, k^{(2)})$.

In step 1, to update $\theta^{(1)}$ and $\theta^{(2)}$, we partition the indices $\{1, \ldots, p\}$ into four groups: $G_{ab} = \{j : \delta_j^{(1)} = a, \delta_j^{(2)} = b\}$ for $a, b = 0, 1$. To update the components of $\theta^{(1)}_{G_{00}}$ and $\theta^{(2)}_{G_{00}}$, for any $j \in G_{00}$ we first draw a common standard normal random variables Z_j, and then obtain $\bar{\theta}_j^{(i)} = t_{k^{(i)}} \rho_0^{-1} Z_j$ for $i = 1, 2$. To update the components of $\theta^{(1)}_{G_{01}}$ and $\theta^{(2)}_{G_{01}}$, for any $j \in G_{01}$ we again first draw a common standard normal random variables Z_j, and then obtain $\bar{\theta}_j^{(1)} = t_{k^{(1)}} \rho_0^{-1} Z_j$, and simultaneously draw $\bar{\theta}_j^{(2)}$ using MALA with proposal $\theta_j^{(2)} + \eta_{k^{(2)}} \nabla \log \pi(\theta_j^{(2)}) + \sqrt{2\eta_{k^{(2)}}} Z_j$, where $\pi(\theta_j^{(2)})$ is the marginal posterior distribution of $\theta_j^{(2)}$. Notice that the joint distribution of $[\theta^{(2)}]_{\delta^{(2)}}$ is given by $W_{k^{(2)}, \delta^{(2)}}$, whose density is proportional to (A.17). A similar update procedure is used for updating the components of $\theta^{(1)}_{G_{10}}$ and $\theta^{(2)}_{G_{10}}$. To update the components of $\theta^{(1)}_{G_{11}}$ and $\theta^{(2)}_{G_{11}}$, we draw reflection-coupled MALA proposals in Biswas et al. (2019), and then for the acceptation step, $\theta^{(1)}_{G_{11}}$ and $\theta^{(2)}_{G_{11}}$ share the same uniform random variables.

Algorithm 3 Adaptive version of simulated tempering for Canonical correlation analysis

Model Input: Matrices $\hat{A}, \hat{B}$, prior parameters $\rho_0, \rho_1, \mathsf{u}$.
MCMC Input: Number of iterations N, Batch size J, temperatures $1 = t_1 < \ldots, < t_K$.
Adaptive MCMC Input: $a = 10$ and $w \in (0, 1)$.
MCMC Initialization: Set $k^{(0)} = 1$. Draw $\delta_j^{(0)} \overset{i.i.d.}{\sim} \mathbf{Ber}(0.5), \forall j = 1, \ldots, p$, and independently $\theta^{(0)} \sim \mathbf{N}(0, I_p)$.
Adaptation Parameters Initialization : . Set $\ell^{(0)} = \mathbf{0} \in \mathbb{R}^K$, $v^{(0)} = (0, \ldots, 0) \in \mathbb{R}^K$, $\nu^{(0)} = (0, \ldots, 0) \in \mathbb{R}^K$, and choose $c^{(0)} \in (0, \infty)^K$.
for $t = 1$ to $N - 1$, given $(\delta^{(t)}, \theta^{(t)}, k^{(t)}) = (\delta, \theta, k)$, $\ell^{(t)} = \ell$, $c^{(t)} = c$, $v^{(t)} = v$, and $\nu^{(t)} = \nu$ **do**

1. **Update** θ **and** ℓ: Draw the components of $[\bar{\theta}]_{\bar{\delta}^c}$independently from $\mathbf{N}(0, \rho_0^{-1} t_k)$. Draw $[\bar{\theta}]_\delta \sim P_{k,\delta}([\theta]_\delta, \cdot)$, where $P_{k,\delta}$ denotes the transition kernel of the MALA with step-size e^{ℓ_k} and invariant distribution given by (A.17). Denote α as the acceptance probability of the MALA update. Set

$$\bar{\ell}_k = \ell_k + v_k^{-0.6}(\alpha - 0.3) \quad \text{and for } i \neq k, \text{ set } \bar{\ell}_i = \ell_i.$$

2. **Update** δ: Uniformly randomly select a subset J from $\{1, \ldots, p\}$ of size J without replacement, and draw $\bar{\delta} \sim Q_{k,\bar{\theta}}^{(\mathsf{J})}(\delta, \cdot)$, where the transition kernel described in (A.20).

3. **Update** k**,** c**,** v **and** ν: Draw $\bar{k} \sim T_{\bar{\delta},\bar{\theta}}(k, \cdot)$, where $T_{\delta,\theta}$ is the transition kernel of the Metropolis-Hastings on $\{1, \ldots, K\}$ with invariant distribution given by (A.21) and random walk proposal with reflection at the boundaries. We then set

$$\bar{c}_{\bar{k}} = c_{\bar{k}} e^a, \quad \bar{v}_{\bar{k}} = v_{\bar{k}} + 1, \ \bar{\nu}_{\bar{k}} = \nu_{\bar{k}} + 1, \ \text{ and for } \ i \neq \bar{k}, \bar{c}_i = c_i, \ \bar{v}_i = v_i, \ \bar{\nu}_i = \nu_i.$$

4. **Update** a **and** ν: If $\|\nu/(\sum_{k=1}^K \nu_k) - 1/K\|_\infty \leq w/K$, then set $a = a/2, \nu = \mathbf{0} \in \mathbb{R}^K$.

5. **New MCMC state**: Set $(\delta^{(t+1)}, \theta^{(t+1)}, k^{(t+1)}) = (\bar{\delta}, \bar{\theta}, \bar{k})$, $\ell^{(t+1)} = \bar{\ell}$, $c^{(t+1)} = \bar{c}$, $v^{(t+1)} = \bar{v}$, and $\nu^{(t+1)} = \bar{\nu}$.

end for
Output: $\{(\delta^{(t)}, \theta^{(t)}, k^{(t)}) : \ 0 \leq t \leq N \quad \text{s.t.} \quad k^{(t)} = 1\}$

In step 2, to update $\delta^{(1)}$ and $\delta^{(2)}$, we first make use of the same randomly drawn subset J. For $i = 1, 2$, drawing $\bar{\delta}^{(i)} \sim Q^{(\mathsf{J})}_{k,\theta}(\delta^{(i)}, \cdot)$ is equivalent to let $\bar{\delta}^{(i)}_{-\mathsf{J}} = \delta^{(i)}_{-\mathsf{J}}$, and for any $j \in \mathsf{J}$, draw $\bar{\delta}^{(i)}_j \sim \mathbf{Ber}(q^{(i)}_j)$ which we implement in the following way. We first draw a common uniform number $u_j \sim \mathbf{Uniform}(0, 1)$, then we obtain $\bar{\delta}^{(i)}_j = \mathbf{1}\{q^{(i)}_j \leq u_j\}$ for $i = 1, 2$.

In step 3, to update $k^{(1)}$ and $k^{(2)}$, we use a common uniform random number to make the proposal move, and a common uniform random number for the acceptation step.

Remark 27. Note that although the empirical mixing time estimation method of Biswas et al. (2019) described above only applies to Markov chains with fixed parameters, we have applied it here to Algorithm 3, which is an MCMC sampler with adaptively tuned parameters. We conjecture that the unbiased MCMC methodology remains approximately valid for well-constructed adaptive MCMC samplers. However the question deserves more research.

A.4.1 Proof of Proposition 3

Using the notations established in Section A.3 the transition kernel of the simulated tempering algorithm on $\mathcal{X} \stackrel{\text{def}}{=} \Delta \times \mathbb{R}^p \times \{1, \ldots, K\}$ is given by

$$P((\delta, \theta, k); (\delta', \mathrm{d}\theta', k')) = P_{\delta,k}(\theta, \mathrm{d}\theta') \left\{ \sum_{\mathsf{J}:\ |\mathsf{J}|=J} \binom{p}{J}^{-1} Q^{(\mathsf{J})}_{k,\theta'}(\delta, \delta') \right\} T_{\delta',\theta'}(k, k'),$$

and we call $\bar{P}$ the transition kernel of the coupled chain on $\mathcal{X} \times \mathcal{X}$ as described in Section A.4. The kernel P is a standard Metropolis-within-Gibbs kernel to sample from the density (2.15) that is positive everywhere. Therefore, P is phi-irreducible, aperiodic and has invariant distribution $\bar{\Pi}(\cdot|\mathbf{Z})$ by construction. Furthermore, for any nonempty compact set $\mathcal{C}$ of $\mathbb{R}^p$, the set $\bar{\mathcal{C}} \stackrel{\text{def}}{=} \mathcal{C} \times \Delta \times \{1, \ldots, K\}$ is a small set for P,

and it is easy to see from the construction of the coupled chain that,

$$\min_{x,y\in\bar{\mathcal{C}}} \bar{P}^{n_0}\left((x,y);\mathcal{D}\right) > 0, \quad \text{with } n_0 = \max\left(K, \frac{p}{J}\right).$$

Therefore, according to Proposition 4 of (Jacob et al. (2017)) to establish the finiteness of the average meeting time $\mathbb{E}(\tau^{(L)})$, it suffices to show that there exist a drift function $V:\ \mathcal{X} \to [1,\infty)$, $\lambda \in (0,1)$, and $b < \infty$ such that

$$PV(x) \leq \lambda V(x) + b\mathbf{1}_{\bar{\mathcal{C}}}(x), \quad \text{for all} \quad x = (\delta,\theta,k) \in \mathcal{X}, \tag{A.22}$$

for some small $\bar{\mathcal{C}}$ of the form $\bar{\mathcal{C}}_L \stackrel{\text{def}}{=} \{x \in \mathcal{X}:\ V(x) \leq L\}$. We show (A.22) in three steps, with

$$V(\delta,\theta,k) \stackrel{\text{def}}{=} 1 + \frac{1}{t_k}\|\theta_\delta\|_2^2.$$

Step 1: Action of the kernel $T_{\delta,\theta}$ We first show that for all $(\theta,\delta,k) \in \mathcal{X}$,

$$T_{\delta,\theta}V(\delta,\theta,k) \leq V(\delta,\theta,k) + c_0, \tag{A.23}$$

for some constant c_0. To show this, we find it easy to reason in a slightly more general terms. Consider a discrete distribution on $\{1,\ldots,K\}$, given by

$$\pi(k) \propto \frac{1}{c_k} e^{-U/t_k},$$

for some increasing sequence $\{c_k,\ 1 \leq k \leq K\}$, and for some nonnegative constant U. Consider a Metropolis-Hastings algorithm to sample from π with a proposal q on $\{1,\ldots,K\}$ such that at j, we propose to move only to $j-1$ or $j+1$ for equal probability (at 1 we propose to move only to 2, and at K we propose to move only to $K-1$). Call M the transition kernel of that Metropolis-Hastings, and for some

nonnegative constant B, define

$$V(j) = \frac{B}{t_j}, \;\; j = 1, \ldots, K.$$

By the definition of the Metropolis-Hasting kernel, we have

$$\begin{aligned} MV(j) = V(j) + \sum_{j'=1}^{K} (V(j') - V(j)) \min\left(1, \frac{\pi(j')q(j',j)}{\pi(j)q(j,j')}\right) q(j,j') \\ = V(j) + \sum_{j'=1}^{K} R(j,j')q(j,j'), \end{aligned}$$

where

$$R(j,j') = (V(j') - V(j)) \min\left(1, \frac{\pi(j')q(j',j)}{\pi(j)q(j,j')}\right).$$

Note that $V(j+1) \leq V(j)$, and therefore $R(j, j+1) \leq 0$. Whereas

$$\begin{aligned} R(j,j-1) &= B\left(\frac{1}{t_{j-1}} - \frac{1}{t_j}\right) \min\left(1, \frac{c_j}{c_{j-1}} \frac{q(j-1,j)}{q(j,j-1)} e^{-\left(\frac{1}{t_{j-1}} - \frac{1}{t_j}\right)U}\right) \\ &\leq \left(\frac{B}{U}\right) \frac{2c_j}{c_{j-1}} \left(\frac{1}{t_{j-1}} - \frac{1}{t_j}\right) U e^{-\left(\frac{1}{t_{j-1}} - \frac{1}{t_j}\right)U} \\ &\leq 2e^{-1} \left(\max_j \frac{c_j}{c_{j-1}}\right) \frac{B}{U} = C, \end{aligned}$$

where we use the fact that $q(j',j)/q(j,j') \leq 2$, and the observation that $te^{-t} \leq e^{-1}$ for all $t \geq 0$. Using these we have

$$MV(1) = V(1) + R(1,2) \leq V(1).$$

$$MV(K) = V(K) + R(K, K-1) \leq V(K) + C.$$

For $2 \leq j \leq K-1$,

$$MV(j) = V(j) + \frac{1}{2}R(j,j-1) + \frac{1}{2}R(j,j+1) \leq V(j) + \frac{C}{2}.$$

Hence, for all $1 \leq j \leq K$, it holds

$$MV(j) \leq V(j) + 2e^{-1} \left(\max_j \frac{c_j}{c_{j-1}} \right) \frac{B}{U}.$$

We can apply this result to the kernel $T_{\delta,\theta}$ with $U = \frac{\rho_1}{2}\|\theta_\delta\|_2^2 + \frac{\rho_0}{2}\|\theta - \theta_\delta\|_2^2 - \sigma_n \mathsf{R}_n(\theta_\delta; \mathbf{Z}) - \mathsf{a}\|\delta\|_0$, and $B = \|\theta_\delta\|_2^2$. Under assumption H2, $\mathsf{R}_n(\theta_\delta; \mathbf{Z}) \leq 1$. Hence for $\rho_1\|\theta_\delta\|_2^2 \geq 4(\sigma_n + \mathsf{a}p)$, the chosen U is non-negative and we get

$$\begin{aligned} T_{\delta,\theta}V(\delta,\theta,k) &\leq V(\delta,\theta,k) \\ &\quad + 2e^{-1} \left(\max_j \frac{c_j}{c_{j-1}} \right) \frac{\|\theta_\delta\|_2^2}{\frac{\rho_1}{2}\|\theta_\delta\|_2^2 + \frac{\rho_0}{2}\|\theta - \theta_\delta\|_2^2 - \sigma_n \mathsf{R}_n(\theta_\delta; \mathbf{Z}) - \mathsf{a}\|\delta\|_0} \\ &\leq V(\delta,\theta,k) + \frac{8e^{-1}}{\rho_1} \left(\max_j \frac{c_j}{c_{j-1}} \right), \end{aligned}$$

for $\|\theta_\delta\|_2 \geq L$, for L taken large enough.

