Randomized methods in statistics

by

Feynman Liang

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Statistics

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Associate Professor Alan Hammond, Co-chair
Associate Adjunct Professor Michael Mahoney, Co-chair
Professor Fraydoun Rezakhanlou
Assistant Professor Shirsendu Ganguly

Fall 2022

Randomized methods in statistics

Thanks to my advisor Michael Mahoney, to whom I owe the past half decade of mentorship and research opportunities. In addition, I am grateful for my colleagues and mentors who have supported me along the way. In no particular order: Liam Hodgkinson, Zhenyu Liao, Michał Dereziński, Nimar Arora, Michael Tingley, Erik Meijer, Preben Thorø, Alexy Khrabrov, Reilly Bodycomb, Ryan Moody, Cassidee Moyer, Armin Eghdami, Xiangrui Meng, Joseph Bradley, Ameet Talwalkar, Ion Stoica, James Koo, Ben Searchinger, Debo Olaosebikan, Dennis Jiang, Emin Arakelian, Andrew Tsai, YB Cho, Dan Rasmusson. And last but not least, thanks to my family and especially my loving wife. You all helped push this over the finish line.

Contents

Acknowledgments

Feynman Liang was supported by a PhD fellowship from the Graduate Fellowship for STEM Diversity (GFSD).

Chapter 1

Introduction

When faced with difficult problems, randomized approximations are a tool commonly employed by quantitative scientists which can offer algernative algorithms and analyses. In this dissertation, we consider a range of problems at the intersection of randomized methods and statistics. Throughout our work, randomized methods are a unifying theme used in the first section to construct tractable approximations and later as tools for automating computational Bayesian statistics.

Initially in the first three chapters, we develop novel analyses to recent problems in experimental design, double descent, and random projections by using determinantal point processes (DPP). In chapter 2, we consider approximately optimal Bayesian experimental design using an adaptive row sampling algorithm based on DPPs and show that it provides good approximations. Through generalizing the previous chapter's proof techniques, in chapter 3 an extension of the DPP-based design is analyzed in closed-form for the over-parameterized $n < d$ regime and predicts a double-descent phenomenon in linear regression which closely matches empirical experiments. In chapter 4 we isolate the part of the proof involving concentration of bilinear forms of matrix resolvents away from the DPP-based design in order to obtain bounds on expected projections $X_S^\dagger X_S$ when $X_S = SX$ is obtained by sub-Gaussian sketching matrix S. These chapters correspond to the following publications:

- Michał Dereziński, Feynman Liang, and Michael Mahoney. "Bayesian experimental design using regularized determinantal point processes". In: *International Conference on Artificial Intelligence and Statistics*. 2020, pp. 3197–3207
- Michał Dereziński, Feynman Liang, and Michael W Mahoney. "Exact expressions for double descent and implicit regularization via surrogate random design". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 5152–5164
- Michał Dereziński, Feynman Liang, Zhenyu Liao, and Michael W Mahoney. "Precise expressions for random projections: Low-rank approximation and randomized Newton". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 18272–18283

In the later chapters we focus on probabilistic programming and developing theory and tools to automate statistical inference using randomized algorithms based on Monte Carlo Markoc chain (MCMC) and variational inference (VI). The second three chapters concern the setting of probabilistic programming, a computational tool for specifying probabilistic models and automating inference through MCMC and VI. In particular, we study approximating a target density $p(x)$ with approximations $q(x)$ for the purposes of importance sampling and variational inference. In chapter 5, we consider parameterizing $q(x)$ by graph neural networks which condition each node on its Markov blanket. This reduces the conditioning sets for a node, resulting in improvements over [LBW17] and run-times which depend on sparsity in the graphical model rather than the length of execution traces. When the target $p(x)$ is both multivariate and heavy tailed, chapter 6 considers the problem of tail anisotropy through both a theoretical and practical perspective. We establish that prior fat-tailed estimators [Jai+20] are tail isotropic, propose an anisotropic approximation (fat-tailed variational inference, FTVI) where an anisotropic product base measures is pushed forwards through a bijective neural network, and confirm that in practice FTVI improves both density estimation as well as variational inference. However, we find in practice FTVI and other tail-adaptive approximations often have trouble optimizing the tail parameter. In chapter 7, we consider addressing this issue during static analysis of a probabilistic program's source code. We define generalized Gamma tail asymptotics for a number of elementary distributions and establish how the tail asymptotics are transformed under algebraic transformations such as sums and products. This enables a priori computation of tail parameters, which we show improves the stability and convergence of a number of inference tasks. These chapters correspond to the following publications:

- Feynman Liang, Nimar Arora, Nazanin Tehrani, Yucen Li, Michael Tingley, and Erik Meijer. "Accelerating Metropolis-Hastings with Lightweight Inference Compilation". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 181–189
- Feynman Liang, Liam Hodgkinson, and Michael Mahoney. "Fat–Tailed Variational Inference with Anisotropic Tail Adaptive Flows". In: *Proceedings of the 39th International Conference on Machine Learning*. Vol. 162. 2022, p. 132
- Feynman Liang, Liam Hodgkinson, and Michael Mahoney. "Static Analysis of Tail Behaviour with a Generalized Gamma Algebra". In: *Submitted to AISTATS 2023* (2023)

Chapter 2

Bayesian experimental design with regularized determinantal point processes

In this chapter, we establish a fundamental connection between Bayesian experimental design and determinantal point processes (DPPs). Experimental design is a classical task in combinatorial optimization, where we wish to select a small subset of d-dimensional vectors to minimize a statistical optimality criterion. We show that a new regularized variant of DPPs can be used to design efficient algorithms for finding $(1+\epsilon)$-approximate solutions to experimental design under four commonly used optimality criteria: A-, C-, D- and V-optimality. A key novelty is that we offer improved guarantees under the Bayesian framework. Our algorithm returns a $(1+\epsilon)$-approximate solution when the subset size k is $\Omega(\frac{d_{\mathbf{A}}}{\epsilon} + \frac{\log 1/\epsilon}{\epsilon^2})$, where $d_{\mathbf{A}} \ll d$ is an effective dimension determined by prior knowledge (via a precision matrix $\mathbf{A}$). This is the first approximation guarantee where the dependence on d is replaced by an effective dimension. Moreover, the time complexity of our algorithm significantly improves on existing approaches with comparable guarantees. Some of the results here were initially published in Michał Dereziński, Feynman Liang, and Michael Mahoney. "Bayesian experimental design using regularized determinantal point processes". In: *International Conference on Artificial Intelligence and Statistics*. 2020, pp. 3197–3207.

2.1 Introduction

Consider a collection of n experiments parameterized by d-dimensional vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$, and let $\mathbf{X}$ denote the $n \times d$ matrix with rows $\mathbf{x}_i^\top$. The outcome of the ith experiment is a random variable $y_i = \mathbf{x}_i^\top \mathbf{w} + \xi_i$, where $\mathbf{w}$ is the parameter vector of a linear model with prior distribution $\mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{A}^{-1})$, and $\xi_i \sim \mathcal{N}(0, \sigma^2)$ is independent noise. In experimental design, we have access to the vectors $\mathbf{x}_i^\top$, for $i \in \{1, \ldots, n\} = [n]$, but we are allowed to observe only a small number of outcomes y_i for experiments we choose. Suppose that we observe

the outcomes from a subset $S \subseteq [n]$ of $|S| = k$ experiments. The posterior distribution of $\mathbf{w}$ given $\mathbf{y}_S$ (the vector of outcomes in S) is:

$$\begin{aligned} \mathbf{w} \mid \mathbf{y}_S &\sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \\ \text{where } \boldsymbol{\mu} &= (\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A})^{-1} \mathbf{X}_S^\top \mathbf{y}_S, \\ \boldsymbol{\Sigma} &= \sigma^2 (\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A})^{-1}. \end{aligned}$$

Here, $\mathbf{X}_S$ is the $k \times d$ matrix with rows $\mathbf{x}_i^\top$ for $i \in S$.

In Bayesian experimental design [CV95], the prior precision matrix $\mathbf{A}$ is used to encode prior knowledge and our goal is to choose S so as to minimize a function (a.k.a. an optimality criterion) measuring the "size" of the posterior covariance matrix $\boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y}_S} = \sigma^2 (\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A})^{-1}$. Note that $\boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y}_S}$ is well defined even if $\mathbf{A}$ is not invertible (i.e., an "improper prior"). In particular, it includes classical experimental design as the special case $\mathbf{A} = \mathbf{0}$, as well as the ridge-regularized case for $\mathbf{A} = \lambda \mathbf{I}$. Denoting $\boldsymbol{\Sigma}$ as the subset covariance $\mathbf{X}_S^\top \mathbf{X}_S$, we will use $f_{\mathbf{A}}(\boldsymbol{\Sigma})$ to represent the following standard Bayesian optimality criteria [CV95; Puk06]:

1. A-optimality: $f_{\mathbf{A}}(\boldsymbol{\Sigma}) = \operatorname{tr}\big((\boldsymbol{\Sigma} + \mathbf{A})^{-1}\big)$;
2. C-optimality: $f_{\mathbf{A}}(\boldsymbol{\Sigma}) = \mathbf{c}^\top (\boldsymbol{\Sigma} + \mathbf{A})^{-1} \mathbf{c}$ for $\mathbf{c} \in \mathbb{R}^d$;
3. D-optimality: $f_{\mathbf{A}}(\boldsymbol{\Sigma}) = \det(\boldsymbol{\Sigma} + \mathbf{A})^{-1/d}$;
4. V-optimality: $f_{\mathbf{A}}(\boldsymbol{\Sigma}) = \frac{1}{n} \operatorname{tr}\big(\mathbf{X}(\boldsymbol{\Sigma} + \mathbf{A})^{-1} \mathbf{X}^\top\big)$.

Applications including clinical trials [RDP15; DRM08; Spi+04; Ber+02; SB98; Flo93], medical imaging [Owe+16], materials science [FW16; Uen+16; TUM12], and biological process models [RDP+16] all use these optimality criteria and thus stand to benefit from our contributions.

The general task we consider is the following combinatorial optimization problem, where $[n]$ denotes $\{1, ..., n\}$:

Bayesian experimental design. Given an $n \times d$ matrix $\mathbf{X}$, a criterion $f_{\mathbf{A}}(\cdot)$ and $k \in [n]$, efficiently compute or approximate

$$\operatorname*{argmin}_{S \subseteq [n]} f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) \quad \text{subject to} \quad |S| = k.$$

We denote the value at the optimal solution as OPT_k. The prior work around this problem can be grouped into two research questions. The first question asks when does there exist a polynomial time algorithm for finding a $(1+\epsilon)$-approximation for OPT_k. The second question asks what we can infer about OPT_k just from the spectral information about the problem, which is contained in the data covariance matrix $\boldsymbol{\Sigma}_{\mathbf{X}} = \mathbf{X}^\top \mathbf{X}$.

Question 2.1 *Given $\mathbf{X}$, $f_{\mathbf{A}}$ and k, can we efficiently find a $(1+\epsilon)$-approximation for OPT_k?*

Question 2.2 *Given only $\boldsymbol{\Sigma}_{\mathbf{X}}$, $f_{\mathbf{A}}$ and k, what is the upper bound on OPT_k?*

A key aspect of both of these questions is how large the subset size k has to be for us to provide useful answers. As a baseline, we should expect meaningful results when k is at least $\Omega(d)$ [see discussion in All+17], and in fact, for classical experimental design (i.e., when $\mathbf{A} = \mathbf{0}$), the problem becomes ill-defined when $k < d$. In the Bayesian setting we should be able to exploit the additional prior knowledge to achieve strong results even for $k \ll d$. Intuitively, the larger the prior precision matrix $\mathbf{A}$, the fewer degrees of freedom we have in the problem. To measure this, we use the statistical notion of *effective dimension* [AM15].

Definition 2.1 *For $d \times d$ positive semi-definite (psd) matrices $\mathbf{A}$ and $\mathbf{\Sigma}$, let the $\mathbf{A}$-effective dimension of $\mathbf{\Sigma}$ be defined as $d_{\mathbf{A}}(\mathbf{\Sigma}) = \mathrm{tr}\big(\mathbf{\Sigma}(\mathbf{\Sigma} + \mathbf{A})^{-1}\big) \leq d$. We will use the shorthand $d_{\mathbf{A}}$ when referring to $d_{\mathbf{A}}(\mathbf{\Sigma}_{\mathbf{X}})$.*

[GK17] showed that $d_{\mathbf{A}}$ can be orders of magnitude smaller than the actual dimension d when the eigenvalues of $\mathbf{\Sigma}_{\mathbf{X}}$ exhibit fast decay, which is often the case in real datasets [GM16]. Recently, [DW18b] obtained bounds on Bayesian A/V-optimality criteria for $k \geq d_{\mathbf{A}}$, suggesting that $d_{\mathbf{A}}$ is the right notion of degrees of freedom for this problem.

Main results

Our main results provide new answers to Questions 1 and 2 by proposing a novel algorithm for Bayesian experimental design with strong theoretical guarantees.

Answer to Question 2.1 We propose an efficient $(1+\epsilon)$-approximation algorithm for A/C/D/V-optimal Bayesian experimental design:

Theorem 2.1 *Let $f_{\mathbf{A}}$ be A/C/D/V-optimality and $\mathbf{X}$ be $n \times d$. If $k = \Omega\big(\frac{d_{\mathbf{A}}}{\epsilon} + \frac{\log 1/\epsilon}{\epsilon^2}\big)$ for some $\epsilon \in (0,1)$, then we can find in polynomial time a subset S of size k s.t.*

$$f_{\mathbf{A}}\big(\mathbf{X}_S^\top \mathbf{X}_S\big) \leq (1+\epsilon) \cdot OPT_k.$$

Remark 2.1 *The algorithm referred to in Theorem 2.1 first solves a convex relaxation of the task via a semi-definite program (SDP) to find a weight vector $p \in [0,1]^n$, then uses our new randomized algorithm to round the weights to $\{0,1\}$, obtaining the subset S. The expected cost after SDP is $O(ndk + k^2d^2)$.*

A number of recent works studied $(1+\epsilon)$-approximate SDP-based algorithms for classical and Bayesian experimental design (see Table 2.1 and Section 2.2 for a comparison). Unlike *all* prior work on this topic, we are able to eliminate the dependence of the subset size k on the dimension d, replacing it with the potentially much smaller effective dimension $d_{\mathbf{A}}$. Our result also improves over the existing approaches in terms of the computational cost of the rounding procedure that is performed after solving the SDP. A number of different methods can be used to solve the SDP relaxation (see Section 2.5). For example, [All+17] suggest using an iterative optimizer called entropic mirror descent, which is known to exhibit fast convergence and can run in $O(nd^2T)$ time, where T is the number of iterations.

	Criteria	Bayesian	k	Cost after SDP
[WYS17]	A,V	✗	d^2/ϵ	$n^2 \cdot d$
[All+17]	A,C,D,E,G,V	✓	d/ϵ^2	$n \cdot kd^2$
[NST19]	A,D	✗	d/ϵ	$n^4 \cdot k^2 d$
this paper	A,C,D,V	✓	$d_{\mathbf{A}}/\epsilon$	$n \cdot kd + k^2 d^2$

Table 2.1: Comparison of SDP-based $(1+\epsilon)$-approximation algorithms for classical and Bayesian experimental design (X-mark means that only the classical setting applies). In the cost analysis, n could be replaced by the number of non-zero weights in the SDP solution. For simplicity we omit the log terms and assume that $\epsilon = \Omega(\frac{1}{d_{\mathbf{A}}})$. Our approach beats other methods both in terms of the runtime and the dependence of k on d (when $d_{\mathbf{A}} = o(d)$).

Answer to Question 2.2 By performing a careful theoretical analysis of the performance of our algorithm, we are able to give an improved upper bound on OPT_k. In the below result, we use a more refined notion of effective dimensionality for Bayesian experimental design, $d_{\frac{n}{k}\mathbf{A}}$ (where the precision matrix $\mathbf{A}$ is scaled by factor $\frac{n}{k}$), which is smaller than $d_{\mathbf{A}}$ and therefore leads to a tighter bound.

Theorem 2.2 *Let $f_{\mathbf{A}}$ be A/C/D/V-optimality and $\mathbf{X}$ be $n \times d$. For any k such that $k \geq 4d_{\frac{n}{k}\mathbf{A}}$,*

$$OPT_k \leq \left(1 + 8\,\frac{d_{\frac{n}{k}\mathbf{A}}}{k} + 8\sqrt{\frac{\ln(k/d_{\frac{n}{k}\mathbf{A}})}{k}}\right) \cdot f_{\mathbf{A}}\big(\tfrac{k}{n}\mathbf{\Sigma}_{\mathbf{X}}\big).$$

Remark 2.2 *We give a (randomized) algorithm which (with probability 1) finds the subset S that certifies this bound and has expected time complexity $O(ndk + k^2d^2)$.*

In particular, this means that if $k \geq 4d_{\frac{n}{k}\mathbf{A}}$ then there is S of size k which satisfies $f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) = O(1) \cdot f_{\mathbf{A}}(\frac{k}{n}\mathbf{\Sigma}_{\mathbf{X}})$. This not only improves on [DW18b] in terms of the supported range of sizes k, but also in terms of the obtained bound (see Section 2.2 for a comparison). In Section 2.5, we we provide numerical evidence suggesting that for many real datasets the quantity $f_{\mathbf{A}}(\frac{k}{n}\mathbf{\Sigma}_{\mathbf{X}})$ provides a good estimate for OPT_k to within a factor of 2.

Comparison of different effective dimensions

Theorem 2.2 suggests that the right notion of degrees of freedom for Bayesian experimental design can in fact be smaller than $d_{\mathbf{A}}$. Intuitively, since $d_{\mathbf{A}}$ is computed using the full data covariance $\mathbf{\Sigma}_{\mathbf{X}}$, it is not in the same scale as the smaller covariance $\mathbf{X}_S^\top \mathbf{X}_S$ based on the subset S of size $k \ll n$. In our result this is corrected by increasing the regularization on $\mathbf{\Sigma}_{\mathbf{X}}$ from $\mathbf{A}$ to $\frac{n}{k}\mathbf{A}$ and using $d_{\frac{n}{k}\mathbf{A}} = d_{\frac{n}{k}\mathbf{A}}(\mathbf{\Sigma}_{\mathbf{X}})$ as the degrees of freedom. Note that $d_{\frac{n}{k}\mathbf{A}} \leq d_{\mathbf{A}}$ and this gap can be very large for some problems.

Consider the two definitions we are comparing:

Full effective dimension $d_{\mathbf{A}} = \operatorname{tr}\big(\mathbf{\Sigma_X}(\mathbf{A}+\mathbf{\Sigma_X})^{-1}\big)$,

Scaled effective dimension $d_{\frac{n}{k}\mathbf{A}} = \operatorname{tr}\big(\mathbf{\Sigma_X}(\frac{n}{k}\mathbf{A}+\mathbf{\Sigma_X})^{-1}\big)$.

Here, we demonstrate that these two effective dimensions can be very different for some matrices and quite similar on others. For simplicity, we consider two diagonal data covariance matrices as our examples: *identity covariance*, $\mathbf{\Sigma}_1 = \mathbf{I}$, and an *approximately low-rank covariance*, $\mathbf{\Sigma}_2 = (1-\epsilon)\frac{d}{s}\mathbf{I}_S + \epsilon\mathbf{I}$, where $\mathbf{I}_S$ is the diagonal matrix with ones on the entries indexed by subset $S \subseteq [d]$ of size $s < d$ and zeros everywhere else. The second matrix is scaled in such way so that $\operatorname{tr}(\mathbf{\Sigma}_1) = \operatorname{tr}(\mathbf{\Sigma}_2)$. We use $d = 100$, $s = 10$ and $\epsilon = 10^{-2}$. The prior precision matrix is $\mathbf{A} = 10^{-2}\,\mathbf{I}$. Figure 2.1 plots the scaled effective dimension $d_{\frac{n}{k}\mathbf{A}}$ as a function of k, against the full effective dimension for both examples. Unsurprisingly, for the identity covariance the full effective dimension is almost d, and the scaled effective dimension goes up very quickly to match it. On the other hand, for the approximately low-rank covariance, $d_{\mathbf{A}} \approx 55$ is considerably less then $d = 100$. Interestingly, the gap between the $d_{\frac{n}{k}\mathbf{A}}$ and $d_{\mathbf{A}}$ for moderately small values of k is even bigger. Our theory suggests that $d_{\frac{n}{k}\mathbf{A}}$ is a valid indicator of Bayesian degrees of freedom when $k \geq C \cdot d_{\frac{n}{k}\mathbf{A}}$ for some small constant C (Theorem 2.2 has $C = 4$, but we believe this can be improved to 1). While for the identity covariance the condition $k \geq d_{\frac{n}{k}\mathbf{A}}$ is almost equivalent to $k \geq d_{\mathbf{A}}$, in the approximately low-rank case, $k \geq d_{\frac{n}{k}\mathbf{A}}$ holds for k as small as 20, much less than $d_{\mathbf{A}}$.

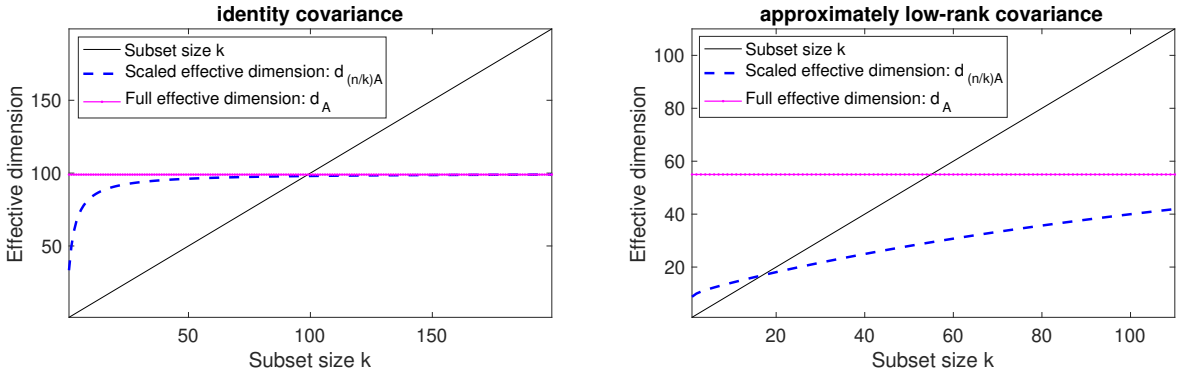


Figure 2.1: Scaled effective dimension compared to the full effective dimension for two diagonal data covariance matrices, with $\mathbf{A} = 10^{-2}\,\mathbf{I}$.

Technical contributions

To establish Theorems 2.1 and 2.2, we develop a theoretical framework for a new sampling distribution which can be seen as a *regularized* variant of a determinantal point process (DPP). DPPs are a well-studied family of distributions with numerous applications in sampling diverse subsets of negatively correlated elements [see KT12].

Given a psd matrix $\mathbf{A}$ and a weight vector $p = (p_1, ..., p_n) \in [0,1]^n$, we define $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ as a distribution over subsets $S \subseteq [n]$ (of all sizes) such that (see Definition 2.2):

$$\Pr(S) \propto \det(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A}) \cdot \prod_{i \in S} p_i \cdot \prod_{i \notin S} (1 - p_i).$$

A number of regularized DPPs have been proposed recently [Der19; DW18b], mostly within the context of Randomized Numerical Linear Algebra (RandNLA) [Mic11; DM16; DM17]. To our knowledge, ours is the first such definition that strictly falls under the umbrella of traditional DPPs [KT12]. We show this in Section 2.3, where we also prove that regularized DPPs can be decomposed into a low-rank DPP plus i.i.d. Bernoulli sampling (Theorem 2.3). This decomposition reduces the sampling cost from $O(n^3)$ to $O(nd^2)$, and involves a more general result about DPPs defined via a correlation kernel (Lemma 2.3), which is of independent interest.

In Section 2.4 we demonstrate a fundamental connection between an $\mathbf{A}$-regularized DPP and Bayesian experimental design with precision matrix $\mathbf{A}$. For simplicity of exposition, let the weight vector p be uniformly equal $(\frac{k}{n}, ..., \frac{k}{n})$. If $S \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ and $f_{\mathbf{A}}$ is any one of the A/C/D/V-optimality criteria, then:

$$\mathbb{E}\big[f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S)\big] \leq f_{\mathbf{A}}\big(\tfrac{k}{n}\mathbf{\Sigma}_{\mathbf{X}}\big) \quad \text{and} \quad \mathbb{E}\big[|S|\big] \leq d_{\frac{n}{k}\mathbf{A}} + k.$$

The proof of Theorem 2.2 relies on these two inequalities and a concentration bound for the subset size $|S|$, whereas to obtain Theorem 2.1 we additionally use the SDP relaxation to find the optimal weight vector p. When $\mathbf{A} = \mathbf{0}$, then $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ bears a lot of similarity to *proportional volume sampling* which is an (unregularized) determinantal distribution proposed by [NST19]. Our algorithm not only extends it to the Bayesian setting but also offers a drastic time complexity improvement from the $O(n^4 dk^2 \log k)$ required by [NST19] down to the $O(nd^2)$ required for sampling from $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$, and recent advances in RandNLA for DPP sampling [DWH18; DWH19a; Der19] suggest that $O(nd \log n + \mathrm{poly}(d))$ time is also possible.

2.2 Related work

A number of works proposed $(1+\epsilon)$-approximation algorithms for experimental design which start with solving a convex relaxation of the problem, and then use some rounding strategy to obtain a discrete solution (see Table 2.1 for comparison). In this line of work we wish to find the smallest k for which a polynomial time approximation algorithm is possible. For example, [WYS17] gave an approximation algorithm for classical A/V-optimality with $k = \Omega(\frac{d^2}{\epsilon})$, where the rounding is done in a greedy fashion, and some randomized rounding strategies are also discussed. [NST19] suggested *proportional volume sampling* for the rounding step and obtained approximation algorithms for classical A/D-optimality with $k = \Omega(\frac{d}{\epsilon} + \frac{\log 1/\epsilon}{\epsilon^2})$. Their approach is particularly similar to ours (when $\mathbf{A} = \mathbf{0}$). However, as discussed earlier, while

their algorithms run in polynomial time, they scale very poorly with the number of experiments n (see Table 2.1). [All+17] proposed an efficient algorithm with a $(1+\epsilon)$-approximation guarantee for a wide range of optimality criteria, including A/C/D/E/V/G-optimality, both classical and Bayesian, when $k = \Omega(\frac{d}{\epsilon^2})$. Our results (in Theorem 2.1) improve on this work in two important ways:

- In terms of the dependence on ϵ for A/C/D/V-optimality,
- In terms of the dependence on the dimension (by replacing d with $d_{\mathbf{A}}$) in the Bayesian setting.

A lower bound shown by [NST19] implies that our Theorem 2.1 cannot be directly extended to E-optimality, but a similar lower bound does not exist for G-optimality. We remark that the approximation approaches relying on a convex relaxation can generally be converted to an upper bound on OPT_k akin to our Theorem 2.2, however, unlike our bound, none of them apply to the regime of $k \leq d$.

Non-trivial bounds for the *classical* A-optimality criterion (i.e., OPT_k with $\mathbf{A} = \mathbf{0}$) were first given by [AB13], where they show that for any $k \geq d$, $\mathrm{OPT}_k \leq (1 + \frac{d-1}{k-d+1}) \cdot f_{\mathbf{0}}(\frac{k}{n}\mathbf{\Sigma_X})$ and the subset S attaining the bound can be found in polynomial time. The result was later extended [DW17; DW18b; DW18a] to the case where $\mathbf{A} = \lambda\mathbf{I}$, proving that for any $k \geq d_{\lambda\mathbf{I}}$, we have $\mathrm{OPT}_k \leq (1 + \frac{d_{\lambda\mathbf{I}}-1}{k-d_{\lambda\mathbf{I}}+1}) \cdot f_{\frac{k}{n}\lambda\mathbf{I}}(\frac{k}{n}\mathbf{\Sigma_X})$, and also a faster $O(nd^2)$ time algorithm was provided. In comparison, our results (in Theorem 2.2) offer the following improvements for upper bounding OPT_k:

- We cover a wider range of subset sizes, because $d_{\frac{n}{k}\lambda\mathbf{I}} \leq d_{\lambda\mathbf{I}}$,
- Our upper bound can be much tighter because $f_{\lambda\mathbf{I}}(\frac{k}{n}\mathbf{\Sigma_X}) \leq f_{\frac{k}{n}\lambda\mathbf{I}}(\frac{k}{n}\mathbf{\Sigma_X})$.

Additionally, [Der+19] propose a new notion of *minimax* experimental design, which is related to A/V-optimality. They also use a determinantal distribution for subset selection, however, due to different assumptions, their bounds are incomparable.

Purely greedy approximation algorithms have been shown to provide guarantees in a number of special cases for experimental design. One example is classical D-optimality criterion, which can be converted to a submodular function [BGS10]. Also, greedy algorithms for Bayesian A/V-optimality criteria have been considered by [Bia+17] and [CR18]. These methods can only provide a constant factor approximation guarantee (as opposed to $1+\epsilon$), and the factor is generally problem dependent (which means it could be arbitrarily large). Finally, a number of heuristics with good empirical performance have been proposed, such as Fedorov's exchange method [CN80]. However, in this work we focus on methods that provide theoretical approximation guarantees.

2.3 A new regularized determinantal point process

In this section we develop the theory for a novel regularized extension of determinantal point processes (DPP) which we use as the sampling distribution for obtaining guarantees in Bayesian experimental design. DPPs form a family of distributions which are used to model repulsion between elements in a random set, with many applications in machine learning [KT12]. Here, we focus on the setting where we are sampling out of all 2^n subsets $S \subseteq [n]$. Traditionally, a DPP is defined by a correlation kernel, which is an $n \times n$ psd matrix $\mathbf{K}$ with eigenvalues between 0 and 1, i.e., such that $\mathbf{0} \preceq \mathbf{K} \preceq \mathbf{I}$. Given a correlation kernel $\mathbf{K}$, the corresponding DPP is defined as

$$S \sim \mathrm{DPP}_{\mathrm{cor}}(\mathbf{K}) \quad \text{iff} \quad \Pr(T \subseteq S) = \det(\mathbf{K}_{T,T}) \ \ \forall_{T \in [n]},$$

where $\mathbf{K}_{T,T}$ is the submatrix of $\mathbf{K}$ with rows and columns indexed by T. Another way of defining a DPP, popular in the machine learning community, is via an ensemble kernel $\mathbf{L}$. Any psd matrix $\mathbf{L}$ is an ensemble kernel of a DPP defined as:

$$S \sim \mathrm{DPP}_{\mathrm{ens}}(\mathbf{L}) \quad \text{iff} \quad \Pr(S) \propto \det(\mathbf{L}_{S,S}).$$

Crucially, every $\mathrm{DPP}_{\mathrm{ens}}$ is also a $\mathrm{DPP}_{\mathrm{cor}}$, but not the other way around. Specifically, $\mathrm{DPP}_{\mathrm{ens}}(\mathbf{L}) = \mathrm{DPP}_{\mathrm{cor}}(\mathbf{K})$ when:

$$\text{(a)} \ \ \mathbf{L} = \mathbf{K}(\mathbf{I} - \mathbf{K})^{-1}, \qquad\qquad \text{(b)} \ \ \mathbf{K} = \mathbf{I} - (\mathbf{I} + \mathbf{L})^{-1},$$

but (a) requires that $\mathbf{I} - \mathbf{K}$ be invertible which is not true for some DPPs. (This will be important in our analysis.) The classical algorithm for sampling from a DPP requires the eigendecomposition of either matrix $\mathbf{K}$ or $\mathbf{L}$, which in general costs $O(n^3)$, followed by a sampling procedure which costs $O(n\,|S|^2)$ [Hou+06; KT12].

We now define our regularized DPP and describe its connection with correlation and ensemble DPPs.

Definition 2.2 *Given matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$, a sequence $p = (p_1, \ldots, p_n) \in [0,1]^n$ and a psd matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ such that $\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top + \mathbf{A}$ is full rank, let $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ be a distribution over $S \subseteq [n]$:*

$$\Pr(S) = \frac{\det(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A})}{\det\big(\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top + \mathbf{A}\big)} \cdot \prod_{i \in S} p_i \cdot \prod_{i \notin S} (1 - p_i). \tag{2.1}$$

The fact that this is a proper distribution (i.e., that it sums to one) can be restated as a determinantal expectation formula: if $b_i \sim \mathrm{Bernoulli}(p_i)$ are independent Bernoulli random variables, then

$$\begin{aligned}
&\sum_{S \subseteq [n]} \det(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A}) \prod_{i \in S} p_i \prod_{i \notin S} (1 - p_i) \\
&= \mathbb{E}\bigg[\det\Big(\sum_i b_i \mathbf{x}_i \mathbf{x}_i^\top + \mathbf{A}\Big)\bigg] \overset{(*)}{=} \det\Big(\sum_i \mathbb{E}[b_i] \mathbf{x}_i \mathbf{x}_i^\top + \mathbf{A}\Big),
\end{aligned}$$

where $(*)$ follows from Lemma 7 of Dereziński et al. [DM19].

The main theoretical contribution in this section is the following efficient algorithm for $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ which reduces it to sampling from a correlation DPP and unioning with i.i.d. Bernoulli samples:

Theorem 2.3 *For any $\mathbf{X} \in \mathbb{R}^{n\times d}$, $p \in [0,1]^n$ and a psd matrix $\mathbf{A}$ s.t. $\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top + \mathbf{A}$ is full rank, let*

$$T \sim \mathrm{DPP}_{\mathrm{cor}}\big(\mathbf{D}_p^{1/2}\mathbf{X}(\mathbf{A} + \mathbf{X}^\top \mathbf{D}_p \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{D}_p^{1/2}\big),$$
$$\textit{where} \quad \mathbf{D}_p = \mathrm{diag}(p).$$

If $b_i \sim \mathrm{Bernoulli}(p_i)$ are independent random variables, then $T \cup \{i : b_i = 1\} \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$.

Remark 2.3 *Figure 2.2 illustrates how to exploit this result to build an efficient sampling algorithm. Since the correlation kernel matrix has rank at most d, the preprocessing cost of eigendecomposition is $O(nd^2)$. Then, each sample costs only $O(n\,|T|^2)$.*

We prove the theorem in three steps. First, we express $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ as an ensemble DPP, which requires some additional assumptions on $\mathbf{A}$ and p to be possible. Then, we convert the ensemble to a correlation kernel (eliminating the extra assumptions), and finally show that this kernel can be decomposed into a rank d kernel plus Bernoulli sampling. In the process, we establish several novel theoretical properties regarding the representation, decomposition, and closure properties of regularized DPPs which may be of independent interest.

Sampling $S \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$

Input: $\mathbf{X} \in \mathbb{R}^{n\times d}$, psd $\mathbf{A} \in \mathbb{R}^{d\times d}$, $p \in [0,1]^n$
Compute $\mathbf{Z} \leftarrow \mathbf{A} + \mathbf{X}^\top \mathbf{D}_p \mathbf{X}$
Compute SVD of $\mathbf{B} = \mathbf{D}_p^{1/2}\mathbf{X}\mathbf{Z}^{-1/2}$
Sample $T \sim \mathrm{DPP}_{\mathrm{cor}}(\mathbf{B}\mathbf{B}^\top)$ [Hou+06]
Sample $b_i \sim \mathrm{Bernoulli}(p_i)$ for $i \in [n]$
return $S = T \cup \{i : b_i = 1\}$

Figure 2.2: Algorithm which exploits Theorem 2.3 to sample $S \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ in $O(nd^2)$ time.

Lemma 2.1 *Given $\mathbf{X}$, $\mathbf{A}$ and $\mathbf{D}_p$ as in Theorem 2.3, assume that $\mathbf{A}$ and $\mathbf{I} - \mathbf{D}_p$ are invertible. Then,*

$$\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A}) = \mathrm{DPP}_{\mathrm{ens}}\big(\widetilde{\mathbf{D}} + \widetilde{\mathbf{D}}^{1/2}\mathbf{X}\mathbf{A}^{-1}\mathbf{X}^\top \widetilde{\mathbf{D}}^{1/2}\big),$$
$$\textit{where} \quad \widetilde{\mathbf{D}} = \mathbf{D}_p(\mathbf{I} - \mathbf{D}_p)^{-1}.$$

Proof Let $S \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$. By Definition 2.2 and the fact that $\det(\mathbf{AB}+\mathbf{I}) = \det(\mathbf{BA}+\mathbf{I})$,

$$\begin{aligned}
\Pr(S) &\propto \det(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A}) \cdot \prod_{i \in S} p_i \cdot \prod_{i \notin S} (1 - p_i) \\
&= \det(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A}) \cdot \prod_{i \in S} \frac{p_i}{1 - p_i} \cdot \prod_{i=1}^{n} (1 - p_i) \\
&\propto \det\big(\mathbf{A}(\mathbf{A}^{-1}\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{I})\big) \det(\widetilde{\mathbf{D}}_{S,S}) \\
&= \det(\mathbf{A}) \det(\mathbf{A}^{-1}\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{I}) \det(\widetilde{\mathbf{D}}_{S,S}) \\
&\propto \det(\mathbf{X}_S \mathbf{A}^{-1} \mathbf{X}_S^\top + \mathbf{I}) \det(\widetilde{\mathbf{D}}_{S,S}) \\
&= \det\Big(\big[\widetilde{\mathbf{D}}^{1/2} \mathbf{X} \mathbf{A}^{-1} \mathbf{X}^\top \widetilde{\mathbf{D}}^{1/2} + \widetilde{\mathbf{D}}\big]_{S,S} \Big),
\end{aligned}$$

which matches the definition of the L-ensemble DPP. ■

At this point, to sample from $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$, we could simply invoke any algorithm for sampling from an ensemble DPP. However, this would only work for invertible $\mathbf{A}$, which in particular excludes the important case of $\mathbf{A} = \mathbf{0}$ corresponding to classical experimental design. Moreover, the standard algorithm would require computing the eigendecomposition of the ensemble kernel, which (at least if done naïvely) costs $O(n^3)$. Even after this is done, the sampling cost would still be $O(n\,|S|^2)$ which can be considerably more than $O(nd^2)$. We first address the issue of invertibility of matrix $\mathbf{A}$ by expressing our distribution via a correlation DPP.

Lemma 2.2 *Given* $\mathbf{X}$*,* $\mathbf{A}$*, and* $\mathbf{D}_p$ *as in Theorem 2.3 (without any additional assumptions), we have*

$$\begin{aligned}
&\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A}) = \mathrm{DPP}_{\mathrm{cor}}\big(\mathbf{D}_p + \\
&(\mathbf{I} - \mathbf{D}_p)^{1/2}\, \mathbf{D}_p^{1/2} \mathbf{X} (\mathbf{A} + \mathbf{X}^\top \mathbf{D}_p \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{D}_p^{1/2} (\mathbf{I} - \mathbf{D}_p)^{1/2}\big).
\end{aligned}$$

When $\mathbf{A}$ and $\mathbf{I} - \mathbf{D}_p$ are invertible, then the proof is a straightforward calculation. Then, we use a limit argument with $p_\epsilon = (1 - \epsilon)p$ and $\mathbf{A}_\epsilon = \mathbf{A} + \epsilon \mathbf{I}$, where $\epsilon \to 0$.

Proof First, we show this under the invertibility assumptions of Lemma 2.1, i.e., given that $\mathbf{A}$ and $\mathbf{I} - \mathbf{D}_p$ are invertible. In this case $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A}) = \mathrm{DPP}_{\mathrm{ens}}(\mathbf{L})$, where

$$\mathbf{L} = \widetilde{\mathbf{D}} + \widetilde{\mathbf{D}}^{1/2} \mathbf{X} \mathbf{A}^{-1} \mathbf{X}^\top \widetilde{\mathbf{D}}^{1/2} \quad \text{and} \quad \widetilde{\mathbf{D}} = \mathbf{D}_p (\mathbf{I} - \mathbf{D}_p)^{-1}. \tag{2.2}$$

Converting this to a correlation kernel $\mathbf{K}$ and denoting $\widetilde{\mathbf{X}} = \mathbf{D}_p^{1/2}\mathbf{X}$, we obtain

$$\begin{aligned}
\mathbf{K} &= \mathbf{I} - (\mathbf{I}+\mathbf{L})^{-1} \\
&= \mathbf{I} - \big(\mathbf{I} + (\mathbf{I}-\mathbf{D}_p)^{-1}\mathbf{D}_p + (\mathbf{I}-\mathbf{D}_p)^{-1/2}\widetilde{\mathbf{X}}\mathbf{A}^{-1}\widetilde{\mathbf{X}}^\top(\mathbf{I}-\mathbf{D}_p)^{-1/2}\big)^{-1} \\
&= \mathbf{I} - \big((\mathbf{I}-\mathbf{D}_p)^{-1} + (\mathbf{I}-\mathbf{D}_p)^{-1/2}\widetilde{\mathbf{X}}\mathbf{A}^{-1}\widetilde{\mathbf{X}}^\top(\mathbf{I}-\mathbf{D}_p)^{-1/2}\big)^{-1} \\
&= \mathbf{I} - (\mathbf{I}-\mathbf{D}_p)^{1/2}(\mathbf{I} + \widetilde{\mathbf{X}}\mathbf{A}^{-1}\widetilde{\mathbf{X}}^\top)^{-1}(\mathbf{I}-\mathbf{D}_p)^{1/2} \\
&\overset{(*)}{=} \mathbf{I} - (\mathbf{I}-\mathbf{D}_p)^{1/2}\big(\mathbf{I} - \widetilde{\mathbf{X}}\mathbf{A}^{-1/2}(\mathbf{I}+\mathbf{A}^{-1/2}\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}}\mathbf{A}^{-1/2})^{-1}\mathbf{A}^{-1/2}\widetilde{\mathbf{X}}^\top\big)(\mathbf{I}-\mathbf{D}_p)^{1/2} \\
&= \mathbf{I} - (\mathbf{I}-\mathbf{D}_p) + (\mathbf{I}-\mathbf{D}_p)^{1/2}\widetilde{\mathbf{X}}(\mathbf{A}+\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^\top(\mathbf{I}-\mathbf{D}_p)^{1/2} \\
&= \mathbf{D}_p + (\mathbf{I}-\mathbf{D}_p)^{1/2}\widetilde{\mathbf{X}}(\mathbf{A}+\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^\top(\mathbf{I}-\mathbf{D}_p)^{1/2},
\end{aligned}$$

where $(*)$ follows from Fact 2.16.19 in [Ber11]. Note that converting from $\mathbf{L}$ to $\mathbf{K}$ got rid of the inverses $\mathbf{A}^{-1}$ and $(\mathbf{I}-\mathbf{D}_p)^{-1}$ appearing in (2.2). The intuition is that when $\mathbf{A}$ or $\mathbf{I}-\mathbf{D}_p$ is non-invertible, then $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X},\mathbf{A})$ is not an L-ensemble but it is still a correlation DPP. To show this, we use a limit argument. For $\epsilon \in [0,1]$, let $p_\epsilon = (1-\epsilon)p$ and $\mathbf{A}_\epsilon = \mathbf{A} + \epsilon\mathbf{I}$. Observe that if $\epsilon > 0$ then $\mathbf{A}_\epsilon$ and $\mathbf{I}-\mathbf{D}_{p_\epsilon}$ are always invertible even if $\mathbf{A}$ and $\mathbf{I}-\mathbf{D}_p$ are not. Denote $\mathbf{K}_\epsilon$ as the above correlation kernel with p replaced by p_ϵ and $\mathbf{A}$ replaced by $\mathbf{A}_\epsilon$. Note that all matrix operations defining kernel $\mathbf{K}_\epsilon$ are continuous w.r.t. $\epsilon \in [0,1]$, including the inverse, since $\mathbf{A}+\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}}$ is assumed to be invertible. Therefore, the following equalities hold (with limits taken point-wise and $\epsilon > 0$):

$$\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X},\mathbf{A}) = \lim_{\epsilon\to 0}\mathrm{DPP}^{p_\epsilon}_{\mathrm{reg}}(\mathbf{X},\mathbf{A}_\epsilon) = \lim_{\epsilon\to 0}\mathrm{DPP}_{\mathrm{cor}}(\mathbf{K}_\epsilon) = \mathrm{DPP}_{\mathrm{cor}}(\mathbf{K}),$$

where we did not have to assume invertibility of $\mathbf{A}$ or $\mathbf{I}-\mathbf{D}_p$. ■

Finally, we show that the correlation DPP arrived at in Lemma 2.2 can be decomposed into a smaller DPP plus Bernoulli sampling. In fact, in the following lemma we obtain a more general recipe for combining DPPs with Bernoulli sampling, which may be of independent interest. Note that if $b_i \sim \mathrm{Bernoulli}(p_i)$ are independent random variables then $\{i : b_i = 1\} \sim \mathrm{DPP}_{\mathrm{cor}}(\mathbf{D}_p)$.

Lemma 2.3 *Let* $\mathbf{K}$ *and* $\mathbf{D}$ *be* $n \times n$ *psd matrices with eigenvalues between 0 and 1, and assume that* $\mathbf{D}$ *is diagonal. If* $T \sim \mathrm{DPP}_{\mathrm{cor}}(\mathbf{K})$ *and* $R \sim \mathrm{DPP}_{\mathrm{cor}}(\mathbf{D})$*, then*

$$T \cup R \sim \mathrm{DPP}_{\mathrm{cor}}\big(\mathbf{D} + (\mathbf{I}-\mathbf{D})^{1/2}\mathbf{K}(\mathbf{I}-\mathbf{D})^{1/2}\big).$$

Proof For this proof we will use the shorthand $\mathbf{K}_A$ for $\mathbf{K}_{A,A}$. If $\mathbf{D}$ has no zeros on the

diagonal then $\det(\mathbf{D}_A) > 0$ for all $A \subseteq [n]$ and

$$\begin{aligned}
\Pr(A \subset T \cup R) &= \sum_{B \subset A} \Pr(R \cap A = A \setminus B)\ \Pr(B \subseteq T) \\
&= \sum_{B \subset A} \det(\mathbf{D}_{A \setminus B}) \det\big([\mathbf{I} - \mathbf{D}]_B\big)\ \det(\mathbf{K}_B) \\
&= \sum_{B \subset A} \det(\mathbf{D}_{A \setminus B}) \det\Big(\big[(\mathbf{I} - \mathbf{D})^{1/2}\mathbf{K}(\mathbf{I} - \mathbf{D})^{1/2}\big]_B\Big) \\
&= \det(\mathbf{D}_A) \sum_{B \subset A} \det\Big(\big[\mathbf{D}^{-1/2}(\mathbf{I} - \mathbf{D})^{1/2}\mathbf{K}(\mathbf{I} - \mathbf{D})^{1/2}\mathbf{D}^{-1/2}\big]_B\Big) \\
&\overset{(*)}{=} \det(\mathbf{D}_A) \det\Big(\mathbf{I} + \big[\mathbf{D}^{-1/2}(\mathbf{I} - \mathbf{D})^{1/2}\mathbf{K}(\mathbf{I} - \mathbf{D})^{1/2}\mathbf{D}^{-1/2}\big]_A\Big) \\
&= \det\Big(\big[\mathbf{D} + (\mathbf{I} - \mathbf{D})^{1/2}\mathbf{K}(\mathbf{I} - \mathbf{D})^{1/2}\big]_A\Big),
\end{aligned}$$

where $(*)$ follows from a standard determinantal identity used to compute the L-ensemble partition function [KT12, Theorem 2.1]. If $\mathbf{D}$ has zeros on the diagonal, a similar limit argument as in Lemma 2.2 with $\mathbf{D}_\epsilon = \mathbf{D} + \epsilon\,\mathbf{I}$ holds. ■

Theorem 2.3 now follows by combining Lemmas 2.2 and 2.3.

2.4 Guarantees for Bayesian experimental design

In this section we prove our main results regarding Bayesian experimental design (Theorems 2.1 and 2.2). First, we establish certain properties of the regularized DPP distribution that make it effective in this setting. Even though the size of the sampled subset $S \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ is random and can be as large as n, it is also highly concentrated around its expectation, which can be bounded in terms of the $\mathbf{A}$-effective dimension. This is crucial, since both of our main results require a subset of deterministically bounded size. Recall that the effective dimension is defined as a function $d_{\mathbf{A}}(\mathbf{\Sigma}) = \mathrm{tr}\big(\mathbf{\Sigma}(\mathbf{A} + \mathbf{\Sigma})^{-1}\big)$.

Lemma 2.4 *Given any $\mathbf{X} \in \mathbb{R}^{n \times d}$, $p \in [0,1]^n$ and a psd matrix $\mathbf{A}$ s.t. $\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top + \mathbf{A}$ is full rank, let $S = T \cup \{i : b_i = 1\} \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ be defined as in Theorem 2.3. Then*

$$\begin{aligned}
\mathbb{E}\big[|S|\big] &\leq \mathbb{E}\big[|T|\big] + \mathbb{E}\Big[\sum_i b_i\Big] \\
&= d_{\mathbf{A}}\Big(\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top\Big) + \sum_i p_i.
\end{aligned}$$

Proof For correlation kernels it is known that the expected size of $\mathrm{DPP}_{\mathrm{cor}}(\mathbf{K})$ is $\mathrm{tr}(\mathbf{K})$. Thus, using $\mathbf{D}_p = \mathrm{diag}(p)$, we can invoke Lemma 2.2 to obtain

$$\begin{aligned}\mathbb{E}\big[|S|\big] &= \mathrm{tr}\big(\mathbf{D}_p + (\mathbf{I}-\mathbf{D}_p)^{1/2}\,\mathbf{D}_p^{1/2}\mathbf{X}(\mathbf{A}+\mathbf{X}^\top\mathbf{D}_p\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{D}_p^{1/2}(\mathbf{I}-\mathbf{D}_p)^{1/2}\big)\\ &\leq \mathrm{tr}(\mathbf{D}_p) + \mathrm{tr}\big(\mathbf{D}_p^{1/2}\mathbf{X}(\mathbf{A}+\mathbf{X}^\top\mathbf{D}_p\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{D}_p^{1/2}\big)\\ &= \mathrm{tr}(\mathbf{D}_p) + \mathrm{tr}\big(\mathbf{X}^\top\mathbf{D}_p\mathbf{X}(\mathbf{A}+\mathbf{X}^\top\mathbf{D}_p\mathbf{X})^{-1}\big) = \mathrm{tr}(\mathbf{D}_p) + d_{\mathbf{A}}(\mathbf{X}^\top\mathbf{D}_p\mathbf{X}),\end{aligned}$$

from which the claim follows. ■

Next, we show two expectation inequalities for the matrix inverse and matrix determinant, which hold for the regularized DPP. We use them to bound the Bayesian optimality criteria in expectation.

Lemma 2.5 *Whenever $S \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X},\mathbf{A})$ is a well-defined distribution it holds that*

$$\mathbb{E}\Big[\big(\mathbf{X}_S^\top\mathbf{X}_S+\mathbf{A}\big)^{-1}\Big] \preceq \Big(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A}\Big)^{-1}, \tag{2.3}$$

$$\mathbb{E}\Big[\det\big(\mathbf{X}_S^\top\mathbf{X}_S+\mathbf{A}\big)^{-1}\Big] \leq \det\Big(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A}\Big)^{-1}. \tag{2.4}$$

Proof For a square matrix $\mathbf{M}$, define its adjugate, denoted $\mathrm{adj}(\mathbf{M})$, as a matrix whose i,j-th entry is $(-1)^{i+j}\det(\mathbf{M}_{-j,-i})$, where $\mathbf{M}_{-j,-i}$ is the matrix $\mathbf{M}$ without jth row and ith column. If $\mathbf{M}$ is invertible, then $\mathrm{adj}(\mathbf{M}) = \det(\mathbf{M})\mathbf{M}^{-1}$. Now, let $b_i \sim \mathrm{Bernoulli}(p_i)$ be independent random variables. As seen in previous section, the identity $\mathbb{E}[\det(\sum_i b_i\mathbf{x}_i\mathbf{x}_i^\top + \mathbf{A})] = \det(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})$ gives us the normalization constant for $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X},\mathbf{A})$. Moreover, as noted in a different context by [DM19], when applied entrywise to the adjugate matrix, this identity implies that $\mathbb{E}[\mathrm{adj}(\sum_i b_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})] = \mathrm{adj}(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})$. Let $\mathcal{I}$ denote the set of all subsets $S\subseteq[n]$ such that $\mathbf{X}_S^\top\mathbf{X}_S+\mathbf{A}$ is invertible. We have

$$\begin{aligned}\mathbb{E}\Big[\big(\mathbf{X}_S^\top\mathbf{X}_S+\mathbf{A}\big)^{-1}\Big] &= \sum_{S\in\mathcal{I}}\big(\mathbf{X}_S^\top\mathbf{X}_S+\mathbf{A}\big)^{-1}\frac{\det(\mathbf{X}_S^\top\mathbf{X}_S+\mathbf{A})}{\det(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})}\prod_{i\in S}p_i\prod_{i\notin S}(1-p_i)\\ &= \sum_{S\in\mathcal{I}}\frac{\mathrm{adj}\big(\mathbf{X}_S^\top\mathbf{X}_S+\mathbf{A}\big)}{\det(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})}\prod_{i\in S}p_i\prod_{i\notin S}(1-p_i)\\ &\preceq \sum_{S\subseteq[n]}\frac{\mathrm{adj}\big(\mathbf{X}_S^\top\mathbf{X}_S+\mathbf{A}\big)}{\det(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})}\prod_{i\in S}p_i\prod_{i\notin S}(1-p_i)\\ &= \frac{\mathbb{E}\big[\mathrm{adj}(\sum_i b_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})\big]}{\det(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})}\\ &= \frac{\mathrm{adj}(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})}{\det(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A})} = \Big(\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top+\mathbf{A}\Big)^{-1}.\end{aligned}$$

Note that if $\mathcal{I}$ contains all subsets of $[n]$, for example when $\mathbf{A} \succ \mathbf{0}$, then the inequality turns into equality. Thus, we showed (2.3), and (2.4) follows even more easily:

$$\mathbb{E}\Big[\det\big(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A}\big)^{-1}\Big] = \sum_{S\in\mathcal{I}} \frac{1}{\det(\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top + \mathbf{A})} \prod_{i\in S} p_i \prod_{i\notin S}(1-p_i) \leq \det\Big(\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top\Big)^{-1},$$

where the equality holds if $\mathcal{I}$ consists of all subsets of $[n]$. ■

Corollary 2.1 *Let $f_{\mathbf{A}}$ be A/C/D/V-optimality. Whenever $S \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ is well-defined,*

$$\mathbb{E}\big[f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S)\big] \leq f_{\mathbf{A}}\Big(\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top\Big).$$

Proof In the case of A-, C-, and V-optimality, the function $f_{\mathbf{A}}$ is a linear transformation of the matrix $(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A})^{-1}$ so the bound follows from (2.3). For D-optimality, we apply (2.4) as follows:

$$\begin{aligned}
\mathbb{E}\big[f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S)\big] &= \mathbb{E}\Big[\det\big(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A}\big)^{-1/d}\Big] \\
&\leq \mathbb{E}\bigg[\Big(\det\big(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A}\big)^{-1/d}\Big)^d\bigg]^{1/d} \\
&= \mathbb{E}\Big[\det\big(\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A}\big)^{-1}\Big]^{1/d} \\
&\leq \det\Big(\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top\Big)^{-1/d},
\end{aligned}$$

which completes the proof. ■

Finally, we present the key lemma that puts everything together. This result is essentially a generalization of Theorem 2.2 from which also follows Theorem 2.1.

Lemma 2.6 *Let $f_{\mathbf{A}}$ be A/C/D/V-optimality and $\mathbf{X}$ be $n \times d$. For some $w = (w_1, \ldots, w_n) \in [0,1]^n$, let $\mathbf{\Sigma}_w = \sum_i w_i \mathbf{x}_i \mathbf{x}_i^\top$ and assume that $\sum_i w_i = k \in [n]$. If $k \geq 4\, d_{\mathbf{A}}(\mathbf{\Sigma}_w)$, then a subset $S \subseteq [n]$ of size k can be found in $O(ndk + k^2 d^2)$ time that satisfies*

$$\begin{aligned}
&f_{\mathbf{A}}\big(\mathbf{X}_S^\top \mathbf{X}_S\big) \\
&\leq \left(1 + 8\,\frac{d_{\mathbf{A}}(\mathbf{\Sigma}_w)}{k} + 8\sqrt{\frac{\ln(k/d_{\mathbf{A}}(\mathbf{\Sigma}_w))}{k}}\right) \cdot f_{\mathbf{A}}\big(\mathbf{\Sigma}_w\big).
\end{aligned}$$

Proof Let $p = (p_1, \ldots, p_n)$ be defined so that $p_i = \frac{w_i}{1+\epsilon}$, and suppose that $S \sim \mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$. Then, using Corollary 2.1, we have

$$\begin{aligned}
&\Pr\big(|S| \le k\big)\, \mathbb{E}\Big[f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) \mid |S| \le k\Big] \\
&\quad \le \mathbb{E}\big[f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S)\big] \\
&\quad \le f_{\mathbf{A}}\Big(\sum_i p_i \mathbf{x}_i \mathbf{x}_i^\top\Big) \\
&\quad \le (1+\epsilon) \cdot f_{\mathbf{A}}\Big(\sum_i w_i \mathbf{x}_i \mathbf{x}_i^\top\Big).
\end{aligned}$$

Using Lemma 2.4 we can bound the expected size of S as follows:

$$\begin{aligned}
\mathbb{E}\big[|S|\big] &\le d_{\mathbf{A}}(\mathbf{\Sigma}_w) + \sum_i p_i \\
&= d_{\mathbf{A}}(\mathbf{\Sigma}_w) + \frac{k}{1+\epsilon} \\
&= k \cdot \Big(1 + \frac{d_{\mathbf{A}}(\mathbf{\Sigma}_w)}{k} - \frac{\epsilon}{1+\epsilon}\Big).
\end{aligned}$$

Let $d_w = d_{\mathbf{A}}(\mathbf{\Sigma}_w)$ and $\alpha = 1 + \frac{d_w}{k} - \frac{\epsilon}{1+\epsilon}$. If $1 \ge \epsilon \ge \frac{4d_w}{k}$, then $\alpha \le 1 + \frac{\epsilon}{4} - \frac{\epsilon}{2} = 1 - \frac{\epsilon}{4}$. Since $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ is a determinantal point process, $|S|$ is a Poisson binomial r.v. so for $\epsilon \ge 6\sqrt{\frac{\ln(k/d_w)}{k}}$,

$$\Pr(|S| > k) \le \mathrm{e}^{-\frac{(k-\alpha k)^2}{2k}} = \mathrm{e}^{-\frac{k}{2}(1-\alpha)^2} \le \mathrm{e}^{-\frac{k\epsilon^2}{32}} \le \frac{d_w}{k}.$$

For any $\epsilon \ge 4\,\frac{d_w}{k} + 6\sqrt{\frac{\ln(k/d_w)}{k}}$, we have

$$\begin{aligned}
&\mathbb{E}\big[f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) \mid |S| \le k\big] \\
&\quad \le \frac{1+\epsilon}{1 - \frac{d_w}{k}} \cdot f_{\mathbf{A}}(\mathbf{\Sigma}_w) \\
&\quad \le \left(1 + \frac{\epsilon + \frac{d_w}{k}}{1 - \frac{d_w}{k}}\right) \cdot f_{\mathbf{A}}(\mathbf{\Sigma}_w) \\
&\quad \le \left(1 + 7\,\frac{d_w}{k} + 8\sqrt{\frac{\ln(k/d_w)}{k}}\right) \cdot f_{\mathbf{A}}(\mathbf{\Sigma}_w).
\end{aligned}$$

Denoting $\mathbb{E}\big[f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) \mid |S| \le k\big]$ as F_k, Markov's inequality implies that

$$\Pr\Big(f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) \ge (1+\delta) F_k \;\mid\; |S| \le k\Big) \le \frac{1}{1+\delta}.$$

Also, we showed that $\Pr(|S| \leq k) \geq 1 - \frac{d_w}{k} \geq \frac{3}{4}$. Setting $\delta = \frac{d_w}{Ck}$ for sufficiently large C we obtain that with probability $\Omega(\frac{d_w}{k})$, the random set S has size at most k and

$$\begin{aligned} & f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) \\ & \leq \left(1 + \frac{d_w}{Ck}\right) \cdot \left(1 + 7\,\frac{d_w}{k} + 8\sqrt{\frac{\ln(k/d_w)}{k}}\right) \cdot f_{\mathbf{A}}(\mathbf{\Sigma}_w) \\ & \leq \left(1 + 8\,\frac{d_w}{k} + 8\sqrt{\frac{\ln(k/d_w)}{k}}\right) \cdot f_{\mathbf{A}}(\mathbf{\Sigma}_w). \end{aligned}$$

We can sample from $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ conditioned on $|S| \leq k$ and $f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S)$ bounded as above by rejection sampling. When $|S| < k$, the set is completed to k with arbitrary indices. On average, $O(\frac{k}{d_w})$ samples from $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ are needed, so the cost is $O(nd^2)$ for the eigendecomposition, $O(\frac{k}{d_w} \cdot nd_w^2) = O(nd_w k)$ for sampling and $O(\frac{k}{d_w} \cdot kd^2)$ for recomputing $f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S)$. ■

To prove the main results, we use Lemma 2.6 with appropriately chosen weights w.

Proof of Theorem 2.1 As discussed by [All+17] and [BV04], the following convex relaxation of experimental design can be written as a semi-definite program and solved using standard SDP solvers:

$$w^* \;=\; \operatorname*{argmin}_{w} \quad f_{\mathbf{A}}\Big(\sum_{i=1}^n w_i \mathbf{x}_i \mathbf{x}_i^\top\Big), \tag{2.5}$$

$$\text{subject to} \quad \forall_i \;\; 0 \leq w_i \leq 1, \quad \sum_i w_i = k. \tag{2.6}$$

The solution w^* satisfies $f_{\mathbf{A}}\big(\mathbf{\Sigma}_{w^*}\big) \leq \mathrm{OPT}_k$. If we use w^* in Lemma 2.6, then observing that $d_{\mathbf{A}}(\mathbf{\Sigma}_{w^*}) \leq d_{\mathbf{A}}$, and setting $k \geq C(\frac{d_{\mathbf{A}}}{\epsilon} + \frac{\log 1/\epsilon}{\epsilon^2})$ for sufficiently large C, the algorithm in the lemma finds subset S such that

$$f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) \leq (1+\epsilon) \cdot f_{\mathbf{A}}(\mathbf{\Sigma}_{w^*}) \leq (1+\epsilon) \cdot \mathrm{OPT}_k.$$

Note that we did not need to solve the SDP exactly, so approximate solvers could be used instead. ■

Proof of Theorem 2.2 Let $w = (\frac{k}{n}, ..., \frac{k}{n})$ in Lemma 2.6. Then, we have $\mathbf{\Sigma}_w = \frac{k}{n}\mathbf{\Sigma}_{\mathbf{X}}$ and also $d_{\mathbf{A}}(\mathbf{\Sigma}_w) = d_{\frac{n}{k}\mathbf{A}}$. Since for any set S of size k, we have $\mathrm{OPT}_k \leq f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S)$, the result follows. ■

2.5 Experiments

We confirm our theoreticala results with experiments on real world data from `libsvm` datasets [CL11] (more details in Appendix 4.6). For all our experiments, the prior precision matrix is

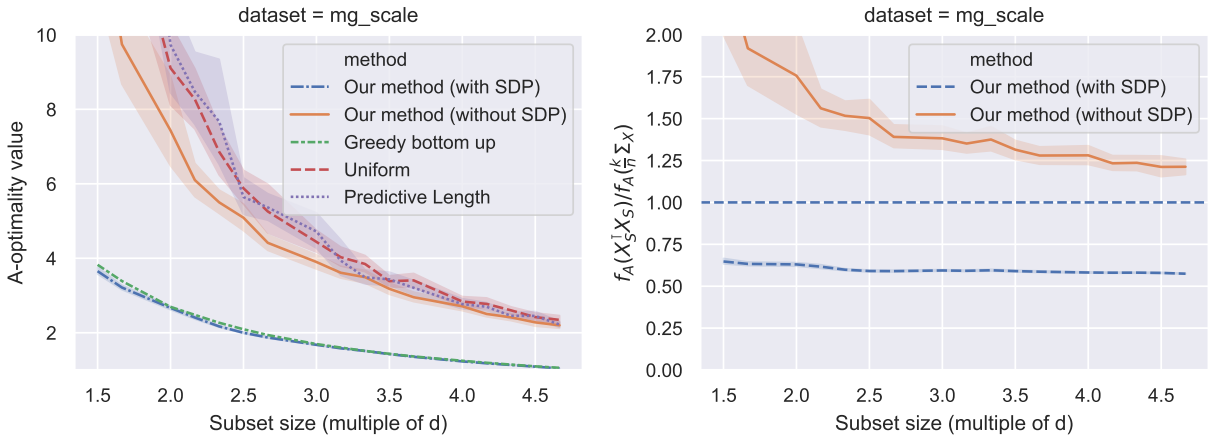


Figure 2.3: (left) A-optimality value obtained by the various methods on the `mg_scale` dataset [CL11] with prior precision $\mathbf{A} = 10^{-5}\,\mathbf{I}$, (right) A-optimality value for our method (with and without SDP) divided by $f_{\mathbf{A}}(\frac{k}{n}\mathbf{\Sigma_X})$, the baseline estimate suggested by Theorem 2.2.

set to $\mathbf{A} = n^{-1}\mathbf{I}$ and we consider sample sizes $k \in [d, 5d]$. Each experiment is averaged over 25 trials and bootstrap 95% confidence intervals are shown. The quality of our method, as measured by the A-optimality criterion,

$$f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S) = \operatorname{tr}\left((\mathbf{X}_S^\top \mathbf{X}_S + \mathbf{A})^{-1}\right),$$

is compared against several baselines and recently proposed methods for A-optimal design that have been shown to perform well in practice. Note that none of these algorithms come with theoretical guarantees as strong as those offered by our approach. The list of implemented methods is as follows:

Our method (with SDP) uses the efficient algorithms developed in proving Theorem 2.1 to sample $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ constrained to subset size k with $p = w^*$, see (2.6), obtained using a recently developed first order convex cone solver called Splitting Conical Solver [SCS, see ODo+16]. We chose SCS because it can handle the SDP constraints in (2.6) and has provable termination guarantees, while also finding solutions faster [ODo+16] than alternative off-the-shelf optimization software libraries such as SDPT3 and Sedumi.

Our method (without SDP) samples $\mathrm{DPP}^p_{\mathrm{reg}}(\mathbf{X}, \mathbf{A})$ with uniform probabilities $p \equiv \frac{k}{n}$.

Greedy bottom-up adds an index $i \in [n]$ to the sample S maximizing the increase in A-optimality criterion [Bia+17; CR17].

Uniform samples every size k subset $S \subseteq [n]$ with equal probability.

Predictive length sampling [Zhu+15] samples each row $\mathbf{x}_i$ of $\mathbf{X}$ with probability $\propto \|\mathbf{x}_i\|$.

Figure 2.3 reveals that our method (without SDP) is superior to both uniform and predictive length sampling, producing designs which achieve lower A-optimality criteria values for all sample sizes. As Theorem 2.3 shows that our method (without SDP) only differs from uniform sampling by an additional DPP sample with controlled expected size (see Lemma 2.4), we may conclude that adding even a small DPP sample can improve a uniformly sampled design.

Consistent with prior observations [WYS17; CR17], the greedy bottom up method achieves surprisingly good performance, despite the limited theoretical guarantees it offers. However, if our method is used in conjunction with an SDP solution, then we are able to match and even slightly exceed the performance of the greedy bottom up method. Furthermore, the overall run-time costs (see Appendix 4.6) between the two are comparable. As the majority of the runtime of our method (with SDP) is occupied by solving the SDP, an interesting future direction is to investigate alternative solvers such as interior point methods as well as terminating the solvers early once an approximate solution is reached.

Figure 2.3 (right) numerically evaluates the tightness of the bound obtained in Theorem 2.2 by plotting the ratio:

$$\frac{f_{\mathbf{A}}(\mathbf{X}_S^\top \mathbf{X}_S)}{f_{\mathbf{A}}(\frac{k}{n}\mathbf{\Sigma}_{\mathbf{X}})}$$

for subsets returned by our method (with and without SDP). Note that the line for our method with SDP on Figure 2.3 (right) shows that the ratio never goes below 0.5, and we saw similar behavior across all examined datasets (see Appendix 4.6). This evidence suggests that for many real datasets OPT_k is within only a small constant factor away from $f_{\mathbf{A}}(\frac{k}{n}\mathbf{\Sigma}_{\mathbf{X}})$, matching the upper bound of Theorem 2.2.

In addition to the `mg_scale` dataset presented in Section 4.6, we also benchmarked on three other data sets described in Table 2.2.

Table 2.2: Datasets used in the experiments [CL11].

	`mg_scale`	`bodyfat_scale`	`mpg_scale`	`housing_scale`
n	1385	252	392	506
d	6	14	7	13

The A-optimality values obtained are illustrated in Figure 2.4. The general trend observed in Section 4.6 of our method (without SDP) outperforming independent sampling methods (uniform and predictive length) and our method (with SDP) matching the performance of the greedy bottom up method continues to hold across the additional datasets considered.

The relative ranking and overall order of magnitude differences between runtimes (Figure 2.5) are also similar across the various datasets. An exception to the rule is on `mg_scale`,

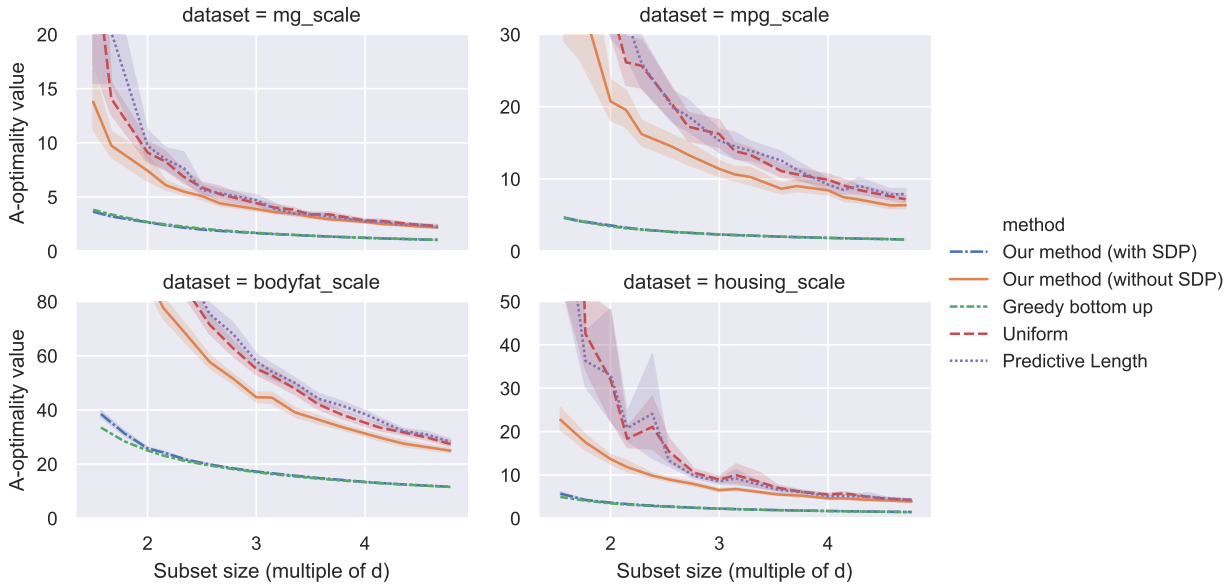


Figure 2.4: A-optimality values achieved by the methods compared. In all cases considered, we found our method (without SDP) to be superior to independent sampling methods like uniform and predictive length sampling. After paying the price to solve an SDP, our method (with SDP) is able to consistently match the performance of a greedy method which has been noted [CR17] to work well empirically.

where we see that our method (without SDP) costs more than the greedy method (whereas everywhere else it costs less).

The claim that $f_{\mathbf{A}}(\frac{k}{n}\mathbf{\Sigma}_{\mathbf{X}})$ is an appropriate quantity to summarize the contribution of problem-dependent factors on the performance of Bayesian A-optimal designs is further evidenced in Figure 2.6. Here, we see that after normalizing the A-optimality values by this quantity, the remaining quantities are all on the same scale and close to 1.

2.6 Conclusions

We proposed a new algorithm for finding $(1+\epsilon)$-approximate Bayesian experimental designs by leveraging a fundamental connection with determinantal point processes. Compared to the state-of-the-art approaches, our method provides stronger theoretical guarantees in terms of the allowed range of subset sizes, as well as offering significantly better time complexity guarantees. At the same time, our experiments show that on the task of A-optimal design the proposed algorithm performs as well as or better than several methods that are used in practice.

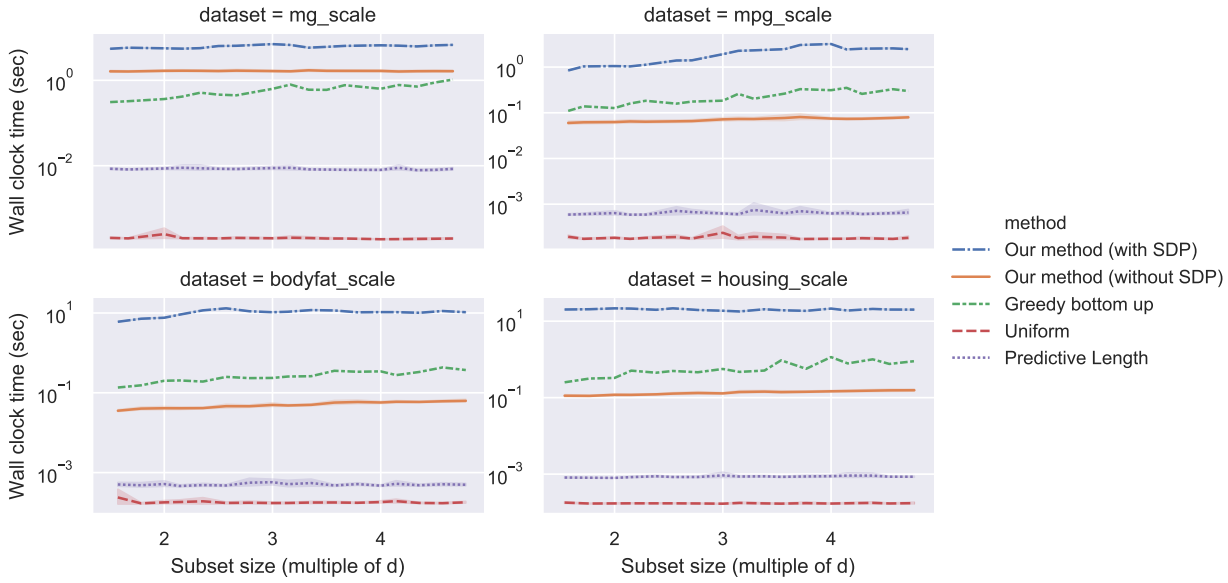


Figure 2.5: Runtimes of the methods compared. Our method (without SDP) is within an order of magnitude of greedy bottom up and faster in 3 out of 4 cases. The gap between our method with and without SDP is attributable to the SDP solver, making investigation of more efficient solvers and approximate solutions an interesting direction for future work.

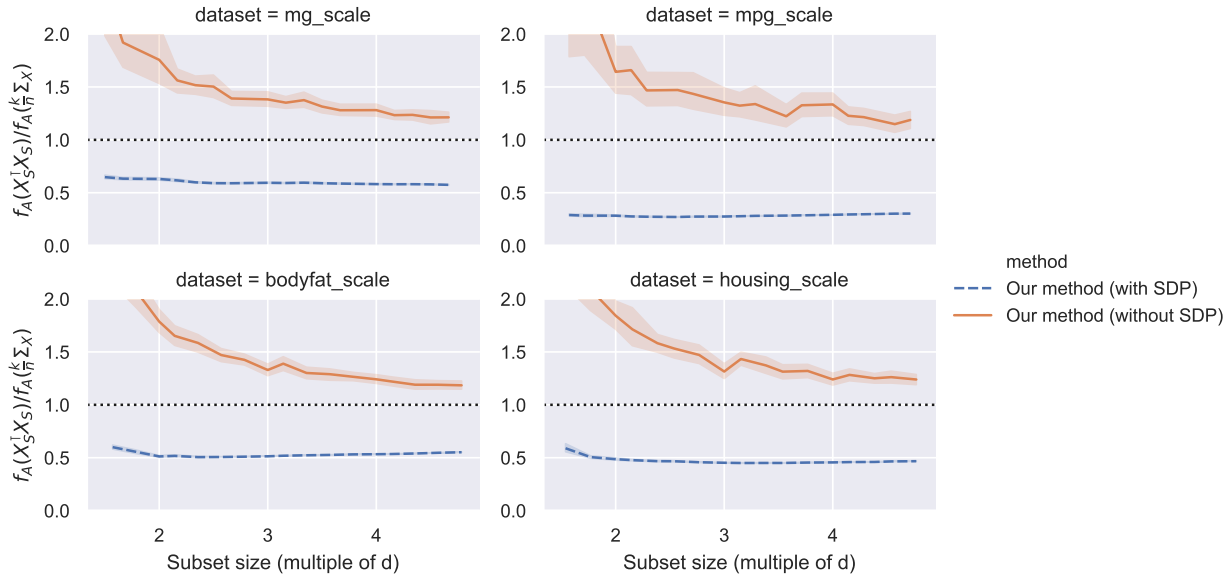


Figure 2.6: The ratio controlled by Lemma 2.6. This ratio converges to 1 as $k \to n$ and is close to 1 across all real world datasets, suggesting that $f_{\mathbf{A}}(\frac{k}{n}\mathbf{\Sigma_X})$ is an appropriate problem-dependent scale for Bayesian A-optimal experimental design.

Chapter 3

Exact expressions for double descent in determinantal random designs

Building on the theory established in chapter 2, in this chapter we analyze a determinantal "surrogate" random design to develop new non-asymptotic theory on double descent in overparameterized linear models. Double descent refers to the phase transition that is exhibited by the generalization error of unregularized learning models when varying the ratio between the number of parameters and the number of training samples. The recent success of highly over-parameterized machine learning models such as deep neural networks has motivated a theoretical analysis of the double descent phenomenon in classical models such as linear regression which can also generalize well in the over-parameterized regime. We provide the first exact non-asymptotic expressions for double descent of the minimum norm linear estimator. Our approach involves constructing a special determinantal point process which we call surrogate random design, to replace the standard i.i.d. design of the training sample. This surrogate design admits exact expressions for the mean squared error of the estimator while preserving the key properties of the standard design. We also establish an exact implicit regularization result for over-parameterized training samples. In particular, we show that, for the surrogate design, the implicit bias of the unregularized minimum norm estimator precisely corresponds to solving a ridge-regularized least squares problem on the population distribution. In our analysis we introduce a new mathematical tool of independent interest: the class of random matrices for which determinant commutes with expectation. Some of the results presented in this chapter were first published in Michał Dereziński, Feynman Liang, and Michael W Mahoney. "Exact expressions for double descent and implicit regularization via surrogate random design". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 5152–5164.

3.1 Introduction

Classical statistical learning theory asserts that to achieve generalization one must use training sample size that sufficiently exceeds the complexity of the learning model, where the latter is typically represented by the number of parameters [or some related structural parameter; see FHT01]. In particular, this seems to suggest the conventional wisdom that one should not use models that fit the training data exactly. However, modern machine learning practice often seems to go against this intuition, using models with so many parameters that the training data can be perfectly interpolated, in which case the training error vanishes. It has been shown that models such as deep neural networks, as well as certain so-called interpolating kernels and decision trees, can generalize well in this regime. In particular, [Bel+19] empirically demonstrated a phase transition in generalization performance of learning models which occurs at an *interpolation thershold*, i.e., a point where training error goes to zero (as one varies the ratio between the model complexity and the sample size). Moving away from this threshold in either direction tends to reduce the generalization error, leading to the so-called *double descent* curve.

To understand this surprising phenomenon, in perhaps the simplest possible setting, we study it in the context of linear or least squares regression. Consider a full rank $n \times d$ data matrix $\mathbf{X}$ and a vector $\mathbf{y}$ of responses corresponding to each of the n data points (the rows of $\mathbf{X}$), where we wish to find the best linear model $\mathbf{X}\mathbf{w} \approx \mathbf{y}$, parameterized by a d-dimensional vector $\mathbf{w}$. The simplest example of an estimator that has been shown to exhibit the double descent phenomenon [BHX19] is the Moore-Penrose estimator, $\widehat{\mathbf{w}} = \mathbf{X}^\dagger \mathbf{y}$: in the so-called over-determined regime, i.e., when $n > d$, it corresponds to the least squares solution, i.e., $\operatorname{argmin}_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$; and in the under-determined regime (also known as over-parameterized or interpolating), i.e., when $n < d$, it corresponds to the minimum norm solution to the linear system $\mathbf{X}\mathbf{w} = \mathbf{y}$. Given the ubiquity of linear regression and the Moore-Penrose solution, e.g., in kernel-based machine learning, studying the performance of this estimator can shed some light on the effects of over-parameterization/interpolation in machine learning more generally. Of particular interest are results that are exact (i.e., not upper/lower bounds) and non-asymptotic (i.e., for large but still finite n and d).