Step 2: Accounting for the kernel $\bar{Q}$ For consistency in the notation, we write summations as integrals with respect to the counting measure. Using (A.23) and the definition of $\bar{Q}$, we have for all $(\delta, \theta, k) \in \mathcal{X}$ such that $\|\theta_\delta\|_2 \geq L$ for some appropriately large L,

$$\begin{aligned} \int_\Delta \bar{Q}_{\theta,k}(\delta, \mathrm{d}\delta') \int T_{\theta,\delta'}(k, \mathrm{d}k') V(\theta, \delta', k') &\leq \int_\Delta \bar{Q}_{\theta,k}(\delta, \mathrm{d}\delta') V(\theta, \delta', k) + c_0 \\ &= \sum_{\mathsf{J}:\, |\mathsf{J}| = J} \binom{p}{J}^{-1} \int_\Delta Q^{(\mathsf{J})}_{\theta,k}(\delta, \mathrm{d}\delta') V(\theta, \delta', k) + c_0. \end{aligned}$$

Given a selection $\mathsf{J} = \{j_1, \ldots, j_J\} \subseteq \{1, \ldots, p\}$, and $j_i \in \mathsf{J}$, we have

$$\int_\Delta \tilde{Q}^{(j_i)}_{k,\theta}(\delta, \mathrm{d}\delta') V(\theta, \delta', k) = V(\theta, \delta, k) + (q_{j_i} - \delta_{j_i}) \frac{\theta_{j_i}^2}{t_k} \leq V(\theta, \delta, k) + (1 - \delta_{j_i}) \frac{\theta_{j_i}^2}{t_k},$$

where q_j is as given in (A.18). It follows that

$$\int_\Delta Q^{(\mathsf{J})}_{\theta,k}(\delta, \mathrm{d}\delta')V(\theta,\delta',k) \leq V(\theta,\delta,k) + \frac{1}{t_k}\sum_{i=1}^{J} \theta^2_{j_i}\mathbf{1}_{\{\delta_{j_i}=0\}}.$$

We conclude that for $\|\theta_\delta\|_2 \geq L$,

$$\int_\Delta \bar{Q}_{\theta,k}(\delta,\mathrm{d}\delta')\int T_{\theta,\delta'}(k,\mathrm{d}k')V(\theta,\delta',k') \leq V(\theta,\delta,k) + \frac{1}{t_k}\sum_{\mathsf{J}:\,|\mathsf{J}|=J}\binom{p}{J}^{-1}\left\{\sum_{i=1}^{J}\theta^2_{\mathsf{J}_i}\mathbf{1}_{\{\delta_{\mathsf{J}_i}=0\}}\right\} + c_0. \quad \text{(A.24)}$$

Step 3: Drift condition for P We recall that under the kernel $P_{\delta,k}$ the components $\{\theta'_j,\ j:\ \delta_j = 0\}$ are drawn independently from the Gaussian distribution $\mathbf{N}(0,\rho_0^{-1})$. Therefore, for $\|\theta_\delta\|_2 \geq L$, using (A.24), we have

$$\begin{aligned}\int_{\mathcal{X}} P\left((\theta,\delta,k);(\mathrm{d}\theta',\mathrm{d}\delta',\mathrm{d}k')\right)V(\theta',\delta',k') &\leq \int_{\mathbb{R}^p} P_{\delta,k}(\theta,\mathrm{d}\theta')\left[V(\theta',\delta,k) + \frac{1}{t_k}\sum_{\mathsf{J}:\,|\mathsf{J}|=J}\binom{p}{J}^{-1}\left\{\sum_{i=1}^{J}(\theta'_{\mathsf{J}_i})^2\mathbf{1}_{\{\delta_{\mathsf{J}_i}=0\}}\right\} + c_0\right] \\ &\leq 1 + \frac{1}{t_k}\int_{\mathbb{R}^{\|\delta\|_0}} M_{k,\delta}(\theta_\delta,\mathrm{d}v)\|v\|_2^2 + \frac{J}{\rho_0 t_k} + c_0, \quad \text{(A.25)}\end{aligned}$$

where $M_{k,\delta}$ is the kernel of the MALA with target distribution proportional to

$$u \mapsto \exp\left(-\frac{\rho_1}{2t_k}\|u\|_2^2 + \frac{\sigma_n}{t_k}\mathsf{R}_n((u,0)_\delta;\mathbf{Z})\right).$$

It remains to deal with the term $\int_{\mathbb{R}^{\|\delta\|_0}} M_{k,\delta}(\theta_\delta,\mathrm{d}v)\|v\|_2^2$. For clarity sake let's work is a slightly more general setting. Suppose that we have a density on $\mathbb{R}^d$, $d \geq 1$, that is proportional to $e^{-m(u)}$ for some function m of the form

$$m(u) \stackrel{\text{def}}{=} \frac{\rho}{2}\|u\|_2^2 + \ell(u),$$

for some bounded function ℓ. Let $q_\eta(u,\cdot)$ be the density of the proposal distribution $\mathbf{N}\left(\left(1-\frac{\rho\eta^2}{2}\right)u, \eta^2 I_d\right)$, and define $\mathsf{R}(u) \stackrel{\text{def}}{=} \{v \in \mathbb{R}^d : \alpha(u,v) < 1\}$, where

$$\alpha(u,v) = \min\left(1, \frac{e^{-m(v)}q_\eta(v,u)}{e^{-m(u)}q_\eta(u,v)}\right).$$

Let L denote the resulting transition kernel on $\mathbb{R}^p$. We have

$$\begin{aligned}\int_{\mathbb{R}^d} L(u,\mathrm{d}v)\|v\|_2^2 &= \|u\|_2^2 + \int \alpha(x,y)\left(\|v\|_2^2 - \|u\|_2^2\right) q_\eta(u,v)\mathrm{d}v \\ &= \|u\|_2^2 + \int_{\mathsf{R}(u)}\left[\frac{e^{-m(v)}q_\eta(v,u)}{e^{-m(u)}q_\eta(u,v)} - 1\right]\left(\|v\|_2^2 - \|u\|_2^2\right) q_\eta(u,v)\mathrm{d}v \\ &\quad + \int_{\mathbb{R}^d}\left(\|v\|_2^2 - \|u\|_2^2\right) q_\eta(u,v)\mathrm{d}v.\end{aligned}$$

We can write

$$\|v\|_2^2 - \|u\|_2^2 = 2\langle u, v-u\rangle + \|v-u\|_2^2.$$

Integrating both sides, we get

$$\int_{\mathbb{R}^d}\left(\|v\|_2^2 - \|u\|_2^2\right) q_\eta(u,v)\mathrm{d}v = -\rho\eta^2\|u\|_2^2 + \frac{\rho^2\eta^4}{4}\|u\|_2^2 + \eta^2 d \le -\frac{3}{4}\rho\eta^2\|u\|_2^2 + \eta^2 d, \tag{A.26}$$

by choosing η such that $\eta^2\rho \le 1/2$. We also have

$$\begin{aligned}&\frac{e^{-m(v)}q_\eta(v,u)}{e^{-m(u)}q_\eta(u,v)} \\ &\qquad = \exp\left(m(u) - m(v) - \frac{1}{2\eta^2}\left\|u - v + \frac{\rho\eta^2}{2}v\right\|^2 + \frac{1}{2\eta^2}\left\|v - u + \frac{\rho\eta^2}{2}u\right\|^2\right).\end{aligned}$$

If $v \in \mathsf{R}(u)$, we necessarily have $\frac{e^{-m(v)}q_\eta(v,u)}{e^{-m(u)}q_\eta(u,v)} < 1$, which translates to:

$$m(v) - m(u) > -\frac{1}{2\eta^2}\left\|u - v + \frac{\rho\eta^2}{2}v\right\|^2 + \frac{1}{2\eta^2}\left\|v - u + \frac{\rho\eta^2}{2}u\right\|^2.$$

Noting that $m(u) = (\rho/2)\|u\|_2^2 + \ell(u)$, where ℓ is bounded by b_0, we infer that for $v \in \mathsf{R}(u)$,

$$\begin{aligned}\frac{\rho}{2}\left(\|u\|_2^2 - \|v\|_2^2\right) &\leq 2b_0 + \frac{1}{2\eta^2}\left\|u - v + \frac{\rho\eta^2}{2}v\right\|^2 - \frac{1}{2\eta^2}\left\|v - u + \frac{\rho\eta^2}{2}u\right\|^2 \\ &= 2b_0 + \frac{\rho^2\eta^2}{8}\left(\|v\|_2^2 - \|u\|_2^2\right) - \frac{\rho}{2}\langle v - u, v + u\rangle \\ &= 2b_0 + \frac{\rho^2\eta^2}{8}\left(\|v\|_2^2 - \|u\|_2^2\right) + \frac{\rho}{2}\left(\|u\|_2^2 - \|v\|_2^2\right).\end{aligned}$$

Hence, for $v \in \mathsf{R}(u)$,

$$\|u\|_2^2 - \|v\|_2^2 \leq \frac{16b_0}{\rho^2\eta^2},$$

which we use to write

$$\begin{aligned}&\int_{\mathsf{R}(u)}\left[\frac{e^{-m(v)}q_\eta(v,u)}{e^{-m(u)}q_\eta(u,v)} - 1\right]\left(\|v\|_2^2 - \|u\|_2^2\right)q_\eta(u,v)\mathrm{d}v \\ &\qquad = \int_{\mathsf{R}(u)}\left[1 - \frac{e^{-m(v)}q_\eta(v,u)}{e^{-m(u)}q_\eta(u,v)}\right]\left(\|u\|_2^2 - \|v\|_2^2\right)q_\eta(u,v)\mathrm{d}v \ \leq \frac{16b_0}{\rho^2\eta^2}. \quad \text{(A.27)}\end{aligned}$$

We combine (A.26)-(A.27) to conclude that

$$\int_{\mathbb{R}^d} L(u,\mathrm{d}v)\|v\|_2^2 \leq \|u\|_2^2 - \frac{3}{4}\rho\eta^2\|u\|_2^2 + \eta^2 d + \frac{16b_0}{\rho^2\eta^2}.$$

Hence we can find b_1 (for instance $b_1 = 4\rho^{-1}(d + 16b_0\rho^{-2}\eta^{-4})$) such that for $\|u\|_2^2 > b_1$, it holds

$$\int_{\mathbb{R}^d} L(u,\mathrm{d}v)\|v\|_2^2 \leq \left(1 - \frac{\rho\eta^2}{2}\right)\|u\|_2^2.$$

This results applied to $M_{k,\delta}$ together with (A.25) implies that there exist $\lambda \in (0,1)$ (for instance $\lambda = \frac{\rho\eta^2}{2t_K}$), and $L < \infty$ such that

$$PV(\delta,\theta,k) \leq \lambda V(\delta,\theta,k), \quad \text{for all}(\delta,\theta,k) \notin \bar{\mathcal{C}}_L.$$

With similar (but simpler) calculations we check that

$$\sup_{(\delta,\theta,k)\in\bar{\mathcal{C}}_L} PV(\delta,\theta,k) \leq b,$$

for some finite constant b. This establishes the drift condition

$$PV(\delta,\theta,k) \leq \lambda V(\delta,\theta,k) + b\mathbf{1}_{\bar{\mathcal{C}}_L}(\delta,\theta,k), \quad \text{for all } (\delta,\theta,k) \in \mathcal{X}.$$

Hence the result.

Appendix B

Supplemental Materials to Chapter 3

B.1 Additional details for the results of Section 3.4.1

Data generation: In both experiment, we generate $\mathcal{D}_1 = \{(\mathbf{x}_i, \mathbf{y}_i),\ 1 \leq i \leq n\}$ and $\mathcal{D}_2 = \{(B\mathbf{x}_i, \mathbf{y}_i),\ 1 \leq i \leq n\}$ as in (3.4), with $n = 200$, where f is the linear model in (3.2), and where $\mathbb{R}$ is the elastic-net regularization $\mathbb{R}_0$ given in (3.19) with $d_x = d_y = 100$, and $\lambda_1 = 1$, $\lambda_2 = 1$. The matrix A in (3.2) is generated with i.i.d. standard normal entries, and $v^2 = 0.001$. In the second experiment, the non-singular matrix B is generated by computing the dot product of a full rank matrix with entries sampled from $N(0, 0.025)$. $\mathcal{D}_2 = \{(\mathbf{x}_i, \mathbf{y}_i),\ 1 \leq i \leq n\}$ is obtained by solving $\mathbf{x}_i = B^{-1}(B\mathbf{x}_i)$.

Model details: In each experiment, we specify H_W as a fully-connected FNN of depth $D_1 = 2$ with layer $(p_0, p_1, p_2) = (d_x, 2d_x, d_x)$, and depth $D_2 = 4$ with layer $(p_0, p_1, p_2, p_3, p_4) = (d_x, d_x, 2d_x, d_x, d_x)$, respectively. The step-size γ of GDN in each experiment is taken as $\gamma_1 = \frac{2v^2}{\lambda_{\mathsf{max}}(A'A)}$ and $\gamma_2 = \frac{5v^2}{\lambda_{\mathsf{max}}(A'A)}$, respectively, where $\lambda_{\mathsf{max}}(A'A)$ is the largest eigenvalue of $A'A$. The unrolling depth is $D' = 1$ (GDN1), $D' = 5$ (GDN2), $D' = 10$ (GDN3), and $D' = 20$ (GDN4) (only for simulation 1). The total number of model parameters is $q = 700$ and $q = 20700$, respectively.

Training details: In the first experiment, for the Bayesian prior we choose $\rho_0 = n$, $\rho_1 = 1$, and $\mathsf{u} = 1$. We choose $\sigma = 0.001$ in (3.14), and run the SA-SGLD with a constant step-size $2 * 10^{-6}$ for GDN. In the first experiment, for the Bayesian prior we choose $\rho_0 = n$, $\rho_1 = 1$, and $\mathsf{u} = 1$. We choose $\sigma = 0.001$ in (3.14), and run the

SA-SGLD with a constant step-size $2 * 10^{-6}$ for GDN. In the second experiment, we use $\rho_0 = n$, $\rho_1 = 1$, and $\mathsf{u} = 800$. We choose $\sigma = 0.001$ in (3.14), and we run the SA-SGLD with a constant step-size $2 * 10^{-14}$ for GDN. The mini-batch size is set to 100 in both cases. The initial value $\mathbf{x}^{(0)}$ of GDN is set to 0. The MCMC samplers are implemented on a Nvidia Tesla V100 GPU system with 384 GB GPU memory running Pytorch. We run the sampler for 10^4 iterations.

B.2 Additional details for the results of Section 3.4.2

We generate 500 simulated data which are 16×16 matrices partitioned into 4 blocks with the upper left and lower right diagonal matrices with diagonal elements sampled from $N(20, 0.5)$ and $N(-10, 0.1)$, the upper right a matrix with entries sampled from $N(10, 0.1)$, and the lower left a matrix with entries sampled from $N(-10, 5)$.

The 3-layer network consists of 3 convolutional layers with respective sizes $3, 3, 1$, respective number of filters $32, 64, 1$. The 6-layer network consists of 2 convolutional layers, 1 channel-wise fully connected layer, 3 deconvolutional layers with respective sizes $5, 3, 2, 4, 5, 3$, respective number of filters $32, 64, 64, 64, 32, 1$. Each layer except the last layer in both network is followed by a LayerNorm layer and a ReLu layer. The layers are padded to keep the image size constant.

For the Bayesian prior, we use $\rho_0 = n, \rho_1 = 1$, and $\mathsf{u} = 8000$. We choose $\sigma^2 = 0.01$ in (3.14), and run the SA-SAGLD with a constant step-size 2×10^{-8} for both FNN and GDN. The mini-batch size is set to 50 in both cases. The MCMC sampler are implemented on a Nvidia Tesla V100 GPU system with 384 GB GPU memory running Pytorch. We run both samplers for 10^4 iterations.

For the comparison, we generate 1000 test samples and evaluate the prediction errors of FNN and GDN by computing for each test sample, the squared 2-norm between the prediction and the true value $\mathbf{x}$, averaged by the dimension d_x.

B.3 Additional details for the results of Section 3.4.3

We generate the corresponding observed measurements $\mathbf{y}_i$ through the linear forward model (3.2), where A is a Gaussian blurring convolution matrix with variance 6.25, and where $v^2 = 0.01$ leading to a highly ill-conditioned inverse problem. We fit the nonparametric regression model (3.7) with a GDN where H_W in (3.10) is taken as 3-layer relu-convolutional neural network. We use the same architecture as in Section 3.4.2, except for the convolution kernel size that we take as $4, 4, 2$, and the number of filters that we take here as $256, 256, 1$. The depth of the GDN is either $D' = 4$ (GDN1), or $D' = 12$ (GDN2). The total number of parameters in both cases is $q = 267,777$. We compare this model with a feedforward architecture that doe not make use of the forward model, and has 3 convolutional layers (with size $4, 4, 2$, filter number $256, 256, 32$, and strides $1, 2, 1$) followed by 4 corresponding deconvolutional layers (with kernel size $2, 4, 4, 2$, filter number $32, 256, 256, 1$, and strides $1, 2, 1, 1$). The total number of parameter of the FNN is $q = 2,271,297$. The rest of the implementation is the same as above (see Appendix B.3), except for the step-size, taken here as $\gamma = 10^{-9}$ for FNN and GDN2, $\gamma = 10^{-8}$ for GDN1, and the mini-batch size taken as $B = 164$.

We run the algorithms for $80,0000$ iterations using a Nvidia Tesla V100 GPU system with 384 GB GPU memory running Matlab 2022a.

B.4 Proofs

B.4.1 Proof of Theorem 10

Proof. We follow the same general steps outlined above in Lemma 12. We recall that the dataset is $\mathcal{D} \stackrel{\text{def}}{=} (\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_n, \mathbf{y}_n)$. For $W \in \mathcal{W}$, we define

$$f_W(\mathcal{D}) \stackrel{\text{def}}{=} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n \|\mathbf{x}_i - g_W(\mathbf{y}_i)\|_2^2\right),$$

$$\text{and} \quad f_\star(\mathcal{D}) \stackrel{\text{def}}{=} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n \|\mathbf{x}_i - g(\mathbf{y}_i)\|_2^2\right).$$

We recall that $\Theta = \mathsf{W} \times \mathcal{S}$. For any measurable set $A \subset \Theta$, we can write the posterior probability $\Pi(A|\mathcal{D})$ as

$$\Pi(A|\mathcal{D}) = \frac{\int_A \frac{f_{\Lambda \odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda, \mathrm{d}W)}{\int_\Theta \frac{f_{\Lambda \odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda, \mathrm{d}W)}. \tag{B.1}$$

We will repeatedly use the following observation. For $W \in \mathsf{W}$, we have

$$\begin{aligned}\log\left(\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\right) &= \frac{1}{2\sigma^2}\sum_{i=1}^n \left(\|\mathbf{x}_i - g(\mathbf{y}_i)\|_2^2 - \|\mathbf{x}_i - g_W(\mathbf{y}_i)\|_2^2\right) \\ &= -\frac{n}{2\sigma^2}\|g_W - g\|_n^2 - \frac{1}{\sigma^2}\sum_{i=1}^n \langle \mathbf{x}_i - g(\mathbf{y}_i), g(\mathbf{y}_i) - g_W(\mathbf{y}_i)\rangle\,. \end{aligned} \tag{B.2}$$

Given $s_0 \geq 1$, $\beta_0 \geq 0$, we set

$$\Theta(s_0, \beta_0) \stackrel{\text{def}}{=} \{(\Lambda, W) \in \Theta : \ \|\Lambda\|_0 \leq s_0, \ \text{ and } \|\Lambda \odot W\|_\infty \leq \beta_0\},$$

and

$$\mathcal{W}(s_0, \beta_0) \stackrel{\text{def}}{=} \{W \in \mathcal{W} : \ \|W\|_0 \leq s_0, \quad \|W\|_\infty \leq \beta_0\}.$$

We set

$$s \stackrel{\text{def}}{=} \frac{1}{\mathsf{u}} + \left(1 + \frac{5}{\mathsf{u}} + \frac{\log(n\mathsf{L}_\star)}{\mathsf{u}\log(q)}\right)s_\star + \frac{4n\varpi_\star^2}{\sigma^2\mathsf{u}\log(q)}, \ \text{ and } \ \mathsf{r} \stackrel{\text{def}}{=} \bar{\sigma}\sqrt{\frac{\mathsf{s}\log(\mathsf{q}\mathsf{L}_\mathsf{s})}{\mathsf{n}}},$$

and

$$\bar{\alpha} \stackrel{\text{def}}{=} \mathsf{u}s - 1,$$

where $\mathsf{L}_\star \stackrel{\text{def}}{=} L(2s_\star^{1/2}\beta_\star)$, $\mathsf{L}_s \stackrel{\text{def}}{=} L(2s^{1/2}b_s)$, and $b_s \stackrel{\text{def}}{=} \rho_1^{-1}(1+\mathsf{u})(s+1)\log(q)$, and where L is as in Assumption 7. Fix $M \geq 2$. For $j \geq 1$ we also set

$$\mathcal{W}_j(s_0,\beta_0) \stackrel{\text{def}}{=} \{W \in \mathcal{W}(s_0,\beta_0):\ j(M\mathsf{r}) < \|\mathsf{g}_W - \mathsf{g}\|_{\mathsf{n}} \leq (\mathsf{j}+\mathsf{1})\mathsf{M}\mathsf{r}\}.$$

We shall apply the same idea as in Lemma 12. Specifically, let

$$B \stackrel{\text{def}}{=} \{(\Lambda, W) \in \Theta:\ \|g_{\Lambda\odot W} - g\|_n \leq M\mathsf{r}\},$$

and consider the $\mathcal{E}$

$$\mathcal{E} = \left\{\mathcal{D}:\ \int_\Theta \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) > \frac{1}{4q^{\bar{\alpha}}},\quad \int_{\mathcal{A}(s)} \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) \leq \frac{1}{q^{\mathsf{u}s}}\right.$$
$$\left.\text{and}\quad \sup_{W\in\mathcal{W}_j(s,b_s)} [\log f_W(\mathcal{D}) - \log f_\star(\mathcal{D})] \leq -\frac{n(jM\mathsf{r})^2}{8\sigma^2},\ \text{ for all }\ j \geq 1\right\},$$

where $\mathcal{A}(s)$ denotes the complement of $\Theta(s,b_s)$. We note if $(\Lambda,W) \in B^c \cap \Theta(s,b_s)$, then $\Lambda\odot W \in \cup_{j\geq 1}\mathcal{W}_j(s,b_s)$. Let $\check{\Pi}_0$ be the distribution of $\Lambda\odot W$, when $(\Lambda,W)\sim\Pi_0$. Starting from (B.1), and following the same argument leading to (3.23), for $\mathcal{D}\in\mathcal{E}$, we have

$$\begin{aligned}
\Pi(B^c|\mathcal{D}) &\leq 4q^{\bar{\alpha}}\int_{B^c}\frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W)\\
&\leq 4q^{\bar{\alpha}}\left(\int_{\mathcal{A}(s)}\frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) + \int_{B^c\cap\Theta(s,b_s)}\frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W)\right)\\
&\leq 4e^{\bar{\alpha}\log(q)}\left(\frac{1}{q^{\mathsf{u}s}} + \int_{B^c\cap\Theta(s,b_s)}\frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W)\right)\\
&\leq 4e^{\bar{\alpha}\log(q)}\left(\frac{1}{q^{\mathsf{u}s}} + \sum_{j\geq 1}\int_{\mathcal{W}_j(s,b_s)}\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\check{\Pi}_0(\mathrm{d}W)\right)\\
&\leq 4e^{\bar{\alpha}\log(q)}\left(e^{-\mathsf{u}s\log(q)} + \sum_{j\geq 1}e^{-\frac{n(jM\mathsf{r})^2}{8\sigma^2}}\right)\\
&\leq 4e^{\bar{\alpha}\log(q)}\left(e^{-\mathsf{u}s\log(q)} + 2e^{-\frac{n(M\mathsf{r})^2}{8\sigma^2}}\right).
\end{aligned}$$

By the definition of s and r above, we have $\mathsf{u}s = \bar{\alpha} + 1$, and

$$n(M\mathsf{r})^2 \geq \mathsf{M}^2\bar{\sigma}^2\mathsf{s}\log(\mathsf{q}) = \mathsf{M}^2\bar{\sigma}^2\left(\frac{1+\bar{\alpha}}{\mathsf{u}}\right)\log(\mathsf{q}) \geq 8\sigma^2(1+\bar{\alpha})\log(\mathsf{q}),$$

by taking $M^2 \geq 8\mathsf{u}(\sigma^2/\bar{\sigma}^2)$. Hence for $\mathcal{D} \in \mathcal{E}$,

$$\Pi(B^c|\mathcal{D}) \leq \frac{12}{q}.$$

This implies that with probability at least $\mathbb{P}(\mathcal{D} \in \mathcal{E})$, we have

$$\Pi(B^c|\mathcal{D}) \leq \frac{12}{q}.$$

We show in Lemma 29 below that

$$\mathbb{P}\left[\int_\Theta \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda, \mathrm{d}W) \leq \frac{1}{4q^{\bar{\alpha}}} \mid \mathbf{y}_{1:n}\right] \leq \frac{4}{q^{s_\star}},$$

and we show in Lemma 28 below that

$$\mathbb{P}\left[\int_{\mathcal{A}} \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda, \mathrm{d}W) > \frac{1}{q^{\mathsf{u}s}} \mid \mathbf{y}_{1:n}\right] \leq \frac{3}{q^{\mathsf{u}}}.$$

It follows that

$$\begin{aligned}\mathbb{P}(\mathcal{D} \notin \mathcal{E} \mid \mathbf{y}_{1:n}) \leq{}& \frac{4}{q^{s_\star}} + \frac{3}{q^{\mathsf{u}}} \\ &+ \mathbb{P}\left[\bigcup_{j\geq 1}\left\{\sup_{W\in\mathcal{W}_j(s,b_s)} [\log f_W(\mathcal{D}) - \log f_\star(\mathcal{D})] > -\frac{n(jM\mathsf{r})^2}{8\sigma^2}\right\} \mid \mathbf{y}_{1:n}\right].\end{aligned}$$

By Lemma 30 applied with $\mathcal{W}_0 = \mathcal{W}(s, b_s)$, the rightmost term in the last display is bounded from above by $e^{-c_0 n} + 4e^{-n(M\mathsf{r})^2/(\mathsf{c}_0\bar{\sigma}^2)}$, for some absolute constant c_0 provided that the term r defined above satisfies

$$\frac{288}{\sqrt{n}}\int_{\frac{x^2}{32\bar{\varsigma}}}^{x} \sqrt{\log \mathcal{N}\left(\epsilon, \mathcal{W}^{(x)}(s, b_s), \|\cdot\|_n\right)}\mathrm{d}\epsilon \leq \frac{x^2}{\bar{\sigma}}, \quad \text{for all} \quad x \geq \mathsf{r}, \tag{B.3}$$

where for $s_0 \geq 1$, $x \geq 0$, $\beta_0 \geq 0$ we define

$$\mathcal{W}^{(x)}(s_0, \beta_0) \stackrel{\text{def}}{=} \{W \in \mathcal{W}(s_0, \beta_0),\ \|g_W - g\|_n \leq x\},$$

and given $\epsilon > 0$, and $A \subset \mathcal{W}$, $\mathcal{N}(\epsilon, A, \|\cdot\|_n)$ denotes the cardinality of a smallest ϵ-cover of A in the pseudo-metric $\|\cdot\|_n$ defined as $\|W - W'\|_n \stackrel{\text{def}}{=} \|g_W - g_{W'}\|_n$. We therefore reach the conclusion that with probability at least $1 - e^{-c_0 n} - c_1/q$,

$$\Pi(B^c|\mathcal{D}) \leq \frac{12}{q}.$$

for some absolute constants c_0, c_1. It remains to check (B.3). First we use the majoration

$$\int_{\frac{x^2}{32\bar{\varsigma}}}^{x} \sqrt{\log \mathcal{N}\left(\epsilon, \mathcal{W}^{(x)}(s, b_s), \|\cdot\|_n\right)} \mathrm{d}\epsilon \leq x \sqrt{\log \mathcal{N}\left(\frac{x^2}{32\bar{\varsigma}}, \mathcal{W}(s, b_s), \|\cdot\|_n\right)}.$$

We recall that our notation $\|W\|_2$ denotes the Euclidean norm of the vectorized parameter W. For $W \in \mathcal{W}(s, b_s)$, $\|W\|_2 \leq s^{1/2} b_s$. Hence, assumption Assumption 6, and the definition of $\mathsf{L}_s = L(s^{1/2} b_s)$ implies that for all $W, W' \in \mathcal{W}(s, b_s)$, we have

$$\|W - W'\|_n = \|g_W - g_{W'}\|_n \leq \mathsf{L}_s \|W - W'\|_2.$$

Therefore, we can use the metric entropy of the s-sparse ball of $\mathbb{R}^q$ with radius $s^{1/2} b_s / \mathsf{L}_s$ with respect to the Euclidean norm to get

$$\mathcal{N}(\epsilon, \mathcal{W}(s, b_s), \|\cdot\|_n) \leq q^s \left(1 + \frac{2 s^{1/2} b_s \mathsf{L}_s}{\epsilon}\right)^s.$$

Hence

$$\frac{288}{\sqrt{n}} \int_{\frac{x^2}{32\bar{\varsigma}}}^{x} \sqrt{\log \mathcal{N}(\epsilon, \mathcal{W}^{(x)}(s, b_s), \|\cdot\|_n)} \mathrm{d}\epsilon \leq 288 x \sqrt{\frac{s \log(q)}{n} + \frac{s \log\left(1 + \frac{64\bar{\varsigma} s^{1/2} b_s \mathsf{L}_s}{x^2}\right)}{n}}.$$

We can insist to search for $x \geq \sqrt{128\bar{\varsigma}/q}$, and conclude that the right hand side of the last display is always upper bounded by

$$288 x \sqrt{\frac{s \log(q)}{n} + \frac{s \log\left(1 + \frac{q s^{1/2} b_s \mathsf{L}_s}{2}\right)}{n}} \leq c_0 x \sqrt{\frac{s \log(q \mathsf{L}_s)}{n}},$$

for some absolute constant c_0. The right hand side of the last display is upper bounded

by $\frac{x^2}{\bar{\sigma}}$ for all

$$x \geq c_0 \bar{\sigma} \sqrt{\frac{s \log(q \mathsf{L}_s)}{n}},$$

hence the theorem, after moving the constant c_0 into M. □

Lemma 28. *Assume Assumption 4, and suppose that $\sigma^2 \geq \max_i \sigma_i^2$. For all integers $s \geq 1$, with $b_s \stackrel{\text{def}}{=} \rho_1^{-1}(1+\mathsf{u})(s+1)\log(q)$, we have*

$$\mathbb{P}\left[\int_{\mathcal{A}(s)} \frac{f_{\Lambda \odot W}(\mathcal{D})}{f_\star(\mathcal{D})} \Pi_0(\mathrm{d}\Lambda, \mathrm{d}W) > \frac{1}{q^{\mathsf{u}s}} \mid \mathbf{y}_{1:n}\right] \leq \frac{3}{q^{\mathsf{u}}},$$

where $\mathcal{A}(s)$ denotes the complement of the set $\Theta(s, b_s)$ where

$$\Theta(s,b) \stackrel{\text{def}}{=} \{(\Lambda, W) \in \Theta : \ \|\Lambda\|_0 \leq s, \quad \textit{and } \|\Lambda \odot W\|_\infty \leq b\}.$$

Proof. We first write

$$\Pi_0(\mathcal{A}(s)) = \Pi_0(\|\Lambda\|_0 > s) + \sum_{\Lambda:\ \|\Lambda\|_0 \leq s} \Pi_0(\Lambda) \times \Pi_0(\|\Lambda \odot W\|_\infty > b_s | \Lambda).$$