We build on methods from Randomized Numerical Linear Algebra (RandNLA) in order to obtain *exact non-asymptotic expressions* for the mean squared error (MSE) of the Moore-Penrose estimator (see Theorem 3.1). This provides a precise characterization of the double descent phenomenon for the linear regression problem. In obtaining these results, we are able to provide precise formulas for the *implicit regularization* induced by minimum norm solutions of under-determined training samples, relating it to classical ridge regularization (see Theorem 3.2). To obtain our precise results, we use a somewhat non-standard random design, based on a specially chosen determinantal point process (DPP), which we term surrogate random design. DPPs are a family of non-i.i.d. sampling distributions which are typically used to induce diversity in the produced samples [KT12]. Our aim in using a DPP as a surrogate design is very different: namely, to make certain quantities (such as the MSE) analytically tractable, while accurately *preserving* the underlying properties of

the original data distribution. This strategy might seem counter-intuitive since DPPs are typically found most useful when they *differ* from the data distribution. However, we show both theoretically (Theorem 3.3) and empirically (Section 3.8), that for many commonly studied data distributions, such as multivariate Gaussians, our DPP-based surrogate design accurately preserves the key properties of the standard i.i.d. design (such as the MSE), and even matches it exactly in the high-dimensional asymptotic limit. In our analysis of the surrogate design, we introduce the concept of *determinant preserving random matrices* (Section 3.4), a class of random matrices for which determinant commutes with expectation, which should be of independent interest.

Main results: double descent and implicit regularization

As the performance metric in our analysis, we use the *mean squared error* (MSE), defined as $\mathrm{MSE}[\widehat{\mathbf{w}}] = \mathbb{E}\big[\|\widehat{\mathbf{w}} - \mathbf{w}^*\|^2\big]$, where $\mathbf{w}^*$ is a fixed underlying linear model of the responses. In analyzing the MSE, we make the following standard assumption that the response noise is homoscedastic.

Assumption 3.1 (Homoscedastic noise) *The noise $\xi = y(\mathbf{x}) - \mathbf{x}^\top \mathbf{w}^*$ has mean 0 and variance σ^2.*

Our main result provides an exact expression for the MSE of the Moore-Penrose estimator under our surrogate design denoted $\bar{\mathbf{X}} \sim S_\mu^n$, where μ is the d-variate distribution of the row vector $\mathbf{x}^\top$ and n is the sample size. This surrogate is used in place of the standard $n \times d$ random design $\mathbf{X} \sim \mu^n$, where n data points (the rows of $\mathbf{X}$) are sampled independently from μ. We form the surrogate by constructing a determinantal point process with μ as the background measure, so that $S_\mu^n(\mathbf{X}) \propto \mathrm{pdet}(\mathbf{X}\mathbf{X}^\top)\mu(\mathbf{X})$, where $\mathrm{pdet}(\cdot)$ denotes the pseudo-determinant (details in Section 3.3). Unlike for the standard design, our MSE formula is fully expressible as a function of the covariance matrix $\mathbf{\Sigma}_\mu = \mathbb{E}_\mu[\mathbf{x}\mathbf{x}^\top]$. To state our main result, we need an additional minor assumption on μ which is satisfied by most standard continuous distributions (e.g., multivariate Gaussians).

Assumption 3.2 (General position) *For $1 \le n \le d$, if $\mathbf{X} \sim \mu^n$, then $\mathrm{rank}(\mathbf{X}) = n$ almost surely.*

Under Assumptions 3.1 and 3.2, we can establish our first main result, stated as the following theorem, where we use $\mathbf{X}^\dagger$ to denote the Moore-Penrose inverse of $\mathbf{X}$.

Theorem 3.1 (Exact non-asymptotic MSE) *If the response noise is homoscedastic (Assumption 3.1) and μ is in general position (Assumption 3.2), then for any $\mathbf{w} \in \mathbb{R}^d$, $\bar{\mathbf{X}} \sim S_\mu^n$ (Definition 3.3), and $\bar{y}_i = y(\bar{\mathbf{x}}_i)$,*

$$\mathrm{MSE}\big[\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}}\big] = \begin{cases} \sigma^2 \,\mathrm{tr}\big((\mathbf{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\big) \cdot \frac{1-\alpha_n}{d-n} \;+\; \frac{\mathbf{w}^{*\top}(\mathbf{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\mathbf{w}^*}{\mathrm{tr}((\mathbf{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1})} \cdot (d-n), & \text{for } n < d, \\ \sigma^2 \,\mathrm{tr}(\mathbf{\Sigma}_\mu^{-1}), & \text{for } n = d, \\ \sigma^2 \,\mathrm{tr}(\mathbf{\Sigma}_\mu^{-1}) \cdot \frac{1-\beta_n}{n-d}, & \text{for } n > d, \end{cases}$$

(a) Surrogate MSE expressions (Theorem 3.1) closely match numerical estimates even for non-isotropic features. Eigenvalue decay leads to a steeper descent curve in the under-determined regime ($n < d$).

(b) The mean of the estimator $\mathbf{X}^\dagger\mathbf{y}$ exhibits shrinkage which closely matches the shrinkage of a ridge-regularized least squares optimum (theory lines), as characterized by Theorem 3.2.

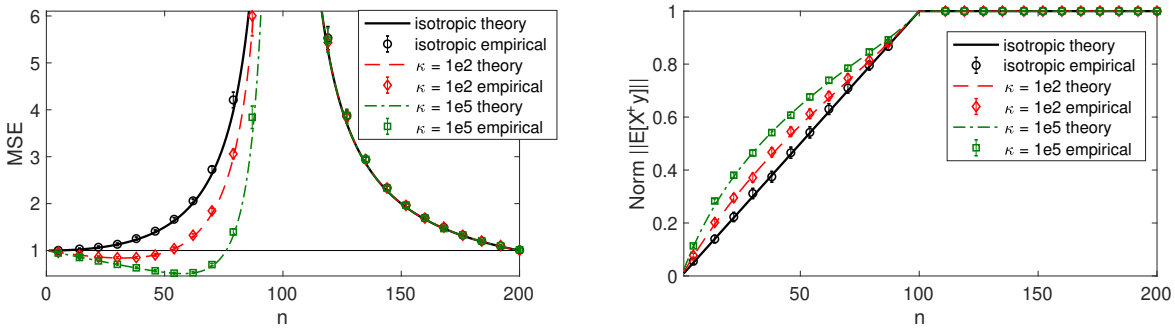


Figure 3.1: Illustration of the main results for $d = 100$ and $\mu = \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ where $\mathbf{\Sigma}$ is diagonal with eigenvalues decaying exponentially and scaled so that $\mathrm{tr}(\mathbf{\Sigma}^{-1}) = d$. We use our surrogate formulas to plot (a) the MSE (Theorem 3.1) and (b) the norm of the expectation (Theorem 3.2) of the Moore-Penrose estimator (*theory* lines), accompanied by the empirical estimates based on the standard i.i.d. design (error bars are three times the standard error of the mean). We consider three different condition numbers κ of $\mathbf{\Sigma}$, with *isotropic* corresponding to $\kappa = 1$, i.e., $\mathbf{\Sigma} = \mathbf{I}$. We use $\sigma^2 = 1$ and $\mathbf{w}^* = \frac{1}{\sqrt{d}}\mathbf{1}$.

with $\lambda_n \geq 0$ *defined by* $n = \mathrm{tr}(\mathbf{\Sigma}_\mu(\mathbf{\Sigma}_\mu + \lambda_n\mathbf{I})^{-1})$, $\alpha_n = \det(\mathbf{\Sigma}_\mu(\mathbf{\Sigma}_\mu + \lambda_n\mathbf{I})^{-1})$ *and* $\beta_n = \mathrm{e}^{d-n}$.

Definition 3.1 *We will use* $\mathcal{M} = \mathcal{M}(\mathbf{\Sigma}_\mu, \mathbf{w}^*, \sigma^2, n)$ *to denote the above expressions for* $\mathrm{MSE}\big[\bar{\mathbf{X}}^\dagger\bar{\mathbf{y}}\big]$.

Proof of Theorem 3.1 is given in Section 3.5. For illustration, we plot the MSE expressions in Figure 3.1a, comparing them with empirical estimates of the true MSE under the i.i.d. design for a multivariate Gaussian distribution $\mu = \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ with several different covariance matrices $\mathbf{\Sigma}$. We keep the number of features d fixed to 100 and vary the number of samples n, observing a double descent peak at $n = d$. We observe that our theory aligns well with the empirical estimates, whereas previously, no such theory was available except for special cases such as $\mathbf{\Sigma} = \mathbf{I}$ (more details in Theorem 3.3 and Section 3.8). The plots show that varying the spectral decay of $\mathbf{\Sigma}$ has a significant effect on the shape of the curve in the under-determined regime. We use the horizontal line to denote the MSE of the null estimator $\mathrm{MSE}[\mathbf{0}] = \|\mathbf{w}^*\|^2 = 1$. When the eigenvalues of $\mathbf{\Sigma}$ decay rapidly, then the Moore-Penrose estimator suffers less error than the null estimator for some values of $n < d$, and the curve exhibits a local optimum in this regime.

One important aspect of Theorem 3.1 comes from the relationship between n and the parameter λ_n, which together satisfy $n = \mathrm{tr}(\mathbf{\Sigma}_\mu(\mathbf{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1})$. This expression is precisely the classical notion of *effective dimension* for ridge regression regularized with λ_n [AM15], and it arises here even though there is no explicit ridge regularization in the problem being considered in Theorem 3.1. The global solution to the ridge regression task (i.e., ℓ_2-regularized least squares) with parameter λ is defined as:

$$\operatorname*{argmin}_{\mathbf{w}} \left\{ \mathbb{E}_{\mu,y}\left[\left(\mathbf{x}^\top \mathbf{w} - y(\mathbf{x})\right)^2\right] + \lambda \|\mathbf{w}\|^2 \right\} \;=\; (\mathbf{\Sigma}_\mu + \lambda \mathbf{I})^{-1} \mathbf{v}_{\mu,y}, \quad \text{where } \mathbf{v}_{\mu,y} = \mathbb{E}_{\mu,y}[y(\mathbf{x})\,\mathbf{x}].$$

When Assumption 3.1 holds, then $\mathbf{v}_{\mu,y} = \mathbf{\Sigma}_\mu \mathbf{w}^*$, however ridge-regularized least squares is well-defined for much more general response models. Our second result makes a direct connection between the (expectation of the) unregularized minimum norm solution on the sample and the global ridge-regularized solution. While the under-determined regime (i.e., $n < d$) is of primary interest to us, for completeness we state this result for arbitrary values of n and d. Note that, just like the definition of regularized least squares, this theorem applies more generally than Theorem 3.1, in that it does *not* require the responses to follow any linear model as in Assumption 3.1 (proof in Section 3.6).

Theorem 3.2 (Implicit regularization of Moore-Penrose estimator) *For μ satisfying Assumption 3.2 and $y(\cdot)$ s.t. $\mathbf{v}_{\mu,y} = \mathbb{E}_{\mu,y}[y(\mathbf{x})\,\mathbf{x}]$ is well-defined, $\bar{\mathbf{X}} \sim S_\mu^n$ (Definition 3.3) and $\bar{y}_i = y(\bar{\mathbf{x}}_i)$,*

$$\mathbb{E}\left[\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}}\right] = \begin{cases} (\mathbf{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1} \mathbf{v}_{\mu,y} & \text{for } n < d, \\ \mathbf{\Sigma}_\mu^{-1} \mathbf{v}_{\mu,y} & \text{for } n \geq d, \end{cases}$$

where, as in Theorem 3.1, λ_n is such that the effective dimension $\mathrm{tr}(\mathbf{\Sigma}_\mu(\mathbf{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1})$ equals n.

That is, when $n < d$, the Moore-Penrose estimator (which itself is not regularized), computed on the random training sample, in expectation equals the global ridge-regularized least squares solution of the underlying regression problem. Moreover, λ_n, i.e., the amount of implicit ℓ_2-regularization, is controlled by the degree of over-parameterization in such a way as to ensure that n becomes the ridge effective dimension (a.k.a. the effective degrees of freedom).

We illustrate this result in Figure 3.1b, plotting the norm of the expectation of the Moore-Penrose estimator. As for the MSE, our surrogate theory aligns well with the empirical estimates for i.i.d. Gaussian designs, showing that the shrinkage of the unregularized estimator in the under-determined regime matches the implicit ridge-regularization characterized by Theorem 3.2. While the shrinkage is a linear function of the sample size n for isotropic features (i.e., $\mathbf{\Sigma} = \mathbf{I}$), it exhibits a non-linear behavior for other spectral decays. Such *implicit regularization* has been studied previously [see, e.g., ML11; M W12]; it has been observed empirically for RandNLA sampling algorithms [PMB15]; and it has also received attention more generally within the context of neural networks [Ney17]. While our implicit

regularization result is limited to the Moore-Penrose estimator, this new connection (and others, described below) between the minimum norm solution of an unregularized underdetermined system and a ridge-regularized least squares solution offers a simple interpretation for the implicit regularization observed in modern machine learning architectures.

Our exact non-asymptotic expressions in Theorem 3.1 and our exact implicit regularization results in Theorem 3.2 are derived for the surrogate design, which is a non-i.i.d. distribution based on a determinantal point process. However, Figure 3.1 suggests that those expressions accurately describe the MSE (up to lower order terms) also under the standard i.i.d. design $\mathbf{X} \sim \mu^n$ when μ is a multivariate Gaussian. As a third result, we verify that the surrogate expressions for the MSE are asymptotically consistent with the MSE of an i.i.d. design, for a wide class of distributions which include multivariate Gaussians.

Theorem 3.3 (Asymptotic consistency of surrogate design) *Let $\mathbf{X} \in \mathbb{R}^{n\times d}$ have i.i.d. rows $\mathbf{x}_i^\top = \mathbf{z}_i^\top \boldsymbol{\Sigma}^{\frac{1}{2}}$ where $\mathbf{z}_i$ has independent zero mean and unit variance sub-Gaussian entries, and suppose that Assumptions 3.1 and 3.2 are satisfied. Furthermore, suppose that there exist $c, C, C^* \in \mathbb{R}_{>0}$ such that $C\mathbf{I} \succeq \boldsymbol{\Sigma} \succeq c\mathbf{I} \succ 0$ and $\|\mathbf{w}^*\| \leq C^*$. Then*

$$\mathrm{MSE}\big[\mathbf{X}^\dagger \mathbf{y}\big] - \mathcal{M}(\boldsymbol{\Sigma}, \mathbf{w}^*, \sigma^2, n) \to 0$$

with probability one as $d, n \to \infty$ with $n/d \to \bar{c} \in (0, \infty) \setminus \{1\}$.

The above result is particularly remarkable since our surrogate design is a determinantal point process. DPPs are commonly used in ML to ensure that the data points in a sample are well spread-out. However, if the data distribution is sufficiently regular (e.g., a multivariate Gaussian), then the i.i.d. samples are already spread-out reasonably well, so rescaling the distribution by a determinant has a negligible effect that vanishes in the high-dimensional regime. Furthermore, our empirical estimates (Figure 3.1) suggest that the surrogate expressions are accurate not only in the asymptotic limit, but even for moderately large dimensions. Based on a detailed empirical analysis described in Section 3.8, we conjecture that the convergence described in Theorem 3.3 has the rate of $O(1/d)$.

3.2 Related work

There is a large body of related work, which for simplicity we cluster into three groups.

Double descent. The double descent phenomenon has been observed empirically in a number of learning models, including neural networks [Bel+19; Gei+19], kernel methods [BMM18; BRT19], nearest neighbor models [BHM18], and decision trees [Bel+19]. The theoretical analysis of double descent, and more broadly the generalization properties of interpolating estimators, have primarily focused on various forms of linear regression [Bar+19; LR19; Has+19; Mut+19]. Note that while we analyze the classical mean squared error, many works focus on the squared prediction error. Also, unlike in our work, some of the literature on double descent deals with linear regression in the so-called *misspecified* setting, where the

set of observed features does not match the feature space in which the response model is linear [BHX19; Has+19; Mit19; MM19b], e.g., when the learner observes a random subset of d features from a larger population.

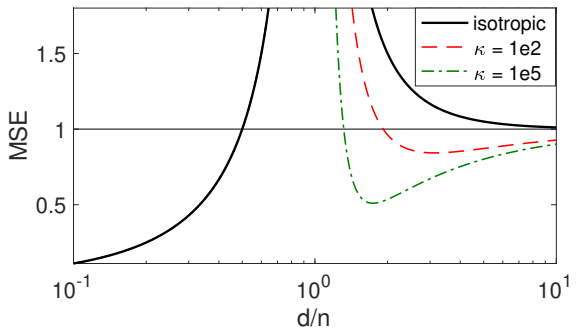


Figure 3.2: Surrogate MSE as a function of d/n, with n fixed to 100 and varying dimension d and condition number κ, for signal-to-noise ratio SNR $= 1$.

The most directly comparable to our setting is the recent work of [Has+19]. They study how varying the feature dimension affects the (asymptotic) generalization error for linear regression, however their analysis is limited to certain special settings such as an isotropic data distribution. As an additional point of comparison, in Figure 3.2 we plot the MSE expressions of Theorem 3.1 when varying the feature dimension d (the setup is the same as in Figure 3.1). Our plots follow the trends outlined by [Has+19] for the isotropic case (see their Figure 2), but the spectral decay of the covariance (captured by our new MSE expressions) has a significant effect on the descent curve. This leads to generalization in the under-determined regime even when the signal-to-noise ratio (SNR $= \|\mathbf{w}^*\|^2/\sigma^2$) is 1, unlike suggested by [Has+19].

RandNLA and DPPs. Randomized Numerical Linear Algebra [DM16; DM17] has traditionally focused on obtaining purely algorithmic improvements for tasks such as least squares regression, but there has been growing interest in understanding the statistical properties of these randomized methods [PMB15; RM16] and a beyond worst-case analysis [Der+20b]. Determinantal point processes [DM21; KT12] have been recently shown to combine strong worst-case regression guarantees with elegant statistical properties [DW17]. However, these results are limited to the over-determined setting [DWH18; DWH19a; Der+19] and ridge regression [DW18b; DLM20a]. Our results are also related to recent work on using DPPs to analyze the expectation of the inverse [DM19; Der+20a] and generalized inverse [MDK20; DKM20] of a subsampled matrix.

Implicit regularization. The term implicit regularization typically refers to the notion that approximate computation can implicitly lead to statistical regularization. See [ML11; PM11; DM14] and references therein for early work on the topic; and see [M W12] for an overview. More recently, often motivated by neural networks, there has been work on

implicit regularization that typically considered SGD-based optimization algorithms. See, e.g., theoretical results [NTS14; Ney17; Sou+18; Gun+17; Aro+19; Kub+19] as well as extensive empirical studies [MM18; MM19a]. The implicit regularization observed by us is different in that it is not caused by an inexact approximation algorithm (such as SGD) but rather by the selection of one out of many exact solutions (e.g., the minimum norm solution). In this context, most relevant are the asymptotic results of [KLS18] and [LJB19].

3.3 Surrogate random designs

In this section, we provide the definition of our surrogate random design S_μ^n, where μ is a d-variate probability measure and n is the sample size. This distribution is used in place of the standard random design μ^n consisting of n row vectors drawn independently from μ.

Preliminaries. For an $n \times n$ matrix $\mathbf{A}$, we use $\mathrm{pdet}(\mathbf{A})$ to denote the pseudo-determinant of $\mathbf{A}$, which is the product of non-zero eigenvalues (repeated eigenvalues are taken to the power of their algebraic multiplicity). For index subsets $\mathcal{I}$ and $\mathcal{J}$, we use $\mathbf{A}_{\mathcal{I},\mathcal{J}}$ to denote the submatrix of $\mathbf{A}$ with rows indexed by $\mathcal{I}$ and columns indexed by $\mathcal{J}$. We may write $\mathbf{A}_{\mathcal{I},*}$ to indicate that we take a subset of rows. We let $\mathbf{X} \sim \mu^k$ denote a $k \times d$ random matrix with rows drawn i.i.d. according to μ, and the ith row is denoted as $\mathbf{x}_i^\top$. We also let $\mathbf{\Sigma}_\mu = \mathbb{E}_\mu[\mathbf{x}\mathbf{x}^\top]$, where $\mathbb{E}_\mu$ refers to the expectation with respect to $\mathbf{x}^\top \sim \mu$, assuming throughout that $\mathbf{\Sigma}_\mu$ is well-defined and positive definite. We use $\mathrm{Poisson}(\gamma)_{\leq a}$ as the Poisson distribution restricted to $[0, a]$, whereas $\mathrm{Poisson}(\gamma)_{\geq a}$ is restricted to $[a, \infty)$. We also let $\#(\mathbf{X})$ denote the number of rows of $\mathbf{X}$.

Definition 3.2 *Let μ satisfy Assumption 3.2 and let K be a random variable over $\mathbb{Z}_{\geq 0}$. A determinantal design $\bar{\mathbf{X}} \sim \mathrm{Det}(\mu, K)$ is a distribution with the same domain as $\mathbf{X} \sim \mu^K$ such that for any event E measurable w.r.t. $\mathbf{X}$, we have*

$$\Pr\{\bar{\mathbf{X}} \in E\} = \frac{\mathbb{E}[\mathrm{pdet}(\mathbf{X}\mathbf{X}^\top)\mathbf{1}_{[\mathbf{X}\in E]}]}{\mathbb{E}[\mathrm{pdet}(\mathbf{X}\mathbf{X}^\top)]}.$$

The above definition can be interpreted as rescaling the density function of μ^K by the pseudo-determinant, and then renormalizing it. We now construct our surrogate design S_μ^n by appropriately selecting the random variable K. The obvious choice of $K = n$ does *not* result in simple closed form expressions for the MSE in the under-determined regime (i.e., $n < d$), which is the regime of primary interest to us. Instead, we derive our random variables K from the Poisson distribution.

Definition 3.3 *For μ satisfying Assumption 3.2, define surrogate design S_μ^n as $\mathrm{Det}(\mu, K)$ where:*

1. *if $n < d$, then $K \sim \mathrm{Poisson}(\gamma_n)_{\leq d}$ with γ_n as the solution of $n = \mathrm{tr}(\mathbf{\Sigma}_\mu(\mathbf{\Sigma}_\mu + \frac{1}{\gamma_n}\mathbf{I})^{-1})$,*
2. *if $n = d$, then we simply let $K = d$,*
3. *if $n > d$, then $K \sim \mathrm{Poisson}(\gamma_n)_{\geq d}$ with $\gamma_n = n - d$.*

Note that the under-determined case, i.e., $n < d$, is restricted to $K \leq d$ so that, under Assumption 3.2, $\mathrm{pdet}(\mathbf{X}\mathbf{X}^\top) = \det(\mathbf{X}\mathbf{X}^\top)$ with probability 1. On the other hand in the over-determined case, i.e., $n > d$, we have $K \geq d$ so that $\mathrm{pdet}(\mathbf{X}\mathbf{X}^\top) = \det(\mathbf{X}^\top\mathbf{X})$. In the special case of $n = d = K$ both of these equations are satisfied: $\mathrm{pdet}(\mathbf{X}\mathbf{X}^\top) = \det(\mathbf{X}^\top\mathbf{X}) = \det(\mathbf{X}\mathbf{X}^\top) = \det(\mathbf{X})^2$.

We first record an important property of the design S_μ^d which can be used to construct an over-determined design for any $n > d$. A similar version of this result was also previously shown by [DWH19b] for a different determinantal design.

Lemma 3.1 *Let $\bar{\mathbf{X}} \sim S_\mu^d$ and $\mathbf{X} \sim \mu^K$, where $K \sim \mathrm{Poisson}(\gamma)$. Then the matrix composed of a random permutation of the rows from $\bar{\mathbf{X}}$ and $\mathbf{X}$ is distributed according to $S_\mu^{d+\gamma}$.*

Proof Let $\widetilde{\mathbf{X}}$ denote the matrix constructed from the permuted rows of $\bar{\mathbf{X}}$ and $\mathbf{X}$. Letting $\mathbf{Z} \sim \mu^{K+d}$, we derive the probability $\Pr\{\widetilde{\mathbf{X}} \in E\}$ by summing over the possible index subsets $S \subseteq [K+d]$ that correspond to the rows coming from $\bar{\mathbf{X}}$:

$$\begin{aligned}
\Pr\{\widetilde{\mathbf{X}} \in E\} &= \mathbb{E}\left[\frac{1}{\binom{K+d}{d}} \sum_{S:\,|S|=d} \frac{\mathbb{E}[\det(\mathbf{Z}_{S,*})^2 \mathbf{1}_{[\mathbf{Z}\in E]} \mid K]}{d!\det(\boldsymbol{\Sigma}_\mu)}\right] \\
&= \sum_{k=0}^{\infty} \frac{\gamma^k \mathrm{e}^{-\gamma}}{k!} \frac{\gamma^d k!}{(k+d)!} \frac{\mathbb{E}\big[\sum_{S:\,|S|=d} \det(\mathbf{Z}_{S,*})^2 \mathbf{1}_{[\mathbf{Z}\in E]} \mid K = k\big]}{\det(\gamma\boldsymbol{\Sigma}_\mu)} \\
&\overset{(*)}{=} \sum_{k=0}^{\infty} \frac{\gamma^{k+d} \mathrm{e}^{-\gamma}}{(k+d)!} \frac{\mathbb{E}[\det(\mathbf{Z}^\top\mathbf{Z}) \mathbf{1}_{[\mathbf{Z}\in E]} \mid K = k]}{\det(\gamma\boldsymbol{\Sigma}_\mu)},
\end{aligned}$$

where $(*)$ uses the Cauchy-Binet formula to sum over all subsets S of size d. Finally, since the sum shifts from k to $k+d$, the last expression can be rewritten as $\mathbb{E}[\det(\mathbf{X}^\top\mathbf{X})\mathbf{1}_{[\mathbf{X}\in E]}]/\det(\gamma\boldsymbol{\Sigma}_\mu)$, where recall that $\mathbf{X} \sim \mu^K$ and $K \sim \mathrm{Poisson}(\gamma)$, matching the definition of $S_\mu^{d+\gamma}$. ■

Another non-trivial property of the surrogate design S_μ^n is that the expected sample size is in fact always equal to n.

Lemma 3.2 *Let $\bar{\mathbf{X}} \sim S_\mu^n$ for any $n > 0$. Then, we have $\mathbb{E}[\#(\bar{\mathbf{X}})] = n$.*

Proof of Lemma 3.2 The result is obvious when $n = d$, whereas for $n > d$ it is an immediate consequence of Lemma 3.1. Finally, for $n < d$ the expected sample size follows as a corollary of Lemma 3.3, which states that

$$\text{(Lemma 3.3)} \qquad \mathbb{E}\big[\mathbf{I} - \bar{\mathbf{X}}^\dagger\bar{\mathbf{X}}\big] = (\gamma_n\boldsymbol{\Sigma}_\mu + \mathbf{I})^{-1},$$

where $\bar{\mathbf{X}}^\dagger\bar{\mathbf{X}}$ is the orthogonal projection onto the subspace spanned by the rows of $\bar{\mathbf{X}}$. Since the rank of this subspace is equal to the number of the rows, we have $\#(\bar{\mathbf{X}}) = \mathrm{tr}(\bar{\mathbf{X}}^\dagger\bar{\mathbf{X}})$, so

$$\mathbb{E}\big[\#(\bar{\mathbf{X}})\big] = d - \mathrm{tr}\big((\gamma_n\boldsymbol{\Sigma}_\mu + \mathbf{I})^{-1}\big) = \mathrm{tr}\big(\gamma_n\boldsymbol{\Sigma}_\mu(\gamma_n\boldsymbol{\Sigma}_\mu + \mathbf{I})^{-1}\big) = n,$$

which completes the proof. ■

Our general template for computing expectations under a surrogate design $\bar{\mathbf{X}} \sim \mathbf{S}_\mu^n$ is to use the following expressions based on the i.i.d. random design $\mathbf{X} \sim \mu^K$:

$$\mathbb{E}[F(\bar{\mathbf{X}})] = \begin{cases} \frac{\mathbb{E}[\det(\mathbf{X}\mathbf{X}^\top)F(\mathbf{X})]}{\mathbb{E}[\det(\mathbf{X}\mathbf{X}^\top)]} & K \sim \text{Poisson}(\gamma_n) \quad \text{for } n < d, \\ \frac{\mathbb{E}[\det(\mathbf{X})^2 F(\mathbf{X})]}{\mathbb{E}[\det(\mathbf{X})^2]} & K = d \quad \text{for } n = d, \\ \frac{\mathbb{E}[\det(\mathbf{X}^\top\mathbf{X})F(\mathbf{X})]}{\mathbb{E}[\det(\mathbf{X}^\top\mathbf{X})]} & K \sim \text{Poisson}(\gamma_n) \quad \text{for } n > d. \end{cases} \tag{3.1}$$

These formulas follow from Definitions 3.2 and 3.3 because the determinants $\det(\mathbf{X}\mathbf{X}^\top)$ and $\det(\mathbf{X}^\top\mathbf{X})$ are non-zero precisely in the regimes $n \leq d$ and $n \geq d$, respectively, which is why we can drop the restrictions on the range of the Poisson distribution. We compute the normalization constants by introducing the concept of determinant preserving random matrices, discussed in Section 3.4.

Proof sketch of Theorem 3.1 We focus here on the under-determined regime (i.e., $n < d$), highlighting the key new expectation formulas we develop to derive the MSE expressions for surrogate designs. A standard decomposition of the MSE yields:

$$\text{MSE}\big[\bar{\mathbf{X}}^\dagger\bar{\mathbf{y}}\big] = \mathbb{E}\big[\|\bar{\mathbf{X}}^\dagger(\bar{\mathbf{X}}\mathbf{w}^* + \boldsymbol{\xi}) - \mathbf{w}^*\|^2\big] = \sigma^2\mathbb{E}\big[\text{tr}\big((\bar{\mathbf{X}}^\top\bar{\mathbf{X}})^\dagger\big)\big] + \mathbf{w}^{*\top}\mathbb{E}\big[\mathbf{I} - \bar{\mathbf{X}}^\dagger\bar{\mathbf{X}}\big]\mathbf{w}^*. \tag{3.2}$$

Thus, our task is to find closed form expressions for the two expectations above. The latter, which is the expected projection onto the complement of the row-span of $\bar{\mathbf{X}}$, is proven in Section 3.6.

Lemma 3.3 *If $\bar{\mathbf{X}} \sim S_\mu^n$ and $n < d$, then we have: $\mathbb{E}\big[\mathbf{I} - \bar{\mathbf{X}}^\dagger\bar{\mathbf{X}}\big] = (\gamma_n\boldsymbol{\Sigma}_\mu + \mathbf{I})^{-1}$.*

No such expectation formula is known for i.i.d. designs, except when μ is an isotropic Gaussian. In Section 3.6, we also prove a generalization of Lemma 3.3 which is then used to establish our implicit regularization result (Theorem 3.2). We next give an expectation formula for the trace of the Moore-Penrose inverse of the covariance matrix for a surrogate design (proof in Section 3.5).

Lemma 3.4 *If $\bar{\mathbf{X}} \sim S_\mu^n$ and $n < d$, then: $\mathbb{E}\big[\text{tr}\big((\bar{\mathbf{X}}^\top\bar{\mathbf{X}})^\dagger\big)\big] = \gamma_n\big(1 - \det\big((\frac{1}{\gamma_n}\mathbf{I} + \boldsymbol{\Sigma}_\mu)^{-1}\boldsymbol{\Sigma}_\mu\big)\big)$.*

Note the implicit regularization term which appears in both formulas, given by $\lambda_n = \frac{1}{\gamma_n}$. Since $n = \text{tr}(\boldsymbol{\Sigma}_\mu(\boldsymbol{\Sigma}_\mu + \lambda_n\mathbf{I})^{-1}) = d - \lambda_n\text{tr}((\boldsymbol{\Sigma}_\mu + \lambda_n\mathbf{I})^{-1})$, it follows that $\lambda_n = (d-n)/\text{tr}((\boldsymbol{\Sigma}_\mu + \lambda_n\mathbf{I})^{-1})$. Combining this with Lemmas 3.3 and 3.4, we recover the surrogate MSE expression in Theorem 3.1.

3.4 Determinant preserving random matrices

In this section, we introduce the key tool for computing expectation formulas of matrix determinants. It is used in our analysis of the surrogate design, and it should be of independent interest.

The key question motivating the following definition is: *When does taking expectation commute with computing a determinant for a square random matrix?*

Definition 3.4 *A random $d \times d$ matrix $\mathbf{A}$ is called determinant preserving (d.p.), if*

$$\mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}})\big] = \det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}}]\big) \quad \textit{for all } \mathcal{I},\mathcal{J} \subseteq [d] \textit{ s.t. } |\mathcal{I}| = |\mathcal{J}|.$$

We next give a few simple examples to provide some intuition. First, note that every 1×1 random matrix is determinant preserving simply because taking a determinant is an identity transfomation in one dimension. Similarly, every fixed matrix is determinant preserving because in this case taking the expectation is an identity transformation. In all other cases, however, Definition 3.4 has to be verified more carefully. Further examples (positive and negative) follow.

Example 3.1 *If $\mathbf{A}$ has i.i.d. Gaussian entries $a_{ij} \sim \mathcal{N}(0,1)$, then $\mathbf{A}$ is d.p. because $\mathbb{E}[\det(\mathbf{A})] = 0$.*

In fact, it can be shown that all random matrices with independent entries are determinant preserving. However, this is not a necessary condition.

Example 3.2 *Let $\mathbf{A} = s\,\mathbf{Z}$, where $\mathbf{Z}$ is fixed with $\operatorname{rank}(\mathbf{Z}) = r$, and s is a scalar random variable. Then for $|\mathcal{I}| = |\mathcal{J}| = r$ we have*

$$\mathbb{E}\big[\det(s\,\mathbf{Z}_{\mathcal{I},\mathcal{J}})\big] = \mathbb{E}[s^r]\det(\mathbf{Z}_{\mathcal{I},\mathcal{J}}) = \det\Big(\big(\mathbb{E}[s^r]\big)^{\frac{1}{r}}\,\mathbf{Z}_{\mathcal{I},\mathcal{J}}\Big),$$

so if $r = 1$ then $\mathbf{A}$ is determinant preserving, whereas if $r > 1$ and $\operatorname{Var}[s] > 0$ then it is not.

We use $\operatorname{adj}(\mathbf{A})$ to denote the adjugate of $\mathbf{A}$, defined as follows: the (i,j)th entry of $\operatorname{adj}(\mathbf{A})$ is $(-1)^{i+j}\det(\mathbf{A}_{[n]\setminus\{j\},[n]\setminus\{i\}})$. We will use two useful identities related to the adjugate: (1) $\operatorname{adj}(\mathbf{A}) = \det(\mathbf{A})\mathbf{A}^{-1}$ for invertible $\mathbf{A}$, and (2) $\det(\mathbf{A} + \mathbf{u}\mathbf{v}^\top) = \det(\mathbf{A}) + \mathbf{v}^\top\operatorname{adj}(\mathbf{A})\mathbf{u}$ [see Fact 2.14.2 in Ber11]. Note that from the definition of an adjugate matrix it immediately follows that if $\mathbf{A}$ is determinant preserving then adjugate commutes with expectation for this matrix:

$$\begin{aligned}
\mathbb{E}\big[\big(\operatorname{adj}(\mathbf{A})\big)_{i,j}\big] &= \mathbb{E}\big[(-1)^{i+j}\det(\mathbf{A}_{[d]\setminus\{j\},[d]\setminus\{i\}})\big] \\
&= (-1)^{i+j}\det\big(\mathbb{E}[\mathbf{A}_{[d]\setminus\{j\},[d]\setminus\{i\}}]\big) \qquad (3.3)\\
&= \big(\operatorname{adj}(\mathbb{E}[\mathbf{A}])\big)_{i,j}. \qquad (3.4)
\end{aligned}$$

The adjugate is useful in our analysis because it connects the determinant and the inverse via the formula $\operatorname{adj}(\mathbf{A}) = \det(\mathbf{A})\mathbf{A}^{-1}$, which holds for any invertible $\mathbf{A}$.

To construct more complex examples, we show that determinant preserving random matrices are closed under addition and multiplication. The proof of this result is an extension of an existing argument, given by [DM19] in the proof of Lemma 7, for computing the expected determinant of the sum of rank-1 random matrices.

Lemma 3.5 (Closure properties) *If* $\mathbf{A}$ *and* $\mathbf{B}$ *are independent and determinant preserving, then:*

1. $\mathbf{A} + \mathbf{B}$ *is determinant preserving,*
2. $\mathbf{AB}$ *is determinant preserving.*

Proof of Lemma 3.5 First, we show that $\mathbf{A} + \mathbf{u}\mathbf{v}^\top$ is d.p. for any fixed $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$. Below, we use the identity for a rank one update of a determinant: $\det(\mathbf{A} + \mathbf{u}\mathbf{v}^\top) = \det(\mathbf{A}) + \mathbf{v}^\top \mathrm{adj}(\mathbf{A})\mathbf{u}$. It follows that for any $\mathcal{I}$ and $\mathcal{J}$ of the same size,

$$\begin{aligned}
\mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{u}_\mathcal{I}\mathbf{v}_\mathcal{J}^\top)\big] &= \mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}}) + \mathbf{v}_\mathcal{J}^\top \operatorname{adj}(\mathbf{A}_{\mathcal{I},\mathcal{J}})\mathbf{u}_\mathcal{I}\big] \\
&\overset{(*)}{=} \det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}}]\big) + \mathbf{v}_\mathcal{J}^\top \operatorname{adj}\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}}]\big)\mathbf{u}_\mathcal{I} \\
&= \det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{u}_\mathcal{I}\mathbf{v}_\mathcal{J}^\top]\big),
\end{aligned}$$

where $(*)$ used (3.4), i.e., the fact that for d.p. matrices, adjugate commutes with expectation. Crucially, through the definition of an adjugate this step implicitly relies on the assumption that all the square submatrices of $\mathbf{A}_{\mathcal{I},\mathcal{J}}$ are also determinant preserving. Iterating this, we get that $\mathbf{A} + \mathbf{Z}$ is d.p. for any fixed $\mathbf{Z}$. We now show the same for $\mathbf{A} + \mathbf{B}$:

$$\begin{aligned}
\mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{B}_{\mathcal{I},\mathcal{J}})\big] &= \mathbb{E}\Big[\mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{B}_{\mathcal{I},\mathcal{J}}) \mid \mathbf{B}\big]\Big] \\
&\overset{(*)}{=} \mathbb{E}\Big[\det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}}] + \mathbf{B}_{\mathcal{I},\mathcal{J}}\big)\Big] \\
&= \det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{B}_{\mathcal{I},\mathcal{J}}]\big),
\end{aligned}$$

where $(*)$ uses the fact that after conditioning on $\mathbf{B}$ we can treat it as a fixed matrix. Next, we show that $\mathbf{AB}$ is determinant preserving via the Cauchy-Binet formula:

$$\begin{aligned}
\mathbb{E}\big[\det\big((\mathbf{AB})_{\mathcal{I},\mathcal{J}}\big)\big] &= \mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},*}\mathbf{B}_{*,\mathcal{J}})\big] \\
&= \mathbb{E}\bigg[\sum_{S\colon |S|=|\mathcal{I}|} \det\big(\mathbf{A}_{\mathcal{I},S}\big)\det\big(\mathbf{B}_{S,\mathcal{J}}\big)\bigg] \\
&= \sum_{S\colon |S|=|\mathcal{I}|} \det\big(\mathbb{E}[\mathbf{A}]_{\mathcal{I},S}\big)\det\big(\mathbb{E}[\mathbf{B}]_{S,\mathcal{J}}\big) \\
&= \det\big(\mathbb{E}[\mathbf{A}]_{\mathcal{I},*}\,\mathbb{E}[\mathbf{B}]_{*,\mathcal{J}}\big) \\
&= \det\big(\mathbb{E}[\mathbf{AB}]_{\mathcal{I},\mathcal{J}}\big),
\end{aligned}$$

where recall that $\mathbf{A}_{\mathcal{I},*}$ denotes the submatrix of $\mathbf{A}$ consisting of its (entire) rows indexed by $\mathcal{I}$. ■

Next, we introduce another important class of d.p. matrices: a sum of i.i.d. rank-1 random matrices with the number of i.i.d. samples being a Poisson random variable. Our use of the Poisson distribution is crucial for the below result to hold. It is an extension of an expectation formula given by [Der19] for sampling from discrete distributions.

Lemma 3.6 *If K is a Poisson random variable and $\mathbf{A}, \mathbf{B}$ are random $K \times d$ matrices whose rows are sampled as an i.i.d. sequence of joint pairs of random vectors, then $\mathbf{A}^\top\mathbf{B}$ is d.p., and so:*

$$\mathbb{E}\big[\det(\mathbf{A}^\top\mathbf{B})\big] = \det\big(\mathbb{E}[\mathbf{A}^\top\mathbf{B}]\big).$$

To prove Lemma 3.6, we will use the following lemma, many variants of which appeared in the literature [e.g., Vaa65]. We use the one given by [DWH19a].