If $(\Lambda, W) \sim \Pi_0$, then Λ is an ensemble of iid random variables drawn from the Bernoulli distribution with success probability $(1+q^{\mathsf{u}+1})^{-1}$. Hence

$$\begin{aligned}\Pi_0(\|\Lambda\|_0 > s) &\leq \sum_{k>s} \binom{q}{k} \left(\frac{1}{1+q^{\mathsf{u}+1}}\right)^k \left(\frac{q^{\mathsf{u}+1}}{1+q^{\mathsf{u}+1}}\right)^{q-k} \\ &\leq \sum_{k>s} \binom{q}{k} \left(\frac{1}{q^{\mathsf{u}+1}}\right)^k \leq 2\left(\frac{1}{q^{\mathsf{u}}}\right)^{s+1},\end{aligned}$$

where we use $\binom{q}{k} \leq q^k$, and $q^{\mathsf{u}} \geq 2$. Given $\Lambda_k = 1$, $|W_k| \sim \mathbf{Exp}(\rho_1)$. Therefore, $\mathbb{P}(|W_k| > t) \leq e^{-\rho_1 t}$ for all $t \geq 0$. Hence by union bound, for $\|\Lambda\|_0 \leq s$, we obtain

$$\Pi_0\left(\|\Lambda \odot W\|_\infty > b_s \mid \Lambda\right) \leq e^{-\rho_1 b_s + \log(s)} \leq \frac{1}{q^{\mathsf{u}(1+s)}}.$$

We conclude that

$$\Pi_0(\mathcal{A}(s)) \leq \frac{3}{q^{\mathsf{u}(1+s)}}. \tag{B.4}$$

By Markov's inequality, and Fubini's theorem, we have

$$\mathbb{P}\left[\int_{\mathcal{A}(s)} \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) > \frac{1}{q^{\mathsf{u}s}} \mid \mathbf{y}_{1:n}\right] \leq q^{\mathsf{u}s}\int_{\mathcal{A}(s)} \mathbb{E}\left[\frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})} \mid \mathbf{y}_{1:n}\right]\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W),$$

and from (B.2) we have

$$\mathbb{E}\left[\frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})} \mid \mathbf{y}_{1:n}\right] = e^{-\frac{n}{2\sigma^2}\|g_{\Lambda\odot W}-g\|_n^2}\mathbb{E}\left[e^{-\frac{1}{\sigma^2}\sum_{i=1}^n\langle \boldsymbol{\xi}_i, g(\mathbf{y}_i)-g_{\Lambda\odot W}(\mathbf{y}_i)\rangle} \mid \mathbf{y}_{1:n}\right].$$

We have assumed in Assumption 4 that $\mathbb{E}(\boldsymbol{\xi}_i|\mathbf{y}_i) = 0$, and $\|\boldsymbol{\xi}_i|\mathbf{y}_i\|_{\psi_2} \leq \sigma_i$. Therefore,

$$\mathbb{E}\left[e^{-\frac{1}{\sigma^2}\sum_{i=1}^n\langle \boldsymbol{\xi}_i, g(\mathbf{y}_i)-g_{\Lambda\odot W}(\mathbf{y}_i)\rangle} \mid \mathbf{y}_{1:n}\right] \leq e^{\frac{1}{\sigma^2}\sum_{i=1}^n \frac{\sigma_i^2 d_i^2}{2\sigma^2}},$$

where d_i is a short for $\|g(\mathbf{y}_i) - g_{\Lambda\odot W}(\mathbf{y}_i)\|_2$. We conclude that

$$\mathbb{E}\left[\frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})} \mid \mathbf{y}_{1:n}\right] \leq \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n\left[\left(1-\frac{\sigma_i^2}{\sigma^2}\right)d_i^2\right]\right).$$

And we easily check that for $\sigma^2 \geq \bar{\sigma}^2$, the right hand size of the last display is bounded from above by 1. We conclude that

$$\mathbb{P}\left[\int_{\mathcal{A}(s)} \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) > \frac{1}{q^{\mathsf{u}s}} \mid \mathbf{y}_{1:n}\right] \leq q^{\mathsf{u}s}\Pi_0(\mathcal{A}) \leq \frac{3}{q^{\mathsf{u}}}.$$

□

The next result lower bounds the normalizing constant of $\Pi(\cdot|\mathcal{D})$.

Lemma 29. *Under the assumption of Theorem 10 it holds,*

$$\mathbb{P}\left[\int_{\Theta} \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) \leq \frac{1}{4q^{\bar{\alpha}}} \mid \mathbf{y}_{1:n}\right] \leq \frac{4}{q^{s_\star}},$$

where

$$\bar{\alpha} \stackrel{\text{def}}{=} \left(\mathsf{u}+5+\frac{\log(n\mathsf{L}_\star)}{\log(q)}\right)s_\star + \frac{4n\varpi_\star^2}{\sigma^2\log(q)}.$$

Proof. By the assumption of Theorem 10, we can find $W_\star$ such that $\|W_\star\|_0 \leq s_\star$,

$\|W_\star\|_\infty \leq \beta_\star$, and $\|g_{W_\star} - g\|_n \leq \varpi_\star$. Let $\Lambda_\star$ denote the sparsity structure of $W_\star$. We set

$$\eta \stackrel{\text{def}}{=} 1 \wedge \frac{\sigma}{\mathsf{L}_\star}\sqrt{\frac{\log(q)}{n}},$$

and

$$\mathcal{N}(\eta) \stackrel{\text{def}}{=} \{W \in \mathcal{W} : \qquad \|W \odot \Lambda_\star - W_\star\|_\infty \leq \eta\}.$$

By definition we have $\eta \leq 1 \leq \beta_\star$. We then have $\|W_\star\|_2 \leq s_\star^{1/2}\beta_\star$. Similarly, for $W \in \mathcal{N}(\eta)$,

$$\|W \odot \Lambda_\star\|_2 \leq \|W_\star\|_2 + \|W_\star - W \odot \Lambda_\star\|_2 \leq s_\star^{1/2}\beta_\star + s_\star^{1/2}\eta \leq 2s_\star^{1/2}\beta_\star.$$

Therefore, by Assumption 6, and with $\mathsf{L}_\star = L(2s_\star^{1/2}\beta_\star)$, for all $W \in \mathcal{N}(\eta)$, we have

$$\max_{1\leq i\leq n} \|g_{\Lambda_\star \odot W}(\mathbf{y}_i) - g_{W_\star}(\mathbf{y}_i)\|_2 \leq \mathsf{L}_\star\|\Lambda_\star \odot W - W_\star\|_2 \leq \mathsf{L}_\star\sqrt{s_\star}\eta \leq \sigma\sqrt{\frac{s_\star \log(q)}{n}}.$$

Hence

$$\max_{1\leq i\leq n} \sup_{W\in\mathcal{N}(\eta)} \|g_{\Lambda_\star \odot W}(\mathbf{y}_i) - g_{W_\star}(\mathbf{y}_i)\|_2 \leq \sigma\sqrt{\frac{s_\star \log(q)}{n}}. \tag{B.5}$$

Switching the sign and taking the conditional expectation in (B.2) using $\mathbb{E}(\mathbf{x}_i|\mathbf{y}_i) = g(\mathbf{y}_i)$, yields

$$\mathbb{E}\left[\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda_\star\odot W}(\mathcal{D})}\right) \mid \mathbf{y}_{1:n}\right] = \frac{1}{2\sigma^2}\sum_{i=1}^n \|g_{\Lambda_\star\odot W}(\mathbf{y}_i) - g(\mathbf{y}_i)\|^2,$$

and we conclude using (B.5) and the definition of $\varpi_\star$ and $\mathcal{W}_\star$ in Theorem 10 that

$$\begin{aligned}\sup_{W\in\mathcal{N}(\eta)} \mathbb{E}\left[\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda_\star\odot W}(\mathcal{D})}\right) \mid \mathbf{y}_{1:n}\right] &\leq \frac{n\varpi_\star^2}{\sigma^2} + \frac{1}{\sigma^2}\sum_{i=1}^n \sup_{W\in\mathcal{N}(\eta)} \|g_{\Lambda_\star\odot W}(\mathbf{y}_i) - g_{W_\star}(\mathbf{y}_i)\|^2 \\ &\leq \frac{n\varpi_\star^2}{\sigma^2} + s_\star\log(q).\end{aligned}$$

Going back to (B.2), we have

$$\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda_\star\odot W}(\mathcal{D})}\right) - \mathbb{E}\left[\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda_\star\odot W}(\mathcal{D})}\right) \mid \mathbf{y}_{1:n}\right] = \frac{1}{\sigma^2}\sum_{i=1}^n \langle \boldsymbol{\xi}_i, g(\mathbf{y}_i) - g_{\Lambda_\star\odot W}(\mathbf{y}_i)\rangle\,. \tag{B.6}$$

We use the notation $\|Z\|_{\psi_2}$ to denote the sub-Gaussian norm of the conditional law of the random variable Z given $\mathbf{y}_{1:n}$. By conditional independence of the error terms $\boldsymbol{\xi}_i$, for all $W \in \mathcal{N}(\eta)$, we have

$$\begin{aligned}
&\left\|\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda_\star\odot W}(\mathcal{D})}\right) - \mathbb{E}\left[\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda_\star\odot W}(\mathcal{D})}\right) \mid \mathbf{y}_{1:n}\right]\right\|_{\psi_2}^2 \\
&\qquad \le \frac{1}{\sigma^4}\sum_{i=1}^n \|\langle \boldsymbol{\xi}_i, g(\mathbf{y}_i) - g_{\Lambda_\star\odot W}(\mathbf{y}_i)\rangle\|_{\psi_2}^2 \\
&\qquad = \frac{1}{\sigma^4}\sum_{i=1}^n \sigma_i^2 \|g(\mathbf{y}_i) - g_{\Lambda_\star\odot W}(\mathbf{y}_i)\|_2^2 \le \frac{2n\varpi_\star^2}{\sigma^2} + 2s_\star\log(q).
\end{aligned}$$

In the sequel, we set

$$a \stackrel{\text{def}}{=} 2\left(\frac{n\varpi_\star^2}{\sigma^2} + s_\star\log(q)\right).$$

Then by Hoeffding's inequality (see e.g. Theorem 2.2.2 of Vershynin (2018)), for all $W \in \mathcal{N}(\eta)$, we have

$$\mathbb{P}\left[\left|\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda_\star\odot W}(\mathcal{D})}\right) - \mathbb{E}\left[\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda_\star\odot W}(\mathcal{D})}\right) \mid \mathbf{y}_{1:n}\right]\right| > a \mid \mathbf{y}_{1:n}\right] \le 2e^{-a/2} \le \frac{2}{q^{s_\star}}.$$

We can rewrite this statement in the following equivalent form. For $W \in \mathcal{W}$, define

$$\mathcal{E}_W \stackrel{\text{def}}{=} \left\{\mathcal{D} :\ \left|\log\left(\frac{f_\star(\mathcal{D})}{f_W(\mathcal{D})}\right) - \mathbb{E}\left[\log\left(\frac{f_\star(\mathcal{D})}{f_W(\mathcal{D})}\right) \mid \mathbf{y}_{1:n}\right]\right| \le a\right\}.$$

We have

$$\sup_{W\in\mathcal{N}(\eta)} \mathbb{P}\left(\mathcal{D} \notin \mathcal{E}_{\Lambda_\star\odot W} \mid \mathbf{y}_{1:n}\right) \le \frac{2}{q^{s_\star}}. \tag{B.7}$$

Using these observations, we have

$$\begin{aligned}\int_\Theta \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) &\geq \int_{\Lambda_\star\times\mathcal{N}(\eta)} e^{-\mathbb{E}\left[\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda\odot W}(\mathcal{D})}\right)\mid \mathbf{y}_{1:n}\right]} \\ \times \exp\left(-\left[\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda\odot W}(\mathcal{D})}\right) - \mathbb{E}\left[\log\left(\frac{f_\star(\mathcal{D})}{f_{\Lambda\odot W}(\mathcal{D})}\right)\mid \mathbf{y}_{1:n}\right]\right]\right)&\mathbf{1}_{\mathcal{E}_{\Lambda\odot W}}(\mathcal{D})\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) \\ &\geq e^{-2a}\int_{\Lambda_\star\times\mathcal{N}(\eta)}\mathbf{1}_{\mathcal{E}_{\Lambda\odot W}}(\mathcal{D})\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) \\ &= e^{-2a}\left(\Pi_0(\Lambda_\star\times\mathcal{N}(\eta)) - \int_{\Lambda_\star\times\mathcal{N}(\eta)}\mathbf{1}_{\mathcal{E}^c_{\Lambda\odot W}}(\mathcal{D})\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W)\right).\end{aligned}$$

Therefore, by Chebyshev's inequality,

$$\begin{aligned}&\mathbb{P}\left[\int_\Theta \frac{f_{\Lambda\odot W}(\mathcal{D})}{f_\star(\mathcal{D})}\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) \leq e^{-2a}\Pi_0(\Lambda_\star\times\mathcal{N}(\eta))/2 \mid \mathbf{y}_{1:n}\right] \\ &\quad\leq \mathbb{P}\left[\int_{\Lambda_\star\times\mathcal{N}(\eta)}\mathbf{1}_{\mathcal{E}^c_{\Lambda\odot W}}(\mathcal{D})\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) \geq \frac{1}{2}\Pi_0(\Lambda_\star\times\mathcal{N}(\eta)) \mid \mathbf{y}_{1:n}\right] \\ &\quad\leq \frac{2}{\Pi_0(\Lambda_\star\times\mathcal{N}(\eta))}\int_{\Lambda_\star\times\mathcal{N}(\eta)}\mathbb{P}\left(\mathcal{D}\notin\mathcal{E}_{\Lambda_\star\odot W}\mid\mathbf{y}_{1:n}\right)\Pi_0(\mathrm{d}\Lambda,\mathrm{d}W) \\ &\quad\leq 2\sup_{W\in\mathcal{N}(\eta)}\mathbb{P}_\star\left(\mathcal{D}\notin\mathcal{E}_{\Lambda_\star\odot W}\mid\mathbf{y}_{1:n}\right) \leq \frac{4}{q^{s_\star}},\end{aligned}$$

using (B.7). To conclude the proof we need to lower bound $\Pi_0(\Lambda_\star\times\mathcal{N}(\eta))$. Since $\log(1-x)\geq -2x$ for all $0\leq x\leq 1/2$, for $q^{\mathsf{u}}\geq 2/\log(2)$, we have

$$\begin{aligned}\Pi_0(\Lambda_\star) &= \left(\frac{1}{1+q^{\mathsf{u}+1}}\right)^{\|\Lambda_\star\|_0}\left(1-\frac{1}{1+q^{\mathsf{u}+1}}\right)^{q-\|\Lambda_\star\|_0} \\ &= \left(\frac{1}{q^{\mathsf{u}+1}}\right)^{\|\Lambda_\star\|_0}\exp\left(q\log\left(1-\frac{1}{1+q^{\mathsf{u}+1}}\right)\right) \\ &\geq \left(\frac{1}{q^{\mathsf{u}+1}}\right)^{\|\Lambda_\star\|_0}\exp\left(-\frac{2q}{1+q^{\mathsf{u}+1}}\right) \geq \frac{1}{2}\left(\frac{1}{q^{\mathsf{u}+1}}\right)^{\|\Lambda_\star\|_0} \geq \frac{1}{2}\left(\frac{1}{q^{\mathsf{u}+1}}\right)^{s_\star}.\end{aligned}$$