Lemma 3.7 ([DWH19a]) *If the rows of random $k \times d$ matrices $\mathbf{A}, \mathbf{B}$ are sampled as an i.i.d. sequence of $k \geq d$ pairs of joint random vectors, then*

$$k^d\,\mathbb{E}\big[\det(\mathbf{A}^\top\mathbf{B})\big] = k^{\underline{d}}\det\big(\mathbb{E}[\mathbf{A}^\top\mathbf{B}]\big). \tag{3.5}$$

Here, we use the following standard shorthand: $k^{\underline{d}} = \frac{k!}{(k-d)!} = k\,(k-1)\cdots(k-d+1)$. Note that the above result almost looks like we are claiming that the matrix $\mathbf{A}^\top\mathbf{B}$ is d.p., but in fact it is not because $k^d \neq k^{\underline{d}}$. The difference in those factors is precisely what we are going to correct with the Poisson random variable. We now present the proof of Lemma 3.6.

Proof of Lemma 3.6 Without loss of generality, it suffices to check Definition 3.4 with both $\mathcal{I}$ and $\mathcal{J}$ equal $[d]$. We first expand the expectation by conditioning on the value of K and letting $\gamma = \mathbb{E}[K]$:

$$\begin{aligned}
\mathbb{E}\big[\det(\mathbf{A}^\top\mathbf{B})\big] &= \sum_{k=0}^{\infty}\mathbb{E}\big[\det(\mathbf{A}^\top\mathbf{B}) \mid K=k\big]\,\Pr(K=k)\\
\text{(Lemma 3.7)}\quad &= \sum_{k=d}^{\infty}\frac{k!k^{-d}}{(k-d)!}\det\big(\mathbb{E}[\mathbf{A}^\top\mathbf{B} \mid K=k]\big)\frac{\gamma^k\mathrm{e}^{-\gamma}}{k!}\\
&= \sum_{k=d}^{\infty}\Big(\frac{\gamma}{k}\Big)^d\det\big(\mathbb{E}[\mathbf{A}^\top\mathbf{B} \mid K=k]\big)\frac{\gamma^{k-d}\mathrm{e}^{-\gamma}}{(k-d)!}.
\end{aligned}$$

Note that $\frac{\gamma}{k}\,\mathbb{E}[\mathbf{A}^\top\mathbf{B} \mid K=k] = \mathbb{E}[\mathbf{A}^\top\mathbf{B}]$, which is independent of k. Thus we can rewrite the above expression as:

$$\det\big(\mathbb{E}[\mathbf{A}^\top\mathbf{B}]\big)\sum_{k=d}^{\infty}\frac{\gamma^{k-d}\mathrm{e}^{-\gamma}}{(k-d)!} = \det\big(\mathbb{E}[\mathbf{A}^\top\mathbf{B}]\big)\sum_{k=0}^{\infty}\frac{\gamma^k\mathrm{e}^{-\gamma}}{k!} = \det\big(\mathbb{E}[\mathbf{A}^\top\mathbf{B}]\big),$$

which concludes the proof. ■

Finally, we show the expectation formula needed for obtaining the normalization constant of the under-determined surrogate design, given in (3.1). The below result is more general than the normalization constant requires, because it allows the matrices $\mathbf{A}$ and $\mathbf{B}$ to be different (the constant is obtained by setting $\mathbf{A} = \mathbf{B} = \mathbf{X} \sim \mu^K$). In fact, we use this more general statement to show Theorems 3.1 and 3.2. The proof uses Lemmas 3.5 and 3.6.

Lemma 3.8 *If K is a Poisson random variable and $\mathbf{A}$, $\mathbf{B}$ are random $K \times d$ matrices whose rows are sampled as an i.i.d. sequence of joint pairs of random vectors, then*

$$\mathbb{E}\big[\det(\mathbf{A}\mathbf{B}^\top)\big] = \mathrm{e}^{-\mathbb{E}[K]} \det\big(\mathbf{I} + \mathbb{E}[\mathbf{B}^\top \mathbf{A}]\big).$$

To prove Lemma 3.8, we use the following standard determinantal formula which is used to derive the normalization constant of a discrete determinantal point process.

Lemma 3.9 ([KT12]) *For any $k \times d$ matrices $\mathbf{A}, \mathbf{B}$ we have*

$$\det(\mathbf{I} + \mathbf{A}\mathbf{B}^\top) = \sum_{S \subseteq [k]} \det(\mathbf{A}_{S,*}\mathbf{B}_{S,*}^\top).$$

Proof of Lemma 3.8 By Lemma 3.6, the matrix $\mathbf{B}^\top\mathbf{A}$ is determinant preserving. Applying Lemma 3.5 we conclude that $\mathbf{I} + \mathbf{B}^\top\mathbf{A}$ is also d.p., so

$$\det\big(\mathbf{I} + \mathbb{E}[\mathbf{B}^\top\mathbf{A}]\big) = \mathbb{E}\big[\det(\mathbf{I} + \mathbf{B}^\top\mathbf{A})\big] = \mathbb{E}\big[\det(\mathbf{I} + \mathbf{A}\mathbf{B}^\top)\big],$$

where the second equality is known as Sylvester's Theorem. We rewrite the expectation of $\det(\mathbf{I} + \mathbf{A}\mathbf{B}^\top)$ by applying Lemma 3.9. Letting $\gamma = \mathbb{E}[K]$, we obtain:

$$\begin{aligned}
\mathbb{E}\big[\det(\mathbf{I} + \mathbf{A}\mathbf{B}^\top)\big] &= \mathbb{E}\bigg[\sum_{S \subseteq [K]} \mathbb{E}\big[\det(\mathbf{A}_{S,*}\mathbf{B}_{S,*}^\top) \mid K\big]\bigg] \\
&\overset{(*)}{=} \sum_{k=0}^{\infty} \frac{\gamma^k \mathrm{e}^{-\gamma}}{k!} \sum_{i=0}^{k} \binom{k}{i} \mathbb{E}\big[\det(\mathbf{A}\mathbf{B}^\top) \mid K = i\big] \\
&= \sum_{i=0}^{\infty} \mathbb{E}\big[\det(\mathbf{A}\mathbf{B}^\top) \mid K = i\big] \sum_{k \geq i}^{\infty} \binom{k}{i} \frac{\gamma^k \mathrm{e}^{-\gamma}}{k!} \\
&= \sum_{i=0}^{\infty} \frac{\gamma^i \mathrm{e}^{-\gamma}}{i!} \mathbb{E}\big[\det(\mathbf{A}\mathbf{B}^\top) \mid K = i\big] \sum_{k \geq i}^{\infty} \frac{\gamma^{k-i}}{(k-i)!} = \mathbb{E}\big[\det(\mathbf{A}\mathbf{B}^\top)\big] \cdot \mathrm{e}^{\gamma},
\end{aligned}$$

where $(*)$ follows from the exchangeability of the rows of $\mathbf{A}$ and $\mathbf{B}$, which implies that the distribution of $\mathbf{A}_{S,*}\mathbf{B}_{S,*}^\top$ is the same for all subsets S of a fixed size k. ■

3.5 Proof of Theorem 3.1

In this section we use Z_μ^n to denote the normalization constant that appears in (3.1) when computing an expectation for surrogate design S_μ^n. We first prove Lemma 3.4.

Lemma 3.10 (restated Lemma 3.4) *If $\bar{\mathbf{X}} \sim S_\mu^n$ for $n < d$, then we have*

$$\mathbb{E}\big[\mathrm{tr}\big((\bar{\mathbf{X}}^\top\bar{\mathbf{X}})^\dagger\big)\big] = \gamma_n\big(1 - \det\big((\tfrac{1}{\gamma_n}\mathbf{I} + \mathbf{\Sigma}_\mu)^{-1}\mathbf{\Sigma}_\mu\big)\big).$$

Proof Let $\mathbf{X} \sim \mu^K$ for $K \sim \mathrm{Poisson}(\gamma_n)$. Note that if $\det(\mathbf{X}\mathbf{X}^\top) > 0$ then using the fact that $\det(\mathbf{A})\mathbf{A}^{-1} = \mathrm{adj}(\mathbf{A})$ for any invertible matrix $\mathbf{A}$, we can write:

$$\begin{aligned}
\det(\mathbf{X}\mathbf{X}^\top)\mathrm{tr}\big((\mathbf{X}^\top\mathbf{X})^\dagger\big) &= \det(\mathbf{X}\mathbf{X}^\top)\mathrm{tr}\big((\mathbf{X}\mathbf{X}^\top)^{-1}\big) \\
&= \mathrm{tr}(\mathrm{adj}(\mathbf{X}\mathbf{X}^\top)) \\
&= \sum_{i=1}^K \det(\mathbf{X}_{-i}\mathbf{X}_{-i}^\top),
\end{aligned}$$

where $\mathbf{X}_{-i}$ is a shorthand for $\mathbf{X}_{[K]\setminus\{i\},*}$. Assumption 3.2 ensures that $\Pr\big\{\det(\mathbf{X}\mathbf{X}^\top) > 0\big\} = 1$, which allows us to write:

$$\begin{aligned}
Z_\mu^n \cdot \mathbb{E}\big[\mathrm{tr}\big((\bar{\mathbf{X}}^\top\bar{\mathbf{X}})^\dagger\big)\big] &= \mathbb{E}\bigg[\sum_{i=1}^K \det(\mathbf{X}_{-i}\mathbf{X}_{-i}^\top) \;\Big|\; \det(\mathbf{X}\mathbf{X}^\top) > 0\bigg] \cdot \overbrace{\Pr\big\{\det(\mathbf{X}\mathbf{X}^\top) > 0\big\}}^{1} \\
&= \sum_{k=0}^d \frac{\gamma_n^k \mathrm{e}^{-\gamma_n}}{k!}\mathbb{E}\Big[\sum_{i=1}^k \det(\mathbf{X}_{-i}\mathbf{X}_{-i}^\top) \;\Big|\; K = k\Big] \\
&= \sum_{k=0}^d \frac{\gamma_n^k \mathrm{e}^{-\gamma_n}}{k!}\, k\; \mathbb{E}\big[\det(\mathbf{X}\mathbf{X}^\top) \mid K = k-1\big] \\
&= \gamma_n \sum_{k=0}^{d-1} \frac{\gamma_n^k \mathrm{e}^{-\gamma_n}}{k!}\mathbb{E}\big[\det(\mathbf{X}\mathbf{X}^\top) \mid K = k\big] \\
&= \gamma_n\Big(\mathbb{E}\big[\det(\mathbf{X}\mathbf{X}^\top)\big] \;-\; \frac{\gamma_n^d \mathrm{e}^{-\gamma_n}}{d!}\mathbb{E}\big[\det(\mathbf{X})^2 \mid K = d\big]\Big) \\
&\overset{(*)}{=} \gamma_n\big(\mathrm{e}^{-\gamma_n}\det(\mathbf{I} + \gamma_n\mathbf{\Sigma}_\mu) - \mathrm{e}^{-\gamma_n}\det(\gamma_n\mathbf{\Sigma}_\mu)\big),
\end{aligned}$$

where $(*)$ uses Lemma 3.8 for the first term and Lemma 3.7 for the second term. We obtain the desired result by dividing both sides by $Z_\mu^n = \mathrm{e}^{-\gamma_n}\det(\mathbf{I} + \gamma_n\mathbf{\Sigma}_\mu)$. ■

In the over-determined regime, a more general matrix expectation formula can be shown (omitting the trace). The following result is related to an expectation formula derived by [DWH19b], however they use a slightly different determinantal design so the results are incomparable.

Lemma 3.11 *If $\bar{\mathbf{X}} \sim S_\mu^n$ and $n > d$, then we have*

$$\mathbb{E}\big[(\bar{\mathbf{X}}^\top \bar{\mathbf{X}})^\dagger\big] = \boldsymbol{\Sigma}_\mu^{-1} \cdot \frac{1 - \mathrm{e}^{-\gamma_n}}{\gamma_n}.$$

Proof Let $\mathbf{X} \sim \mu^K$ for $K \sim \mathrm{Poisson}(\gamma_n)$. Assumption 3.2 implies that for $K \neq d-1$ we have

$$\det(\mathbf{X}^\top \mathbf{X})(\mathbf{X}^\top \mathbf{X})^\dagger = \mathrm{adj}(\mathbf{X}^\top \mathbf{X}), \tag{3.6}$$

however when $k = d-1$ then (3.6) does not hold because $\det(\mathbf{X}^\top \mathbf{X}) = 0$ while $\mathrm{adj}(\mathbf{X}^\top \mathbf{X})$ may be non-zero. It follows that:

$$\begin{aligned}
Z_\mu^n \cdot \mathbb{E}\big[(\bar{\mathbf{X}}^\top \bar{\mathbf{X}})^\dagger\big] &= \mathbb{E}\big[\det(\mathbf{X}^\top \mathbf{X})(\mathbf{X}^\top \mathbf{X})^\dagger\big] \\
&= \mathbb{E}\big[\mathrm{adj}(\mathbf{X}^\top \mathbf{X})\big] - \frac{\gamma_n^{d-1}\mathrm{e}^{-\gamma_n}}{(d-1)!}\mathbb{E}\big[\mathrm{adj}(\mathbf{X}^\top \mathbf{X}) \mid K = d-1\big] \\
&\overset{(*)}{=} \mathrm{adj}\big(\mathbb{E}[\mathbf{X}^\top \mathbf{X}]\big) - \frac{\gamma_n^{d-1}\mathrm{e}^{-\gamma_n}}{(d-1)^{d-1}}\,\mathrm{adj}\big(\mathbb{E}[\mathbf{X}^\top \mathbf{X} \mid K = d-1]\big) \\
&= \mathrm{adj}(\gamma_n \boldsymbol{\Sigma}_\mu) - \mathrm{e}^{-\gamma_n}\,\mathrm{adj}(\gamma_n \boldsymbol{\Sigma}_\mu) \\
&= \det(\gamma_n \boldsymbol{\Sigma}_\mu)\,(\gamma_n \boldsymbol{\Sigma}_\mu)^{-1}(1 - \mathrm{e}^{-\gamma_n}) \\
&= \det(\gamma_n \boldsymbol{\Sigma}_\mu)\,\boldsymbol{\Sigma}_\mu^{-1} \cdot \frac{1 - \mathrm{e}^{-\gamma_n}}{\gamma_n},
\end{aligned}$$

where the first term in $(*)$ follows from Lemma 3.8 and (3.4), whereas the second term comes from Lemma 2.3 of [DWH19b]. Dividing both sides by $Z_\mu^n = \det(\gamma_n \boldsymbol{\Sigma}_\mu)$ completes the proof. ■

Applying the closed form expressions from Lemmas 3.3, 3.4 and 3.11, we derive the formula for the MSE and prove Theorem 3.1 (we defer the proof of Lemma 3.3 to Section 3.6).

Proof of Theorem 3.1 First, assume that $n < d$, in which case we have $\gamma_n = \frac{1}{\lambda_n}$ and moreover

$$\begin{aligned}
n &= \mathrm{tr}\big(\boldsymbol{\Sigma}_\mu(\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\big) \\
&= \mathrm{tr}\big((\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I} - \lambda_n \mathbf{I})(\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\big) \\
&= d - \lambda_n \mathrm{tr}\big((\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\big),
\end{aligned}$$

so we can write λ_n as $(d-n)/\mathrm{tr}((\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1})$. From this and Lemmas 3.3 and 3.10, we obtain the desired expression, where recall that $\alpha_n = \det\big(\boldsymbol{\Sigma}_\mu(\boldsymbol{\Sigma}_\mu + \frac{1}{\gamma_n})^{-1}\big)$:

$$\begin{aligned}
\mathrm{MSE}\big[\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}}\big] &= \sigma^2\,\gamma_n(1 - \alpha_n) + \tfrac{1}{\gamma_n}\,\mathbf{w}^{*\top}(\boldsymbol{\Sigma}_\mu + \tfrac{1}{\gamma_n}\mathbf{I})^{-1}\mathbf{w}^* \\
&\overset{(a)}{=} \sigma^2\,\frac{1 - \alpha_n}{\lambda_n} + \lambda_n\,\mathbf{w}^{*\top}(\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\mathbf{w}^* \\
&\overset{(b)}{=} \sigma^2 \mathrm{tr}\big((\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\big)\frac{1 - \alpha_n}{d - n} + (d-n)\frac{\mathbf{w}^{*\top}(\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\mathbf{w}^*}{\mathrm{tr}\big((\boldsymbol{\Sigma}_\mu + \lambda_n \mathbf{I})^{-1}\big)}.
\end{aligned}$$

While the expression given after (a) is simpler than the one after (b), the latter better illustrates how the MSE depends on the sample size n and the dimension d. Now, assume that $n > d$. In this case, we have $\gamma_n = n - d$ and apply Lemma 3.11:

$$\mathrm{MSE}\big[\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}}\big] = \sigma^2 \operatorname{tr}(\boldsymbol{\Sigma}_\mu^{-1}) \, \frac{1 - \mathrm{e}^{-\gamma_n}}{\gamma_n} = \sigma^2 \operatorname{tr}(\boldsymbol{\Sigma}_\mu^{-1}) \, \frac{1 - \beta_n}{n - d}.$$

The case of $n = d$ was shown in Theorem 2.12 of [DWH19b]. This concludes the proof. ■

3.6 Proof of Theorem 3.2

As in the previous section, we use Z_μ^n to denote the normalization constant that appears in (3.1) when computing an expectation for surrogate design S_μ^n. Recall that our goal is to compute the expected value of $\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}}$ under the surrogate design S_μ^n. Similarly as for Theorem 3.1, the case of $n = d$ was shown in Theorem 2.10 of [DWH19b]. We break the rest down into the under-determined case ($n < d$) and the over-determined case ($n > d$), starting with the former. Recall that we do *not* require any modeling assumptions on the responses.

Lemma 3.12 *If $\bar{\mathbf{X}} \sim S_\mu^n$ and $n < d$, then for any $y(\cdot)$ such that $\mathbb{E}_{\mu,y}[y(\mathbf{x})\,\mathbf{x}]$ is well-defined, denoting $\bar{y}_i$ as $y(\bar{\mathbf{x}}_i)$, we have*

$$\mathbb{E}\big[\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}}\big] = \big(\boldsymbol{\Sigma}_\mu + \tfrac{1}{\gamma_n}\mathbf{I}\big)^{-1} \mathbb{E}_{\mu,y}[y(\mathbf{x})\,\mathbf{x}].$$

Proof Let $\mathbf{X} \sim \mu^K$ for $K \sim \mathrm{Poisson}(\gamma_n)$ and denote $y(\mathbf{x}_i)$ as y_i. Note that when $\det(\mathbf{X}\mathbf{X}^\top) > 0$, then the jth entry of $\mathbf{X}^\dagger \mathbf{y}$ equals $\mathbf{f}_j^\top (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{y}$, where $\mathbf{f}_j$ is the jth column of $\mathbf{X}$, so:

$$\begin{aligned}
\det(\mathbf{X}\mathbf{X}^\top)\,(\mathbf{X}^\dagger \mathbf{y})_j &= \det(\mathbf{X}\mathbf{X}^\top)\,\mathbf{f}_j^\top (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{y} \\
&= \det(\mathbf{X}\mathbf{X}^\top + \mathbf{y}\mathbf{f}_j^\top) - \det(\mathbf{X}\mathbf{X}^\top).
\end{aligned}$$

If $\det(\mathbf{X}\mathbf{X}^\top) = 0$, then also $\det(\mathbf{X}\mathbf{X}^\top + \mathbf{y}\mathbf{f}_j^\top) = 0$, so we can write:

$$\begin{aligned}
Z_\mu^n \cdot \mathbb{E}\big[(\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}})_j\big] &= \mathbb{E}\big[\det(\mathbf{X}\mathbf{X}^\top)(\mathbf{X}^\dagger \mathbf{y})_j\big] \\
&= \mathbb{E}\big[\det(\mathbf{X}\mathbf{X}^\top + \mathbf{y}\mathbf{f}_j^\top) - \det(\mathbf{X}\mathbf{X}^\top)\big] \\
&= \mathbb{E}\big[\det\big([\mathbf{X}, \mathbf{y}][\mathbf{X}, \mathbf{f}_j]^\top\big)\big] - \mathbb{E}\big[\det(\mathbf{X}\mathbf{X}^\top)\big] \\
&\overset{(a)}{=} \mathrm{e}^{-\gamma_n} \det\left(\mathbf{I} + \gamma_n \, \mathbb{E}_{\mu,y}\left[\begin{pmatrix} \mathbf{x}\mathbf{x}^\top & \mathbf{x}\, y(\mathbf{x}) \\ x_j\, \mathbf{x}^\top & x_j\, y(\mathbf{x}) \end{pmatrix}\right]\right) - \mathrm{e}^{-\gamma_n} \det(\mathbf{I} + \gamma_n \boldsymbol{\Sigma}_\mu) \\
&\overset{(b)}{=} \mathrm{e}^{-\gamma_n} \det(\mathbf{I} + \gamma_n \boldsymbol{\Sigma}_\mu) \\
&\qquad \times \Big(\mathbb{E}_{\mu,y}\big[\gamma_n x_j\, y(\mathbf{x})\big] - \mathbb{E}_\mu\big[\gamma_n x_j\, \mathbf{x}^\top\big](\mathbf{I} + \gamma_n \boldsymbol{\Sigma}_\mu)^{-1} \mathbb{E}_{\mu,y}\big[\gamma_n \mathbf{x}\, y(\mathbf{x})\big]\Big),
\end{aligned}$$

where (a) uses Lemma 3.8 twice, with the first application involving two different matrices $\mathbf{A} = [\mathbf{X}, \mathbf{y}]$ and $\mathbf{B} = [\mathbf{X}, \mathbf{f}_j]$, whereas (b) is a standard determinantal identity [see Fact 2.14.2 in Ber11]. Dividing both sides by Z_μ^n and letting $\mathbf{v}_{\mu,y} = \mathbb{E}_{\mu,y}[y(\mathbf{x})\,\mathbf{x}]$, we obtain that:

$$\begin{aligned}\mathbb{E}\big[\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}}\big] &= \gamma_n \mathbf{v}_{\mu,y} - \gamma_n^2 \boldsymbol{\Sigma}_\mu (\mathbf{I} + \gamma_n \boldsymbol{\Sigma}_\mu)^{-1} \mathbf{v}_{\mu,y} \\ &= \gamma_n \big(\mathbf{I} - \gamma_n \boldsymbol{\Sigma}_\mu (\mathbf{I} + \gamma_n \boldsymbol{\Sigma}_\mu)^{-1}\big) \mathbf{v}_{\mu,y} = \gamma_n (\mathbf{I} + \gamma_n \boldsymbol{\Sigma}_\mu)^{-1} \mathbf{v}_{\mu,y},\end{aligned}$$

which completes the proof. ■

We return to Lemma 3.3, regarding the expected orthogonal projection onto the complement of the row-span of $\bar{\mathbf{X}}$, i.e., $\mathbb{E}[\mathbf{I} - \bar{\mathbf{X}}^\dagger \bar{\mathbf{X}}]$, which follows as a corollary of Lemma 3.12.

Proof of Lemma 3.3 We let $y(\mathbf{x}) = x_j$ where $j \in [d]$ and apply Lemma 3.12 for each j, obtaining:

$$\mathbf{I} - \mathbb{E}\big[\bar{\mathbf{X}}^\dagger \bar{\mathbf{X}}\big] = \mathbf{I} - (\boldsymbol{\Sigma}_\mu + \tfrac{1}{\gamma_n}\mathbf{I})^{-1} \boldsymbol{\Sigma}_\mu,$$

from which the result follows by simple algebraic manipulation. ■

We move on to the over-determined case, where the ridge regularization of adding the identity to $\boldsymbol{\Sigma}_\mu$ vanishes. Recall that we assume throughout the paper that $\boldsymbol{\Sigma}_\mu$ is invertible.

Lemma 3.13 *If $\bar{\mathbf{X}} \sim S_\mu^n$ and $n > d$, then for any real-valued random function $y(\cdot)$ such that $\mathbb{E}_{\mu,y}[y(\mathbf{x})\,\mathbf{x}]$ is well-defined, denoting $\bar{y}_i$ as $y(\bar{\mathbf{x}}_i)$, we have*

$$\mathbb{E}\big[\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}}\big] = \boldsymbol{\Sigma}_\mu^{-1} \mathbb{E}_{\mu,y}\big[y(\mathbf{x})\,\mathbf{x}\big].$$

Proof Let $\mathbf{X} \sim \mu^K$ for $K \sim \text{Poisson}(\gamma_n)$ and denote $y_i = y(\mathbf{x}_i)$. Similarly as in the proof of Lemma 3.12, we note that when $\det(\mathbf{X}^\top \mathbf{X}) > 0$, then the jth entry of $\mathbf{X}^\dagger \mathbf{y}$ equals $\mathbf{e}_j^\top (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$, where $\mathbf{e}_j$ is the jth standard basis vector, so:

$$\det(\mathbf{X}^\top \mathbf{X})\,(\mathbf{X}^\dagger \mathbf{y})_j = \det(\mathbf{X}^\top \mathbf{X})\,\mathbf{e}_j^\top (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y} = \det(\mathbf{X}^\top \mathbf{X} + \mathbf{X}^\top \mathbf{y} \mathbf{e}_j^\top) - \det(\mathbf{X}^\top \mathbf{X}).$$

If $\det(\mathbf{X}^\top \mathbf{X}) = 0$, then also $\det(\mathbf{X}^\top \mathbf{X} + \mathbf{X}^\top \mathbf{y} \mathbf{e}_j^\top) = 0$. We proceed to compute the expectation:

$$\begin{aligned}Z_\mu^n \cdot \mathbb{E}\big[(\bar{\mathbf{X}}^\dagger \bar{\mathbf{y}})_j\big] &= \mathbb{E}\big[\det(\mathbf{X}^\top \mathbf{X})(\mathbf{X}^\dagger \mathbf{y})_j\big] \\ &= \mathbb{E}\big[\det(\mathbf{X}^\top \mathbf{X} + \mathbf{X}^\top \mathbf{y} \mathbf{e}_j^\top) - \det(\mathbf{X}^\top \mathbf{X})\big] \\ &= \mathbb{E}\big[\det\big(\mathbf{X}^\top (\mathbf{X} + \mathbf{y} \mathbf{e}_j^\top)\big)\big] - \mathbb{E}\big[\det(\mathbf{X}^\top \mathbf{X})\big] \\ &\overset{(*)}{=} \det\Big(\gamma_n\, \mathbb{E}_{\mu,y}\big[\mathbf{x}(\mathbf{x} + y(\mathbf{x})\mathbf{e}_j)^\top\big]\Big) - \det(\gamma_n \boldsymbol{\Sigma}_\mu) \\ &= \det\big(\gamma_n \boldsymbol{\Sigma}_\mu + \gamma_n \mathbb{E}_{\mu,y}[\mathbf{x}\, y(\mathbf{x})] \mathbf{e}_j^\top\big) - \det(\gamma_n \boldsymbol{\Sigma}_\mu) \\ &= \det(\gamma_n \boldsymbol{\Sigma}_\mu) \cdot \gamma_n \mathbf{e}_j^\top (\gamma_n \boldsymbol{\Sigma}_\mu)^{-1} \mathbb{E}_{\mu,y}\big[y(\mathbf{x})\,\mathbf{x}\big],\end{aligned}$$

where $(*)$ uses Lemma 3.6 twice (the first time, with $\mathbf{A} = \mathbf{X}$ and $\mathbf{B} = \mathbf{X} + \mathbf{y}\mathbf{e}_j^\top$). Dividing both sides by $Z_\mu^n = \det(\gamma_n \boldsymbol{\Sigma}_\mu)$ concludes the proof. ■

We combine Lemmas 3.12 and 3.13 to obtain the proof of Theorem 3.2.

Proof of Theorem 3.2 The case of $n = d$ follows directly from Theorem 2.10 of [DWH19a]. Assume that $n < d$. Then we have $\gamma_n = \frac{1}{\lambda_n}$, so the result follows from Lemma 3.12. If $n > d$, then the result follows from Lemma 3.13. ■

3.7 Proof of Theorem 3.3

The proof of Theorem 3.3 follows the standard decomposition of MSE in Equation 3.2, and in the process, establishes consistency of the variance and bias terms independently. To this end, we introduce the following two useful lemmas that capture the limiting behavior of the variance and bias terms, respectively.

Lemma 3.14 *Under the setting of Theorem 3.3, we have, as $n, d \to \infty$ with $n/d \to \bar{c} \in (0, \infty) \setminus \{1\}$ that*

$$\begin{cases} \mathbb{E}\big[\mathrm{tr}((\mathbf{X}^\top \mathbf{X})^\dagger)\big] - (1 - \alpha_n)\lambda_n^{-1} \to 0, & \text{for } \bar{c} < 1, \\ \mathbb{E}\big[\mathrm{tr}((\mathbf{X}^\top \mathbf{X})^\dagger)\big] - \frac{1-\beta_n}{n-d} \cdot \mathrm{tr}\boldsymbol{\Sigma}^{-1} \to 0, & \text{for } \bar{c} > 1 \end{cases} \tag{3.7}$$

where $\lambda_n \geq 0$ is the unique solution to $n = \mathrm{tr}(\boldsymbol{\Sigma}(\boldsymbol{\Sigma} + \lambda_n \mathbf{I})^{-1})$, $\alpha_n = \det(\boldsymbol{\Sigma}(\boldsymbol{\Sigma} + \lambda_n \mathbf{I})^{-1})$, and $\beta_n = e^{d-n}$.

The second term in the MSE derivation (3.2), $\mathbb{E}[\mathbf{I} - \mathbf{X}^\dagger \mathbf{X}]$, involves the expectation of a projection onto the orthogonal complement of a sub-Gaussian general position sample $\mathbf{X}$. This term is zero when $n > d$, and for $n < d$ we prove in section 3.7 that the surrogate design's bias $\mathcal{B}(\boldsymbol{\Sigma}, n)$ provides an asymptotically consistent approximation to all of the eigenvectors and eigenvalues:

Lemma 3.15 *Under the setting of Theorem 3.3, for $\mathbf{w} \in \mathbb{R}^d$ of bounded Euclidean norm (i.e., $\|\mathbf{w}\| \leq C'$ for all d), we have, as $n, d \to \infty$ with $n/d \to \bar{c} \in (0, 1)$ that*

$$\mathbf{w}^\top \mathbb{E}[\mathbf{I} - \mathbf{X}^\dagger \mathbf{X}]\mathbf{w} - \lambda_n \mathbf{w}^\top (\boldsymbol{\Sigma} + \lambda_n \mathbf{I})^{-1} \mathbf{w} \to 0 \tag{3.8}$$

while $\mathbf{I} - \mathbf{X}^\dagger \mathbf{X} = 0$ for $\bar{c} > 1$.

Proof of lemma 3.14

The $\bar{c} \in (0, 1)$ case

For $n < d$, we first establish (1) $\liminf_n \lambda_n > 0$ and (2) $\alpha_n \to 0$. To prove (1), by hypothesis $\boldsymbol{\Sigma} \succeq c\mathbf{I}$ for all d. Since $\frac{n}{d} < 1$, we have (by definition of λ_n) for some $\delta > 0$

$$1 - \delta > \frac{n}{d} = \frac{1}{d}\mathrm{tr}(\boldsymbol{\Sigma}(\boldsymbol{\Sigma} + \lambda_n \mathbf{I})^{-1}) > \frac{c}{c + \lambda_n}$$

Rearranging, we have $\lambda_n > \frac{\delta c}{1-\delta} > 0$. For (2), let $(\tau_i)_{i\in[d]}$ denote the eigenvalues of $\mathbf{\Sigma}$. Since $1 - x \leq e^{-x}$ and $C\mathbf{I} \succeq \mathbf{\Sigma} \succeq c\mathbf{I}$ for all d,

$$\alpha_n = \prod_{i=1}^{d} \frac{\tau_i}{\tau_i + \lambda_n} \leq \left(\frac{C}{C+\lambda_n}\right)^d = \left(1 - \frac{\lambda_n}{C+\lambda_n}\right)^d \leq \exp\left(-d\frac{\lambda_n}{C+\lambda_n}\right)$$

and since $\lambda_n > 0$ eventually as $d \to \infty$ we have $\alpha_n \to 0$ so that $(1-\alpha_n)\lambda_n^{-1} - \lambda_n^{-1} \to 0$.

As a consequence of (2) and Slutsky's theorem, it suffices to show $\mathrm{tr}(\mathbf{X}^\top\mathbf{X})^\dagger - \lambda_n^{-1} \xrightarrow{d} 0$ as $n, d \to \infty$. To do this, we consider the limiting behavior of $\mathrm{tr}(\mathbf{X}^\top\mathbf{X})^\dagger/n = \mathrm{tr}(\mathbf{X}\mathbf{X}^\top)^\dagger/n$ as $n/d \to \bar{c} \in (0,1)$, for $\mathbf{X} = \mathbf{Z}\mathbf{\Sigma}^{\frac{1}{2}}$ with $\mathbf{Z} \in \mathbb{R}^{n\times d}$ having i.i.d. zero mean, unit variance sub-Gaussian entries, i.e., the behavior of

$$\lim_{n,d\to\infty} \lim_{z\to 0^+} \frac{1}{n}\mathrm{tr}\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1} \tag{3.9}$$

by definition of the pseudo-inverse.

The proof comes in three steps: (i) for fixed $z > 0$, consider the limiting behavior of $\delta(z) \equiv \mathrm{tr}(\mathbf{X}\mathbf{X}^\top/n + z\mathbf{I}_n)^{-1}/n$ as $n, d \to \infty$ and state

$$\lim_{n,d\to\infty} \delta(z) - m(z) \to 0 \tag{3.10}$$

almost surely for some $m(z)$ to be defined; (ii) show that both $\delta(z)$ and its derivate $\delta'(z)$ are uniformly bounded (by some quantity independent of $z > 0$) so that by Arzela-Ascoli theorem, $\delta(z)$ converges uniformly to its limit and we are allowed to take $z \to 0^+$ in (3.10) and state

$$\lim_{z\to 0^+} \lim_{n,d\to\infty} \delta(z) - \lim_{z\to 0^+} m(z) \to 0 \tag{3.11}$$

almost surely, given that the limit $\lim_{z\to 0^+} m(z) \equiv m(0)$ exists and eventually (iii) exchange the two limits in (3.11) with Moore-Osgood theorem, to reach

$$\lim_{n,d\to\infty} \lim_{z\to 0^+} \frac{1}{n}\mathrm{tr}\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1} - m(0) \to 0.$$

Step (i) follows from [SB95] that, we have, for $z > 0$ that

$$\delta(z) \equiv \frac{1}{n}\mathrm{tr}\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1} - m(z) \to 0$$

almost surely as $n, d \to \infty$, for $m(z)$ the unique positive solution to

$$m(z) = \left(z + \frac{1}{n}\mathrm{tr}\mathbf{\Sigma}(\mathbf{I} + m(z)\mathbf{\Sigma})^{-1}\right)^{-1}. \tag{3.12}$$

For the above step (ii), we use the assumption $\mathbf{\Sigma} \succeq c\mathbf{I} \succ 0$ for all d large, so that with $\mathbf{X} = \mathbf{Z}\mathbf{\Sigma}^{\frac{1}{2}}$, we have for large enough n, d that

$$\lambda_{\min}(\mathbf{X}\mathbf{X}^\top/n) \geq \lambda_{\min}(\mathbf{Z}\mathbf{Z}^\top/n)\lambda_{\min}(\mathbf{\Sigma}) \geq \frac{c}{2}(\sqrt{\bar{c}}-1)^2$$

almost surely, where we used Bai-Yin theorem [BY+93], which states that the minimum eigenvalue of $\mathbf{Z}\mathbf{Z}^\top/n$ is almost surely larger than $(\sqrt{\bar{c}}-1)^2/2$ for $n < d$ sufficiently large. Note that here the case $\bar{c} = 1$ is excluded.

Observe that

$$|\delta(z)| = \left|\frac{1}{n}\mathrm{tr}\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1}\right| \leq \frac{1}{\lambda_{\min}(\mathbf{X}\mathbf{X}^\top/n)}$$

and similarly for its derivative, so that we are allowed to take the $z \to 0^+$ limit. Note that the existence of the $\lim_{z\to 0^+} m(z)$ for $m(z)$ defined in (3.12) is well known, see for example [LP11]. Then, by Moore-Osgood theorem we finish step (iii) and by concluding that

$$\mathrm{tr}(\mathbf{X}^\top\mathbf{X})^\dagger - m(0) \to 0$$

for $m(0) = \lambda_n^{-1}$ the unique solution to $\lambda_n^{-1} = \left(\frac{1}{n}\mathrm{tr}\mathbf{\Sigma}(\mathbf{I} + \lambda_n^{-1}\mathbf{\Sigma})^{-1}\right)^{-1}$, or equivalently, to

$$n = \mathrm{tr}\mathbf{\Sigma}(\mathbf{\Sigma} + \lambda_n\mathbf{I})^{-1}$$

as desired.

The $\bar{c} \in (1, \infty)$ case

First note that as $n, d \to \infty$ with $n > d$, we have $\beta_n = e^{d-n} \to 0$ and it it suffices to show

$$\mathrm{tr}(\mathbf{X}^\top\mathbf{X})^\dagger - \frac{1}{n-d}\mathrm{tr}\mathbf{\Sigma}^{-1} \to 0$$

almost surely to conclude the proof.

In the $\bar{c} \in (1, \infty)$ case, it is more convenient to work on the following co-resolvent

$$\lim_{n,d\to\infty}\lim_{z\to 0^+}\frac{1}{n}\mathrm{tr}\left(\frac{1}{n}\mathbf{X}^\top\mathbf{X} + z\mathbf{I}_d\right)^{-1}$$

where we recall $\mathbf{X}^\top\mathbf{X} = \mathbf{\Sigma}^{\frac{1}{2}}\mathbf{Z}^\top\mathbf{Z}\mathbf{\Sigma}^{\frac{1}{2}} \in \mathbb{R}^{d\times d}$ and following the same three-step procedure as in the $\bar{c} < 1$ case above. The only difference is in step (i) we need to assess the asymptotic behavior of $\delta \equiv \mathrm{tr}(\mathbf{X}^\top\mathbf{X}/n + z\mathbf{I}_d)^{-1}/n$. This was established in [BS+98] where it was shown that, for $z > 0$ we have

$$\frac{1}{n}\mathrm{tr}(\mathbf{X}^\top\mathbf{X}/n + z\mathbf{I}_d)^{-1} - \frac{d}{n}m(z) \to 0$$

almost surely as $n, d \to \infty$, for $m(z)$ the unique solution to

$$m(z) = \frac{1}{d}\text{tr}\left(\left(1 - \frac{d}{n} - \frac{d}{n}zm(z)\right)\mathbf{\Sigma} - z\mathbf{I}_d\right)^{-1}$$

so that for $d < n$ by taking $z = 0$ we have

$$m(0) = \frac{n}{d}\frac{1}{n-d}\text{tr}\mathbf{\Sigma}^{-1}.$$

The steps (ii) and (iii) follow exactly the same line of arguments as the $\bar{c} < 1$ case and are thus omitted.

Proof of lemma 3.15

Since $\mathbf{X}^\dagger\mathbf{X} = \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top)^\dagger\mathbf{X}$, to prove lemma 3.15, we are interested in the limiting behavior of the following quadratic form

$$\lim_{n,d\to\infty}\lim_{z\to 0^+}\frac{1}{n}\mathbf{w}^\top\mathbf{X}^\top\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1}\mathbf{X}\mathbf{w}$$

for deterministic $\mathbf{w} \in \mathbb{R}^d$ of bounded Euclidean norm (i.e., $\|\mathbf{w}\| \leq C'$ as $n, d \to \infty$), as $n, d \to \infty$ with $n/d \to \bar{c} \in (0, 1)$. The limiting behavior of the above quadratic form, or more generally, bilinear form of the type $\frac{1}{n}\mathbf{w}_1^\top\mathbf{X}^\top\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1}\mathbf{X}\mathbf{w}_2$ for $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^d$ of bounded Euclidean norm are widely studied in random matrix literature, see for example [Hac+13].

For the proof of Lemma 3.15 we follow the same protocol as that of Lemma 3.14, namely: (i) we consider, for fixed $z > 0$, the limiting behavior of $\frac{1}{n}\mathbf{w}^\top\mathbf{X}^\top\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1}\mathbf{X}\mathbf{w}$. Note that

$$\begin{aligned}\delta(z) &\equiv \frac{1}{n}\mathbf{w}^\top\mathbf{X}^\top\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1}\mathbf{X}\mathbf{w} = \mathbf{w}^\top\left(\frac{1}{n}\mathbf{X}^\top\mathbf{X} + z\mathbf{I}_d\right)^{-1}\frac{1}{n}\mathbf{X}^\top\mathbf{X}\mathbf{w}\\ &= \|\mathbf{w}\|^2 - z\mathbf{w}^\top\left(\frac{1}{n}\mathbf{X}^\top\mathbf{X} + z\mathbf{I}_d\right)^{-1}\mathbf{w}\end{aligned}$$

and it remains to work on the second $z\mathbf{w}^\top\left(\frac{1}{n}\mathbf{X}^\top\mathbf{X} + z\mathbf{I}_d\right)^{-1}\mathbf{w}$ term. It follows from [Hac+13] that

$$z\mathbf{w}^\top\left(\frac{1}{n}\mathbf{X}^\top\mathbf{X} + z\mathbf{I}_d\right)^{-1}\mathbf{w} - \mathbf{w}^\top(\mathbf{I}_d + m(z)\mathbf{\Sigma})^{-1}\mathbf{w}^\top \to 0$$

almost surely as $n, d \to \infty$, where we recall $m(z)$ is the unique solution to (3.12).