If $U\sim\textbf{Laplace}(\rho_1)$, then $P(|U-a|\leq t)\geq P(|a|\leq U\leq |a|+t)$ for all $t\geq 0$. We use this inequality to deduce that

$$\Pi_0(\mathcal{N}(\eta)\mid\Lambda_\star) \geq \left(\int_{\beta_\star}^{\beta_\star+\eta}\frac{\rho_1}{2}e^{-\rho_1 t}\mathrm{d}t\right)^{\|\Lambda_\star\|_0} \geq \left(\frac{e^{-\rho_1\beta_\star}}{2}\left(1-e^{-\rho_1\eta}\right)\right)^{s_\star} \geq \left(\frac{\rho_1\eta}{4\exp(1)}\right)^{s_\star}.$$

The last inequality in the last display uses the assumption that $\rho_1\beta_\star \leq 1$, $\rho_1\eta \leq \rho_1 \leq \log(2)$, and the observation that for all $0 \leq x \leq \log(2)$, $1 - e^{-x} \geq x/2$. We conclude that

$$
\begin{aligned}
e^{-2a}\Pi_0(\Lambda_\star \times \mathcal{N}(\eta)) & \\
&\geq \frac{1}{2}\exp\left(-\frac{4n\varpi_\star^2}{\sigma^2} - 4s_\star\log(q) - (\mathsf{u}+1)s_\star\log(q) - s_\star\log\left(\frac{4e^1}{\rho_1\eta}\right)\right), \\
&\geq \frac{1}{2}\exp\left(-\frac{4n\varpi_\star^2}{\sigma^2} - (\mathsf{u}+5)s_\star\log(q) - s_\star\log(n\mathsf{L}_\star)\right),
\end{aligned}
$$

where we use the assumption that $12/\rho_1 \leq \sqrt{n}\min(\sigma, \sqrt{n})$, which yields $4e^1/(\rho_1\eta) \leq n\mathsf{L}_\star$. Hence the result. □

Lemma 30. *Assume Assumption 4. Let $\mathcal{W}_0$ be some subset of $\mathcal{W}$. Suppose that we can find $\mathsf{r} > 0$ such that for all $x \geq \mathsf{r}$, it holds*

$$
\frac{288}{\sqrt{n}}\int_{\frac{x^2}{16\bar{\sigma}}}^{x}\sqrt{\log\mathcal{N}(\epsilon, \mathcal{W}^{(x)}, \|\cdot\|_n)}\mathrm{d}\epsilon \leq \frac{x^2}{\bar{\sigma}}, \tag{B.8}
$$

where $\mathcal{W}^{(x)} \stackrel{\text{def}}{=} \{W \in \mathcal{W}_0,\ \|g_W - g\|_n \leq x\}$. Then there exists an absolute constant c_0 such that for all $M \geq 1$, such that $n(M\mathsf{r})^2 \geq \mathsf{c_0}\bar{\sigma}^2$, it holds

$$
\mathbb{P}\left[\bigcup_{j\geq 1}\left\{\sup_{W\in\widetilde{\mathcal{W}}^{(j)}}\log\left(\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\right) > -\frac{n(jM\mathsf{r})^2}{8\sigma^2}\right\} \mid \mathbf{y}_{1:n}\right] \leq e^{-c_0 n} + 4e^{-\frac{nM^2\mathsf{r}^2}{\mathsf{c_0}\bar{\sigma}^2}},
$$

where $\widetilde{\mathcal{W}}^{(j)} \stackrel{\text{def}}{=} \{W \in \mathcal{W}_0 :\ jM\mathsf{r} < \|\mathsf{g_W} - \mathsf{g}\|_\mathsf{n} \leq (\mathsf{j}+1)\mathsf{Mr}\}$.

Proof. We proceed as in Lemma 3.2 of Geer et al. (2000). Throughout the proof, all expectations and probability are conditional given $\mathbf{y}_{1:n}$. However to ease notation we omit the conditioning. Let M and r as in the statement, and for each integer j, we set $\mathsf{r_j} = \mathsf{Mrj}$. We recall the definition of the error terms $\boldsymbol{\xi}_i \stackrel{\text{def}}{=} \mathbf{x}_i - g(\mathbf{y}_i)$, and we define

$$
Z_n(g_W) \stackrel{\text{def}}{=} \frac{1}{n\sigma^2}\sum_{i=1}^{n}\langle\boldsymbol{\xi}_i, g(\mathbf{y}_i) - g_W(\mathbf{y}_i)\rangle, \quad W \in \mathcal{W}.
$$

Using (B.2) we can re-express the log-likelihood ratio as

$$
\log\left(\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\right) = -\frac{n}{2\sigma^2}\|g_W - g\|_n^2 - nZ_n(g_W). \tag{B.9}
$$

Let ς_i denote the sub-Gaussian norm of $\|\boldsymbol{\xi}_i\|_2$,

$$\bar{\varsigma} \stackrel{\text{def}}{=} \max_{1\le i\le n} \varsigma_i.$$

The sub-Gaussian assumption on $\boldsymbol{\xi}_i$ implies that $\varsigma_i < \infty$ (see e.g. Theorem 6.3.2 of Vershynin (2018)), and that $\|\boldsymbol{\xi}_i\|_2^2$ is sub-exponential, with sub-exponential norm ς_i^2. We note also that $\mathbb{E}(\|\boldsymbol{\xi}_i\|_2^2) \le 2\varsigma_i^2$. Therefore, by Bernstein inequality (see e.g. Theorem 2.8.1 of Vershynin (2018)),

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^n \|\boldsymbol{\xi}_i\|_2^2 > 3\bar{\varsigma}^2\right) \le \mathbb{P}\left(\sum_{i=1}^n \|\boldsymbol{\xi}_i\|_2^2 - \mathbb{E}(\|\boldsymbol{\xi}_i\|_2^2) > n\bar{\varsigma}^2\right) \le e^{-c_0 n},$$

for some absolute constant c_0. To make use of this bound, we define

$$\mathcal{F}_0 \stackrel{\text{def}}{=} \left\{\mathcal{D} :\ \sum_{i=1}^n \|\boldsymbol{\xi}_i\|_2^2 \le 3n\bar{\varsigma}^2\right\}.$$

Therefore,

$$\mathbb{P}\left[\bigcup_{j\ge 1}\left\{\sup_{W\in\widetilde{\mathcal{W}}^{(j)}} \log\left(\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\right) > -\frac{n\mathsf{r}_{\mathsf{j}}^2}{8\sigma^2}\right\}\right] \le e^{-c_0 n} + \sum_{j\ge 1}\mathbb{P}\left[\mathcal{F}_j\right],$$

where

$$\mathcal{F}_j \stackrel{\text{def}}{=} \mathcal{F}_0 \bigcap \left\{\sup_{W\in\widetilde{\mathcal{W}}^{(j)}} \log\left(\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\right) > -\frac{n\mathsf{r}_{\mathsf{j}}^2}{8\sigma^2}\right\}.$$

For each $j \ge 1$, we set

$$\mathcal{W}^{(j)} \stackrel{\text{def}}{=} \{W \in \mathcal{W}_0 :\ \|g_W - g\|_n \le \mathsf{r}_{\mathsf{j}+1}\},$$

and each $\iota = 1, \ldots$, let $\mathcal{C}_j^{(\iota)} \stackrel{\text{def}}{=} \{g_{j,1}^{(\iota)}, \ldots, g_{j,N_{j,\iota}}^{(\iota)}\}$ be a $(\mathsf{r}_{\mathsf{j}+1}2^{-\iota})$-covering of $\mathcal{W}^{(j)}$. For $\iota = 0$, we set $\mathcal{C}_j^{(0)} = \{g\}$. The definition implies that for any $W \in \mathcal{W}^{(j)}$, we can find $g_{j,W}^{(\iota)} \in \mathcal{C}_j^{(\iota)}$ such that $\|g_W - g_{j,W}^{(\iota)}\|_n \le \mathsf{r}_{\mathsf{j}+1}2^{-\iota}$. Let $\ell_j \ge 0$, be the smallest integer such that

$$\frac{\mathsf{r}_{\mathsf{j}+1}}{2^{\ell_j}} \le \frac{\mathsf{r}_{\mathsf{j}}^2}{16\bar{\varsigma}}.$$

We consider separately the cases $\ell_j = 0$ and $\ell_j > 0$.

`Suppose` $\ell_j = 0$. In that case for any $W \in \mathcal{W}^{(j)}$, $\|g_W - g\|_n \leq \mathsf{r}_{\mathsf{j}+1} \leq \mathsf{r}_{\mathsf{j}}^2/(16\bar{\varsigma})$. Therefore, on the event $\mathcal{F}_0$, we have

$$\sup_{W \in \mathcal{W}_j} |Z_n(g_W)| \leq \frac{1}{n\sigma^2} \sum_{i=1}^n \|\boldsymbol{\xi}_i\|_2 \|g_W(\mathbf{y}_i) - g(\mathbf{y}_i)\|_2 \leq \frac{\sqrt{3\bar{\varsigma}^2}}{\sigma^2} \|g_W - g\|_n \\ \leq \frac{\sqrt{3\bar{\varsigma}^2}}{\sigma^2} \frac{\mathsf{r}_{\mathsf{j}}^2}{16\bar{\varsigma}} \leq \frac{\mathsf{r}_{\mathsf{j}}^2}{8\sigma^2}.$$

Taking this conclusion to (B.9) implies that on $\mathcal{F}_0$,

$$\sup_{W \in \widetilde{\mathcal{W}}^{(j)}} \log\left(\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\right) \leq -\frac{n\mathsf{r}_{\mathsf{j}}^2}{2\sigma^2} + n \sup_{W \in \mathcal{W}_j} |Z_n(g_W)| \leq -\frac{n\mathsf{r}_{\mathsf{j}}^2}{4\sigma^2}.$$

Hence, when $\ell_j = 0$, $\mathbb{P}(\mathcal{F}_j) = 0$.

`Suppose` $\ell_j > 0$. Similarly, on the event $\mathcal{F}_0$, we have

$$\left|Z_n(g_W) - Z_n(g_{j,W}^{(\ell_j)})\right| \leq \frac{1}{n\sigma^2} \sum_{i=1}^n \|\boldsymbol{\xi}_i\|_2 \|g_{j,W}^{(\ell_j)}(\mathbf{y}_i) - g_W(\mathbf{y}_i)\|_2 \\ \leq \frac{\sqrt{3\bar{\varsigma}^2}}{\sigma^2} \|g_{j,W}^{(\ell_j)} - g_W\|_n \leq \frac{\sqrt{3\bar{\varsigma}^2}}{\sigma^2} \frac{\mathsf{r}_{\mathsf{j}+1}}{2^{\ell_j}} \leq \frac{\sqrt{3\bar{\varsigma}^2}}{\sigma^2} \frac{\mathsf{r}_{\mathsf{j}}^2}{16\bar{\varsigma}} \leq \frac{\mathsf{r}_{\mathsf{j}}^2}{8\sigma^2}.$$

This implies that on $\mathcal{F}_0$,

$$\sup_{W \in \widetilde{\mathcal{W}}^{(j)}} \log\left(\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\right) \leq -\frac{n\mathsf{r}_{\mathsf{j}}^2}{2\sigma^2} + n \sup_{W \in \mathcal{W}^{(j)}} \left|Z_n(g_W) - Z_n(g_{j,W}^{(\ell_j)})\right| + n \sup_{W \in \mathcal{W}^{(j)}} \left|Z_n(g_{j,W}^{(\ell_j)})\right| \\ \leq -\frac{3n\mathsf{r}_{\mathsf{j}}^2}{8\sigma^2} + n \sup_{W \in \mathcal{W}^{(j)}} \left|Z_n(g_{j,W}^{(\ell_j)})\right|.$$

Hence

$$\mathbb{P}(\mathcal{F}_j) \leq \mathbb{P}\left[\sup_{W \in \mathcal{W}^{(j)}} \left|Z_n(g_{j,W}^{(\ell_j)})\right| > \frac{\mathsf{r}_{\mathsf{j}}^2}{4\sigma^2}\right].$$

To bound this latter term we introduce

$$\delta_j \stackrel{\text{def}}{=} \int_{\frac{\mathsf{r}_{\mathsf{j}+1}^2}{64\zeta}}^{\mathsf{r}_{\mathsf{j}+1}} \sqrt{\log \mathcal{N}\left(\epsilon, \mathcal{W}^{(j)}, \|\cdot\|_n\right)} \mathrm{d}\epsilon,$$

$$\text{and} \quad \eta_{j,\iota} \stackrel{\text{def}}{=} \max\left(\frac{1}{6}\frac{\iota^{1/2}}{2^\iota}, \frac{\sqrt{\log N_{j,\iota}}}{4\delta_j}\frac{\mathsf{r}_{\mathsf{j}+1}}{2^\iota}\right), \quad \iota = 1, \ldots, \ell_j.$$

and we write $g_{j,W}^{(\ell_j)}$ as a telescoping sum

$$g_{j,W}^{(\ell_j)} - g = \sum_{\iota=1}^{\ell_j} g_{j,W}^{(\iota)} - g_{j,W}^{(\iota-1)},$$

so that

$$\sup_{W\in\mathcal{W}^{(j)}} \left|Z_n(g_{j,W}^{(\ell_j)})\right| \le \sum_{\iota=1}^{\ell_j} \sup_{W\in\mathcal{W}^{(j)}} \left|\frac{1}{n\sigma^2}\sum_{i=1}^n \left\langle \boldsymbol{\xi}_i, g_{j,W}^{(\iota-1)}(\mathbf{y}_i) - g_{j,W}^{(\iota)}(\mathbf{y}_i)\right\rangle\right|.$$

We show below that the sequence $\{\eta_{j,\iota},\ \iota = 1, \ldots, \ell_j\}$ introduced above satisfies

$$\sum_{\iota=1}^{\ell_j} \eta_{j,\iota} \le 1. \tag{B.10}$$

Due to (B.10), we can use the sequence $\{\eta_{j,\iota},\ \iota = 1, \ldots, \ell_j\}$ to say that

$$\mathbb{P}\left[\sup_{W\in\mathcal{W}^{(j)}} \left|Z_n(g_{j,W}^{(\ell_j)})\right| > \frac{\mathsf{r}_{\mathsf{j}}^2}{4\sigma^2}\right]$$
$$\le \sum_{\iota=1}^{\ell_j} \mathbb{P}\left[\sup_{W\in\mathcal{W}^{(j)}} \left|\frac{1}{n\sigma^2}\sum_{i=1}^n \left\langle \boldsymbol{\xi}_i, g_{j,W}^{(\iota-1)}(\mathbf{y}_i) - g_{j,W}^{(\iota)}(\mathbf{y}_i)\right\rangle\right| > \frac{\eta_{j,\iota}\mathsf{r}_{\mathsf{j}}^2}{4\sigma^2}\right].$$

The supremum on the right-hand side of the last display is in fact a max over a finite set of cardinality at most $N_{j,\iota-1} \times N_{j,\iota} \le N_{j,\iota}^2$, and for $W \in \mathcal{W}^{(j)}$,

$$\frac{1}{n^2\sigma^4}\sum_{i=1}^n \sigma_i^2 \|g_{j,W}^{(\iota-1)}(\mathbf{y}_i) - g_{j,W}^{(\iota)}(\mathbf{y}_i)\|_2^2$$
$$\le \frac{2\max_i \sigma_i^2}{n^2\sigma^4}\left(n\|g_W - g_{j,W}^{(\iota)}\|_n^2 + n\|g_W - g_{j,W}^{(\iota-1)}\|_n^2\right) \le \frac{10}{n}\max_i\left(\frac{\sigma_i^2}{\sigma^4}\right)\frac{\mathsf{r}_{\mathsf{j}+1}^2}{2^{2\iota}}.$$

Therefore by Hoefdding's inequality,

$$\mathbb{P}\left[\sup_{W\in\mathcal{W}^{(j)}}\left|Z_n(g_{j,W}^{(\ell_j)})\right|>\frac{\mathsf{r}_{\mathsf{j}}^2}{4\sigma^2}\right]\leq\sum_{\iota=1}^{\ell_j}\exp\left(2\log N_{j,\iota}-\frac{n2^{2\iota}\eta_{j,\iota}^2\mathsf{r}_{\mathsf{j}}^4}{(20\times16)\bar{\sigma}^2\mathsf{r}_{\mathsf{j+1}}^2}\right).$$

By construction,

$$\frac{2^{2\iota}\eta_{j,\iota}^2}{\mathsf{r}_{\mathsf{j+1}}^2}\geq\frac{\log N_{j\iota}}{16\delta_j^2},$$

which gives

$$\frac{n2^{2\iota}\eta_{j,\iota}^2\mathsf{r}_{\mathsf{j}}^4}{(20\times16)\bar{\sigma}^2\mathsf{r}_{\mathsf{j+1}}^2}\geq\frac{n\mathsf{r}_{\mathsf{j}}^4}{(20\times32^2)\bar{\sigma}^2\delta_j^2}\times(4\log N_{j\iota})\geq4\log N_{j\iota},$$

using (B.8). Therefore

$$2\log N_{j,\iota}-\frac{n2^{2\iota}\eta_{j,\iota}^2\mathsf{r}_{\mathsf{j}}^4}{(20\times16)\bar{\sigma}^2\mathsf{r}_{\mathsf{j+1}}^2}\leq-\frac{n2^{2\iota}\eta_{j,\iota}^2\mathsf{r}_{\mathsf{j}}^4}{(20\times32)\bar{\sigma}^2\mathsf{r}_{\mathsf{j+1}}^2}\leq-\frac{n\mathsf{r}_{\mathsf{j}}^2\iota}{(80\times36\times32)\bar{\sigma}^2},$$

where the last inequality uses the fact that $2^{2\iota}\eta_{j,\iota}^2\geq\iota/36$. It follows that

$$\mathbb{P}\left[\sup_{W\in\mathcal{W}^{(j)}}\left|Z_n(g_{j,W}^{(\ell_j)})\right|>\frac{\mathsf{r}_{\mathsf{j}}^2}{4\sigma^2}\right]\leq\sum_{\iota=1}^{\ell_j}\exp\left(-\frac{n\mathsf{r}_{\mathsf{j}}^2\iota}{c_0\bar{\sigma}^2}\right)\leq2\exp\left(-\frac{n\mathsf{r}_{\mathsf{j}}^2}{c_0\bar{\sigma}^2}\right),$$

since $n\mathsf{r}_{\mathsf{j}}^2\geq\mathsf{c}_0\bar{\sigma}^2\log(2)$, for some constant c_0 that can be taken as $c_0=80\times36\times32$. In conclusion,

$$\mathbb{P}\left[\bigcup_{j\geq1}\left\{\sup_{W\in\widetilde{\mathcal{W}}^{(j)}}\log\left(\frac{f_W(\mathcal{D})}{f_\star(\mathcal{D})}\right)>-\frac{n\mathsf{r}_{\mathsf{j}}^2}{8\sigma^2}\right\}\right]$$
$$\leq e^{-c_0n}+2\sum_{j\geq1}\exp\left(-\frac{n\mathsf{r}_{\mathsf{j}}^2}{c_0\bar{\sigma}^2}\right)\leq e^{-c_0n}+4\exp\left(-\frac{nM\mathsf{r}^2}{c_0\bar{\sigma}^2}\right).$$

To check (B.10), we note

$$\sum_{\iota=1}^{\ell_j}\eta_{j,\iota}\leq\frac{1}{6}\sum_{\iota=1}^{\ell_j}\frac{\iota^{1/2}}{2^\iota}+\frac{1}{4\delta_j}\sum_{\iota=1}^{\ell_j}\frac{\mathsf{r}_{\mathsf{j+1}}}{2^\iota}\sqrt{\log N_{j,\iota}}.$$

The function $h(x)=x^{1/2}2^{-x}=x^{\alpha-1}e^{-\beta x}$, with $\alpha=3/2$, $\beta=\log(2)$ is decreasing for

$x \geq 1$. Hence

$$\sum_{\iota \geq 1} \frac{\iota^{1/2}}{2^{\iota}} = \frac{1}{2} + \sum_{\iota \geq 2} h(\iota) \leq \frac{1}{2} + \sum_{k \geq 2} \int_{k-1}^{k} h(x) \mathrm{d}x \leq \frac{1}{2} + \int_{1}^{\infty} x^{\alpha-1} e^{\beta x} \mathrm{d}x \leq 3.$$

Whereas,

$$\begin{aligned}
\sum_{\iota=1}^{\ell_j} \frac{\mathsf{r}_{\mathsf{j}+1}}{2^{\iota}} \sqrt{\log N_{j,\iota}} &= \sum_{\iota=1}^{\ell_j} 2 \int_{\frac{\mathsf{r}_{\mathsf{j}+1}}{2^{\iota+1}}}^{\frac{\mathsf{r}_{\mathsf{j}+1}}{2^{\iota}}} \sqrt{\log \mathcal{N}\left(\frac{\mathsf{r}_{\mathsf{j}+1}}{2^{\iota}}, \mathcal{W}^{(j)}, \|\cdot\|_n\right)} \mathrm{d}\epsilon \\
&\leq 2 \int_{\frac{\mathsf{r}_{\mathsf{j}+1}}{2^{\ell_j+1}}}^{\frac{\mathsf{r}_{\mathsf{j}+1}}{2}} \sqrt{\log \mathcal{N}\left(\epsilon, \mathcal{W}^{(j)}, \|\cdot\|_n\right)} \mathrm{d}\epsilon \\
&\leq 2 \int_{\frac{r_{\mathsf{j}+1}^2}{64\varsigma}}^{\mathsf{r}_{\mathsf{j}+1}} \sqrt{\log \mathcal{N}\left(\epsilon, \mathcal{W}^{(j)}, \|\cdot\|_n\right)} \mathrm{d}\epsilon = 2\delta_j.
\end{aligned}$$

□

B.4.2 Proof of Theorem 8

We apply Theorem 10. The argument has two main steps. First, we show that the function g can be well approximated by elements of the function class $\{g_W,\ W \in \mathcal{W}\}$ constructed in (3.11), and secondly we show that the functions g_W are locally Lipschitz. Both steps rely on a well-known telescoping argument that we outline first (see e.g. Proposition 6 of Taheri et al. (2021)). Given two functions $f = f_K \circ \cdots \circ f_1$, and $g = g_K \circ \cdots \circ g_1$, we write $f - g$ as a telescoping sum

$$f(\mathbf{x}) - g(\mathbf{x}) = \sum_{j=1}^{K} f_K \circ \cdots \circ f_j \left(g_{j-1} \circ \cdots \circ g_1(\mathbf{x})\right) - f_K \circ \cdots \circ f_{j+1} \circ g_j \left(g_{j-1} \circ \cdots \circ g_1(\mathbf{x})\right), \tag{B.11}$$

with the convention that for $j = 1$, $g_{j-1} \circ \cdots \circ g_1$ is the identity map, and for $j = K$, $f_K \circ \cdots \circ f_{j+1}$ is the identity map. A bound on $\|f(\mathbf{x}) - g(\mathbf{x})\|$ can then be derived using the Lipschitz and boundedness properties of the functions f_j, g_j.

Specifically, define $H_W^{(0)}(\mathbf{x}) \stackrel{\text{def}}{=} \mathbf{x}$, and define $H_W^{(\ell)}(\mathbf{x}) \stackrel{\text{def}}{=} \Psi_{W_\ell}^{(\ell)}(H_W^{(\ell-1)}(\mathbf{x}))$ for $1 \leq$

$\ell \leq D$, so that $H_W(\mathbf{x}) = H_W^{(D)}(\mathbf{x})$. We recall that

$$\Psi_M^{(\ell)}(\mathbf{x}) = \mathsf{a}_\ell(M\mathbf{x}),$$

where the activation functions $\mathsf{a}_\ell : \mathbb{R}^{p_\ell} \to \mathbb{R}^{p_\ell}$ are Lipschitz with constant 1. Then for $1 \leq \ell \leq D$, and all $W, W' \in \mathcal{W}$, $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^{d_x}$, by the Lipschitz property of the activation functions a_ℓ, we have

$$\begin{aligned}\|H_W^{(\ell)}(\mathbf{x}_1) - H_W^{(\ell)}(\mathbf{x}_2)\|_2 &\leq \|W_\ell H_W^{(\ell-1)}(\mathbf{x}_1) - W_\ell H_W^{(\ell-1)}(\mathbf{x}_2)\|_2 \\ &\leq \|W_\ell\|_{\mathsf{op}} \|H_W^{(\ell-1)}(\mathbf{x}_1) - H_W^{(\ell-1)}(\mathbf{x}_2)\|_2,\end{aligned}$$

where $\|\cdot\|_{\mathsf{op}}$ denotes the operator norm. Iterating this yields,

$$\|H_W^{(\ell)}(\mathbf{x}_1) - H_W^{(\ell)}(\mathbf{x}_2)\|_2 \leq \prod_{j=1}^{\ell} \|W_j\|_{\mathsf{op}} \|\mathbf{x}_1 - \mathbf{x}_2\|_2. \tag{B.12}$$

Similarly, for any $1 \leq \ell \leq D$, (B.11) gives

$$\begin{aligned}H_W^{(\ell)}(\mathbf{x}) - H_{W'}^{(\ell)}(\mathbf{x}) = \sum_{j=1}^{\ell} &\Psi_{W_\ell}^{(\ell)} \circ \cdots \circ \Psi_{W_{j+1}}^{(j+1)} \circ \Psi_{W_j}^{(j)} \circ \left(\Psi_{W'_{j-1}}^{(j-1)} \circ \cdots \circ \Psi_{W'_1}^{(1)}(\mathbf{x})\right) \\ &- \Psi_{W_\ell}^{(\ell)} \circ \cdots \circ \Psi_{W_{j+1}}^{(j+1)} \circ \Psi_{W'_j}^{(j)} \left(\Psi_{W'_{j-1}}^{(j-1)} \circ \cdots \circ \Psi_{W'_1}^{(1)}(\mathbf{x})\right).\end{aligned}$$

Therefore,

$$\begin{aligned}&\left\|H_W^{(\ell)}(\mathbf{x}) - H_{W'}^{(\ell)}(\mathbf{x})\right\|_2 \\ &\leq \sum_{j=1}^{\ell} \prod_{k=j+1}^{\ell} \|W_k\|_{\mathsf{op}} \left\|\Psi_{W_j}^{(j)}\left(\Psi_{W'_{j-1}}^{(j-1)} \circ \cdots \circ \Psi_{W'_1}^{(1)}(\mathbf{x})\right) - \Psi_{W'_j}^{(j)}\left(\Psi_{W'_{j-1}}^{(j-1)} \circ \cdots \circ \Psi_{W'_1}^{(1)}(\mathbf{x})\right)\right\|_2 \\ &\leq \sum_{j=1}^{\ell} \prod_{k=j+1}^{\ell} \|W_k\|_{\mathsf{op}} \|W_j - W'_j\|_{\mathsf{op}} \left\|\Psi_{W'_{j-1}}^{(j-1)} \circ \cdots \circ \Psi_{W'_1}^{(1)}(\mathbf{x})\right\|_2.\end{aligned}$$

Since $\Psi_M^{(j)}(\mathbf{0}) = \mathbf{0}$, we have the bound

$$\left\| \Psi_{W'_{j-1}}^{(j-1)} \circ \cdots \circ \Psi_{W'_1}^{(1)}(\mathbf{x}) \right\|_2 \leq \|\mathbf{x}\|_2 \prod_{k=1}^{j-1} \|W'_j\|_{\mathsf{op}}.$$

In conclusion, for all $1 \leq \ell \leq D$, $W, W' \in \mathcal{W}$, and for all $\mathbf{x} \in \mathbb{R}^{d_x}$, we have

$$\|H_W^{(\ell)}(\mathbf{x}) - H_{W'}^{(\ell)}(\mathbf{x})\|_2 \leq \|\mathbf{x}\|_2 \sum_{j=1}^{\ell} \|W_j - W'_j\|_{\mathsf{op}} \prod_{k=1}^{j-1} \|W'_k\|_{\mathsf{op}} \prod_{k=j+1}^{\ell} \|W_k\|_{\mathsf{op}}. \tag{B.13}$$

For $\mathbf{x} \in \mathbb{R}^{d_x}$, $\mathbf{y} \in \mathsf{Y}$, we define

$$F_{\mathbf{y}}(\mathbf{x}) \stackrel{\text{def}}{=} \mathrm{Prox}^{\gamma \mathbb{R}}\left(\mathbf{x} - \gamma \nabla_{\mathbf{x}} f(\mathbf{y}|\mathbf{x})\right), \quad \text{and} \quad F_{\mathbf{y},W}(\mathbf{x}) \stackrel{\text{def}}{=} H_W\left(\mathbf{x} - \gamma \nabla_{\mathbf{x}} f(\mathbf{y}|\mathbf{x})\right).$$

We use the notation h^k to denote the function h composed k times with the convention that h^0 is the identity map. Hence $g_W(\mathbf{y}) = F_{\mathbf{y},W}^{D'}(\mathbf{x}^{(0)})$. Assumption 5 implies that $F_{\mathbf{y}}$ is non-expansive.

Lemma 31. *Assume H5. Let $0 \leq \epsilon \leq R$, and $W \in \mathcal{W}$ be such that*

$$\sup_{\|\mathbf{x}\|_2 \leq R} \left\| \mathrm{Prox}^{\gamma \mathbb{R}}(\mathbf{x}) - H_W(\mathbf{x}) \right\|_2 \leq \epsilon.$$

Then with g_W as in (3.11) it holds,

$$\max_{1 \leq i \leq n} \|g_W(\mathbf{y}_i) - g(\mathbf{y}_i)\|_2 \leq R_0 \rho_n^{D'} + D' \sup_{\mathbf{x} \in \mathbb{R}^{d_x},\, \|\mathbf{x}\|_2 \leq R_0''} \|H_W(\mathbf{x}) - \mathrm{Prox}^{\gamma \mathbb{R}}(\mathbf{x})\|_2,$$

for some constant R_0''.