We move on to step (ii), under the assumption that $c \leq \lambda_{\min}(\mathbf{\Sigma}) \leq \lambda_{\max}(\mathbf{\Sigma}) \leq C$ and $\|\mathbf{w}\| \leq C'$, we have

$$\begin{aligned}\lambda_{\max}\left(\frac{1}{n}\mathbf{X}^\top\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + z\mathbf{I}_n\right)^{-1}\mathbf{X}\right) &\leq \frac{\lambda_{\max}(\mathbf{X}\mathbf{X}^\top/n)}{\lambda_{\min}(\mathbf{X}\mathbf{X}^\top/n) + z} \leq \frac{\lambda_{\max}(\mathbf{Z}\mathbf{Z}^\top/n)\lambda_{\max}(\mathbf{\Sigma})}{\lambda_{\min}(\mathbf{Z}\mathbf{Z}^\top/n)\lambda_{\min}(\mathbf{\Sigma})}\\ &\leq 4\frac{(\sqrt{\bar{c}}+1)^2 C}{(\sqrt{\bar{c}}-1)^2 c}\end{aligned}$$

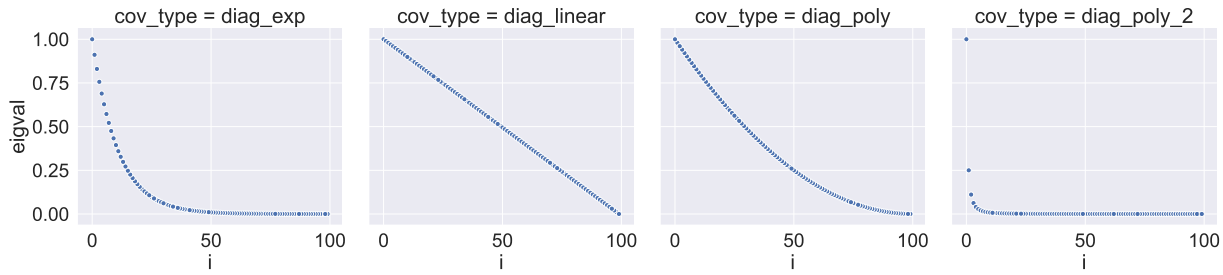


Figure 3.3: Scree-plots of $\mathbf{\Sigma}$ for the eigenvalue decays examined in our empirical valuations.

so that $\delta(z)$ remains bounded and similarly for its derivative $\delta'(z)$, which, by Arzela-Ascoli theorem, yields uniform convergence and we are allowed to take the $z \to 0^+$ limit. Ultimately, in step (iii) we exchange the two limits with Moore-Osgood theorem, concluding the proof.

Finishing the proof of Theorem 3.3

To finish the proof of Theorem 3.3, it remains to write

$$\mathrm{MSE}\big[\mathbf{X}^\dagger \mathbf{y}\big] = \sigma^2 \mathbb{E}\big[\mathrm{tr}\big((\mathbf{X}^\top \mathbf{X})^\dagger\big)\big] + \mathbf{w}^{*\top} \mathbb{E}\big[\mathbf{I} - \mathbf{X}^\dagger \mathbf{X}\big] \mathbf{w}^*$$

Since $\lambda_n = \frac{d-n}{\mathrm{tr}(\mathbf{\Sigma}+\lambda_n \mathbf{I})^{-1}}$, by Lemma 3.14 and Lemma 3.15 we have $\mathrm{MSE}\big[\mathbf{X}^\dagger \mathbf{y}\big] - \mathcal{M}(\mathbf{\Sigma}, \mathbf{w}^*, \sigma^2, n) \to 0$ as $n, d \to \infty$ with $n/d \to \bar{c} \in (0, \infty) \setminus \{1\}$, which concludes the proof of Theorem 3.3.

3.8 Empirical evaluation of asymptotic consistency

In this section, we empirically quantify the convergence rates for the asymptotic result of Theorem 3.3. We focus on the under-determined regime (i.e., $n < d$) and separate the evaluation into the bias and variance terms, following the MSE decomposition given in (3.2). Consider $\mathbf{X} = \mathbf{Z}\mathbf{\Sigma}^{1/2}$, where the entries of $\mathbf{Z}$ are i.i.d. standard Gaussian, and define:

1. Variance discrepancy: $\Big|\frac{\mathbb{E}[\mathrm{tr}((\mathbf{X}^\top \mathbf{X})^\dagger)]}{\mathcal{V}(\mathbf{\Sigma}, n)} - 1\Big|$ where $\mathcal{V}(\mathbf{\Sigma}, n) = \frac{1-\alpha_n}{\lambda_n}$.

2. Bias discrepancy: $\sup_{\mathbf{w} \in \mathbb{R}^d \setminus \{\mathbf{0}\}} \Big|\frac{\mathbf{w}^\top \mathbb{E}[\mathbf{I} - \mathbf{X}^\dagger \mathbf{X}]\mathbf{w}}{\mathbf{w}^\top \mathcal{B}(\mathbf{\Sigma}, n)\mathbf{w}} - 1\Big|$ where $\mathcal{B}(\mathbf{\Sigma}, n) = \lambda_n (\mathbf{\Sigma} + \lambda_n \mathbf{I})^{-1}$.

Recall that $\lambda_n = \frac{d-n}{\mathrm{tr}((\mathbf{\Sigma}+\lambda_n \mathbf{I})^{-1})}$, so our surrogate MSE can be written as $\mathcal{M} = \sigma^2 \mathcal{V}(\mathbf{\Sigma}, n) + \mathbf{w}^{*\top} \mathcal{B}(\mathbf{\Sigma}, n) \mathbf{w}^*$, and when both discrepancies are bounded by ϵ, then $(1-2\epsilon)\mathcal{M} \leq \mathrm{MSE}\big[\mathbf{X}^\dagger \mathbf{y}\big] \leq (1+2\epsilon)\mathcal{M}$. In our experiments, we consider four standard eigenvalue decay profiles for $\mathbf{\Sigma}$, including polynomial and exponential decay (see Figure 3.3 and Section 3.8).

Figure 3.4 (top) plots the variance discrepancy (with $\mathbb{E}[\mathrm{tr}((\mathbf{X}^\top \mathbf{X})^\dagger)]$ estimated via Monte Carlo sampling and bootstrapped confidence intervals) as d increases from 10 to 1000, across a range of aspect ratios n/d. In all cases, we observe that the discrepancy decays to zero at a

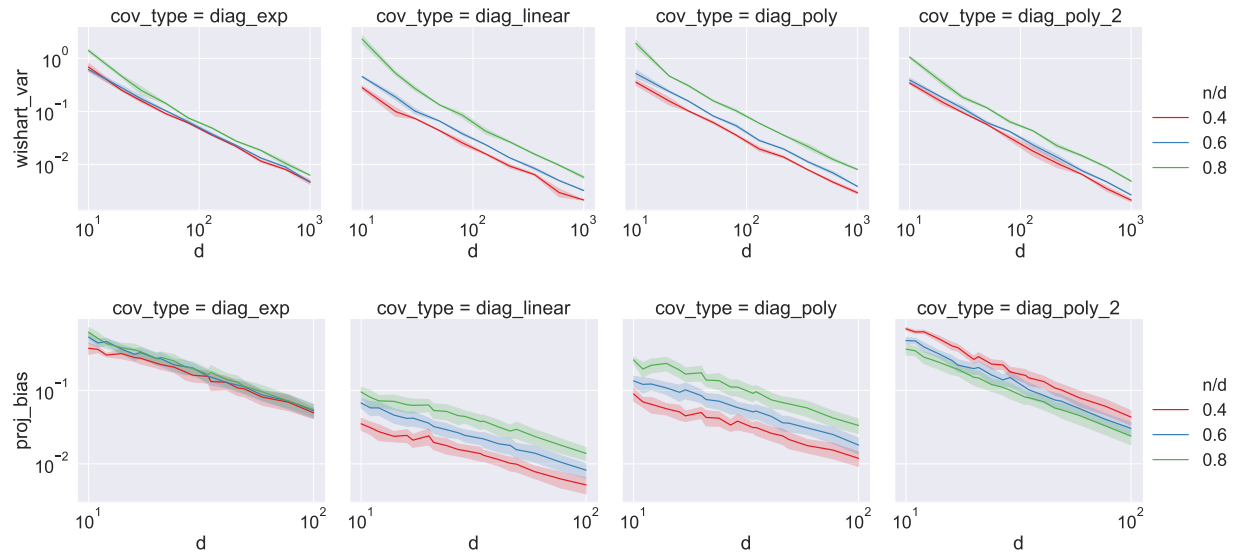


Figure 3.4: Empirical verification of the asymptotic consistency of surrogate MSE. We show the discrepancies for the variance (top) and bias (bottom), with bootstrapped 95% confidence intervals, as d increases and n/d is fixed. We observe $O(1/d)$ decay (linear with slope -1 on a log-log plot).

rate of $O(1/d)$. Figure 3.4 (bottom) plots the bias discrepancy, with the same rate of decay observed throughout. Note that the range of d is smaller than in Figure 3.4 (top) because the large number of Monte Carlo samples (up to two million) required for this experiment made the computations much more expensive (more details in Section 3.8). Based on the above empirical results, we conclude with a conjecture.

Conjecture 3.1 *When μ is a centered multivariate Gaussian and its covariance has a constant condition number, then, for n/d fixed, the surrogate MSE satisfies: $\left|\frac{\mathrm{MSE}[\mathbf{X}^\dagger \mathbf{y}]}{\mathcal{M}} - 1\right| = O(1/d)$.*

Additional details for empirical evaluation

Our empirical investigation of the rate of asymptotic convergence in Theorem 3.3 (and, more specifically, the variance and bias discrepancies defined in Section 3.8), in the context of Gaussian random matrices, is related to open problems which have been extensively studied in the literature. Note that when $\mathbf{X} = \mathbf{Z}\mathbf{\Sigma}^{1/2}$ were $\mathbf{Z}$ has i.i.d. Gaussian entries (as in Section 3.8), then $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$ is known as the pseudo-Wishart distribution (also called the singular Wishart), denoted as $\mathbf{W} \sim \mathcal{PW}(\mathbf{\Sigma}, n)$, and the variance term from the MSE can be written as $\sigma^2 \mathbb{E}[\mathrm{tr}(\mathbf{W}^\dagger)]$. [Sri03] first derived the probability density function of the pseudo-Wishart distribution, and [CF11] computed the first and second moments of generalized inverses. However, for the Moore-Penrose inverse and arbitrary covariance $\mathbf{\Sigma}$, [CF11] claims that the

quantities required to express the mean "do not have tractable closed-form representation." The bias term, $\mathbf{w}^{*\top}\mathbb{E}[\mathbf{I} - \mathbf{X}^\dagger\mathbf{X}]\mathbf{w}^*$, has connections to directional statistics. Using the SVD, we have the equivalent representation $\mathbf{X}^\dagger\mathbf{X} = \mathbf{V}\mathbf{V}^\top$ where $\mathbf{V}$ is an element of the Stiefel manifold $V_{n,d}$ (i.e., orthonormal n-frames in $\mathbb{R}^d$). The distribution of $\mathbf{V}$ is known as the matrix angular central Gaussian (MACG) distribution [Chi90]. While prior work has considered high dimensional limit theorems [Chi91] as well as density estimation and hypothesis testing [Chi98] on $V_{n,d}$, they only analyzed the invariant measure (which corresponds in our setting to $\mathbf{\Sigma} = \mathbf{I}$), and to our knowledge a closed form expression of $\mathbb{E}[\mathbf{V}\mathbf{V}^\top]$ where $\mathbf{V}$ is distributed according to MACG with arbitrary $\mathbf{\Sigma}$ remains an open question.

For analyzing the rate of decay of variance and bias discrepancies (as defined in Section 3.8), it suffices to only consider diagonal covariance matrices $\mathbf{\Sigma}$. This is because if $\mathbf{\Sigma} = \mathbf{Q}\mathbf{D}\mathbf{Q}^\top$ is its eigendecomposition and $\mathbf{X} \sim \mathcal{N}_{n,d}(\mathbf{0}, \mathbf{I}_n \otimes \mathbf{Q}\mathbf{D}\mathbf{Q}^\top)$, then we have for $\mathbf{W} \sim \mathcal{PW}(\mathbf{\Sigma}, n)$ that $\mathbf{W} \overset{d}{=} \mathbf{X}^\top\mathbf{X}$ and hence, defining $\widetilde{\mathbf{X}} \sim \mathcal{N}_{n,d}(\mathbf{0}, \mathbf{I}_n \otimes \mathbf{D})$, by linearity and unitary invariance of trace,

$$\mathbb{E}[\mathrm{tr}(\mathbf{W}^\dagger)] = \mathrm{tr}\big(\mathbb{E}[(\mathbf{X}^\top\mathbf{X})^\dagger]\big) = \mathrm{tr}\Big(\mathbf{Q}\mathbb{E}\big[(\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}})^\dagger\big]\mathbf{Q}^\top\Big) = \mathrm{tr}\Big(\mathbb{E}\big[(\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}})^\dagger\big]\Big) = \mathbb{E}\left[\mathrm{tr}\big((\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}})^\dagger\big)\right].$$

Similarly, we have that $\mathbb{E}[\mathbf{X}^\dagger\mathbf{X}] = \mathbf{Q}\mathbb{E}\big[\widetilde{\mathbf{X}}^\dagger\widetilde{\mathbf{X}}\big]\mathbf{Q}^\top$, and a simple calculation shows that the bias discrepancy is also independent of the choice of matrix $\mathbf{Q}$.

In our experiments, we increase d while keeping the aspect ratio n/d fixed and examining the rate of decay of the discrepancies. We estimate $\mathbb{E}\big[\mathrm{tr}(\mathbf{W}^\dagger)\big]$ (for the variance) and $\mathbb{E}[\mathbf{I}-\mathbf{X}^\dagger\mathbf{X}]$ (for the bias) through Monte Carlo sampling. Confidence intervals are constructed using ordinary bootstrapping for the variance. We rewrite the supremum over $\mathbf{w}$ in bias discrepancy as a spectral norm:

$$\big\|\mathcal{B}(\mathbf{\Sigma}, n)^{-\frac{1}{2}}\mathbb{E}[\mathbf{I} - \mathbf{X}^\dagger\mathbf{X}]\mathcal{B}(\mathbf{\Sigma}, n)^{-\frac{1}{2}} - \mathbf{I}\big\|,$$

and apply existing methods for constructing bootstrapped operator norm confidence intervals described in [LEM19]. To ensure that estimation noise is sufficiently small, we continually increase the number of Monte Carlo samples until the bootstrap confidence intervals are within $\pm 12.5\%$ of the measured discrepancies. We found that while variance discrepancy required a relatively small number of trials (up to one thousand), estimation noise was much larger for the bias discrepancy, and it necessitated over two million trials to obtain good estimates near $d = 100$.

Eigenvalue decay profiles

Letting $\lambda_i(\mathbf{\Sigma})$ be the ith largest eigenvalue of $\mathbf{\Sigma}$, we consider the following eigenvalue profiles (visualized in Figure 3.3):

- `diag_linear`: linear decay, $\lambda_i(\mathbf{\Sigma}) = b - ai$;
- `diag_exp`: exponential decay, $\lambda_i(\mathbf{\Sigma}) = b\,10^{-ai}$;

- `diag_poly`: fixed-degree polynomial decay, $\lambda_i(\mathbf{\Sigma}) = (b - ai)^2$;
- `diag_poly_2`: variable-degree polynomial decay, $\lambda_i(\mathbf{\Sigma}) = bi^{-a}$.

The constants a and b are chosen to ensure $\lambda_{\max}(\mathbf{\Sigma}) = 1$ and $\lambda_{\min}(\mathbf{\Sigma}) = 10^{-4}$ (i.e., the condition number $\kappa(\mathbf{\Sigma}) = 10^4$ remains constant).

3.9 Conclusions

We derived exact non-asymptotic expressions for the MSE of the Moore-Penrose estimator in the linear regression task, reproducing the double descent phenomenon as the sample size crosses between the under- and over-determined regime. To achieve this, we modified the standard i.i.d. random design distribution using a determinantal point process to obtain a surrogate design which admits exact MSE expressions, while capturing the key properties of the i.i.d. design. We also provided a result that relates the expected value of the Moore-Penrose estimator of a training sample in the under-determined regime (i.e., the minimum norm solution) to the ridge-regularized least squares solution for the population distribution, thereby providing an interpretation for the implicit regularization resulting from over-parameterization.

Chapter 4

Exact expectation expressions for sub-Gaussian random projections

It is often desirable to reduce the dimensionality of a large dataset by projecting it onto a low-dimensional subspace. Matrix sketching has emerged as a powerful technique for performing such dimensionality reduction very efficiently. Even though there is an extensive literature on the worst-case performance of sketching, existing guarantees are typically very different from what is observed in practice. Building on the Stieltjes transform methods employed in section 3.7 while establishing asymptotic consistency of surrogate design's MSE, this chapter develops novel techniques that provide provably accurate expressions for the expected value of random projection matrices obtained via sketching. These expressions can be used to characterize the performance of dimensionality reduction in a variety of common machine learning tasks, ranging from low-rank approximation to iterative stochastic optimization. Our results apply to several popular sketching methods, including Gaussian and Rademacher sketches, and they enable precise analysis of these methods in terms of spectral properties of the data. Empirical results show that the expressions we derive reflect the practical performance of these sketching methods, down to lower-order effects and even constant factors. Some of the results here were originally published in Michał Dereziński, Feynman Liang, Zhenyu Liao, and Michael W Mahoney. "Precise expressions for random projections: Low-rank approximation and randomized Newton". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 18272–18283.

4.1 Introduction

Many settings in modern machine learning, optimization and scientific computing require us to work with data matrices that are so large that some form of dimensionality reduction is a necessary component of the process. One of the most popular families of methods for dimensionality reduction, coming from the literature on Randomized Numerical Linear Algebra (RandNLA), consists of data-oblivious sketches [Mic11; HMT11; Woo14]. Consider

a large $m \times n$ matrix $\mathbf{A}$. A *data-oblivious sketch* of size k is the matrix $\mathbf{SA}$, where $\mathbf{S}$ is a $k \times m$ random matrix such that $\mathbb{E}[\frac{1}{k}\mathbf{S}^\top\mathbf{S}] = \mathbf{I}$, whose distribution does not depend on $\mathbf{A}$. This sketch reduces the first dimension of $\mathbf{A}$ from m to a much smaller k (we assume without loss of generality that $k \ll n \leq m$), and an analogous procedure can be defined for reducing the second dimension as well. This approximate representation of $\mathbf{A}$ is central to many algorithms in areas such as linear regression, low-rank approximation, kernel methods, and iterative second-order optimization. While there is a long line of research aimed at bounding the worst-case approximation error of such representations, these bounds are often too loose to reflect accurately the practical performance of these methods. In this paper, we develop new theory which enables more precise analysis of the accuracy of sketched data representations.

A common way to measure the accuracy of the sketch $\mathbf{SA}$ is by considering the k-dimensional subspace spanned by its rows. The goal of the sketch is to choose a subspace that best aligns with the distribution of all of the m rows of $\mathbf{A}$ in $\mathbb{R}^n$. Intuitively, our goal is to minimize the (norm of the) residual when projecting a vector $\mathbf{a} \in \mathbb{R}^n$ onto that subspace, i.e., $\mathbf{a} - \mathbf{Pa} = (\mathbf{I} - \mathbf{P})\mathbf{a}$, where $\mathbf{P} = (\mathbf{SA})^\dagger\mathbf{SA}$ is the orthogonal projection matrix onto the subspace spanned by the rows of $\mathbf{SA}$ (and $(\cdot)^\dagger$ denotes the Moore-Penrose pseudoinverse). For this reason, the quantity that has appeared ubiquitously in the error analysis of RandNLA sketching is what we call the residual projection matrix:

$$\textbf{(residual projection matrix)} \quad \mathbf{P}_\perp \;:=\; \mathbf{I} - \mathbf{P} \;=\; \mathbf{I} - (\mathbf{SA})^\dagger\mathbf{SA}.$$

Since $\mathbf{P}_\perp$ is random, the average performance of the sketch can often be characterized by its expectation, $\mathbb{E}[\mathbf{P}_\perp]$. For example, the low-rank approximation error of the sketch can be expressed as $\mathbb{E}[\|\mathbf{A} - \mathbf{AP}\|_F^2] = \operatorname{tr}\mathbf{A}^\top\mathbf{A}\,\mathbb{E}[\mathbf{P}_\perp]$, where $\|\cdot\|_F$ denotes the Frobenius norm. A similar formula follows for the trace norm error of a sketched Nyström approximation [WS01; GM16]. Among others, this approximation error appears in the analysis of sketched kernel ridge regression [FSS20] and Gaussian process regression [BRV19]. Furthermore, a variety of iterative algorithms, such as randomized second-order methods for convex optimization [Qu+16; QR16; Gow+19; GRB20] and linear system solvers based on the generalized Kaczmarz method [GR15], have convergence guarantees which depend on the extreme eigenvalues of $\mathbb{E}[\mathbf{P}_\perp]$. Finally, a generalized form of the expected residual projection has been recently used to model the implicit regularization of the interpolating solutions in over-parameterized linear models [DLM20b; Bar+19].

Main result

Despite its prevalence in the literature, the expected residual projection is not well understood, even in such simple cases as when $\mathbf{S}$ is a Gaussian sketch (i.e., with i.i.d. standard normal entries). We address this by providing a surrogate expression, i.e., a simple analytically tractable approximation, for this matrix quantity:

$$\mathbb{E}[\mathbf{P}_\perp] \;\overset{\epsilon}{\simeq}\; \bar{\mathbf{P}}_\perp := (\gamma\mathbf{A}^\top\mathbf{A} + \mathbf{I})^{-1}, \quad \text{with} \;\; \gamma > 0 \;\; \text{s.t.} \;\; \operatorname{tr}\bar{\mathbf{P}}_\perp = n - k. \tag{4.1}$$

Here, $\overset{\epsilon}{\preceq}$ means that while the surrogate expression is not exact, it approximates the true quantity up to some ϵ accuracy. Our main result provides a rigorous approximation guarantee for this surrogate expression with respect to a range of sketching matrices $\mathbf{S}$, including the standard Gaussian and Rademacher sketches. We state the result using the positive semi-definite ordering denoted by $\preceq$.

Theorem 4.1 *Let $\mathbf{S}$ be a sketch of size k with i.i.d. mean-zero sub-gaussian entries and let $r = \|\mathbf{A}\|_F^2/\|\mathbf{A}\|^2$ be the stable rank of $\mathbf{A}$. If we let $\rho = r/k$ be a fixed constant larger than 1, then*

$$(1-\epsilon)\,\bar{\mathbf{P}}_\perp \preceq \mathbb{E}[\mathbf{P}_\perp] \preceq (1+\epsilon)\,\bar{\mathbf{P}}_\perp \quad \textit{for} \quad \epsilon = O(\tfrac{1}{\sqrt{r}}).$$

In other words, when the sketch size k is smaller than the stable rank r of $\mathbf{A}$, then the discrepancy between our surrogate expression $\bar{\mathbf{P}}_\perp$ and $\mathbb{E}[\mathbf{P}_\perp]$ is of the order $1/\sqrt{r}$, where the big-O notation hides only the dependence on ρ and on the sub-gaussian constant (see Theorem 4.2 for more details). Our proof of Theorem 4.1 is inspired by the techniques from random matrix theory which have been used to analyze the asymptotic spectral distribution of large random matrices by focusing on the associated matrix resolvents and Stieltjes transforms [HLN+07; BS10]. However, our analysis is novel in several respects:

1. The residual projection matrix can be obtained from the appropriately scaled resolvent matrix $z(\mathbf{A}^\top\mathbf{S}^\top\mathbf{S}\mathbf{A} + z\mathbf{I})^{-1}$ by taking $z \to 0$. Prior work (e.g., [Has+19]) combined this with an exchange-of-limits argument to analyze the asymptotic behavior of the residual projection. This approach, however, does not allow for a precise control in finite-dimensional problems. We are able to provide a more fine-grained, non-asymptotic analysis by working directly with the residual projection itself, instead of the resolvent.

2. We require no assumptions on the largest and smallest singular value of $\mathbf{A}$. Instead, we derive our bounds in terms of the stable rank of $\mathbf{A}$ (as opposed to its actual rank), which implicitly compensates for ill-conditioned data matrices.

3. We obtain upper/lower bounds for $\mathbb{E}[\mathbf{P}_\perp]$ in terms of the positive semi-definite ordering $\preceq$, which can be directly converted to guarantees for the precise expressions of expected low-rank approximation error derived in the following section.

It is worth mentioning that the proposed analysis is significantly different from the sketching literature based on subspace embeddings (e.g., [Sar06; CW17; NN13; Coh+15; CNW16]), in the sense that here our object of interest is not to obtain a worst-case approximation with high probability, but rather, our analysis provides *precise* characterization on the *expected* residual projection matrix that goes *beyond worst-case bounds*. From an application perspective, the subspace embedding property is neither sufficient nor necessary for many numerical implementations of sketching [AMT10; MSM14], or statistical results [RM16; DL19; Yan+20], as well as in the context of iterative optimization and implicit regularization (see Sections 4.1 and 4.1 below), which are discussed in detail as concrete applications of the proposed analysis.

Low-rank approximation

We next provide some immediate corollaries of Theorem 4.1, where we use $x \overset{\epsilon}{\simeq} y$ to denote a multiplicative approximation $|x-y| \leq \epsilon y$. Note that our analysis is new even for the classical Gaussian sketch where the entries of $\mathbf{S}$ are i.i.d. standard normal. However the results apply more broadly, including a standard class of data-base friendly Rademacher sketches where each entry s_{ij} is a ± 1 Rademacher random variable [Ach03]. We start by analyzing the Frobenius norm error $\|\mathbf{A}-\mathbf{AP}\|_F^2 = \operatorname{tr}\mathbf{A}^\top\mathbf{A}\,\mathbf{P}_\perp$ of sketched low-rank approximations. Note that by the definition of γ in (4.1), we have $k = \operatorname{tr}(\mathbf{I}-\bar{\mathbf{P}}_\perp) = \operatorname{tr}\gamma\mathbf{A}^\top\mathbf{A}(\gamma\mathbf{A}^\top\mathbf{A}+\mathbf{I})^{-1}$, so the surrogate expression we obtain for the expected error is remarkably simple.

Corollary 4.1 *Let σ_i be the singular values of $\mathbf{A}$. Under the assumptions of Theorem 4.1, we have:*

$$\mathbb{E}\big[\|\mathbf{A}-\mathbf{AP}\|_F^2\big] \;\overset{\epsilon}{\simeq}\; k/\gamma \quad \textit{for} \;\; \gamma > 0 \;\; \textit{s.t.} \;\; \sum_i \frac{\gamma\sigma_i^2}{\gamma\sigma_i^2+1} = k.$$

Remark 4.1 *The parameter $\gamma = \gamma(k)$ increases at least linearly as a function of k, which is why the expected error will always decrease with increasing k. For example, when the singular values of $\mathbf{A}$ exhibit exponential decay, i.e., $\sigma_i^2 = C\cdot\alpha^{i-1}$ for $\alpha \in (0,1)$, then the error also decreases exponentially, at the rate of $k/(\alpha^{-k}-1)$. We discuss this further in Section 4.5, giving explicit formulas for the error as a function of k under both exponential and polynomial spectral decay profiles.*

The above result is important for many RandNLA methods, and it is also relevant in the context of kernel methods, where the data is represented via a positive semi-definite $m \times m$ kernel matrix $\mathbf{K}$ which corresponds to the matrix of dot-products of the data vectors in some reproducible kernel Hilbert space. In this context, sketching can be applied directly to the matrix $\mathbf{K}$ via an extended variant of the Nyström method [GM16]. A Nyström approximation constructed from a sketching matrix $\mathbf{S}$ is defined as $\tilde{\mathbf{K}} = \mathbf{C}^\top\mathbf{W}^\dagger\mathbf{C}$, where $\mathbf{C} = \mathbf{SK}$ and $\mathbf{W} = \mathbf{SKS}^\top$, and it is applicable to a variety of settings, including Gaussian Process regression, kernel machines and Independent Component Analysis [BRV19; WS01; BJ03]. By setting $\mathbf{A} = \mathbf{K}^{\frac{1}{2}}$, it is easy to see [DKM20] that the trace norm error $\|\mathbf{K}-\tilde{\mathbf{K}}\|_*$ is identical to the squared Frobenius norm error of the low-rank sketch $\mathbf{SA}$, so Corollary 4.1 implies that

$$\mathbb{E}\big[\|\mathbf{K}-\tilde{\mathbf{K}}\|_*\big] \;\overset{\epsilon}{\simeq}\; k/\gamma \quad \text{for} \;\; \gamma > 0 \;\; \text{s.t.} \;\; \sum_i \frac{\gamma\lambda_i}{\gamma\lambda_i+1} = k, \tag{4.2}$$

with any sub-gaussian sketch, where λ_i denote the eigenvalues of $\mathbf{K}$. Our error analysis given in Section 4.5 is particularly relevant here, since commonly used kernels such as the Radial Basis Function (RBF) or the Matérn kernel induce a well-understood eigenvalue decay [San+97; RW06].

Metrics other than the aforementioned Frobenius norm error, such as the spectral norm error [HMT11], are also of significant interest in the low-rank approximation literature. We leave these directions for future investigation.

Randomized iterative optimization

We next turn to a class of iterative methods which take advantage of sketching to reduce the per iteration cost of optimization. These methods have been developed in a variety of settings, from solving linear systems to convex optimization and empirical risk minimization, and in many cases the residual projection matrix appears as a black box quantity whose spectral properties determine the convergence behavior of the algorithms [GR15]. With our new results, we can precisely characterize not only the rate of convergence, but also, in some cases, the complete evolution of the parameter vector, for the following algorithms:

1. *Generalized Kaczmarz method* [GR15] for approximately solving a linear system $\mathbf{Ax} = \mathbf{b}$;
2. *Randomized Subspace Newton* [Gow+19], a second order method, where we sketch the Hessian matrix.
3. *Jacobian Sketching* [GRB20], a class of first order methods which use additional information via a weight matrix $\mathbf{W}$ that is sketched at every iteration.

We believe that extensions of our techniques will apply to other algorithms, such as that of [LPP19].

We next give a result in the context of linear systems for the generalized Kaczmarz method [GR15], but a similar convergence analysis is given for the methods of [Gow+19; GRB20] in Section 4.2.

Corollary 4.2 *Let $\mathbf{x}^*$ be the unique solution of $\mathbf{Ax}^* = \mathbf{b}$ and consider the iterative algorithm:*

$$\mathbf{x}^{t+1} = \operatorname*{argmin}_{\mathbf{x}} \|\mathbf{x} - \mathbf{x}^t\|^2 \quad \text{subject to} \quad \mathbf{SAx} = \mathbf{Sb}.$$

Under the assumptions of Theorem 4.1, with γ defined in (4.1) and $r = \|\mathbf{A}\|_F^2/\|\mathbf{A}\|^2$, we have:

$$\mathbb{E}\big[\mathbf{x}^{t+1} - \mathbf{x}^*\big] \overset{\epsilon}{\simeq} (\gamma \mathbf{A}^\top \mathbf{A} + \mathbf{I})^{-1}\, \mathbb{E}\big[\mathbf{x}^t - \mathbf{x}^*\big] \quad \textit{for} \quad \epsilon = O(\tfrac{1}{\sqrt{r}}).$$

The corollary follows from Theorem 4.1 combined with Theorem 4.1 in [GR15]. Note that when $\mathbf{A}^\top\mathbf{A}$ is positive definite then $(\gamma\mathbf{A}^\top\mathbf{A} + \mathbf{I})^{-1} \prec \mathbf{I}$, so the algorithm will converge from any starting point, and the worst-case convergence rate of the above method can be obtained by evaluating the largest eigenvalue of $(\gamma\mathbf{A}^\top\mathbf{A} + \mathbf{I})^{-1}$. However the result itself is much stronger, in that it can be used to describe the (expected) trajectory of the iterates for any starting point $\mathbf{x}^0$. Moreover, when the spectral decay profile of $\mathbf{A}$ is known, then the explicit expressions for γ as a function of k derived in Section 4.5 can be used to characterize the convergence properties of generalized Kaczmarz as well as other methods discussed above.

Implicit regularization

Setting $\mathbf{x}^t = \mathbf{0}$, we can view one step of the iterative method in Corollary 4.2 as finding a minimum norm interpolating solution of an under-determined linear system $(\mathbf{SA}, \mathbf{Sb})$. Recent interest in the generalization capacity of over-parameterized machine learning models has motivated extensive research on the statistical properties of such interpolating solutions [e.g., Bar+19; Has+19; DLM20b]. In this context, Theorem 4.1 provides new evidence for the implicit regularization conjecture posed by [DLM20b] (see their Theorem 2 and associated discussion), with the amount of regularization equal $\frac{1}{\gamma}$, where γ is implicitly defined in (4.1):

$$\underbrace{\mathbb{E}\Big[\operatorname*{argmin}_{\mathbf{x}} \|\mathbf{x}\|^2 \;\text{ s.t. }\; \mathbf{SAx} = \mathbf{Sb}\Big] - \mathbf{x}^*}_{\text{Bias of sketched minimum norm solution}} \;\overset{\epsilon}{\simeq}\; \underbrace{\operatorname*{argmin}_{\mathbf{x}} \Big\{\|\mathbf{Ax} - \mathbf{b}\|^2 + \tfrac{1}{\gamma}\|\mathbf{x}\|^2\Big\} - \mathbf{x}^*}_{\text{Bias of } l_2\text{-regularized solution}}.$$

While implicit regularization has received attention recently in the context of SGD algorithms for overparameterized machine learning models, it was originally discussed in the context of approximation algorithms more generally [M W12]. Recent work has made precise this notion in the context of RandNLA [DLM20b], and our results here can be viewed in terms of implicit regularization of scalable RandNLA methods.

Related work

A significant body of research has been dedicated to understanding the guarantees for low-rank approximation via sketching, particularly in the context of RandNLA [DM16; DM18]. This line of work includes i.i.d. row sampling methods [BMD08; AM15] which preserve the structure of the data, and data-oblivious methods such as Gaussian and Rademacher sketches [Mic11; HMT11; Woo14]. However, all of these results focus on worst-case upper bounds on the approximation error. One exception is a recent line of works on non-i.i.d. row sampling with Determinantal Point Processes (DPP, [DM21]). In this case, exact analysis of the low-rank approximation error [DKM20], as well as precise convergence analysis of stochastic second order methods [MDK20], have been obtained. Remarkably, the expressions they obtain are analogous to (4.1), despite using completely different techniques. However, their analysis is limited only to DPP-based sketches, which are considerably more expensive to construct and thus much less widely used. The connection between DPPs and Gaussian sketches was recently explored by [DLM20b] in the context of analyzing the implicit regularization effect of choosing a minimum norm solution in under-determined linear regression. They conjectured that the expectation formulas obtained for DPPs are a good proxy for the corresponding quantities obtained under a Gaussian distribution. Similar observations were made by [Der+20a] in the context of sketching for regularized least squares and second order optimization. While both of these works only provide empirical evidence for this particular claim, our Theorem 4.1 can be viewed as the first theoretical non-asymptotic justification of that conjecture.

The effectiveness of sketching has also been extensively studied in the context of second order optimization. These methods differ depending on how the sketch is applied to the Hessian matrix, and whether or not it is applied to the gradient as well. The class of methods discussed in Section 4.1, including Randomized Subspace Newton and the Generalized Kaczmarz method, relies on projecting the Hessian downto a low-dimensional subspace, which makes our results directly applicable. A related family of methods uses the so-called Iterative Hessian Sketch (IHS) approach [PW16b; LP19]. The similarities between IHS and the Subspace Newton-type methods (see [Qu+16] for a comparison) suggest that our techniques could be extended to provide precise convergence guarantees also to the IHS. Finally, yet another family of Hessian sketching methods has been studied by [RM19; WGM17; XRM17; Yao+18b; Roo+18; Wan+17a; DM19]. These methods preserve the rank of the Hessian, and so their convergence guarantees do not rely on the residual projection.

4.2 Convergence analysis of randomized iterative methods

Here, we discuss how our surrogate expressions for the expected residual projection can be used to perform convergence analysis for several randomized iterative optimization methods discussed in Section 4.1.

Generalized Kaczmarz method

Generalized Kaczmarz [GR15] is an iterative method for solving an $m \times n$ linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$, which uses a $k \times m$ sketching matrix $\mathbf{S}_t$ to reduce the linear system and update an iterate $\mathbf{x}^t$ as follows:

$$\mathbf{x}^{t+1} = \operatorname*{argmin}_{\mathbf{x}} \|\mathbf{x} - \mathbf{x}^t\|^2 \quad \text{subject to} \quad \mathbf{S}_t\mathbf{A}\mathbf{x} = \mathbf{S}_t\mathbf{b}.$$

Assume that $\mathbf{x}^*$ is the unique solution to the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$. In Theorems 4.1 and 4.6, [GR15] show that the expected trajectory of the generalized Kaczmarz iterates, as they converge to $\mathbf{x}^*$, is controlled by the projection matrix $\mathbf{P} = (\mathbf{S}_t\mathbf{A})^\dagger\mathbf{S}_t\mathbf{A}$ as follows:

([GR15], Theorem 4.1) $$\mathbb{E}[\mathbf{x}^{t+1} - \mathbf{x}^*] = \big(\mathbf{I} - \mathbb{E}[\mathbf{P}]\big)\,\mathbb{E}[\mathbf{x}^t - \mathbf{x}^*],$$

([GR15], Theorem 4.6) $$\mathbb{E}\big[\|\mathbf{x}^{t+1} - \mathbf{x}^*\|^2\big] \leq (1-\kappa)\,\mathbb{E}\big[\|\mathbf{x}^t - \mathbf{x}^*\|^2\big], \text{ where } \kappa = \lambda_{\min}\big(\mathbb{E}[\mathbf{P}]\big).$$

Both of these results depend on the expected projection $\mathbb{E}[\mathbf{P}]$. The first one describes the expected trajectory of the iterate, whereas the second one gives the worst-case convergence rate in terms of the so-called *stochastic condition number* κ. We next demonstrate how Theorem 4.1 can be used in combination with the above results to obtain convergence analysis for generalized Kaczmarz which is formulated in terms of the spectral properties of $\mathbf{A}$. This includes precise expressions for both the expected trajectory and κ. The following result is a more detailed version of Corollary 4.2 from Section 4.1.