Proof. For any $\mathbf{y} \in \mathsf{Y}$, we can write

$$g(\mathbf{y}) - g_W(\mathbf{y}) = g(\mathbf{y}) - F_{\mathbf{y},W}^{D'}(\mathbf{x}^{(0)}) = g(\mathbf{y}) - F_{\mathbf{y}}^{D'}(\mathbf{x}^{(0)}) + F_{\mathbf{y}}^{D'}(\mathbf{x}^{(0)}) - F_{\mathbf{y},W}^{D'}(\mathbf{x}^{(0)}).$$

By Assumption 5, we have

$$\left\| g(\mathbf{y}_i) - F_{\mathbf{y}_i}^{D'}(\mathbf{x}^{(0)}) \right\|_2 \leq R_0 \rho_n^{D'}.$$

Since $F_{\mathbf{y}}$ is non-expansive, by the telescoping argument (B.11), we have

$$\begin{aligned}\left\|F_{\mathbf{y}_i}^{D'}(\mathbf{x}^{(0)}) - F_{\mathbf{y}_i,W}^{D'}(\mathbf{x}^{(0)})\right\|_2 &\leq \sum_{j=1}^{D'} \|F_{\mathbf{y}_i,W}\left(F_{\mathbf{y}_i,W}^{j-1}(\mathbf{x}^{(0)})\right) - F_{\mathbf{y}_i}\left(F_{\mathbf{y}_i,W}^{j-1}(\mathbf{x}^{(0)})\right)\|_2, \\ &\leq D' \sup_{\mathbf{x}\in\mathbb{R}^{d_x},\, \|\mathbf{x}\|_2\leq R_0'} \|F_{\mathbf{y}_i,W}(\mathbf{x}) - F_{\mathbf{y}_i}(\mathbf{x})\|_2, \\ &\leq D' \sup_{\mathbf{x}\in\mathbb{R}^{d_x},\, \|\mathbf{x}\|_2\leq R_0''} \|H_W(\mathbf{x}) - \mathrm{Prox}^{\gamma\mathbb{R}}(\mathbf{x})\|_2,\end{aligned}$$

where the last two inequalities use Assumption 5-(2)-(3), for constants R_0' and R_0''. The result follows by taking the max over i. □

Lemma 32. *Assume Assumption 5, Assumption 6, and let $\{g_W,\ W\in\mathcal{W}\}$ be as in (3.11). For any $\eta>0$, and any $W,W'\in\mathcal{W}$, such that $\max(\|W\|_2,\|W'\|_2)\leq\eta$, we have*

$$\max_{1\leq i\leq n} \|g_W(\mathbf{y}_i) - g_{W'}(\mathbf{y}_i)\|_2 \leq L(\eta)\|W-W'\|_2,$$

where

$$L(\eta) \stackrel{\text{def}}{=} R_0''\left(1+\frac{\eta^2}{D}\right)^{D(D'+1)}.$$

Proof. We recall that the convexity of $\mathbf{x}\mapsto f(\mathbf{y}|\mathbf{x})$ assumed in Assumption 5 implies that the function $\mathbf{x}\mapsto \mathbf{x}-\gamma\nabla_{\mathbf{x}}f(\mathbf{y}|\mathbf{x})$ is non-expansive on $\mathbb{R}^{d_x}$. First, we apply (B.12) to obtain that for all $\mathbf{x}_1,\mathbf{x}_2\in\mathbb{R}^{d_x}$,

$$\|H_W(\mathbf{x}_1) - H_W(\mathbf{x}_2)\|_2 \leq \lambda_W\|\mathbf{x}_1-\mathbf{x}_2\|_2, \quad \text{where} \quad \lambda_W \stackrel{\text{def}}{=} \prod_{\ell=1}^{D}\|W_\ell\|_{\mathsf{op}}\vee 1.$$

It follows that for all $\mathbf{y}\in\mathsf{Y}$, $\mathbf{x}_1,\mathbf{x}_2\in\mathbb{R}^{d_x}$,

$$\begin{aligned}&\|F_{\mathbf{y},W}(\mathbf{x}_1) - F_{\mathbf{y},W}(\mathbf{x}_2)\|_2 \\ &\qquad\leq \lambda_W\|\mathbf{x}_1-\mathbf{x}_2-\gamma\left(\nabla_{\mathbf{x}}f(\mathbf{y}|\mathbf{x}_1)\right) - \nabla_{\mathbf{x}}f(\mathbf{y}|\mathbf{x}_2))\|_2 \leq \lambda_W\|\mathbf{x}_1-\mathbf{x}_2\|_2. \quad \text{(B.14)}\end{aligned}$$

Using (B.14), and the telescoping identity (B.11), we obtain

$$\|g_W(\mathbf{y}) - g_{W'}(\mathbf{y})\|_2 \leq \sum_{j=1}^{D'}\lambda_W^{D'-j}\left\|F_{\mathbf{y},W}\left(F_{\mathbf{y},W'}^{j-1}(\mathbf{x}^{(0)})\right) - F_{\mathbf{y},W'}\left(F_{\mathbf{y},W'}^{j-1}(\mathbf{x}^{(0)})\right)\right\|_2. \quad \text{(B.15)}$$

We set $G_{\gamma,\mathbf{y}}(\mathbf{x}) \stackrel{\text{def}}{=} \mathbf{x} - \gamma\nabla f(\mathbf{y}|\mathbf{x})$. We apply (B.12) with $\mathbf{x}_2 = \mathbf{0}$ and the non-expansiveness of $G_{\gamma,\mathbf{y}}$ to write that for all $\mathbf{x} \in \mathbb{R}^{d_x}$

$$\begin{aligned}\|F_{\mathbf{y},W}(\mathbf{x})\|_2 = \|H_W\left(G_{\gamma,\mathbf{y}}(\mathbf{x})\right)\|_2 &\leq \lambda_W \|G_{\gamma,\mathbf{y}}(\mathbf{x}) - G_{\gamma,\mathbf{y}}(\mathbf{x}^{(0)}) + G_{\gamma,\mathbf{y}}(\mathbf{x}^{(0)})\|_2 \\ &\leq \lambda_W \left(\|\mathbf{x}\|_2 + \|\mathbf{x}^{(0)}\|_2 + \|G_{\gamma,\mathbf{y}}(\mathbf{x}^{(0)})\|_2\right).\end{aligned}$$

By iterating this inequality we obtain that for all for all $\mathbf{x} \in \mathbb{R}^{d_x}$

$$\|F^j_{\mathbf{y},W}(\mathbf{x}^{(0)})\|_2 \leq \left(\sum_{\ell=1}^{j} \lambda_W^\ell\right) \left(\|\mathbf{x}^{(0)}\|_2 + \|G_{\gamma,\mathbf{y}}(\mathbf{x}^{(0)})\|_2\right). \tag{B.16}$$

Setting $\lambda_{W,W'} \stackrel{\text{def}}{=} \sqrt{D}\lambda_W\lambda_{W'}\|W - W'\|_2$, we then apply (B.13) to write that for all $\mathbf{x} \in \mathbb{R}^{d_x}$

$$\begin{aligned}&\left\|F_{\mathbf{y},W}\left(F^{j-1}_{\mathbf{y},W'}(\mathbf{x}^{(0)})\right) - F_{\mathbf{y},W'}\left(F^{j-1}_{\mathbf{y},W'}(\mathbf{x}^{(0)})\right)\right\|_2 \\ &\qquad = \left\|H_W \circ G_{\gamma,\mathbf{y}}\left(F^{j-1}_{\mathbf{y},W'}(\mathbf{x}^{(0)})\right) - H_{W'} \circ G_{\gamma,\mathbf{y}}\left(F^{j-1}_{\mathbf{y},W'}(\mathbf{x}^{(0)})\right)\right\|_2 \\ &\qquad\qquad \leq \lambda_{W,W'} \|G_{\gamma,\mathbf{y}}\left(F^{j-1}_{\mathbf{y},W'}(\mathbf{x}^{(0)})\right)\|_2 \\ &\qquad \leq \lambda_{W,W'} \left(\|F^{j-1}_{\mathbf{y},W'}(\mathbf{x}^{(0)})\|_2 + \|\mathbf{x}^{(0)}\|_2 + \|G_{\gamma,\mathbf{y}}(\mathbf{x}^{(0)}\|_2\right) \\ &\qquad\qquad \leq \lambda_{W,W'} \left(\sum_{\ell=0}^{j-1} \lambda_{W'}^\ell\right) \left(\|\mathbf{x}^{(0)}\|_2 + \|G_{\gamma,\mathbf{y}}(\mathbf{x}^{(0)})\|_2\right).\end{aligned}$$

The last display together with (B.15) yields,

$$\max_{1\leq i\leq n} \|g_W(\mathbf{y}_i) - g_{W'}(\mathbf{y}_i)\|_2 \leq R_n\sqrt{D}\lambda_W\lambda_{W'}\|W - W'\|_2 \left(\sum_{j=1}^{D'}\sum_{\ell=0}^{j-1} \lambda_W^{D'-j}\lambda_{W'}^\ell\right), \tag{B.17}$$

where $R_n = \|\mathbf{x}^{(0)}\|_2 + \max_{1\leq i\leq n} \|G_{\gamma,\mathbf{y}_i}(\mathbf{x}^{(0)})\|_2$ as introduced in H5. Since the geometric mean is never larger than the arithmetic mean, we have

$$\lambda_W \stackrel{\text{def}}{=} \prod_{j=1}^{D} 1 \vee \|W_j\|_{\mathsf{op}} \leq \left(\frac{1}{D}\sum_{j=1}^{D} 1 \vee \|W_j\|^2_{\mathsf{op}}\right)^{D/2} \leq \left(1 + \frac{\|W\|_2^2}{D}\right)^{D/2}.$$

Hence, if $\max(\|W\|_2, \|W'\|_2) \leq \eta$, then we can conclude from that

$$\|g_W(\mathbf{y}_i) - g_{W'}(\mathbf{y}_i)\|_2 \leq R_0'' \sqrt{D} \lambda_W \lambda_{W'} \|W - W'\|_2 \sum_{j=0}^{D'-1} \lambda_W^j \leq L(\eta) \, \|W - W'\|_2, \quad \text{(B.18)}$$

where

$$L(\eta) \stackrel{\text{def}}{=} R_0'' D' \sqrt{D} \left(1 + \frac{\eta^2}{D}\right)^{D(D'+1)/2} \leq R_0'' \left(1 + \frac{\eta^2}{D}\right)^{D(D'+1)},$$

using the fact that $A^x \geq x$ for all $x \geq 1$, and $A \geq 1$. □

Proof of Theorem 8

Proof. We recall the notation $a \lesssim b$ means that $a \leq cb$, for some constant c that does not depend on the sample size n. Fix

$$\varpi_\star = \log(n) \sqrt{d_x} \left(\frac{\log(q)}{n}\right)^{\frac{1}{2+\beta_2}}, \quad \text{and} \quad \frac{\log\left(\frac{2R_0}{\varpi_\star}\right)}{-\log(\rho)} \leq D' \leq n.$$

By Assumption 6, and Lemma 31, by taking a deep neural network function H_W, with depth $D = D_0 \log(2D'\sqrt{d_x}/\varpi_\star)$, layer size $(p_0, \ldots, p_D)$ all at most $N_0(2D'\sqrt{d_x}/\varpi_\star)^{\beta_1}$, and $W \in \mathcal{W}$ with sparsity at most $s_\star = s_0(2D'\sqrt{d_x}/\varpi_\star)^{\beta_2}$, and we achieve

$$\begin{aligned}\max_{1 \leq i \leq n} \|g_W(\mathbf{y}_i) - g(\mathbf{y}_i)\|_2 &\leq R_0 \rho^{D'} \\ &\quad + D' \sup_{\mathbf{x}:\, \|\mathbf{x}\|_2 \leq R_0'} \|H_W(\mathbf{x}) - \mathrm{Prox}^{\gamma \mathbb{R}}(\mathbf{x})\|_2 \\ &\leq R_0 \frac{\varpi_\star}{2R_0} + D' \frac{\varpi_\star}{2D'} = \varpi_\star.\end{aligned}$$

Then by Lemma 32, the term $\mathsf{L}_\star$ in Theorem 10 scales like

$$\mathsf{L}_\star = L(2s_\star^{1/2}) = R_0'' \left(1 + \frac{4s_\star}{D}\right)^{D(D'+1)}.$$

It follows that

$$\frac{\log \mathsf{L}_\star}{\log(q)} \lesssim DD' \lesssim D' \log(n), \quad \text{and} \quad s_\star = s_0 \left(\frac{2D'\sqrt{d_x}}{\varpi_\star}\right)^{\beta_2} \lesssim \left(\frac{D'}{\log(n)}\right)^{\beta_2} \left(\frac{n}{\log(q)}\right)^{\frac{\beta_2}{2+\beta_2}},$$

and the term s in Theorem 10 is of order

$$s = \frac{1}{\mathsf{u}} + s_\star \left(1 + \frac{5}{\mathsf{u}} + \frac{\log(n)}{\mathsf{u}\log(q)} + \frac{\log(\mathsf{L}_\star)}{u\log(q)}\right) + \frac{4n\varpi_\star^2}{\sigma^2 \mathsf{u}\log(q)}$$
$$\lesssim \left[(1 + D'\log(n))\left(\frac{D'}{\log(n)}\right)^{\beta_2} + (\log(n))^2\right]\left(\frac{n}{\log(q)}\right)^{\frac{\beta_2}{2+\beta_2}}$$
$$\lesssim (D')^{1+\beta_2}\log(n)\left(\frac{n}{\log(q)}\right)^{\frac{\beta_2}{2+\beta_2}}.$$

Therefore the term L_s in Theorem 10 scales like

$$\mathsf{L}_s = L(s^{1/2}b_s) = R_0'(D + sb_s^2)^{DD'},$$

which gives,

$$\log(s^{1/2}b_s\mathsf{L}_s) \lesssim \log(\mathsf{L}_s) \lesssim DD'\log(n) \lesssim D'(\log(n))^2.$$

Noting that

$$\frac{s}{n} \lesssim (D')^{1+\beta_2}\left(\frac{\log(q)}{n}\right)^{\frac{2}{2+\beta_2}}\frac{\log(n)}{\log(q)},$$

we deduce that the conclusion of Theorem 10 holds with a rate

$$\mathsf{r} = \bar{\sigma}\sqrt{\frac{\mathsf{s}\log(\mathsf{n}) + \mathsf{s}\log(\mathsf{q}) + \mathsf{s}\log(\mathsf{s}^{1/2}\mathsf{b}_\mathsf{s}\mathsf{L}_\mathsf{s})}{\mathsf{n}}}$$
$$\lesssim \bar{\sigma}\,(D')^{1+\beta_2/2}\,(\log(n))^{3/2}\left(\frac{\log(q)}{n}\right)^{\frac{1}{2+\beta_2}}.$$

□

References

Aggarwal, H., Mani, M., and Jacob, M. (2017). Modl: Model based deep learning architecture for inverse problems. *IEEE Transactions on Medical Imaging*, PP.

Andrew, G., Arora, R., Bilmes, J., and Livescu, K. (2013). Deep canonical correlation analysis. In *Proc. ICML*, pages 1247–1255.

Andrieu, C. and Thoms, J. (2008). A tutorial on adaptive mcmc. *Statistics and Computing*, 18(4):343–373.

Ardizzone, L., Kruse, J., Rother, C., and Kothe, U. (2019). Analyzing inverse problems with invertible neural networks. In *International Conference on Learning Representations*.

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 214–223.

Arnold, B. C., Castillo, E., and Sarabia, J. M. (2001). Conditionally specified distributions: An introduction. *Statistical Science*, 16(3):249–265.

Arora, S., Ge, R., Liang, Y., Ma, T., and Zhang, Y. (2017). Generalization and equilibrium in generative adversarial nets (GANs). In Precup, D. and Teh, Y. W., editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 224–232, International Convention Centre, Sydney, Australia. PMLR.

Atchadé, Y. and Rosenthal, J. (2005). On adaptive markov chain monte carlo algorithms. *Bernoulli*, 11.

Atchadé, Y. F. and Liu, J. S. (2010). The wang-landau algorithm in general state spaces: Applications and convergence analysis. *Statistica Sinica*, 20(1):209–233.

Atchadé, Y. and Bhattacharyya, A. (2019). An approach to large-scale quasi-bayesian inference with spike-and-slab priors. *ArXiv* https://arxiv.org/abs/1803.10282

Atchadé, Y., Roberts, G., and Rosenthal, J. (2011). Towards optimal scaling of metropolis-coupled markov chain monte carlo. *Statistics and Computing*, 21:555–568.