Corollary 4.3 *Let σ_i denote the singular values of $\mathbf{A}$, and let k denote the size of sketch $\mathbf{S}_t$. Define:*

$$\Delta_t = \mathbf{x}^t - \mathbf{x}^* \quad \textit{and} \quad \bar{\Delta}_{t+1} = (\gamma \mathbf{A}^\top \mathbf{A} + \mathbf{I})^{-1} \mathbb{E}[\Delta_t] \quad \textit{s.t.} \quad \sum_i \frac{\gamma \sigma_i^2}{\gamma \sigma_i^2 + 1} = k.$$

Suppose that $\mathbf{S}_t$ has i.i.d. mean-zero sub-gaussian entries and let $r = \|\mathbf{A}\|_F^2 / \|\mathbf{A}\|^2$ be the stable rank of $\mathbf{A}$. Assume that $\rho = r/k$ is a constant larger than 1. Then, the expected trajectory satisfies:

$$\big\| \mathbb{E}[\Delta_{t+1}] - \bar{\Delta}_{t+1} \big\| \leq \epsilon \cdot \|\bar{\Delta}_{t+1}\|, \quad \textit{for} \quad \epsilon = O\big(\tfrac{1}{\sqrt{r}}\big). \tag{4.3}$$

Moreover, we obtain the following worst-case convergence guarantee:

$$\mathbb{E}\big[\|\Delta_{t+1}\|^2\big] \leq \big(1 - (\bar{\kappa} - \epsilon)\big)\, \mathbb{E}\big[\|\Delta_t\|^2\big], \quad \textit{where} \quad \bar{\kappa} = \frac{\sigma_{\min}^2}{\sigma_{\min}^2 + 1/\gamma}. \tag{4.4}$$

Remark 4.2 *Our worst-case convergence guarantee* (4.4) *requires the matrix $\mathbf{A}$ to be sufficiently well-conditioned so that $\bar{\kappa} - \epsilon > 0$. However, we believe that our surrogate expression $\bar{\kappa}$ for the stochastic condition number is far more accurate than suggested by the current analysis.*

Proof of Corollary 4.3 Using Theorem 4.1, for $\bar{\mathbf{P}}_\perp$ as defined in (4.1), we have

$$(1-\epsilon)\bar{\mathbf{P}}_\perp \preceq \mathbf{I} - \mathbb{E}[\mathbf{P}] = \mathbb{E}[\mathbf{P}_\perp] \preceq (1+\epsilon)\bar{\mathbf{P}}_\perp, \quad \text{where} \quad \epsilon = O\big(\tfrac{1}{\sqrt{r}}\big).$$

In particular, this implies that $\|\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}(\mathbb{E}[\mathbf{P}_\perp] - \bar{\mathbf{P}}_\perp)\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\| \leq \epsilon$. Moreover, in the proof of Theorem 4.2 we showed that $\frac{\rho-1}{\rho}\mathbf{I} \preceq \bar{\mathbf{P}}_\perp \preceq \mathbf{I}$, see (4.6), so it follows that:

$$\bar{\mathbf{P}}_\perp^{-1}(\mathbb{E}[\mathbf{P}_\perp] - \bar{\mathbf{P}}_\perp)^2\bar{\mathbf{P}}_\perp^{-1} \preceq \frac{\rho}{\rho-1}\big(\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}(\mathbb{E}[\mathbf{P}_\perp] - \bar{\mathbf{P}}_\perp)\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\big)^2 \preceq \frac{\rho}{\rho-1}\,\epsilon^2 \cdot \mathbf{I},$$

where note that $\frac{\rho}{\rho-1}\,\epsilon^2 = O(1/r)$, since ρ is treated as a constant. Thus we conclude that:

$$\begin{aligned}\|\mathbb{E}[\Delta_{t+1}] - \bar{\Delta}_{t+1}\|^2 &= \mathbb{E}[\Delta_t]^\top(\mathbb{E}[\mathbf{P}_\perp] - \bar{\mathbf{P}}_\perp)^2\mathbb{E}[\Delta_t] \\ &\leq O(1/r) \cdot \mathbb{E}[\Delta_t]^\top \bar{\mathbf{P}}_\perp^2 \mathbb{E}[\Delta_t] = O(1/r) \cdot \|\bar{\Delta}_{t+1}\|^2,\end{aligned}$$

which completes the proof of (4.3). To show (4.4), it suffices to observe that

$$\lambda_{\min}(\mathbb{E}[\mathbf{P}]) = 1 - \lambda_{\max}(\mathbb{E}[\mathbf{P}_\perp]) \geq 1 - (1+\epsilon)\lambda_{\max}(\bar{\mathbf{P}}_\perp) \geq \lambda_{\min}(\mathbf{I} - \bar{\mathbf{P}}_\perp) - \epsilon,$$

which completes the proof since $\mathbf{I} - \bar{\mathbf{P}}_\perp = \gamma \mathbf{A}^\top \mathbf{A}(\gamma \mathbf{A}^\top \mathbf{A} + \mathbf{I})^{-1}$. ■ Corollaries 4.4 and 4.5 follow analogously from Theorem 4.1.

Randomized Subspace Newton

Randomized Subspace Newton (RSN, [Gow+19]) is a randomized Newton-type method for minimizing a smooth, convex and twice differentiable function $f : \mathbb{R}^d \times \mathbb{R}$. The iterative update for this algorithm is defined as follows:

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \frac{1}{L}\mathbf{S}_t^\top(\mathbf{S}_t\mathbf{H}(\mathbf{x}^t)\mathbf{S}_t^\top)^\dagger\mathbf{S}_t\mathbf{g}(\mathbf{x}^t),$$

where $\mathbf{H}(\mathbf{x}^t)$ and $\mathbf{g}(\mathbf{x}^t)$ are the Hessian and gradient of f at $\mathbf{x}^t$, respectively, whereas $\mathbf{S}_t$ is a $k \times d$ sketching matrix (with $k \ll d$) which is refreshed at every iteration. Here, L denotes the *relative smoothness* constant defined by [Gow+19] in Assumption 1, which also defines relative strong convexity, denoted by μ. In Theorem 2, they prove the following convergence guarantee for RSN:

$$\mathbb{E}[f(\mathbf{x}^t)] - f(\mathbf{x}^*) \leq \left(1 - \kappa\frac{\mu}{L}\right)^t (f(\mathbf{x}^0) - f(\mathbf{x}^*)),$$

where $\kappa = \min_{\mathbf{x}} \kappa(\mathbf{x})$ and $\kappa(\mathbf{x}) = \lambda_{\min}^+(\mathbb{E}[\mathbf{P}(\mathbf{x})])$ is the smallest positive eigenvalue of the expectation of the projection matrix $\mathbf{P}(\mathbf{x}) = \mathbf{H}^{\frac{1}{2}}(\mathbf{x})\mathbf{S}_t^\top(\mathbf{S}_t\mathbf{H}(\mathbf{x})\mathbf{S}_t^\top)^\dagger\mathbf{S}_t\mathbf{H}^{\frac{1}{2}}(\mathbf{x})$. Our results lead to the following surrogate expression for this expected projection when the sketch is sub-gaussian:

$$\mathbb{E}[\mathbf{P}(\mathbf{x})] \simeq \mathbf{H}(\mathbf{x})\big(\mathbf{H}(\mathbf{x}) + \tfrac{1}{\gamma(\mathbf{x})}\mathbf{I}\big)^{-1} \quad \text{for} \quad \gamma(\mathbf{x}) > 0 \quad \text{s.t.} \quad \operatorname{tr}\mathbf{H}(\mathbf{x})\big(\mathbf{H}(\mathbf{x}) + \tfrac{1}{\gamma(\mathbf{x})}\mathbf{I}\big)^{-1} = k.$$

Thus, the condition number κ of RSN can be estimated using the following surrogate expression:

$$\kappa \simeq \bar{\kappa} := \min_{\mathbf{x}} \frac{\lambda_{\min}^+(\mathbf{H}(\mathbf{x}))}{\lambda_{\min}^+(\mathbf{H}(\mathbf{x})) + 1/\gamma(\mathbf{x})}.$$

Just as in Corollary 4.3, an approximation of the form $|\bar{\kappa} - \kappa| \leq \epsilon$ can be shown from Theorem 4.1.

Corollary 4.4 *Suppose that sketch $\mathbf{S}_t$ has size k and i.i.d. mean-zero sub-gaussian entries. Let $r = \min_{\mathbf{x}} \operatorname{tr}\mathbf{H}(\mathbf{x})/\|\mathbf{H}(\mathbf{x})\|$ be the (minimum) stable rank of the (square root) Hessian and assume that $\rho = r/k$ is a constant larger than 1. Then,*

$$|\kappa - \bar{\kappa}| \leq O\big(\tfrac{1}{\sqrt{r}}\big).$$

Jacobian Sketching

Jacobian Sketching (JacSketch, [GRB20]) defines an $n \times n$ positive semi-definite weight matrix $\mathbf{W}$, and combines it with an $k \times n$ sketching matrix $\mathbf{S}$ (which is refreshed at every iteration of the algorithm), to implicitly construct the following projection matrix:

$$\Pi_{\mathbf{S}} = \mathbf{S}^\top(\mathbf{S}\mathbf{W}\mathbf{S}^\top)^\dagger\mathbf{S}\mathbf{W},$$

which is used to sketch the Jacobian at the current iterate (for the complete method, we refer to their Algorithm 1). The convergence rate guarantee given in their Theorem 3.6 for JacSketch is given in terms of the Lyapunov function:

$$\Psi^t = \|\mathbf{x}^t - \mathbf{x}^*\|^2 + \frac{\alpha}{2\mathcal{L}_2}\|\mathbf{J}^t - \nabla F(\mathbf{x}^*)\|^2_{\mathbf{W}^{-1}},$$

where α is the step size used by the algorithm. Under appropriate choice of the step-size, Theorem 3.6 states that:

$$\mathbb{E}[\Psi^t] \leq \left(1 - \mu \min\left\{\frac{1}{4\mathcal{L}_1}, \frac{\kappa}{4\mathcal{L}_2\rho/n^2 + \mu}\right\}\right)^t \cdot \Psi^0,$$

where $\kappa = \lambda_{\min}(\mathbb{E}[\Pi_{\mathbf{S}}])$ is the *stochastic condition number* analogous to the one defined for the Generalized Kaczmarz method, n is the data size and parameters ρ, $\mathcal{L}_1$, $\mathcal{L}_2$ and μ are problem dependent constants defined in Theorem 3.6. Similarly as before, we can use our surrogate expressions for the expected residual projection to obtain a precise estimate for the stochastic condition number κ under sub-gaussian sketching:

$$\kappa \simeq \bar{\kappa} := \frac{\lambda_{\min}(\mathbf{W})}{\lambda_{\min}(\mathbf{W}) + 1/\gamma} \quad \text{for} \quad \gamma > 0 \quad \text{s.t.} \quad \operatorname{tr}\mathbf{W}(\mathbf{W} + \tfrac{1}{\gamma}\mathbf{I})^{-1} = k.$$

Corollary 4.5 *Suppose $\mathbf{S}_t$ has size k and i.i.d. mean-zero sub-gaussian entries. Let $r = \operatorname{tr}\mathbf{W}/\|\mathbf{W}\|$ be the stable rank of $\mathbf{W}^{\frac{1}{2}}$ and assume that $\rho = r/k$ is a constant larger than 1. Then,*

$$|\kappa - \bar{\kappa}| \leq O\big(\tfrac{1}{\sqrt{r}}\big).$$

4.3 Precise analysis of the residual projection

In this section, we give a detailed statement of our main technical result, along with a sketch of the proof. First, recall the definition of sub-gaussian random variables and vectors.

Definition 4.1 *We say that x is a K-sub-gaussian random variable if its sub-gaussian Orlicz norm $\|x\|_{\psi_2} \leq K$, where $\|x\|_{\psi_2} := \inf\{t > 0 : \mathbb{E}[\exp(x^2/t^2)] \leq 2\}$. Similarly, we say that a random vector $\mathbf{x}$ is K-sub-gaussian if for all $\|\mathbf{a}\| \leq 1$ we have $\|\mathbf{x}^\top\mathbf{a}\|_{\psi_2} \leq K$.*

For convenience, we state the main result in a slightly different form than Theorem 4.1. Namely, we replace the $m \times n$ matrix $\mathbf{A}$ with a positive semi-definite $n \times n$ matrix $\mathbf{\Sigma}^{\frac{1}{2}}$. Furthermore, instead of a sketch $\mathbf{S}$ with i.i.d. sub-gaussian entries, we use a random matrix $\mathbf{Z}$ with i.i.d. sub-gaussian *rows*, which is a strictly weaker condition because it allows for the entries of each row to be correlated. Since the rows of $\mathbf{Z}$ are also assumed to have mean zero and identity covariance, each row of $\mathbf{Z}\mathbf{\Sigma}^{\frac{1}{2}}$ has covariance $\mathbf{\Sigma}$. In Section 4.3 we show how to convert this statement back to the form of Theorem 4.1.

Theorem 4.2 *Let* $\mathbf{P}_\perp = \mathbf{I} - \mathbf{X}^\dagger\mathbf{X}$ *for* $\mathbf{X} = \mathbf{Z}\mathbf{\Sigma}^{\frac{1}{2}}$*, where* $\mathbf{Z} \in \mathbb{R}^{k\times n}$ *has i.i.d. K-sub-gaussian rows with zero mean and identity covariance, and* $\mathbf{\Sigma}$ *is an* $n \times n$ *positive semi-definite matrix. Define:*

$$\bar{\mathbf{P}}_\perp = (\gamma\mathbf{\Sigma} + \mathbf{I})^{-1}, \quad \textit{such that} \quad \operatorname{tr}\bar{\mathbf{P}}_\perp = n - k.$$

Let $r = \operatorname{tr}(\mathbf{\Sigma})/\|\mathbf{\Sigma}\|$ *be the stable rank of* $\mathbf{\Sigma}^{\frac{1}{2}}$ *and fix* $\rho = r/k > 1$*. There exists a constant* $C_\rho > 0$*, depending only on* ρ *and* K*, such that if* $r \geq C_\rho$*, then*

$$\Big(1 - \frac{C_\rho}{\sqrt{r}}\Big) \cdot \bar{\mathbf{P}}_\perp \preceq \mathbb{E}[\mathbf{P}_\perp] \preceq \Big(1 + \frac{C_\rho}{\sqrt{r}}\Big) \cdot \bar{\mathbf{P}}_\perp. \tag{4.5}$$

We first provide the following informal derivation of the expression for $\bar{\mathbf{P}}_\perp$ given in Theorem 4.2. Let us use $\mathbf{P}$ to denote the matrix $\mathbf{X}^\dagger\mathbf{X} = \mathbf{I} - \mathbf{P}_\perp$. Using a rank-one update formula for the Moore-Penrose pseudoinverse (see Lemma 4.1 in the appendix) we have

$$\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp] = \mathbb{E}[\mathbf{P}] = \mathbb{E}\big[(\mathbf{X}^\top\mathbf{X})^\dagger\mathbf{X}^\top\mathbf{X}\big] = \sum_{i=1}^{k} \mathbb{E}[(\mathbf{X}^\top\mathbf{X})^\dagger\mathbf{x}_i\mathbf{x}_i^\top] = k\,\mathbb{E}\left[\frac{(\mathbf{I} - \mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top}{\mathbf{x}_k^\top(\mathbf{I} - \mathbf{P}_{-k})\mathbf{x}_k}\right],$$

where we use $\mathbf{x}_i^\top$ to denote the i-th row of $\mathbf{X}$, and $\mathbf{P}_{-k} = \mathbf{X}_{-k}^\dagger\mathbf{X}_{-k}$, where $\mathbf{X}_{-i}$ is the matrix $\mathbf{X}$ without its i-th row. Due to the sub-gaussianity of $\mathbf{x}_k$, the quadratic form $\mathbf{x}_k^\top(\mathbf{I} - \mathbf{P}_{-k})\mathbf{x}_k$ in the denominator concentrates around its expectation (with respect to $\mathbf{x}_k$), i.e., $\operatorname{tr}\mathbf{\Sigma}(\mathbf{I} - \mathbf{P}_{-k})$, where we use $\mathbb{E}[\mathbf{x}_k\mathbf{x}_k^\top] = \mathbf{\Sigma}$. Further note that, with $\mathbf{P}_{-k} \simeq \mathbf{P}$ for large k and $\frac{1}{k}\operatorname{tr}\mathbf{\Sigma}(\mathbf{I} - \mathbf{P}_{-k}) \simeq \frac{1}{k}\operatorname{tr}\mathbf{\Sigma}\mathbb{E}[\mathbf{P}_\perp]$ from a concentration argument, we conclude that

$$\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp] \simeq \frac{k\mathbb{E}[\mathbf{P}_\perp]\mathbf{\Sigma}}{\operatorname{tr}\mathbf{\Sigma}\mathbb{E}[\mathbf{P}_\perp]} \qquad \Longrightarrow \qquad \mathbb{E}[\mathbf{P}_\perp] \simeq \Big(\frac{k\mathbf{\Sigma}}{\operatorname{tr}\mathbf{\Sigma}\mathbb{E}[\mathbf{P}_\perp]} + \mathbf{I}\Big)^{-1},$$

and thus $\mathbb{E}[\mathbf{P}_\perp] \simeq \bar{\mathbf{P}}_\perp$ for $\bar{\mathbf{P}}_\perp = (\gamma\mathbf{\Sigma} + \mathbf{I})^{-1}$ and $\gamma^{-1} = \frac{1}{k}\operatorname{tr}\mathbf{\Sigma}\bar{\mathbf{P}}_\perp$. This leads to the (implicit) expression for $\bar{\mathbf{P}}_\perp$ and γ given in Theorem 4.2.

Proof sketch of Theorem 4.2

To make the above intuition rigorous, we next present a proof sketch for Theorem 4.2, with the detailed proof deferred to Appendix 4.4. The proof can be divided into the following three steps.

Step 1. First note that, to obtain the lower and upper bound for $\mathbb{E}[\mathbf{P}_\perp]$ in the sense of symmetric matrix as in Theorem 4.2, it suffices to bound the spectral norm $\|\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1}\| \leq \frac{C_\rho}{\sqrt{r}}$, so that, with $\frac{\rho-1}{\rho}\mathbf{I} \preceq \bar{\mathbf{P}}_\perp \preceq \mathbf{I}$ for $\rho = r/k > 1$ from the definition of $\bar{\mathbf{P}}_\perp$, we have

$$\|\mathbf{I} - \bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\| = \|\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}(\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}^{-1})\bar{\mathbf{P}}_\perp^{\frac{1}{2}}\| \leq \frac{C_\rho}{\sqrt{r}}\sqrt{\frac{\rho}{\rho-1}} =: \epsilon.$$

This means that all eigenvalues of the p.s.d. matrix $\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}$ lie in the interval $[1-\epsilon, 1+\epsilon]$, so $(1-\epsilon)\mathbf{I} \preceq \bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-\frac{1}{2}} \preceq (1+\epsilon)\mathbf{I}$. Multiplying by $\bar{\mathbf{P}}_\perp^{\frac{1}{2}}$ from both sides, we obtain the desired bound.

Step 2. Then, we carefully design an event E that (i) is provable to occur with high probability and (ii) ensures that the denominators in the following decomposition are bounded away from zero:

$$\begin{aligned}\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1} &= \mathbb{E}[\mathbf{P}] - \gamma\mathbb{E}[\mathbf{P}_\perp]\boldsymbol{\Sigma} = \mathbb{E}[\mathbf{P}\cdot\mathbf{1}_E] + \mathbb{E}[\mathbf{P}\cdot\mathbf{1}_{\neg E}] - \gamma\mathbb{E}[\mathbf{P}_\perp]\boldsymbol{\Sigma} \\ &= \gamma\underbrace{\mathbb{E}\left[(\bar{s}-\hat{s})\cdot\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\cdot\mathbf{1}_E\right]}_{\mathbf{T}_1} - \gamma\underbrace{\mathbb{E}[(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top\cdot\mathbf{1}_{\neg E}]}_{\mathbf{T}_2} \\ &\quad + \gamma\underbrace{\mathbb{E}[\mathbf{P}-\mathbf{P}_{-k}]\boldsymbol{\Sigma}}_{\mathbf{T}_3} + \underbrace{\mathbb{E}[\mathbf{P}\cdot\mathbf{1}_{\neg E}]}_{\mathbf{T}_4},\end{aligned}$$

where we let $\hat{s} = \mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k$ and $\bar{s} = k/\gamma$.

Step 3. It then remains to bound the spectral norms of $\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3, \mathbf{T}_4$ respectively to reach the conclusion. More precisely, the terms $\|\mathbf{T}_2\|$ and $\|\mathbf{T}_4\|$ are proportional to $\Pr(\neg E)$, while the term $\|\mathbf{T}_3\|$ can be bounded using the rank-one update formula for the pseudoinverse (Lemma 4.1 in the appendix). The remaining term $\|\mathbf{T}_1\|$ is more subtle and can be bounded with a careful application of the Hanson-Wright type [RV13] sub-gaussian concentration inequalities (Lemmas 4.2 and 4.3 in the appendix). This allows for a bound on the operator norm $\|\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1}\|$ and hence the conclusion.

Proof of Theorem 4.1

We now discuss how Theorem 4.1 can be obtained from Theorem 4.2. The crucial difference between the statements is that in Theorem 4.1 we let $\mathbf{A}$ be an arbitrary rectangular matrix, whereas in Theorem 4.2 we instead use a square, symmetric and positive semi-definite matrix $\boldsymbol{\Sigma}$. To convert between the two notations, consider the SVD decomposition $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ of $\mathbf{A} \in \mathbb{R}^{m\times n}$ (recall that we assume $m \geq n$), where $\mathbf{U} \in \mathbb{R}^{m\times n}$ and $\mathbf{V} \in \mathbb{R}^{n\times n}$ have orthonormal columns and $\mathbf{D}$ is a diagonal matrix. Now, let $\mathbf{Z} = \mathbf{S}\mathbf{U}$, $\boldsymbol{\Sigma} = \mathbf{D}^2$ and $\mathbf{X} = \mathbf{Z}\boldsymbol{\Sigma}^{\frac{1}{2}} = \mathbf{S}\mathbf{U}\mathbf{D}$. Using the fact that $\mathbf{V}^\top\mathbf{V} = \mathbf{V}\mathbf{V}^\top = \mathbf{I}$, it follows that:

$$\mathbf{I} - (\mathbf{S}\mathbf{A})^\dagger\mathbf{S}\mathbf{A} = \mathbf{V}(\mathbf{I} - \mathbf{X}^\dagger\mathbf{X})\mathbf{V}^\top \quad \text{and} \quad (\gamma\mathbf{A}^\top\mathbf{A} + \mathbf{I})^{-1} = \mathbf{V}(\gamma\boldsymbol{\Sigma} + \mathbf{I})^{-1}\mathbf{V}^\top.$$

Note that since $\|\mathbf{U}\mathbf{v}\| = \|\mathbf{v}\|$, the rows of $\mathbf{Z}$ are sub-gaussian with the same constant as the rows of $\mathbf{S}$. Moreover, using the fact that $\mathbf{B} \preceq \mathbf{C}$ implies $\mathbf{V}\mathbf{B}\mathbf{V}^\top \preceq \mathbf{V}\mathbf{C}\mathbf{V}^\top$ for any p.s.d. matrices $\mathbf{B}$ and $\mathbf{C}$, Theorem 4.1 follows as a corollary of Theorem 4.2.

4.4 Proof of Theorem 4.2

We first introduce the following technical lemmas.

Lemma 4.1 *For $\mathbf{X} \in \mathbb{R}^{k\times n}$ with $k < n$, denote $\mathbf{P} = \mathbf{X}^\dagger\mathbf{X}$ and $\mathbf{P}_{-k} = \mathbf{X}_{-k}^\dagger\mathbf{X}_{-k}$, with $\mathbf{X}_{-i} \in \mathbb{R}^{(k-1)\times n}$ the matrix $\mathbf{X}$ without its i-th row $\mathbf{x}_i \in \mathbb{R}^n$. Then, conditioned on the event $E_k : \left\{\left|\frac{\mathrm{tr}\mathbf{\Sigma}(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k} - 1\right| \le \frac{1}{2}\right\}$:*

$$(\mathbf{X}^\top\mathbf{X})^\dagger\mathbf{x}_k = \frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}, \quad \mathbf{P}-\mathbf{P}_{-k} = \frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}.$$

Proof Since conditioned on E_k we have $\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k \neq 0$, from [Mey73, Theorem 1] we deduce

$$\begin{aligned}(\mathbf{X}^\top\mathbf{X})^\dagger = (\mathbf{A}+\mathbf{x}_k\mathbf{x}_k^\top)^\dagger = \mathbf{A}^\dagger - \frac{\mathbf{A}^\dagger\mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k} - \frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top\mathbf{A}^\dagger}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k} \\ + (1+\mathbf{x}_k^\top\mathbf{A}^\dagger\mathbf{x}_k)\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{(\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k)^2}\end{aligned}$$

for $\mathbf{A} = \mathbf{X}_{-k}^\top\mathbf{X}_{-k}$ so that $\mathbf{I}-\mathbf{P}_{-k} = \mathbf{I}-\mathbf{A}^\dagger\mathbf{A}$, where we used the fact that $\mathbf{I}-\mathbf{P}_{-k}$ is a projection matrix so that $(\mathbf{I}-\mathbf{P}_{-k})^2 = \mathbf{I}-\mathbf{P}_{-k}$. As a consequence, multiplying by $\mathbf{x}_k$ and simplifying we get

$$(\mathbf{X}^\top\mathbf{X})^\dagger\mathbf{x}_k = \frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}.$$

By definition of the pseudoinverse, $\mathbf{P} = \mathbf{X}^\dagger\mathbf{X} = (\mathbf{X}^\top\mathbf{X})^\dagger\mathbf{X}^\top\mathbf{X}$ so that

$$\mathbf{P}-\mathbf{P}_{-k} = \mathbf{X}^\dagger\mathbf{X} - \mathbf{X}_{-k}^\dagger\mathbf{X}_{-k} = \frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}$$

where we used $\mathbf{A}(\mathbf{I}-\mathbf{P}_{-k}) = \mathbf{A}-\mathbf{A}\mathbf{A}^\dagger\mathbf{A} = 0$ and thus the conclusion. ■

Lemma 4.2 *For a K-sub-gaussian random vector $\mathbf{x} \in \mathbb{R}^n$ with $\mathbb{E}[\mathbf{x}] = 0$, $\mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \mathbf{I}_n$ and positive semi-definite matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$, we have*

$$\Pr\left[|\mathbf{x}^\top\mathbf{A}\mathbf{x} - \mathrm{tr}\mathbf{A}| \ge \frac{1}{3}\mathrm{tr}\mathbf{A}\right] \le 2\exp\left(-\min\left\{\frac{r_\mathbf{A}}{9C^2K^4}, \frac{\sqrt{r_\mathbf{A}}}{3CK^2}\right\}\right)$$

with $r_\mathbf{A} = \mathrm{tr}\mathbf{A}/\|\mathbf{A}\|$ the stable rank of $\mathbf{A}$, and

$$\mathbb{E}\left[(\mathbf{x}^\top\mathbf{A}\mathbf{x} - \mathrm{tr}\mathbf{A})^2\right] \le c\ K^4\ \mathrm{tr}\mathbf{A}^2$$

for some $C, c > 0$ independent of K.

Proof This follows from a Hanson-Wright type [RV13] sub-gaussian concentration inequality. More precisely, from [Zaj18, Corollary 2.9] we have, for K-sub-gaussian $\mathbf{x} \in \mathbb{R}^n$ with $\mathbb{E}[\mathbf{x}] = 0$, $\mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \mathbf{I}_n$ and symmetric positive semi-definite $\mathbf{A} \in \mathbb{R}^{n\times n}$ that

$$\Pr\left\{|\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathrm{tr}\mathbf{A}| \geq t\right\} \leq 2\exp\left(-\min\left\{\frac{t^2}{C^2K^4\mathrm{tr}\mathbf{A}^2}, \frac{t}{CK^2\sqrt{\mathrm{tr}\mathbf{A}^2}}\right\}\right)$$

for some universal constant $C > 0$. Taking $t = \frac{1}{3}\mathrm{tr}\mathbf{A}$ we have

$$\frac{t^2}{C^2K^4\mathrm{tr}\mathbf{A}^2} = \frac{(\mathrm{tr}\mathbf{A})^2}{9C^2K^4\mathrm{tr}\mathbf{A}^2} \geq \frac{\mathrm{tr}\mathbf{A}}{9C^2K^4\|\mathbf{A}\|} = \frac{r_{\mathbf{A}}}{9C^2K^4}, \quad \frac{t}{CK^2\sqrt{\mathrm{tr}\mathbf{A}^2}} \geq \frac{\sqrt{r_{\mathbf{A}}}}{3CK^2}$$

where we use the fact that $\mathrm{tr}\mathbf{A}^2 \leq \|\mathbf{A}\|\mathrm{tr}\mathbf{A}$.

Integrating this bound yields:

$$\mathbb{E}\left[(\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathrm{tr}\mathbf{A})^2\right] \leq c\ K^4\ \mathrm{tr}\mathbf{A}^2$$

and thus the conclusion. ■

Lemma 4.3 *With the notations of Lemma 4.1, for* $X = \mathrm{tr}\,\boldsymbol{\Sigma}(\mathbf{P}_{-k} - \mathbb{E}[\mathbf{P}_{-k}])$ *and* $\|\boldsymbol{\Sigma}\| = 1$, *we have*

$$\mathbb{E}[X^2] \leq Ck \quad \textit{and} \quad \Pr\{|X| \geq t\} \leq 2\mathrm{e}^{-\frac{t^2}{ck}}.$$

for some universal constant $C, c > 0$.

Proof To simplify notations, we work on $\mathbf{P}$ instead of $\mathbf{P}_{-k}$, the same line of argument applies to $\mathbf{P}_{-k}$ by changing the sample size k to $k-1$.

First note that

$$\begin{aligned} X &= \mathrm{tr}\boldsymbol{\Sigma}(\mathbf{P} - \mathbb{E}\mathbf{P}) = \mathbb{E}_k[\mathrm{tr}\boldsymbol{\Sigma}\mathbf{P}] - \mathbb{E}_0[\mathrm{tr}\boldsymbol{\Sigma}\mathbf{P}] \\ &= \sum_{i=1}^{k} (\mathbb{E}_i[\mathrm{tr}\boldsymbol{\Sigma}\mathbf{P}] - \mathbb{E}_{i-1}[\mathrm{tr}\boldsymbol{\Sigma}\mathbf{P}]) = \sum_{i=1}^{k} (\mathbb{E}_i - \mathbb{E}_{i-1})\mathrm{tr}\boldsymbol{\Sigma}(\mathbf{P} - \mathbf{P}_{-i}) \end{aligned}$$

where we used the fact that $\mathbb{E}_i[\mathrm{tr}\boldsymbol{\Sigma}\mathbf{P}_{-i}] = \mathbb{E}_{i-1}[\mathrm{tr}\boldsymbol{\Sigma}\mathbf{P}_{-i}]$, for $\mathbb{E}_i[\cdot]$ the conditional expectation with respect to $\mathcal{F}_i$ the σ-field generating the rows $\mathbf{x}_1 \ldots, \mathbf{x}_i$ of $\mathbf{X}$. This forms a martingale difference sequence (it is a difference sequence of the Doob martingale for $\mathrm{tr}\boldsymbol{\Sigma}(\mathbf{P} - \mathbf{P}_{-i})$ with respect to filtration $\mathcal{F}_i$) hence it falls within the scope of the Burkholder inequality [Bur73], recalled as follows.

Lemma 4.4 *For* $\{x_i\}_{i=1}^k$ *a real martingale difference sequence with respect to the increasing* σ *field* $\mathcal{F}_i$, *we have, for* $L > 1$, *there exists* $C_L > 0$ *such that*

$$\mathbb{E}\Big[\Big|\sum_{i=1}^{k} x_i\Big|^L\Big] \leq C_L\mathbb{E}\Big[\Big(\sum_{i=1}^{k} |x_i|^2\Big)^{L/2}\Big].$$

From Lemma 4.1, $\mathbf{P}-\mathbf{P}_{-i}=\frac{(\mathbf{I}-\mathbf{P}_{-i})\mathbf{x}_i\mathbf{x}_i^\top(\mathbf{I}-\mathbf{P}_{-i})}{\mathbf{x}_i^\top(\mathbf{I}-\mathbf{P}_{-i})\mathbf{x}_i}$ is positive semi-definite, we have $\mathrm{tr}\boldsymbol{\Sigma}(\mathbf{P}-\mathbf{P}_{-i})\le\|\boldsymbol{\Sigma}\|=1$ so that with Lemma 4.4 we obtain with $x_i=(\mathbb{E}_i-\mathbb{E}_{i-1})\mathrm{tr}\boldsymbol{\Sigma}(\mathbf{P}-\mathbf{P}_{-i})$ that, for $L>1$

$$\mathbb{E}|X|^L\le C_L k^{L/2}.$$

In particular, for $L=2$, we obtain $\mathbb{E}|X|^2\le Ck$.

For the second result, since we have almost surely bounded martingale differences ($|x_i|\le 2$), by the Azuma-Hoeffding inequality

$$\Pr\{|X|\ge t\}\le 2\mathrm{e}^{\frac{-t^2}{8k}}$$

as desired. ■

Complete proof of Theorem 4.2

Equipped with the lemmas above, we are ready to prove Theorem 4.2. First note that:

1. Since $\mathbf{X}^\dagger\mathbf{X}\overset{d}{=}(\alpha\mathbf{X})^\dagger(\alpha\mathbf{X})$ for any $\alpha\in\mathbb{R}\setminus\{0\}$, we can assume without loss of generality (after rescaling $\bar{\mathbf{P}}_\perp$ correspondingly) that $\|\boldsymbol{\Sigma}\|=1$.

2. According to the definition of $\bar{\mathbf{P}}_\perp$ and γ, the following bounds hold

$$\frac{1}{\gamma+1}\mathbf{I}\preceq\bar{\mathbf{P}}_\perp\preceq\mathbf{I},\quad\gamma\le\frac{k}{r-k}=\frac{1}{\rho-1}\tag{4.6}$$

   for $r\equiv\frac{\mathrm{tr}\boldsymbol{\Sigma}}{\|\boldsymbol{\Sigma}\|}=\mathrm{tr}\boldsymbol{\Sigma}$ and $\rho\equiv\frac{r}{k}>1$, where we used the fact that

$$k=n-\mathrm{tr}\,\bar{\mathbf{P}}_\perp=\mathrm{tr}\,\bar{\mathbf{P}}_\perp(\gamma\boldsymbol{\Sigma}+\mathbf{I})-\mathrm{tr}\,\bar{\mathbf{P}}_\perp=\gamma\mathrm{tr}\,\bar{\mathbf{P}}_\perp\boldsymbol{\Sigma}\ge\frac{\gamma}{\gamma+1}\mathrm{tr}\,\boldsymbol{\Sigma},$$

   so that $r=\mathrm{tr}\boldsymbol{\Sigma}\le k\cdot\frac{\gamma+1}{\gamma}$.

3. As already discussed in Section 4.3, to obtain the lower and upper bound for $\mathbb{E}[\mathbf{P}_\perp]$ in the sense of symmetric matrix as in Theorem 4.2, it suffices to bound the following spectral norm

$$\|\mathbf{I}-\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1}\|\le\frac{C_\rho}{\sqrt{r}},\tag{4.7}$$

   so that, with $\frac{\rho-1}{\rho}\mathbf{I}\preceq\bar{\mathbf{P}}_\perp\preceq\mathbf{I}$ from (4.6), we have

$$\|\mathbf{I}-\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\|=\|\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}(\mathbf{I}-\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1})\bar{\mathbf{P}}_\perp^{\frac{1}{2}}\|\le\frac{C_\rho}{\sqrt{r}}\sqrt{\frac{\rho}{\rho-1}}.$$

Defining $\epsilon = \frac{C_\rho}{\sqrt{r}}\sqrt{\frac{\rho}{\rho-1}}$, this means that all eigenvalues of the p.s.d. matrix $\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-\frac{1}{2}}$ lie in the interval $[1-\epsilon, 1+\epsilon]$, and

$$(1-\epsilon)\mathbf{I} \preceq \bar{\mathbf{P}}_\perp^{-\frac{1}{2}}\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-\frac{1}{2}} \preceq (1+\epsilon)\mathbf{I}.$$

so that by multiplying $\bar{\mathbf{P}}_\perp^{\frac{1}{2}}$ on both sides, we obtain the desired bound.

As a consequence of the above observations, we only need to prove (4.7) under the setting $\|\mathbf{\Sigma}\| = 1$. The proof comes in the following two steps:

1. For $\mathbf{P}_{-i} = \mathbf{X}_{-i}^\dagger\mathbf{X}_{-i}$, with $\mathbf{X}_{-i} \in \mathbb{R}^{(k-1)\times n}$ the matrix $\mathbf{X}$ without its i-th row, we define, for $i \in \{1, \ldots, k\}$, the following events

$$E_i : \left\{ \left| \frac{\mathrm{tr}(\mathbf{I}-\mathbf{P}_{-i})\mathbf{\Sigma}}{\mathbf{x}_i^\top(\mathbf{I}-\mathbf{P}_{-i})\mathbf{x}_i} - 1 \right| \leq \frac{1}{2} \right\}, \tag{4.8}$$

where we recall $\mathbf{x}_i \in \mathbb{R}^n$ is the i-th row of $\mathbf{X}$ so that $\mathbb{E}[\mathbf{x}_i] = 0$ and $\mathbb{E}[\mathbf{x}_i\mathbf{x}_i^\top] = \mathbf{\Sigma}$. With Lemma 4.2, we can bound the probability of $\neg E_i$, and consequently that of $\neg E$ for $E = \bigwedge_{i=1}^k E_i$;

2. We then bound, conditioned on E and $\neg E$ respectively, the spectral norm $\|\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1}\|$. More precisely, since

$$\begin{aligned}
\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1} &= \mathbb{E}[\mathbf{P}] - \gamma\mathbb{E}[\mathbf{P}_\perp]\mathbf{\Sigma} \\
&= \mathbb{E}[\mathbf{P}\cdot\mathbf{1}_E] + \mathbb{E}[\mathbf{P}\cdot\mathbf{1}_{\neg E}] - \gamma\mathbb{E}[\mathbf{P}_\perp]\mathbf{\Sigma} \\
&= k\,\mathbb{E}\left[\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\cdot\mathbf{1}_E\right] - \gamma\mathbb{E}[\mathbf{P}_\perp]\mathbf{\Sigma} \;+\; \mathbb{E}[\mathbf{P}\cdot\mathbf{1}_{\neg E}] \\
&= \gamma\,\underbrace{\mathbb{E}\left[(\bar{s}-\hat{s})\cdot\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\cdot\mathbf{1}_E\right]}_{\mathbf{T}_1} - \gamma\underbrace{\mathbb{E}[(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top\cdot\mathbf{1}_{\neg E}]}_{\mathbf{T}_2} \\
&\quad + \gamma\underbrace{\mathbb{E}[\mathbf{P}-\mathbf{P}_{-k}]\mathbf{\Sigma}}_{\mathbf{T}_3} + \underbrace{\mathbb{E}[\mathbf{P}\cdot\mathbf{1}_{\neg E}]}_{\mathbf{T}_4},
\end{aligned}$$

where we used Lemma 4.1 for the third equality and denote $\hat{s} = \mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k$ as well as $\bar{s} = \mathrm{tr}\bar{\mathbf{P}}_\perp\mathbf{\Sigma} = k/\gamma$. It then remains to bound the spectral norms of $\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3, \mathbf{T}_4$ to reach the conclusion.

Another important relation that will be constantly used throughout the proof is

$$\mathrm{tr}(\mathbf{I}-\mathbf{P}_{-k})\mathbf{\Sigma} = \mathrm{tr}\mathbf{\Sigma}^{\frac{1}{2}}(\mathbf{I}-\mathbf{P}_{-k})^2\mathbf{\Sigma}^{\frac{1}{2}} = \|\mathbf{\Sigma}^{\frac{1}{2}} - \mathbf{\Sigma}^{\frac{1}{2}}\mathbf{X}_{-k}^\dagger\mathbf{X}_{-k}\|_F^2 \geq \sum_{i\geq k}\lambda_i(\mathbf{\Sigma}) \geq r-k \tag{4.9}$$

where we used the fact that $\mathrm{rank}(\mathbf{X}_{-k}^{\dagger}\mathbf{X}_{-k}) \leq \mathrm{rank}(\mathbf{X}_{-k}) \leq k-1$ and arranged the eigenvalues $1 = \lambda_1(\mathbf{\Sigma}) \geq \ldots \geq \lambda_n(\mathbf{\Sigma})$ in a non-increasing order. As a consequence, we also have

$$\frac{\mathrm{tr}(\mathbf{I}-\mathbf{P}_{-k})\mathbf{\Sigma}}{\|(\mathbf{I}-\mathbf{P}_{-k})\mathbf{\Sigma}\|} \geq \mathrm{tr}(\mathbf{I}-\mathbf{P}_{-k})\mathbf{\Sigma} \geq r-k. \tag{4.10}$$

For the first step, we have, with Lemma 4.2 and (4.10) that

$$\begin{aligned}\Pr(\neg E_i) &\leq \Pr\left\{|\mathbf{x}_i^{\top}(\mathbf{I}-\mathbf{P}_{-i})\mathbf{x}_i - \mathrm{tr}\mathbf{\Sigma}(\mathbf{I}-\mathbf{P}_{-i})| \geq \frac{1}{3}\mathrm{tr}\mathbf{\Sigma}(\mathbf{I}-\mathbf{P}_{-i})\right\} \\ &\leq 2\mathrm{e}^{-\min\left\{\frac{r-k}{9C^2K^4}, \frac{\sqrt{r-k}}{3CK^2}\right\}}.\end{aligned}$$

so that with the union bound we obtain

$$\Pr(\neg E) \leq 2k\mathrm{e}^{-\min\left\{\frac{r-k}{9C^2K^4}, \frac{\sqrt{r-k}}{3CK^2}\right\}} \leq \frac{k}{(r-k)^2}\cdot 2(r-k)^2\mathrm{e}^{-\min\left\{\frac{r-k}{9C^2K^4}, \frac{\sqrt{r-k}}{3CK^2}\right\}} \leq \frac{C_\rho}{r-k} \tag{4.11}$$

where we used the fact that, for $\alpha > 0$, $x^2\mathrm{e}^{-\alpha x} \leq \frac{4\mathrm{e}^{-2}}{\alpha^2}$ and $x^4\mathrm{e}^{-\alpha x} \leq \frac{256\mathrm{e}^{-4}}{\alpha^4}$ on $x > 0$. Also, denote $c_\rho = \frac{r-k}{r} = \frac{\rho-1}{\rho} > 0$, we have

$$\Pr(\neg E) \leq \frac{C_\rho}{r-k} = \frac{C_\rho}{c_\rho r} = \frac{C'_\rho}{r} \tag{4.12}$$

for some $C'_\rho > 0$ that depends on $\rho = r/k > 1$ and the sub-gaussian norm K.

At this point, note that, conditioned on the event E, we have for $i \in \{1, \ldots, k\}$

$$\frac{1}{2}\frac{1}{\mathrm{tr}(\mathbf{I}-\mathbf{P}_{-i})\mathbf{\Sigma}} \leq \frac{1}{\mathbf{x}_i^{\top}(\mathbf{I}-\mathbf{P}_{-i})\mathbf{x}_i} \leq \frac{3}{2}\frac{1}{\mathrm{tr}(\mathbf{I}-\mathbf{P}_{-i})\mathbf{\Sigma}}, \tag{4.13}$$

Also, with (4.12) and the fact that $\|\mathbf{P}\| \leq 1$, we have $\|\mathbf{T}_4\| \leq \frac{C_\rho}{r}$ for some $C_\rho > 0$ that depends on ρ and K. To handle non-symmetric matrix $\mathbf{T}_2$, note that $\mathbf{T}_2 + \mathbf{T}_2^{\top}$ is symmetric and

$$-\mathbb{E}[(\mathbf{I}-\mathbf{P}_{-k})\cdot\mathbf{1}_{\neg E}] - \mathbb{E}[(\mathbf{x}_k^{\top}\mathbf{x}_k)\mathbf{x}_k\mathbf{x}_k^{\top}\cdot\mathbf{1}_{\neg E}] \preceq \mathbf{T}_2 + \mathbf{T}_2^{\top} \preceq \mathbb{E}[(\mathbf{I}-\mathbf{P}_{-k})\cdot\mathbf{1}_{\neg E}] + \mathbb{E}[(\mathbf{x}_k^{\top}\mathbf{x}_k)\mathbf{x}_k\mathbf{x}_k^{\top}\cdot\mathbf{1}_{\neg E}] \tag{4.14}$$

with $-(\mathbf{A}\mathbf{A}^{\top} + \mathbf{B}\mathbf{B}^{\top}) \preceq \mathbf{A}\mathbf{B}^{\top} + \mathbf{B}\mathbf{A}^{\top} \preceq \mathbf{A}\mathbf{A}^{\top} + \mathbf{B}\mathbf{B}^{\top}$. To obtain an upper bound for operator norm of $\mathbb{E}[(\mathbf{x}_k^{\top}\mathbf{x}_k)\mathbf{x}_k\mathbf{x}_k^{\top}\cdot\mathbf{1}_{\neg E}]$, note that

$$\begin{aligned}\|\mathbb{E}[(\mathbf{x}_k^{\top}\mathbf{x}_k)\mathbf{x}_k\mathbf{x}_k^{\top}\cdot\mathbf{1}_{\neg E}]\| &\leq \mathbb{E}[(\mathbf{x}_k^{\top}\mathbf{x}_k)^2\cdot\mathbf{1}_{\neg E}] = \int_0^{\infty}\Pr(\mathbf{x}^{\top}\mathbf{x}\cdot\mathbf{1}_{\neg E} \geq \sqrt{t})dt \\ &\leq \int_0^{X}\Pr(\mathbf{x}^{\top}\mathbf{x}\cdot\mathbf{1}_{\neg E} \geq \sqrt{t})dt + \int_X^{\infty}\Pr(\mathbf{x}^{\top}\mathbf{x} \geq \sqrt{t})dt \\ &\leq X\cdot\Pr(\neg E) + \int_X^{\infty}\mathrm{e}^{-\min\left\{\frac{t}{C^2K^4r}, \frac{\sqrt{t}}{CK^2\sqrt{r}}\right\}}dt \leq \frac{c_\rho}{r}\end{aligned}$$

where we recall $\mathbb{E}[\mathbf{x}^\top\mathbf{x}] = \mathrm{tr}\boldsymbol{\Sigma} = r$ and take $X \geq C^2K^4r$, the third line follows from the proof of Lemma 4.2 and the forth line from the same argument as in (4.11). Moreover, since $\|\mathbf{T}_2\| \leq \|\mathbf{T}_2 + \mathbf{T}_2^\top\|$ (see for example [Ser10, Proposition 5.11]), we conclude that $\|\mathbf{T}_2\| \leq \frac{C_\rho}{r}$.

And it thus remains to handle the terms $\mathbf{T}_1$ and $\mathbf{T}_3$ to obtain a bound on $\|\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1}\|$.

To bound $\mathbf{T}_3$, with $\mathbf{P} - \mathbf{P}_{-k} = \frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}$ in Lemma 4.1, we have

$$\begin{aligned}\|\mathbf{T}_3\| &\leq \left\|\mathbb{E}\left[\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\cdot\mathbf{1}_E\right]\right\| + \|\mathbb{E}[(\mathbf{P}-\mathbf{P}_{-k})\cdot\mathbf{1}_{\neg E}]\| \\ &\leq \frac{3}{2}\mathbb{E}\left[\frac{1}{\mathrm{tr}(\mathbf{I}-\mathbf{P}_{-k})\boldsymbol{\Sigma}}\right] + \frac{c_\rho}{r-k} \leq \frac{C_\rho}{r-k} = \frac{C'_\rho}{r}\end{aligned}$$

where we used the fact that $\mathrm{tr}\,(\mathbf{I}-\mathbf{P}_{-k})\boldsymbol{\Sigma} \geq r-k$ from (4.9) and recall $\rho \equiv r/k > 1$.

For $\mathbf{T}_1$ we write

$$\begin{aligned}\|\mathbf{T}_1\| &\leq \mathbb{E}\left[\|\mathbf{I}-\mathbf{P}_{-k}\|\cdot\left\|\mathbb{E}\Big[|\bar{s}-\hat{s}|\cdot\frac{\mathbf{x}_k\mathbf{x}_k^\top}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\cdot\mathbf{1}_E \mid \mathbf{P}_{-k}\Big]\right\|\right] \\ &\leq \frac{3}{2}\frac{1}{r-k}\cdot\mathbb{E}\left[\sup_{\|\mathbf{v}\|=1}\mathbb{E}\Big[|\bar{s}-\hat{s}|\cdot\mathbf{v}^\top\mathbf{x}_k\mathbf{x}_k^\top\mathbf{v}\cdot\mathbf{1}_E \mid \mathbf{P}_{-k}\Big]\right] \\ &\leq \frac{C_\rho}{r}\cdot\mathbb{E}\Bigg[\underbrace{\sqrt{\mathbb{E}\big[(\bar{s}-\hat{s})^2\cdot\mathbf{1}_E \mid \mathbf{P}_{-k}\big]}}_{T_{1,1}}\cdot\underbrace{\sup_{\|\mathbf{v}\|=1}\sqrt{\mathbb{E}\big[(\mathbf{v}^\top\mathbf{x}_k)^4\big]}}_{T_{1,2}}\Bigg]\end{aligned}$$

where we used Jensen's inequality for the first inequality, the relation in (4.9) for the second inequality, and Cauchy–Schwarz for the third inequality.

We first bound $T_{1,2}$ by definition of sub-gaussian random vectors. We have for $\mathbf{x}_k$ a K-sub-gaussian and $\|\mathbf{v}\| = 1$ that, $\mathbf{v}^\top\mathbf{x}_k$ is a sub-gaussian random variable with $\|\mathbf{v}^\top\mathbf{a}\|_{\psi_2} \leq K$. As such, $T_{1,2} \leq CK^2$ for some absolute constant $C > 0$, see for example [Ver18, Section 2.5.2].

For $T_{1,1}$ we have

$$\sqrt{\mathbb{E}[(\bar{s}-\hat{s})^2\cdot\mathbf{1}_E \mid \mathbf{P}_{-k}]} = \sqrt{(\bar{s}-s)^2 + \mathbb{E}[(s-\hat{s})^2\cdot\mathbf{1}_E]}$$

where we denote $s = \mathbb{E}[\hat{s}] = \mathrm{tr}\,\mathbb{E}[\mathbf{I}-\mathbf{P}_{-k}]\boldsymbol{\Sigma}$. Note that

$$\begin{aligned}\mathbb{E}\big[(s-\hat{s})^2\big] &= \mathbb{E}\big[\big(\mathrm{tr}\,\boldsymbol{\Sigma}(\mathbf{P}_{-k}-\mathbb{E}[\mathbf{P}_{-k}])\big)^2\big] + \mathbb{E}\big[(\mathrm{tr}\,(\mathbf{I}-\mathbf{P}_{-k})\boldsymbol{\Sigma} - \mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k)^2\big] \\ &\leq C_1k + C_2\mathbb{E}\big[\mathrm{tr}\,(\boldsymbol{\Sigma}-\mathbf{P}_{-k}\boldsymbol{\Sigma})^2\big] \leq C(k+s) \leq C\big(k+\bar{s}+|s-\bar{s}|\big)\end{aligned}$$

where we used Lemma 4.3 and Lemma 4.2. Recall that $\bar{s} = \mathrm{tr}\bar{\mathbf{P}}_\perp\boldsymbol{\Sigma} \leq \mathrm{tr}\boldsymbol{\Sigma} = r$ and $k < r$, we have

$$T_{1,1} \leq \sqrt{(\bar{s}-s)^2 + C(|\bar{s}-s| + 2r)} \tag{4.15}$$

It remains to bound $|\bar{s} - s|$. Note that $\mathbf{P} = (\mathbf{X}^\top\mathbf{X})^\dagger\mathbf{X}^\top\mathbf{X} = \mathbf{X}^\top\mathbf{X}(\mathbf{X}^\top\mathbf{X})^\dagger$ and is symmetric, so

$$\begin{aligned}
&\mathbf{I} - \mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1} + \mathbf{I} - \bar{\mathbf{P}}_\perp^{-1}\mathbb{E}[\mathbf{P}_\perp] = 2\mathbb{E}[\mathbf{P}] - \mathbb{E}[\gamma\mathbf{P}_\perp\boldsymbol{\Sigma}] - \mathbb{E}[\gamma\boldsymbol{\Sigma}\mathbf{P}_\perp] \\
&= \sum_{i=1}^{k} \mathbb{E}\big[(\mathbf{X}^\top\mathbf{X})^\dagger\mathbf{x}_i\mathbf{x}_i^\top + \mathbf{x}_i\mathbf{x}_i^\top(\mathbf{X}^\top\mathbf{X})^\dagger\big] - \gamma(\mathbb{E}[\mathbf{P}_\perp]\boldsymbol{\Sigma} + \boldsymbol{\Sigma}\mathbb{E}[\mathbf{P}_\perp]) \\
&= \gamma\,\mathbb{E}\left[\bar{s}\cdot\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top + \mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\right] - \gamma\,\mathbb{E}\left[\hat{s}\cdot\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top + \mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\right] \\
&+ \gamma\,(\mathbb{E}[(\mathbf{I}-\mathbf{P}_{-k})\boldsymbol{\Sigma}] + \mathbb{E}[\boldsymbol{\Sigma}(\mathbf{I}-\mathbf{P}_{-k})]) - \gamma(\mathbb{E}[\mathbf{P}_\perp]\boldsymbol{\Sigma} + \boldsymbol{\Sigma}\mathbb{E}[\mathbf{P}_\perp]) \\
&= \gamma\,\mathbb{E}\left[(\bar{s}-\hat{s})\cdot\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top + \mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\right] + \gamma(\mathbb{E}[\mathbf{P}-\mathbf{P}_{-k}]\boldsymbol{\Sigma} + \boldsymbol{\Sigma}\mathbb{E}[\mathbf{P}-\mathbf{P}_{-k}]).
\end{aligned}$$

Moreover, using the fact that $\bar{\mathbf{P}}_\perp\boldsymbol{\Sigma} \preceq \frac{1}{\gamma+1}\mathbf{I}$ and $\bar{\mathbf{P}}_\perp\boldsymbol{\Sigma} = \boldsymbol{\Sigma}\bar{\mathbf{P}}_\perp$, we obtain that

$$\begin{aligned}
&|\bar{s} - s| = |\mathrm{tr}(\bar{\mathbf{P}}_\perp - \mathbb{E}[\mathbf{I}-\mathbf{P}_{-k}])\boldsymbol{\Sigma}| \le |\mathrm{tr}(\bar{\mathbf{P}}_\perp - \mathbb{E}[\mathbf{P}_\perp])\boldsymbol{\Sigma}| + |\mathrm{tr}\mathbb{E}[\mathbf{P}-\mathbf{P}_{-k}]\boldsymbol{\Sigma}| \\
&= \frac{1}{2}\big|\mathrm{tr}\,(\mathbf{I}-\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1})\bar{\mathbf{P}}_\perp\boldsymbol{\Sigma} + \mathrm{tr}\,\bar{\mathbf{P}}_\perp(\mathbf{I}-\bar{\mathbf{P}}_\perp^{-1}\mathbb{E}[\mathbf{P}_\perp])\boldsymbol{\Sigma}\big| + \mathrm{tr}\,\mathbb{E}\left[\frac{(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\right]\boldsymbol{\Sigma} \\
&\le \frac{1}{2}\big|\mathrm{tr}\,(\mathbf{I}-\mathbb{E}[\mathbf{P}_\perp]\bar{\mathbf{P}}_\perp^{-1} + \mathbf{I} - \bar{\mathbf{P}}_\perp^{-1}\mathbb{E}[\mathbf{P}_\perp])\bar{\mathbf{P}}_\perp\boldsymbol{\Sigma}\big| + 1 \\
&\le \frac{\gamma}{2}\,\mathbb{E}\left[|\bar{s}-\hat{s}|\cdot\frac{\mathrm{tr}\,((\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top + \mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k}))\bar{\mathbf{P}}_\perp\boldsymbol{\Sigma}}{\mathrm{tr}\,(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top}\right] \\
&+ \gamma\,\mathbb{E}\left[\frac{\mathrm{tr}\,(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\bar{\mathbf{P}}_\perp\boldsymbol{\Sigma}}{\mathrm{tr}\,(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k\mathbf{x}_k^\top}\right] + 1 \\
&\le \frac{\gamma}{\gamma+1}\left(\mathbb{E}\left[|\bar{s}-\hat{s}|\cdot\frac{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}{\mathbf{x}_k^\top(\mathbf{I}-\mathbf{P}_{-k})\mathbf{x}_k}\right] + 1\right) + 1 \le \frac{\gamma}{\gamma+1}\Big(|\bar{s}-s| + \mathbb{E}\big[|s-\hat{s}|\big] + 1\Big) + 1 \\
&\le \frac{\gamma}{\gamma+1}\Big(|\bar{s}-s| + C\sqrt{|\bar{s}-s|} + C\sqrt{2r} + 1\Big) + 1.
\end{aligned}$$

Solving for $|\bar{s} - s|$, we deduce that

$$|\bar{s} - s| \le C_1\sqrt{r} + C_2,$$

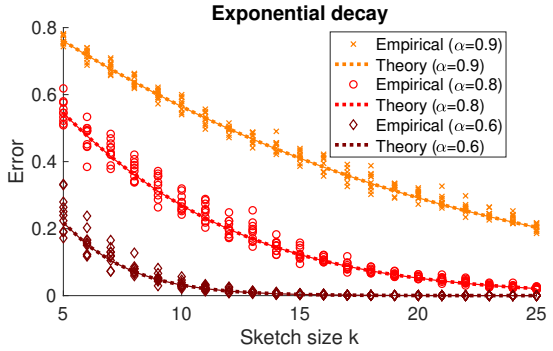
so plugging back to (4.15) we get $T_{1,1} \le C\sqrt{r}$ and $\|\mathbf{T}_1\| \le \frac{C_\rho}{\sqrt{r}}$, thus completing the proof.

4.5 Explicit formulas under known spectral decay

The expression we give for the expected residual projection, $\mathbb{E}[\mathbf{P}_\perp] \simeq (\gamma\mathbf{A}^\top\mathbf{A} + \mathbf{I})^{-1}$, is implicit in that it depends on the parameter γ which is the solution of the following equation:

$$\sum_{i\ge 1}\frac{\gamma\sigma_i^2}{\gamma\sigma_i^2 + 1} = k, \qquad \text{where } \sigma_i \text{ are the singular values of } \mathbf{A}. \tag{4.16}$$

(a) Singular values are given by $\sigma_i^2 = C \cdot \alpha^{i-1}$.



(b) Singular values are given by $\sigma_i^2 = C \cdot i^{-\beta}$.

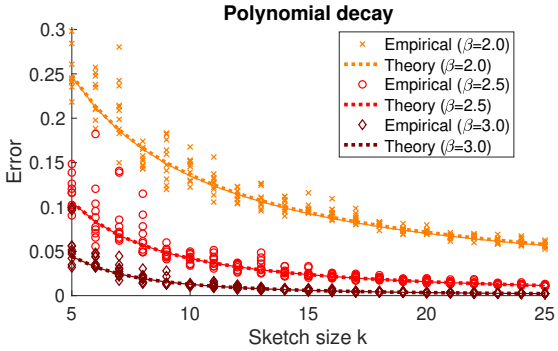


Figure 4.1: Theoretical predictions of low-rank approximation error of a Gaussian sketch under known spectral decays, compared to the empirical results. The constant C is scaled so that $\|\mathbf{A}\|_F^2 = 1$ and we let $n = m = 1000$. For the theory, we plot the explicit formulas (4.17) and (4.18) (dashed lines), as well as the implicit expression from Corollary 4.1 (thin solid lines) obtained by numerically solving (4.16). Observe that the explicit and implicit predictions are nearly (but not exactly) identical.

In general, it is impossible to solve this equation analytically, i.e., to write γ as an explicit formula of n, k and the singular values of $\mathbf{A}$. However, we show that when the singular values exhibit a known rate of decay, then it is possible to obtain explicit formulas for γ. In particular, this allows us to provide precise and easily interpretable rates of decay for the low-rank approximation error of a sub-gaussian sketch.

Matrices that have known spectral decay, most commonly with either exponential or polynomial rate, arise in many machine learning problems [MDK20]. Such behavior can be naturally occurring in data, or it can be induced by feature expansion using, say, the RBF kernel (for exponential decay) or Matérn (for polynomial decay) kernels [San+97; RW06]. Understanding these two classes of decay plays an important role in distinguishing the properties of light-tailed and heavy-tailed data distributions. Note that in the kernel setting we may often represent our data via the $m \times m$ kernel matrix $\mathbf{K}$, instead of the $m \times n$ data matrix $\mathbf{A}$, and study the sketched Nyström method [GM16] for low-rank approximation. To handle the kernel setting in our analysis, it suffices to replace the squared singular values σ_i^2 of $\mathbf{A}$ with the eigenvalues of $\mathbf{K}$.

Exponential spectral decay

Suppose that the squared singular values of $\mathbf{A}$ exhibit exponential decay, i.e. $\sigma_i^2 = C \cdot \alpha^{i-1}$, where C is a constant and $\alpha \in (0, 1)$. For simplicity of presentation, we will let $m, n \to \infty$. Under this spectral decay, we can approximate the sum in (4.16) by the analytically computable integral $\int_y^\infty \frac{1}{1+(C\gamma)^{-1}\alpha^{-x}} dx$, obtaining $\gamma \approx (\alpha^{-k} - 1)\sqrt{\alpha}/C$. Applying this to the

formula from Corollary 4.1, we can express the low-rank approximation error for a sketch of size k as follows:

$$\mathbb{E}\big[\|\mathbf{A}-\mathbf{AP}\|_F^2\big] \approx \frac{C}{\sqrt{\alpha}} \cdot \frac{k}{\alpha^{-k}-1}, \quad \text{when} \quad \sigma_i^2 = C \cdot \alpha^{i-1} \ \text{ for all } i. \tag{4.17}$$

In Figure 4.1a, we plot the above formula against the numerically obtained implicit expression from Corollary 4.1, as well as empirical results for a Gaussian sketch. First, we observe that the theoretical predictions closely align with empirical values even after the sketch size crosses the stable rank $r \approx \frac{1}{1-\alpha}$, suggesting that Theorem 4.1 can be extended to this regime. Second, while it is not surprising that the error decays at a similar rate as the singular values, our predictions offer a much more precise description, down to lower order effects and even constant factors. For instance, we observe that the error (normalized by $\|\mathbf{A}\|_F^2$, as in the figure) only starts decaying exponentially after k crosses the stable rank, and until that point it decreases at a linear rate with slope $-\frac{1-\alpha}{2\sqrt{\alpha}}$.

Polynomial spectral decay

We now turn to polynomial spectral decay, which is a natural model for analyzing heavy-tailed data distributions. Let $\mathbf{A}$ have squared singular values $\sigma_i^2 = C \cdot i^{-\beta}$ for some $\beta \geq 2$, and let $m, n \to \infty$. As in the case of exponential decay, we use the integral $\int_y^\infty \frac{1}{1+(C\gamma)^{-1}x^{-\beta}}dx$ to approximate the sum in (4.16), and solve for γ, obtaining $\gamma \approx \big((k+\frac{1}{2})\frac{\beta}{\pi}\sin(\frac{\pi}{\beta})\big)^\beta$. Combining this with Corollary 4.1 we get:

$$\mathbb{E}\big[\|\mathbf{A}-\mathbf{AP}\|_F^2\big] \approx C \cdot \frac{k}{(k+\frac{1}{2})^\beta}\left(\frac{\pi/\beta}{\sin(\pi/\beta)}\right)^\beta, \quad \text{when} \quad \sigma_i^2 = C \cdot i^{-\beta} \ \text{ for all } i. \tag{4.18}$$

Figure 4.1b compares our predictions to the empirical results for several values of β. In all of these cases, the stable rank is close to 1, and yet the theoretical predictions align very well with the empirical results. Overall, the asymptotic rate of decay of the error is $k^{1-\beta}$. However it is easy to verify that the lower order effect of $(k+\frac{1}{2})^\beta$ appearing instead of k^β in (4.18) significantly changes the trajectory for small values of k. Also, note that as β grows large, the constant $\big(\frac{\pi/\beta}{\sin(\pi/\beta)}\big)^\beta$ goes to 1, but it plays a significant role for $\beta = 2$ or 3 (roughly, scaling the expression by a factor of 2). Finally, we remark that for $\beta \in (1,2)$, our integral approximation of (4.16) becomes less accurate. We expect that a corrected expression is possible, but likely more complicated and less interpretable.

4.6 Empirical results

In this section, we numerically verify the accuracy of our theoretical predictions for the low-rank approximation error of sketching on benchmark datasets from the libsvm repository

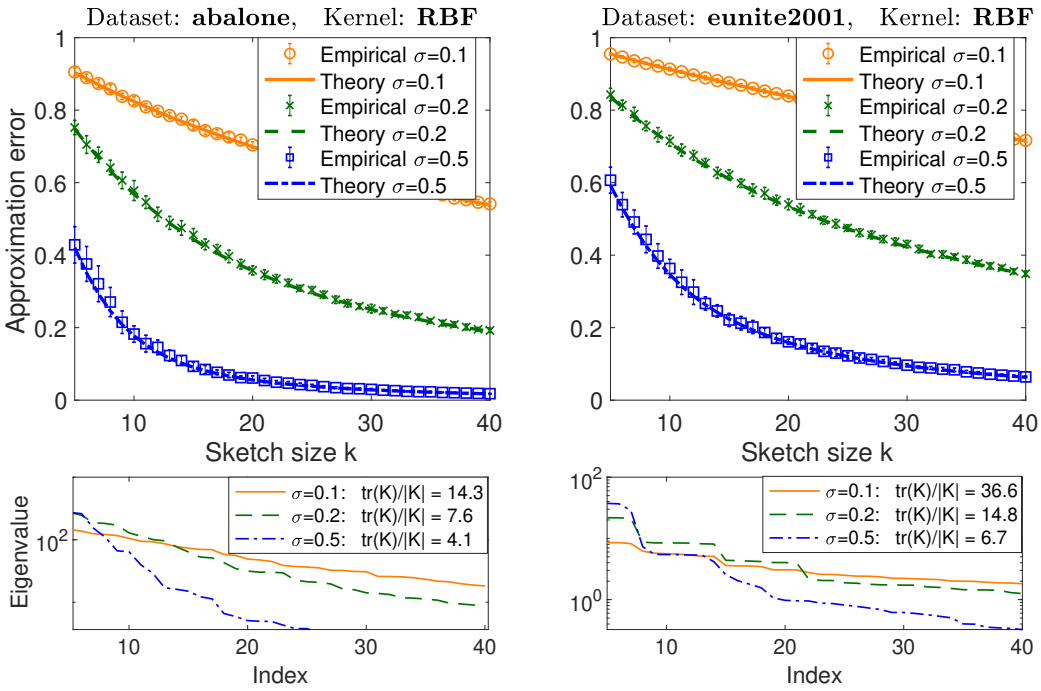


Figure 4.2: Theoretical predictions versus approximation error for the sketched Nyström with the RBF kernel (spectral decay shown at the bottom).

[CL11] (further numerical results are in Appendix 4.6). We repeated every experiment 10 times, and plot both the average and standard deviation of the results. We use the following $k \times m$ sketching matrices $\mathbf{S}$:

1. *Gaussian sketch:* with i.i.d. standard normal entries;

2. *Rademacher sketch:* with i.i.d. entries equal 1 with probability 0.5 and -1 otherwise.

Varying spectral decay. To demonstrate the role of spectral decay and the stable rank on the approximation error, we performed feature expansion using the radial basis function (RBF) kernel $k(\mathbf{a}_i, \mathbf{a}_j) = \exp(-\|\mathbf{a}_i - \mathbf{a}_j\|^2/(2\sigma^2))$, obtaining an $m \times m$ kernel matrix $\mathbf{K}$. We used the sketched Nyström method to construct a low-rank approximation $\tilde{\mathbf{K}} = \mathbf{K}\mathbf{S}^\top(\mathbf{S}\mathbf{K}\mathbf{S}^\top)^\dagger\mathbf{S}\mathbf{K}$, and computed the normalized trace norm error $\|\mathbf{K} - \tilde{\mathbf{K}}\|_*/\|\mathbf{K}\|_*$. The theoretical predictions are coming from (4.2), which in turn uses Theorem 4.1. Following [GM16], we use the RBF kernel because varying the scale parameter σ allows us to observe the approximation error under qualitatively different spectral decay profiles of the kernel. In Figure 4.2, we present the results for the Gaussian sketch on two datasets, with three values of σ, and in all cases our theory aligns with the empirical results. Furthermore, as smaller σ leads to slower spectral decay and larger stable rank, it also makes the approximation error decay more linearly for

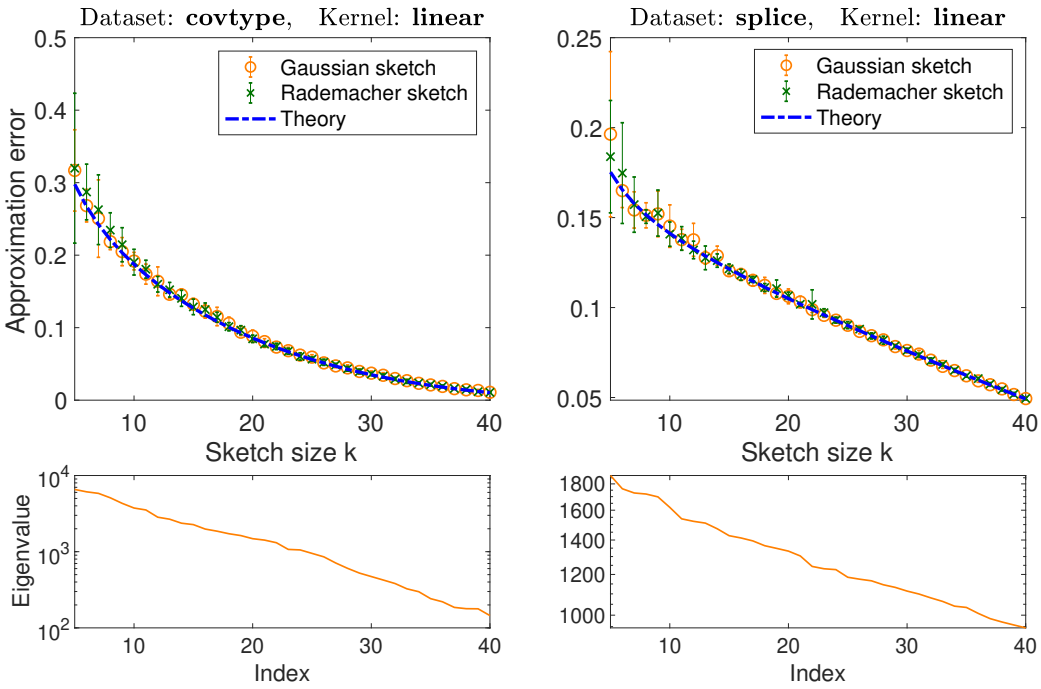


Figure 4.3: Theoretical predictions versus approximation error for the Gaussian and Rademacher sketches (spectral decay shown at the bottom).

small sketch sizes. This behavior is predicted by our explicit expressions (4.17) for the error under exponential spectral decay from Section 4.5. Once the sketch sizes are sufficiently larger than the stable rank of $\mathbf{K}^{\frac{1}{2}}$, the error starts decaying at an exponential rate. Note that Theorem 4.1 only guarantees accuracy of our expressions for sketch sizes below the stable rank, however the predictions are accurate regardless of this constraint.

Varying sketch type. In the next set of empirical results, we compare the performance of Gaussian and Rademacher sketches, and also verify the theory when sketching the data matrix $\mathbf{A}$ without kernel expansion, plotting $\|\mathbf{A} - \mathbf{A}(\mathbf{SA})^{\dagger}\mathbf{SA}\|_F^2/\|\mathbf{A}\|_F^2$. Since both of the sketching methods have sub-gaussian entries, Corollary 4.1 predicts that they should have comparable performance in this task and match our expressions. This is exactly what we observe in Figure 4.3 for two datasets and a range of sketching sizes, as well as in other empirical results shown in Section 4.6.

Additional empirical results on libsvm datasets

We complement the results of Section 4.6 with empirical results on four additional libsvm datasets [CL11] (bringing the total number of benchmark datasets to eight), which further establish the accuracy of our surrogate expressions for the low-rank approximation error.

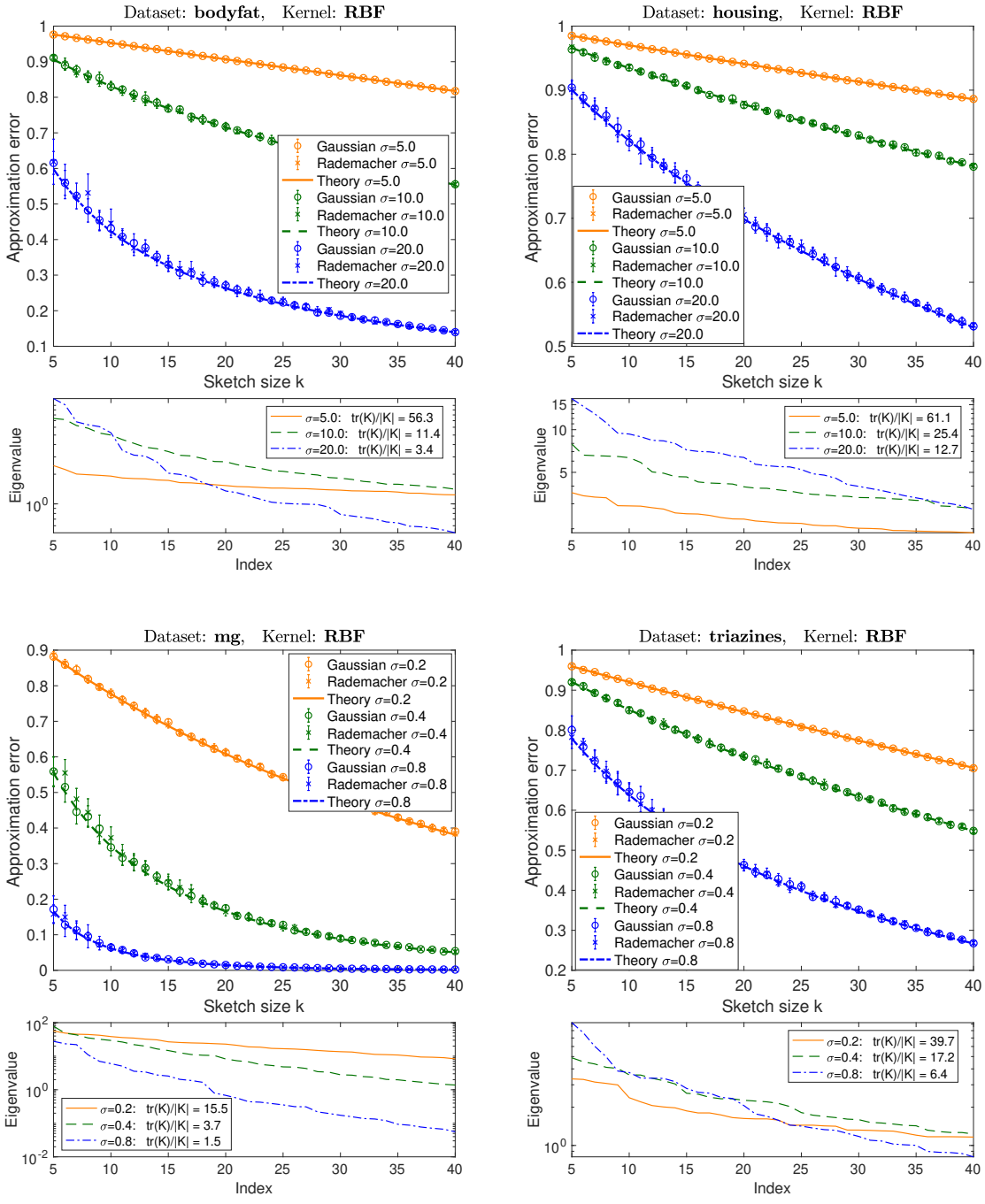


Figure 4.4: Theoretical predictions versus approximation error for the sketched Nyström with the RBF kernel, using Gaussian and Rademacher sketches (spectral decay shown at the bottom).

Similary as in Figure 4.2, we use the sketched Nyström method [GM16] with the RBF kernel $k(\mathbf{a}_i, \mathbf{a}_j) = \exp(-\|\mathbf{a}_i - \mathbf{a}_j\|^2/(2\sigma^2))$, for several values of the parameter σ. The values of σ were chosen so as to demonstrate the effectiveness of our theoretical predictions both when the stable rank is moderately large and when it is very small.

In Figure 4.4 we show the results for both Gaussian and Rademacher sketches. These results reinforce the conclusions we made in Section 4.6: our theoretical estimates are very accurate in all cases, for both sketching methods, and even when the stable rank is close to 1 (a regime that is not supported by the current theory).

4.7 Conclusions

We derived the first theoretically supported precise expressions for the expected residual projection matrix, which is a central component in the analysis of RandNLA dimensionality reduction via sketching. Our analysis provides a new understanding of low-rank approximation, the Nyström method, and the convergence properties of many randomized iterative algorithms. As a direction for future work, we conjecture that our main result can be extended to sketch sizes larger than the stable rank of the data matrix.

Chapter 5

Accelerating Metropolis-hastings with lightweight inference compilation

While the subsequent chapters depart from a previously common theme of DPPs, they continue our study of statistical applications of randomized methods. In this chapter, we are concerned with the problem of sampling intractable posteriors of Bayesian graphical models. In order to construct accurate proposers for Metropolis-Hastings Markov Chain Monte Carlo, we integrate ideas from probabilistic graphical models and neural networks in a framework we call Lightweight Inference Compilation (LIC). LIC implements amortized inference within an open-universe declarative probabilistic programming language (PPL). Graph neural networks are used to parameterize proposal distributions as functions of Markov blankets, which during "compilation" are optimized to approximate single-site Gibbs sampling distributions. Unlike prior work in inference compilation (IC), LIC forgoes importance sampling of linear execution traces in favor of operating directly on Bayesian networks. Through using a declarative PPL, the Markov blankets of nodes (which may be non-static) are queried at inference-time to produce proposers. Experimental results show LIC can produce proposers which have less parameters, greater robustness to nuisance random variables, and improved posterior sampling in a Bayesian logistic regression and n-schools inference application. Parts of this chapter were originally published in Feynman Liang, Nimar Arora, Nazanin Tehrani, Yucen Li, Michael Tingley, and Erik Meijer. "Accelerating Metropolis-Hastings with Lightweight Inference Compilation". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 181–189.

5.1 Background

Deriving and implementing samplers has traditionally been a high-effort and application-specific endeavour [Por+08; MAM10], motivating the development of general-purpose probabilistic programming languages (PPLs) where a non-expert can specify a generative model (i.e. joint distribution) $p(\boldsymbol{x}, \boldsymbol{y})$ and the software automatically performs inference to sample latent

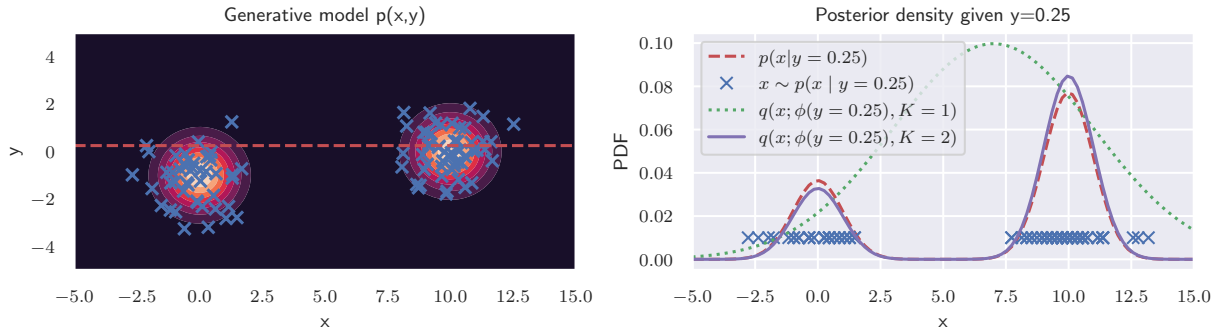


Figure 5.1: Intuition for Lightweight Inference Compilation (LIC). LIC uses samples $(x_i, y_i) \overset{\text{iid}}{\sim} p$ (blue "x" in left) drawn from the joint density $p(x, y)$ to approximate the expected inclusive KL-divergence $\mathbb{E}_{p(y)} D_{\text{KL}}(p(x \mid y) \parallel q(x; \phi(y)))$ between the posterior $p(x \mid y)$ and the LIC proposal distribution $q(x; \phi(y))$. For an observation $y = 0.25$ (dashed red line in left), the posterior $p(x \mid y = 0.25)$ (dashed red line in right) is "approximated" by samples "close" to y (blue "x" in right) to form an empirical inclusive KL-divergence minimized by LIC. As inclusive KL-divergence encourages a mass-covering / mean-seeking fit, the resulting proposal distribution $q(x; \phi(y = 0.25), K = 1)$ (green dotted line in right) when using a single $(K = 1)$ Gaussian proposal density covers both modes and can successfully propose moves which cross between the two mixture components. Using a 2-component $(K = 2)$ GMM proposal density results in $q(x; \phi(y = 0.25), K = 2)$ (purple solid line in right) which captures both the bi-modality of the posterior as well as the low probability region between the two modes. As a result of sampling the generative model, LIC can discover both posterior modes and their appropriate mixture weights (whereas other state of the art MCMC samplers fail, see fig. 5.3).

variables $\boldsymbol{x}$ from the posterior $p(\boldsymbol{x} \mid \boldsymbol{y})$ conditioned on observations $\boldsymbol{y}$. While exceptions exist, modern general-purpose PPLs typically implement variational inference [Bin+19], importance sampling [WMM14; LBW17], or Monte Carlo Markov Chain (MCMC, [WSG11; Teh+20b]).

Our work focuses on MCMC. More specifically, we target lightweight Metropolis-Hastings (LMH, [WSG11]) within a recently developed declarative PPL called `beanmachine` [Teh+20b]. The performance of Metropolis-Hastings critically depends on the quality of the proposal distribution used, which is the primary goal of LIC. Broadly speaking, LIC amortizes MCMC by constructing function approximators (parameterized by graph neural networks) from Markov blankets to proposal distribution parameters which are learned via forward simulation and training to match the full conditionals (equivalently, to minimize the inclusive KL divergence). In doing so, LIC makes the following contributions:

1. We present a novel implementation of inference compilation (IC) within an open-universe declarative PPL which combines Markov blanket structure with graph neural network

architectures. Our primary innovation is the use of a graph neural network to aggregate Markov blankets, which enables proposers to ignore irrelevant variables by construction.

2. We demonstrate LIC's ability to escape local modes, improved robustness to nuisance random variables, and improvements over state-of-the-art methods across a number of metrics in two industry-relevant applications.