Atchadé, Y. and Wang, L. (2021). A fast asynchronous mcmc sampler for sparse bayesian inference. *ArXiv eprints* https://arxiv.org/abs/2108.06446

Barron, A. R. and Klusowski, J. M. (2018). Approximation and estimation for high-dimensional deep learning networks. *ArXiv eprints* https://arxiv.org/abs/1809.03090

Beck, A. and Teboulle, M. (2010). Gradient-based algorithms with applications to signal-recovery problems. In *Convex optimization in signal processing and communications*, pages 42–88. Cambridge Univ. Press, Cambridge.

Bertsimas, D. and Tsitsiklis, J. (1993). Simulated Annealing. *Stat. Sci.*, 8(1):10 – 15.

Bickel, P. J. and Kleijn, B. J. K. (2012). The semiparametric Bernstein-von Mises theorem. *The Annals of Statistics*, 40(1):206–237.

Bissantz, N., Hohage, T., Munk, A., and Ruymgaart, F. (2007). Convergence rates of general regularization methods for statistical inverse problems and applications. *SIAM Journal on Numerical Analysis*, 45(6):2610–2636.

Biswas, N., Jacob, P. E., and Vanetti, P. (2019). Estimating convergence of markov chains with l-lag couplings.

Blanchard, G. and Mücke, N. (2018). Optimal rates for regularization of statistical inverse learning problems. *Found. Comput. Math.*, 18(4):971–1013.

Burger, H. C., Schuler, C. J., and Harmeling, S. (2012). Image denoising: Can plain neural networks compete with bm3d? In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2392–2399.

Carlin, B. P. and Chib, S. (1995). Bayesian model choice via markov chain monte carlo methods. *J. Roy. Stat. Soc. B*, 57(3):473–484.

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2009). Handling sparsity via the horseshoe. In *International Conference on Artificial Intelligence and Statistics*.

Chang, J. R., Li, C.-L., Poczos, B., Vijaya Kumar, B., and Sankaranarayanan, A. C. (2017). One network to solve them all – solving linear inverse problems using deep projection models. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5889–5898.

Chen, X., Liu, J., Wang, Z., and Yin, W. (2018). Theoretical linear convergence of unfolded ista and its practical weights and thresholds. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 31.

Chun, Y. and Fessler, J. A. (2018). Deep bcd-net using identical encoding-decoding cnn structures for iterative image recovery. In *2018 IEEE 13th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*, pages 1–5.

Combettes, P. and Wajs, V. (2005). Signal recovery by proximal forward-backward splitting. *Multiscale Modeling and Simulation*, 4(4):1168–1200.

De Sa, C., Olukotun, K., and Ré, C. (2016). Ensuring rapid mixing and low bias for asynchronous gibbs sampling. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 1567–1576.

DeVore, R., Hanin, B., and Petrova, G. (2021). Neural network approximation. *Acta Numerica*, 30:327–444.

Dong, W., Zhang, L., Shi, G., and Wu, X. (2011). Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Transactions on Image Processing*, 20(7):1838–1857.

Ee, W., Ma, C., and Wang, Q. (2020). Rademacher complexity and the generalization error of residual networks. *Communications in Mathematical Sciences*, 18:1755–1774.

Friedman, M. J. (1983). Control of malaria virulence by alpha 1-acid glycoprotein (orosomucoid), an acute-phase (inflammatory) reactant. *Proceedings of the National Academy of Sciences*, 80(17):5421–5424.

Gao, C., Ma, Z., Ren, Z., and Zhou, H. H. (2015). Minimax estimation in sparse canonical correlation analysis. *Annals of Statistics,* 43(5):2168–2197.

Gao, C., Ma, Z., and Zhou, H. H. (2017). Sparse cca: Adaptive estimation and computational barriers. *Annals of Statistics*, 45(5):2074–2101.

Ge, R., Lee, H., and Risteski, A. (2018). Simulated Tempering Langevin Monte Carlo II: An Improved Proof using Soft Markov Chain Decomposition. *arXiv e-prints*, page arXiv:1812.00793.

Geer, S., van de Geer, S., Gill, R., Ripley, B., Ross, S., Silverman, B., Williams, D., and Stein, M. (2000). *Empirical Processes in M-Estimation*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

George, E. I. and McCulloch, R. E. (1993). Variable selection via gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889.

George, E. I. and McCulloch, R. E. (1997). Approaches for bayesian variable selection. *Statistica Sinica*, 7(2):339–373.

Geyer, C. J. and Thompson, E. A. (1995). Annealing markov chain monte carlo with applications to ancestral inference. *J. Amer. Stat. Assoc.*, 90(431):909–920.

Gilton, D., Ongie, G., and Willett, R. M. (2020). Neumann networks for linear inverse problems in imaging. *IEEE Transactions on Computational Imaging*, 6:328–343.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.

Gregor, K. and LeCun, Y. (2010). Learning fast approximations of sparse coding. In *Proceedings of the 27th international conference on international conference on machine learning*, pages 399–406.

Hanche-Olsen, H. and Holden, H. (2010). The kolmogorov–riesz compactness theorem. *Expositiones Mathematicae*, 28(4):385–394.

Hardoon, D. and Shawe-Taylor, J. (2011). Sparse canonical correlation analysis. *Machine Learning*, 83:331–353.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika*.

Jacob, P., O'Leary, J., and Atchadé, Y. (2017). Unbiased markov chain monte carlo with couplings. *Journal of the Royal Statistical Society. Series B*.

Karpievitch, Y. V., Polpitiya, A. D., Anderson, G. A., Smith, R. D., and Dabney, A. R. (2010). Liquid chromatography mass spectrometry-based proteomics: Biological and technological aspects. *Annals of Applied Statistics*, 4(4):1797–1823.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598):671–680.

Knapik, B., Vaart, A., and Zanten, J. (2011). Bayesian inverse problems with gaussian priors. *The Annals of Statistics*, 39.

Li, S. (2011). Concise formulas for the area and volume of a hyperspherical cap. *Asian Journal of Mathematics and Statistics*, 4(1):66–70.

Li, Y., Tofighi, M., Geng, J., Monga, V., and Eldar, Y. C. (2020). Efficient and interpretable deep blind image deblurring via algorithm unrolling. *IEEE Transactions on Computational Imaging*, 6:666–681.

Liu, J. S. (2008). *Monte Carlo strategies in scientific computing*. Springer Science & Business Media.

Liu, R., Cheng, S., Ma, L., Fan, X., and Luo, Z. (2019). Deep proximal unrolling: Algorithmic framework, convergence analysis and applications. *IEEE Transactions on Image Processing*, 28(10):5013–5026.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Lucas, A., Iliadis, M., Molina, R., and Katsaggelos, A. K. (2018). Using deep neural networks for inverse problems in imaging: Beyond analytical methods. *IEEE Signal Processing Magazine*, 35(1):20–36.

Mardia, K., Kent, J., and Bibby, J. (1979). *Multivariate analysis*. Probability and mathematical statistics. Acad. Press, London [u.a.].

Miasojedow, B., Moulines, E., and Vihola, M. (2013). An adaptive parallel tempering algorithm. *Journal of Computational and Graphical Statistics*, 22(3):649–664.

Mitchell, T. J. and Beauchamp, J. J. (1988). Bayesian variable selection in linear regression. *J. Amer. Stat. Assoc.*, 83(404):1023–1032.

Müller, A. (1997). Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443.

Mo, Q., Shen, R., Guo, C., Vannucci, M., Chan, K. S., and Hilsenbeck, S. G. (2017). A fully bayesian latent variable model for integrative clustering analysis of multi-type omics data. *Biostatistics*, 19(1):71–86.

Monga, V., Li, Y., and Eldar, Y. C. (2021). Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44.

Necoara, I., Nesterov, Y., and Glineur, F. (2019). Linear convergence of first order methods for non-strongly convex optimization. *Math. Program.*, 175:69—-107.

Nickl, R. (2017). Bernstein - von mises theorems for statistical inverse problems i: Schrödinger equation. *Journal of the European Mathematical Society*, 22.

Nickl, R. and Sohl, J. (2019). Bernstein - von mises theorems for statistical inverse problems ii: Compound poisson processes. *Electronic Journal of Statistics*, 13:3513 – 3571.

Nowozin, S., Cseke, B., and Tomioka, R. (2016). F-gan: Training generative neural samplers using variational divergence minimization. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 271–279, Red Hook, NY, USA. Curran Associates Inc.

O'Brien, J. J., Gunawardena, H. P., Paulo, J. A., Chen, X., Ibrahim, J. G., Gygi, S. P., and Qaqish, B. F. (2018). The effects of nonignorable missing data on label-free mass spectrometry proteomics experiments. *Ann. Appl. Stat.*, 12(4):2075–2095.

Ongie, G., Jalal, A., Baraniuk, C., Dimakis, A., and Willett, R. (2020). Deep learning techniques for inverse problems in imaging. *IEEE Journal on Selected Areas in Information Theory*, PP:1–1.

Parkhomenko, E., Tritchler, D., and Beyene, J. (2009). Sparse canonical correlation analysis with application to genomic data integration. *Statistical applications in genetics and molecular biology*, 8:Article 1.

Perkins, S. and Leslie, D. (2014). Stochastic fictitious play with continuous action sets. *Journal of Economic Theory*, 152:179 – 213.

Polson, N. G. and Ročková, V. (2018). Posterior concentration for sparse deep learning. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31*, pages 930–941. Curran Associates, Inc.

Rappoport, N. and Shamir, R. (2018). Multi-omic and multi-view clustering algorithms: review and cancer benchmark. *Nucleic acids research*, 46(20):10546–10562.

Raskutti, G., Wainwright, M. J., and Yu, B. (2010). Restricted eigenvalue properties for correlated gaussian designs. *J. Mach. Learn. Res.*, 11:2241–2259.

Rastogi, A., Blanchard, G., and Mathé, P. (2020). Convergence analysis of Tikhonov regularization for non-linear statistical inverse problems. *Electronic Journal of Statistics*, 14(2):2798 – 2841.

Ravishankar, S., Chun, I. Y., and Fessler, J. A. (2017). Physics-driven deep training of dictionary-based algorithms for mr image reconstruction. In *2017 51st Asilomar Conference on Signals, Systems, and Computers*, pages 1859–1863.

Roberts, G. O. and Tweedie, R. L. (1996). Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363.

Rousu, J., Agranoff, D. D., Sodeinde, O., Shawe-Taylor, J., and Fernandez-Reyes, D. (2013). Biomarker discovery by sparse canonical correlation analysis of complex clinical phenotypes of tuberculosis and malaria. *PLOS Computational Biology*, 9(4):1–10.

Rudelson, M. and Zhou, S. (2013). Reconstruction from anisotropic random measurements. *IEEE Trans. Inf. Theor.*, 59(6):3434–3447.

Saatchi, Y. and Wilson, A. G. (2017). Bayesian gan. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 3622–3631. Curran Associates, Inc.

Sahu, B. R., Kampa, R. K., Padhi, A., and Panda, A. K. (2020). C-reactive protein: A promising biomarker for poor prognosis in covid-19 infection. *Clinica Chimica Acta*, 509:91 – 94.

Samworth, R. J. (2018). Recent Progress in Log-Concave Density Estimation. *Statistical Science*, 33(4):493 – 509.

Schauder, J. (1930). Der fixpunktsatz in funktionalraümen. *Studia Mathematica*, 2(1):171–180.

Schmidt-Hieber, J. (2020). Nonparametric regression using deep neural networks with relu activation function. *Annals of Statistics*, 48.

Shen, B., Yi, X., Sun, Y., Bi, X., Du, J., Zhang, C., Quan, S., Zhang, F., Sun, R., Qian, L., Ge, W., Liu, W., Liang, S., Chen, H., Zhang, Y., Li, J., Xu, J., He, Z., Chen, B., Wang, J., Yan, H., Zheng, Y., Wang, D., Zhu, J., Kong, Z., Kang, Z., Liang, X., Ding, X., Ruan, G., Xiang, N., Cai, X., Gao, H., Li, L., Li, S., Xiao, Q., Lu, T., Zhu, Y., Liu, H., Chen, H., and Guo, T. (2020). Proteomic and metabolomic characterization of covid-19 patient sera. *Cell*, 182(1):59 – 72.e15.

Shen, X. and Wasserman, L. (2001). Rates of convergence of posterior distributions. *The Annals of Statistics*, 29(3):687 – 714.

Shlezinger, N., Whang, J., Eldar, Y. C., and Dimakis, A. G. (2021). Model-based deep learning: Key approaches and design guidelines. In *2021 IEEE Data Science and Learning Workshop (DSLW)*, pages 1–6.

Sreter, H. and Giryes, R. (2018). Learned convolutional sparse coding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2191–2195. IEEE.

Stuart, A. M. (2010). Inverse problems: A bayesian perspective. *Acta Numerica*, 19:451–559.

Sulam, J., Aberdam, A., Beck, A., and Elad, M. (2020). On multi-layer basis pursuit, efficient algorithms and convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(8):1968–1980.

Taheri, M., Xie, F., and Lederer, J. (2021). Statistical guarantees for regularized neural networks. *Neural Networks*, 142:148–161.

Tan, K. M., Wang, Z., Liu, H., and Zhang, T. (2018). Sparse generalized eigenvalue problem: Optimal statistical rates via truncated rayleigh flow. *J. R. Stat. Soc. Ser. B*, 80(5):1057–1086.

Tao, S., Boley, D., and Zhang, S. (2016). Local linear convergence of ista and fista on the lasso problem. *SIAM Journal on Optimization*, 26(1):313–336.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Tolooshams, B., Song, A., Temereanca, S., and Ba, D. (2020). Convolutional dictionary learning based auto-encoders for natural exponential-family distributions. In *International Conference on Machine Learning*, pages 9493–9503. PMLR.

Vaart, A. W. v. d. (1998). *Asymptotic Statistics*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Vershynin, R. (2018). *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Vu, V. Q. and Lei, J. (2013). Minimax sparse principal subspace estimation in high dimensions. *Ann. Statist.*, 41(6):2905–2947.

Waaijenborg, S. and Zwinderman, A. (2009). Sparse canonical correlation analysis for identifying, connecting and completing gene-expression networks. *BMC bioinformatics*, 10:315.

Wainwright, M. J. (2019). *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Welling, M. and Teh, Y. W. (2011). Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pages 681–688, USA. Omnipress.

Wiesel, A., Kliger, M., and Hero, A. (2008). A greedy approach to sparse canonical correlation analysis.

Witten, D. M. and Tibshirani, R. J. (2009). Extensions of sparse canonical correlation analysis with applications to genomic data. *Statistical applications in genetics and molecular biology*, 8(1):1–27.

Woodard, D. B., Schmidler, S. C., and Huber, M. (2009). Conditions for rapid mixing of parallel and simulated tempering on multimodal distributions. *Ann. Appl. Probab.*, 19(2):617–640.

Xie, J., Xu, L., and Chen, E. (2012). Image denoising and inpainting with deep neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.

Yang, y., Sun, J., Li, H., and Xu, Z. (2016). Deep admm-net for compressive sensing mri. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Yarotsky, D. (2017). Error bounds for approximations with deep relu networks. *Neural Networks*, 94:103–114.

Yoon, G., Carroll, R. J., and Gaynanova, I. (2018). Sparse semiparametric canonical correlation analysis for data of mixed types. *arXiv: Methodology*.

Zhang, K., Zuo, W., Gu, S., and Zhang, L. (2017). Learning deep cnn denoiser prior for image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3929–3938.

Zou, H. and Hastie, T. (2005a). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 67(2):301–320.

Zou, H. and Hastie, T. (2005b). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 67(2):301–320.

CURRICULUM VITAE