Declarative Probabilistic Programming

To make Bayesian inference accessible to non-experts, PPLs provide user-friendly primitives in high-level programming languages for abstraction and composition in model representation [Goo13; Gha15]. Existing PPLs can be broadly classified based on the representation inference is performed over, with declarative PPLs [Lun+00; Plu+03; Mil+07; Teh+20b] performing inference over Bayesian graphical networks and imperative PPLs conducting importance sampling [WMM14] or MCMC [WSG11] on linearized execution traces. Because an execution trace is a topological sort of an instantiated Bayesian network, declarative PPLs naturally preserve additional model structure such as Markov blanket relationships while imperative PPLs require additional dependency tracking instrumentation [MSP14] or analysis tooling [Gor+20; Cus+19] to achieve similar functionality.

Definition 5.1 *The* Markov Blanket $\mathrm{MB}(x_i)$ *of a node* x_i *is the minimal set of random variables such that*

$$p(x_i \mid \boldsymbol{x}_{-i}, \boldsymbol{y}) = p(x_i \mid \mathrm{MB}(x_i)) \tag{5.1}$$

In a Bayesian network, $\mathrm{MB}(x_i)$ *consists of the parents, children, and children's parents of* x_i *[Pea87].*

Inference Compilation

Amortized inference [GG14] refers to the re-use of initial up-front learning to improve future queries. In context of Bayesian inference [MYM18; Zha+18] and IC [PW16a; Wei+19; Har+19], this means using acceleration performing multiple inferences over different observations $\boldsymbol{y}$ to amortize a one-time "compilation time." While compilation in both trace-based IC [PW16a; LBW17; Har+19] and LIC consists of drawing forward samples from the generative model $p(\boldsymbol{x}, \boldsymbol{y})$ and training neural networks to minimize inclusive KL-divergence, trace-based IC uses the resulting neural network artifacts to parameterize proposal distributions for importance sampling while LIC uses them for MCMC proposers.

Lightweight Metropolis Hastings

Lightweight Metropolis Hastings (LMH, [WSG11; RSG16]) is a simple but general method of implementing a PPL. In LMH, random variables are assigned string identifiers and their

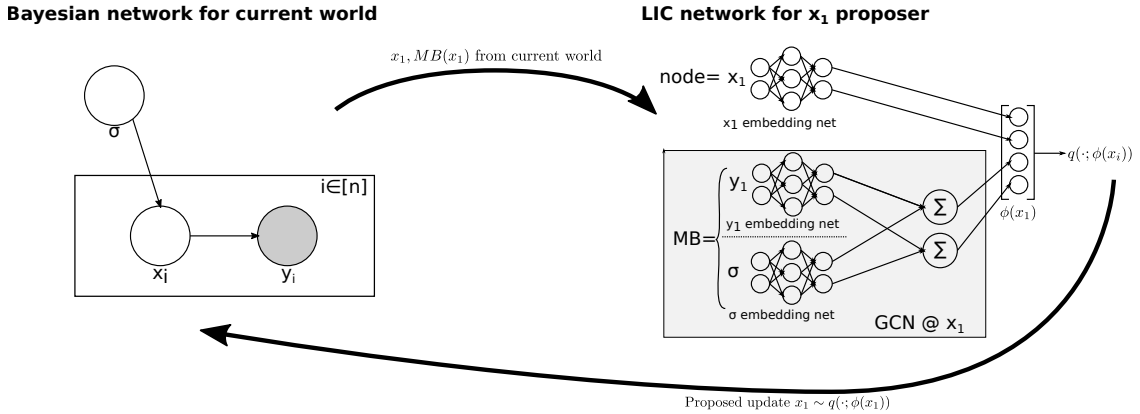


Figure 5.2: The Markov blankets in the Bayesian network for eq. (5.3) (left, expressed in plate notation) are available in a declarative PPL, and are used as inputs to LIC. The LIC proposer for node x_1 (right) is obtained by first performing neural network embedding of x_1 and every node in its Markov blanket, followed by a graph convolutional network aggregation over the Markov blanket of x_1. The resulting vectors are then combined to yield a parameter vector $\phi(x_1)$ for a proposal distribution $q(\cdot; \phi(x_1))$ which is then sampled for proposing an update within Metropolis-Hastings.

values and likelihoods are stored in a database (`World` in our implementation). MCMC is performed in a Metropolis-within-Gibbs manner where: (1) a single random variable is modified according to a proposal distribution while all others remain fixed, (2) the computations dependant upon the modified random variable (e.g. the continuation in a continuation-passing-style implementation) is re-executed to generate the remaining trace (re-using the database values for all other random variables) and the new trace's likelihood, and finally (3) a Metropolis-Hastings accept/reject correction is performed.

While a number of choices for proposal distribution exist, the single-site Gibbs sampler which proposes from eq. (5.1) enjoys a 100% acceptance probability [Pea87] and provides a good choice when available [Lun+00; Plu+03]. Unfortunately, outside of discrete models they are oftentimes intractable to directly sample so another proposal distribution must be used. LIC seeks to approximate these single-site Gibbs distributions using tractable neural network approximations.

Related Works

Prior work on IC in imperative PPLs can be broadly classified based on the order in which nodes are sampled. "Backwards" methods approximate an inverse factorization, starting at observations and using IC artifacts to propose propose parent random variables. Along these lines, [PW16a] use neural autoregressive density estimators but heuristically invert the model by expanding parent sets. [Web+18] proposes a more principled approach utilizing minimal

I-maps and demonstrate that minimality of inputs to IC neural networks can be beneficial; an insight also exploited through LIC's usage of Markov blankets. Unfortunately, model inversion is not possible in universal PPLs [LBW17].

The other group of "forwards" methods operate in the same direction as the probabilistic model's dependency graph. Starting at root nodes, these methods produce inference compilation artifacts which map an execution trace's prefix to a proposal distribution. In [RHG16], a user-specified model-specific guide program parameterizes the proposer's architecture and results in more interpretable IC artifacts at the expense of increased user effort. [LBW17] automates this by using a recurrent neural network (RNN) to summarize the execution prefix while constructing a node's proposal distribution. This approach suffers from well-documented RNN limitations learning long distance dependencies [Hoc98], requiring mechanisms like attention [Har+19] to avoid degradation in the presence of long execution trace prefixes (e.g. when nuisance random variables are present).

With respect to prior work, LIC is most similar to the attention-based extension [Har+19] of [LBW17]. Both methods minimize inclusive KL-divergence empirically approximated by samples from the generative model $p(\boldsymbol{x}, \boldsymbol{y})$, and both methods use neural networks to produce a parametric proposal distribution from a set of inputs sufficient for determining a node's posterior distribution. However, important distinctions include (1) LIC's use of a declarative PPL implies Markov blanket relationships are directly queryable and ameliorates the need for also learning an attention mechanism, and (2) LIC uses a graph neural network to summarize the Markov blanket rather than a RNN over the execution trace prefix. [Wan+17b] is also closely related to LIC as both are methods for amortizing MCMC by learning Markov blanket parameterized neural Gibbs proposers, but a key difference is that LIC exploits permutation-invariance of graph neural networks to address the issue where "Markov blankets... might not be consistent across all instantiations" whereas [Wan+17b] restricts "focus on hand-identified common structur[al motifs]" for constructing proposers.

5.2 Lightweight Inference Compilation

Architecture

Figure 5.2 shows a sketch of LIC's architecture. For every latent node x_i, LIC constructs a mapping $(x_i, \mathrm{MB}(x_i)) \mapsto \phi(x_i)$ parameterized by feedforward and graph neural networks to produce a parameter vector $\phi(x_i)$ for a parametric density estimator $q(\cdot; \phi(x_i))$. Every node x_i has feedforward "node embedding network" used to map the value of the underlying random variable into a vector space of common dimensionality. The set of nodes in the Markov blanket are then summarized to a fixed-length vector following section 5.2, and a feedforward neural network ultimately maps the concatenation of the node's embedding with its Markov blanket summary to proposal distribution parameters $\phi(x_i)$.

Dynamic Markov Blanket embeddings

Because a node's Markov blanket may vary in both its size and elements (e.g. in a GMM, a data point's component membership may change during MCMC), $\text{MB}(x_i)$ is a non-static set of vectors (albeit all of the same dimension after node embeddings are applied) and a feed-forward network with fixed input dimension is unsuitable for computing a fixed-length proposal parameter vector $\phi(x_i)$. Furthermore, Markov blankets (unlike execution trace prefixes) are unordered sets and lack a natural ordering hence use of a RNN as done in [LBW17] is inappropriate. Instead, LIC draws motivation from graph neural networks [Sca+08; DDS16] which have demonstrated superior results in representation learning [Bru+13] and performs summarization of Markov Blankets following [KW16] by defining

$$\phi(x_i) = \sigma\left(\boldsymbol{W} \underset{x_j \in \text{MB}(x_i)}{\square} \frac{1}{\sqrt{|\text{MB}(x_i)||\text{MB}(x_j)|}} f_j(x_j)\right)$$

where $f_j(x_j)$ denotes the output of the node embedding network for node x_j when provided its current value as an input, $\square$ is any differentiable permutation-invariant function (summation in LIC's case), and σ is an activation function. This technique is analogous to the "Deep sets trick" [Zah+17], and we expect it to perform well when elements of the Markov blanket are conditionally exchangeable. However, we note this assumption of permutation invariance may not always hold hence additional investigation into more sophisticated aggregation schemes (e.g. identifying exchangeable elements, using permutation-dependent aggregators like RNNs) is important future work.

Parameterized density estimation

The resulting parameter vectors $\phi(x_i)$ of LIC are ultimately used to parameterize proposal distributions $q(x_i; \phi(\text{MB}(x_i)))$ for MCMC sampling. For discrete x_i, LIC directly estimates logit scores over the support. For continuous x_i, LIC transforms continuous x_i to unconstrained space following [Car+17] and models the density using a Gaussian mixture model (GMM). Note that although more sophisticated density estimators such as masked autoregressive flows [Kin+16; PPM17] can equally be used.

Objective Function

To "compile" LIC, parameters are optimized to minimize the inclusive KL-divergence between the posterior distributions and inference compilation proposers: $D_{\text{KL}}(p(\boldsymbol{x} \mid \boldsymbol{y}) \parallel q(\boldsymbol{x} \mid \boldsymbol{y}; \phi))$. Consistent with [LBW17], observations $\boldsymbol{y}$ are sampled from the marginal distribution $p(\boldsymbol{y})$, but note that this may not be representative of observations $\boldsymbol{y}$ encountered at inference. The

resulting objective function is given by

$$\begin{aligned}
&\mathbb{E}_{p(\boldsymbol{y})}\left[D_{\mathrm{KL}}(p(\boldsymbol{x} \mid \boldsymbol{y}) \parallel q(\boldsymbol{x} \mid \boldsymbol{y}; \phi))\right] \\
&= \mathbb{E}_{p(\boldsymbol{x},\boldsymbol{y})}\left[\log \frac{p(\boldsymbol{x} \mid \boldsymbol{y})}{q(\boldsymbol{x} \mid \boldsymbol{y}; \phi)}\right] \\
&\propto \mathbb{E}_{p(\boldsymbol{x},\boldsymbol{y})}\left[-\log q(\boldsymbol{x} \mid \boldsymbol{y}; \phi)\right] \\
&\approx \sum_{i=1}^{N} -\log q(\boldsymbol{x} \mid \boldsymbol{y}; \phi), \qquad (\boldsymbol{x}, \boldsymbol{y}) \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, \boldsymbol{y}) \\
&=: \mathcal{L}(\phi)
\end{aligned} \tag{5.2}$$

where we have neglected a conditional entropy term independent of ϕ and performed Monte-Carlo approximation to an expectation. The intuition for this objective is shown in Figure 5.1, which shows how samples from $p(\boldsymbol{x}, \boldsymbol{y})$ (left) form an empirical joint approximation where "slices" (at $y = 0.25$ in fig. 5.1) yield posterior approximations which the objective is computed over (right).

5.3 Experiments

To validate LIC's competitiveness, we conducted experiments benchmarking a variety of desired behaviors and relevant applications. In particular:

- Training on samples from the joint distribution $p(\boldsymbol{x}, \boldsymbol{y})$ should enable discovery of distant modes, so LIC samplers should be less likely to to get "stuck" in a local mode. We validate this in section 5.3 using a GMM mode escape experiment, where we see LIC escape not only escape a local mode but also yield accurate mixture component weights.

- When there is no approximation error (i.e. the true posterior density is within the family of parametric densities representable by LIC), we expect LIC to closely approximate the posterior at least for the range of observations $\boldsymbol{y}$ sampled during compilation (eq. (5.2)) with high probability under the prior $p(\boldsymbol{y})$. Section 5.3 shows this is indeed the case in a conjugate Gaussian-Gaussian model where a closed form expression for the posterior is available.

- Because Markov blankets can be explicitly queried, we expect LIC's performance to be unaffected by the presence of nuisance random variables (i.e. random variables which extend the execution trace but are statistically independent from the observations and queried latent variables). This is confirmed in section 5.3 using the probabilistic program from [Har+19], where we see trace-based IC suffering an order of magnitude increase in model parameters and compilation time while yielding an effective sample size almost 5× smaller (Table 5.1).

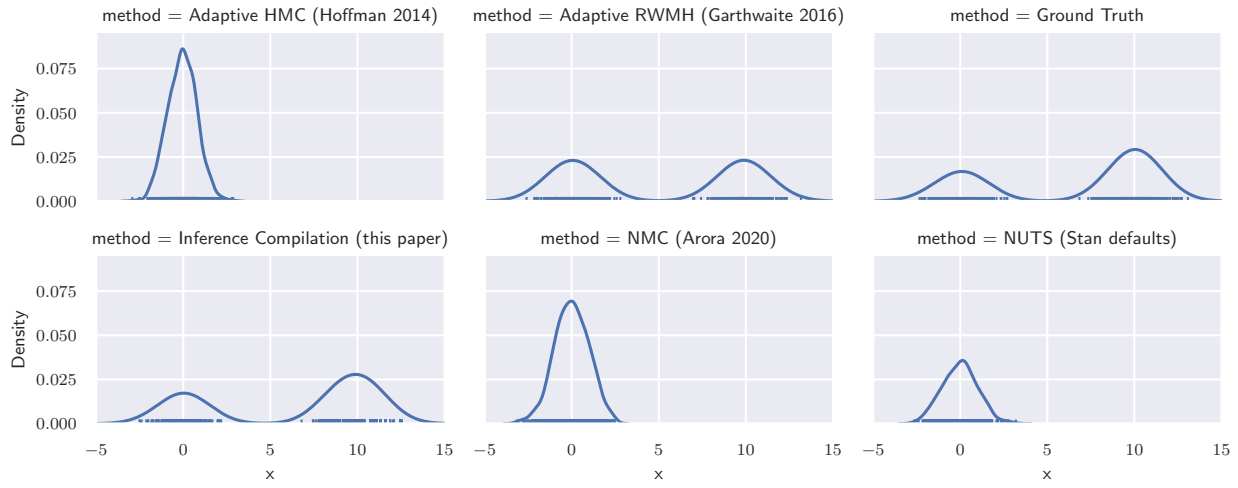


Figure 5.3: When sampling the bi-modal posterior density from fig. 5.1, only inference compilation (IC, this paper) and adaptive step-size random walk Metropolis-Hastings (Adaptive RWMH, [GFS16]) are able to recover both posterior modes. Whereas RWMH's posterior samples erroneously assign approximately equal probability to both modes, IC's samples faithfully reproduce the ground truth and yields higher probability for the mode at $x = 10$ than the mode at $x = 0$.

- To verify LIC yields competitive performance in applications of interest, we benchmark LIC against other state-of-the-art MCMC methods on a Bayesian logistic regression problem (section 5.3) and on a generalization of the classical eight schools problem [Rub81] called n-schools (section 5.3) which is used in production at a large internet company for Bayesian meta-analysis [SA01]. We find that LIC exceeds the performance of adaptive random walk Metropolis-Hastings [GFS16] and Newtonian Monte Carlo [Aro+20] and yields comparable performance to NUTS [HG14] despite being implemented in an interpreted (Python) versus compiled (C++) language.

A reference implementation for LIC and code to reproduce our experiments have been made publically available[1].

GMM mode escape

Consider the multi-modal posterior resulting from conditioning on $y = 0.25$ in the 2-dimensional GMM in fig. 5.1, which is comprised of two Gaussian components with greater mixture probability on the right-hand component and a large energy barrier separating the two components. Because LIC is compiled by training on samples from the joint distribution $p(x, y)$, it is reasonable to expect LIC's proposers to assign high probability to values for

[1] `https://github.com/facebookresearch/lightweight-inference-compilation`

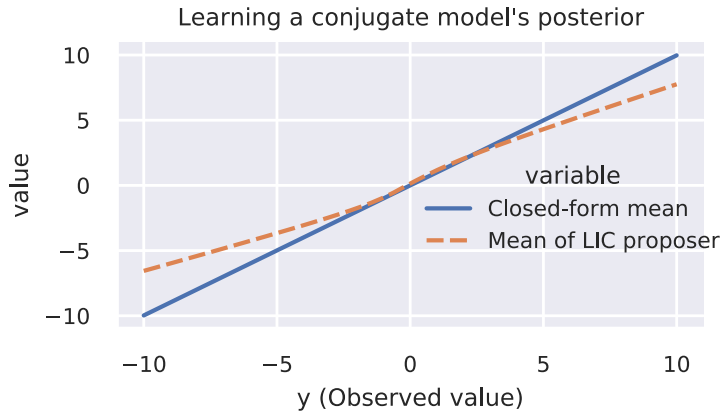


Figure 5.4: In a conjugate normal-normal model, LIC's proposal distribution mean (dashed orange line) closely follows the closed-form posterior mean (blue solid line) across a wide range of observed values y.

the latent variable x from both modes. In contrast, uncompiled methods such as random walk Metropolis-Hastings (RWMH) and NUTS may encounter difficulty crossing the low-probability energy barrier and be unable to escape the basin of attraction of the mode closest to their initialization.

This intuition is confirmed in fig. 5.3, which illustrates kernel density estimates of 1,000 posterior samples obtained by a variety of MCMC algorithms as well as ground truth samples. HMC with adaptive step size [HG14], NMC, and NUTS with the default settings as implemented in Stan [Car+17] are all unable to escape the mode they are initialized in. While both LIC and RWMH with adaptive step size escape the local mode, RWMH's samples erroneously imply equal mixture component probabilities whereas LIC's samples faithfully reproduce a higher component probability for the right-hand mode.

Conjugate Gaussian-Gaussian Model

We next consider a Gaussian likelihood with a Gaussian mean prior, a conjugate model with closed-form posterior given by:

$$\begin{aligned} x \sim \mathcal{N}(0, \sigma_x), \qquad y \sim \mathcal{N}(x, \sigma_y) \\ \Pr[x \mid y, \sigma_x, \sigma_y] \sim \mathcal{N}\left(\frac{\sigma_y^{-2}}{\sigma_x^{-2} + \sigma_y^{-2}} y, \frac{1}{\sigma_x^{-2} + \sigma_y^{-2}}\right) \end{aligned} \tag{5.3}$$

There is minimal approximation error because the posterior density is in the same family as LIC's GMM proposal distributions and the relationship between the Markov blanket $\mathrm{MB}(x) = \{y\}$ and the posterior mean is a linear function easily approximated (locally)

by neural networks. As a result, we expect LIC's proposal distribution to provide a good approximation to the true posterior and LIC to approximately implement a direct posterior sampler.

To confirm this behavior, we trained LIC with a $K = 1$ component GMM proposal density on 1,000 samples and show the resulting LIC proposer's mean as the observed value y varies in fig. 5.4. Here $\sigma_x = 2$ and $\sigma_y = 0.1$, so the marginal distribution of y (i.e. the observations sampled during compilation in eq. (5.2)) is Gaussian with mean 0 and standard deviation $\sqrt{\sigma_x^2 + \sigma_y^2} \approx 2.0025$. Consistent with our expectations, LIC provides a good approximation to the true posterior for observed values y well-represented during training (i.e. with high probability under the marginal $p(y)$). While LIC also provides a reasonable proposer by extrapolation to less represented observed values y, it is clear from fig. 5.4 that the approximation is less accurate. This motivates future work into modifying the forward sampling distribution used to approximate eq. (5.2) (e.g. "inflating" the prior) as well as adapting LIC towards the distribution of observations y used at inference time.

Robustness to Nuisance Variables

An important innovation of LIC is its use of a declarative PPL and ability to query for Markov blanket relationships so that only statistically relevant inputs are utilized when constructing proposal distributions. To validate this yields significant improvement over prior work in IC, we reproduced an experiment from [Har+19] where nuisance random variables (i.e. random variables which are statistically independent from the queried latent variables/observations whose only purpose is to extend the execution trace) are introduced and the impact on system performance is measured. As trace-based inference compilation utilizes the execution trace prefix to summarize program state, extending the trace of the program with nuisance random variables typically results in degradation of performance due to difficulties encountered by RNN in capturing long range dependencies as well as the production of irrelevant neural network embedding artifacts.

We reproduce trace-based IC as described in [LBW17] using the author-provided software package [PyP20], and implement Program 1 from [Har+19] with the source code illustrated in listing 5.1 where 100 nuisance random variables are added. Note that although `nuisance` has no relationship to the remainder of the program, the line number where they are instantiated has a dramatic impact on performance. By extending the trace between where `x` and `y` are defined, trace-based IC's RNNs degrade due to difficulty learning a long-range dependency[Hoc98] between the two variables. For LIC, the equivalent program expressed in the `beanmachine` declarative PPL [Teh+20b] is shown in listing 5.2. In this case, the order in which random variable declarations appear is irrelevant as all permutations describe the same probabilistic graphical model.

Listing 5.1: A version of Program 1 from [Har+19] to illustrate nuisance random variables

```
def magnitude(obs):
  x = sample(Normal(0, 10))
```

```
  for _ in range(100):
    nuisance = sample(Normal(0, 10))
  y = sample(Normal(0, 10))
  observe(
    obs**2,
    likelihood=Normal(
      x**2 + y**2,
      0.1))
  return x
```

Listing 5.2: The equivalent program in beanmachine, where independencies are explicit in program specification

```
class NuisanceModel:
  @random_variable
  def x(self):
      return dist.Normal(0, 10)
  @random_variable
  def nuisance(self, i):
      return dist.Normal(0, 10)
  @random_variable
  def y(self):
      return dist.Normal(0, 10)
  @random_variable
  def noisy_sq_length(self):
      return dist.Normal(
        self.x()**2 + self.y()**2,
        0.1)
```

Table 5.1 compares the results between LIC and trace-based IC [LBW17] for this nuisance variable model. Both `pyprob`'s defaults (1 layer 4 dimension sample embedding, 64 dimension address embedding, 8 dimension distribution type embedding, 10 component GMM proposer, 1 layer 512 dimension LSTM) and LIC's defaults (used for all experiments in this paper, 1 layer 4 dimension node embedding, 3 layer 8 dimension Markov blanket embedding, 1 layer node proposal network) with a 10 component GMM proposer are trained on 10,000 samples and subsequently used to draw 100 posterior samples. Although model size is not directly comparable due differences in model architecture, `pyprob`'s resulting models were over 7× larger than those of LIC. Furthermore, despite requiring more than 10× longer time to train, the resulting sampler produced by `pyprob` yields an effective sample size almost 5× smaller than that produced by LIC.

	# params	compile time	ESS
LIC	3,358	44 sec.	49.75
PyProb	21,952	472 sec.	10.99

Table 5.1: Number of parameters, compilation time (10,000 samples), and effective sample size (100 samples) for inference compilation in LIC (this work) versus [PyP20]

Bayesian Logistic Regression

Consider a Bayesian logistic regression model over d covariates with prior distribution $\boldsymbol{\beta} \sim \mathcal{N}_{d+1}(\mathbf{0}_{d+1}, \text{diag}(10, 2.5\mathbf{1}_d))$ and likelihood $y_i \mid \boldsymbol{x}_i \overset{\text{i.i.d.}}{\sim} \text{Bernoulli}(\sigma(\boldsymbol{\beta}^\top \boldsymbol{x}_i))$ where $\sigma(t) = (1 + e^{-t})^{-1}$ is the logistic function. This model is appropriate in a wide range of classification problems where prior knowledge about the regression coefficients $\boldsymbol{\beta}$ are available.

Figure 5.5a shows the results of performing inference using LIC compared against other MCMC inference methods. Results for existing IC approaches [LBW17] are omitted because they are not comparable due to lack of support for vector-valued random variables in the publically available reference implementation [PyP20]. All methods yield similar predictive log-likelihoods on held-out test data, but LIC and NUTS yield significantly higher ESS and $\widehat{R}$s closer to 1.0 suggesting better mixing and more effective sampling.

n-Schools

The eight schools model [Rub81] is a Bayesian hierarchical model originally used to model the effectiveness of schools at improving SAT scores. n-schools is a generalization of this model from 8 to n possible treatments, and is used at a large internet company for performing Bayesian meta-analysis [SA01] to estimate (fixed) effect sizes. Let K denote the total number of schools, n_j the number of districts/states/types, and j_k the district/state/type of school k.

$$
\begin{aligned}
\beta_0 &\sim \text{StudentT}(3, 0, 10) \\
\tau_i &\sim \text{HalfCauchy}(\sigma_i) \qquad \text{for } i \in [\text{district}, \text{state}, \text{type}] \\
\beta_{i,j} &\sim \mathcal{N}(0, \tau_i) \qquad \text{for } i \in [\text{district}, \text{state}, \text{type}], j \in [n_i] \\
y_k &\sim \mathcal{N}(\beta_0 + \sum_i \beta_{i,j_k}, \sigma_k)
\end{aligned}
$$

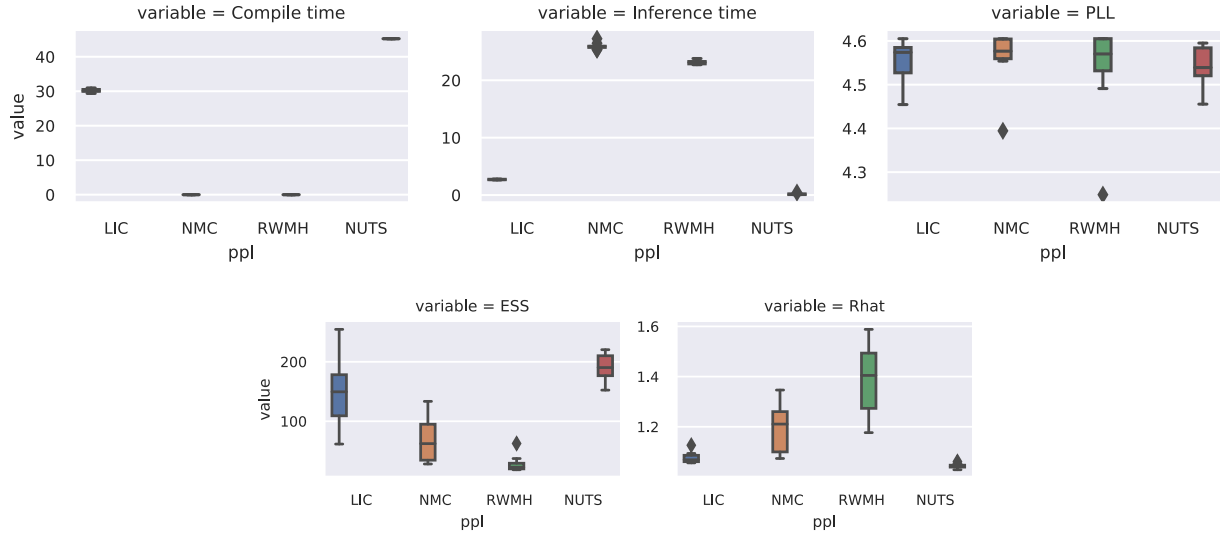
The treatment effects y_k and standard errors σ_i and σ_k are observed.

Intuitively, each "school" corresponds to a set of treatment parameters (here a tuple of district, state, and type) and $\beta_{i,j}$ measures the average change in response y when treatment parameter i is set equal to j (e.g. $\beta_{\text{state,CA}}$ measures the change in SAT scores when a school is located in California).

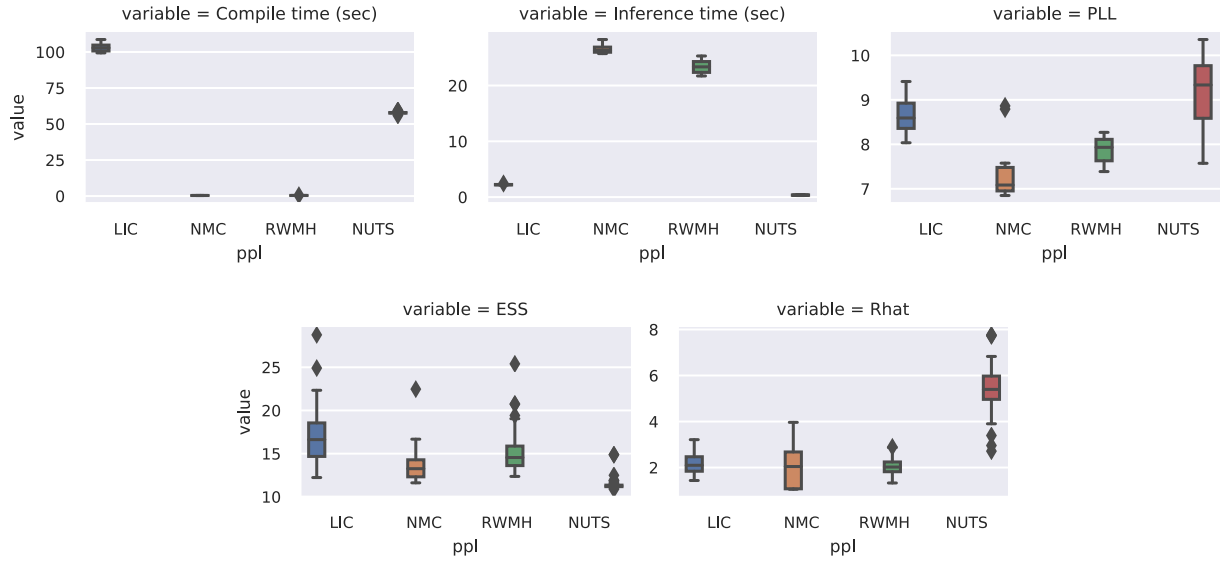
Figure 5.5b presents results in a format analogous to section 5.3. Here, we see that while both LIC and NUTS yield higher PLLs (with NUTS outperforming LIC in this case), LICs

Figure 5.5: Results of MCMC on two Bayesian inference tasks where we compare the compilation time (neural network training for LIC and Stan's C++ codegen / compilation for NUTS), inference time, predictive log-likelihood (PLL) on hold-out data, expected sample size (ESS, higher is better, [Gey11]) and the rank normalized $\widehat{R}$ diagnostic (Rhat, closer to 1 is better, [Veh+20]).

(a) Bayesian logistic regression (2000 rows, 10 features). Both LIC and Stan (NUTS) amortize the upfront compilation cost with accelerated inference times. LIC achieves comparable PLL to NUTS [HG14] and other methods, and yields ESS comparable to NUTS and higher than any other method. The $\widehat{R}$ of LIC is close to 1 (similar to NUTS and lower than all other methods).



(b) n-schools (1000 schools, 8 states, 5 districts, 5 types). Again, increased compilation times are offset by accelerated inference times. In this case, LIC achieves comparable PLL to NUTS while simultaneously producing higher ESS and lower $\widehat{R}$, suggesting that the resulting posterior samples are less autocorrelated and provide a more accurate posterior approximation.



ESS is significantly higher than other compared methods. Additionally, the $\widehat{R}$ of NUTS is also larger than the other methods which suggests that even after 1,000 burn-in samples NUTS has still not properly mixed.

5.4 Conclusion

We introduced Lightweight Inference Compilation (LIC) for building high quality single-site proposal distributions to accelerate Metropolis-Hastings MCMC. LIC utilizes declarative probabilistic programming to retain graphical model structure and graph neural networks to approximate single-site Gibbs sampling distributions. To our knowledge, LIC is the first proposed method for inference compilation within an open-universe declarative probabilistic programming language and an open-source implementation will be released in early 2021. Compared to prior work, LIC's use of Markov blankets resolves the need for attention to handle nuisance random variances and yields posterior sampling comparable to state-of-the-art MCMC samplers such as NUTS and adaptive RWMH.

Chapter 6

Fat-tailed variational inference

In some applications of probabilistic modeling, researchers may desire to explicitly model large deviation "tail" or "black-swan" events. While fat-tailed densities commonly arise as posterior and marginal distributions in robust models and scale mixtures, they present challenges when prior methods (including chapter 5) fails to capture tail decay accurately. In this chapter, we first improve previous theory on tails of Lipschitz flows by quantifying how the tails affect the *rate* of tail decay and by expanding the theory to non-Lipschitz polynomial flows. We then develop an alternative theory for multivariate tail parameters which is sensitive to tail-anisotropy. In doing so, we unveil a fundamental problem which plagues many existing flow-based methods: they can only model tail-isotropic distributions (i.e., distributions having the same tail parameter in every direction). To mitigate this and enable modeling of tail-anisotropic targets, we propose anisotropic tail-adaptive flows (ATAF). Experimental results on both synthetic and real-world targets confirm that ATAF is competitive with prior work while also exhibiting appropriate tail-anisotropy. Parts of this chapter were first presented in Feynman Liang, Liam Hodgkinson, and Michael Mahoney. "Fat–Tailed Variational Inference with Anisotropic Tail Adaptive Flows". In: *Proceedings of the 39th International Conference on Machine Learning*. Vol. 162. 2022, p. 132.

6.1 Introduction

Flow-based methods [Pap+21] have proven to be effective techniques to model complex probability densities. They compete with the state of the art on density estimation [Hua+18; Dur+19; Jai+20], generative modeling [Che+19; KD18], and variational inference [Kin+16; ASD20] tasks. These methods start with a random variable X having a simple and tractable distribution μ, and then apply a learnable transport map f_θ to build another random variable $Y = f_\theta(X)$ with a more expressive *pushforward* probability measure $(f_\theta)_*\mu$ [Pap+21]. In contrast to the implicit distributions [Hus17] produced by generative adversarial networks (GANs), flow-based methods restrict the transport map f_θ to be invertible and to have efficiently-computable Jacobian determinants. As a result, probability density functions can

be tractably computed through direct application of a change of variables

$$p_Y(y) = p_X(f_\theta^{-1}(y)) \left| \det \left. \frac{df_\theta^{-1}(z)}{dz} \right|_{z=y} \right|. \tag{6.1}$$

While recent developments [Che+19; Hua+18; Dur+19] have focused primarily on the transport map f_θ, the base distribution μ has received comparatively less investigation. The most common choice for the base distribution is standard Gaussian $\mu = \mathcal{N}(0, \boldsymbol{I})$. However, in Theorem 6.1, we show this choice results in significant restrictions on the expressivity of the model, limiting its utility for data that exhibits fat-tailed (or heavy-tailed) structure. Prior work addressing heavy-tailed flows [Jai+20] are limited to tail-isotropic base distributions. In Proposition 6.1, we prove flows built on these base distributions are unable to model accurately multivariate anisotropic fat-tailed structure.

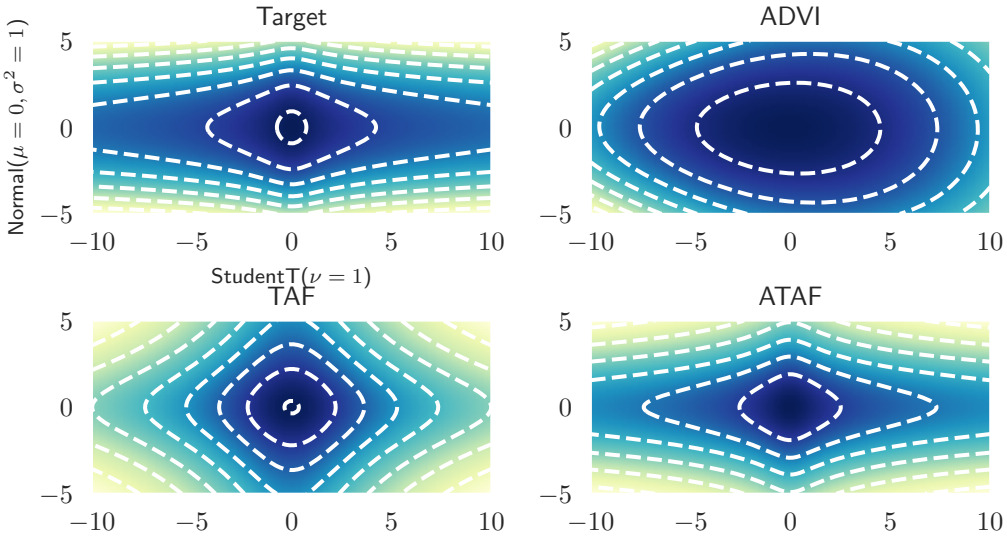


Figure 6.1: Variational inference against a tail-anisotropic target distribution $\mathcal{N}(0, 1) \otimes \text{StudentT}(\nu = 1)$ (top left). Only ATAF (bottom right) is able to correctly reproduce the tail-anisotropy (fat-tailed along x-axis, Gaussian along y-axis). In contrast, ADVI's (top right) Gaussian base distribution and TAF's (bottom left) tail-isotropic $\prod_{i=1}^{2} \text{StudentT}(\nu)$ base distribution can only model tail-isotropic distributions (Proposition 6.1), which erroneously imposes power-law tails with the same rate of decay along both the x and y axes.

Our work here aims to identify and address these deficiencies. To understand the impact of the base distribution μ in flow-based models, we develop and apply theory for fat-tailed random variables and their transformations under Lipschitz-continuous functions. Our approach leverages the theory of concentration functions [Led01, Chapter 1.2] to sharpen significantly and extend prior results [JSY19, Theorem 4] by describing precisely the tail parameters

of the pushforward distribution $(f_\theta)_*\mu$ under both Lipschitz-continuous (Theorem 6.1) and polynomial (Corollary 6.2) transport maps. In the multivariate setting, we develop a theory of direction-dependent tail parameters (Definition 6.4), and we show that tail-isotropic base distributions yield tail-isotropic pushforward measures (Proposition 6.1). As a consequence of Proposition 6.1, prior methods [Jai+20] are limited in that they are unable to capture *tail-anisotropy*. This motivates the construction of *anisotropic tail adaptive flows* (ATAF, Definition 6.5) as a means to alleviate this issue (Remark 6.1) and to improve modeling of tail-anisotropic distributions. Our experiments show that ATAF exhibits correct tail behaviour in synthetic target distributions exhibiting fat-tails (Figure 6.5 of Section 6.4) and tail-anisotropy (Figure 6.1). On realistic targets, we find that ATAF can yield improvements in variational inference (VI) by capturing potential tail-anisotropy (Section 6.4).

Related Work

Fat-Tails in Variational Inference. Recent work in variational autoencoders (VAEs) have considered relaxing Gaussian assumptions to heavier-tailed distributions [Mat+19; Che+19; Boe+20; AO20]. In Mathieu et al. [Mat+19], a StudentT prior distribution $p(z)$ is considered over the latent code z in a VAE with Gaussian encoder $q(z \mid x)$. They argue that the anisotropy of a StudentT product distribution leads to more disentangled representations, as compared to the standard choice of Normal distributions. A similar modification is performed in Chen et al. [CSN20] for a coupled VAE [Cao+22]. This result showed improvements in the marginal likelihoods of reconstructed images. In addition, Boenninghoff et al. [Boe+20] consider a mixture of StudentTs for the prior $p(z)$. To position our work in context, note that the encoder $q(z \mid x)$ may be viewed as a variational approximation to the posterior $p(z \mid x)$ defined by the decoder model $p(x \mid z)$ and the prior $p(z)$. Our work differs from Mathieu et al. [Mat+19], Chen et al. [CSN20], and Boenninghoff et al. [Boe+20], in that we consider fat-tailed variational approximations $q(z \mid x)$ rather than priors $p(z)$. Although Abiri et al. [AO20] also considers a StudentT approximate posterior, our work involves a more general variational family which uses normalizing flows. Similarly, although Wang et al. [WLL18] also deals with fat-tails in variational inference, their goal is to improve α-divergence VI by controlling the moments of importance sampling ratios (which may be heavy-tailed). Our work here adopts Kullback-Leibler divergence and is concerned with enriching the variational family to include anisotropic fat-tailed distributions. More directly comparable recent work [DQV11; FSS17] studies the t-exponential family variational approximation which includes StudentTs and other heavier-tailed densities. Critically, the selection of their parameter t (directly related to the StudentT's degrees of freedom ν), and the issue of tail anisotropy, are not discussed.

Flow-Based Methods. Normalizing flows and other flow-based methods have a rich history within variational inference [Kin+16; RM15; ASD20; Web+19a]. Consistent with our experience (Figure 6.6), Webb et al. [Web+19a] documents normalizing flows can offer improvements over ADVI and NUTS across thirteen different Bayesian linear regression

models from Gelman et al. [GH06]. Agrawal et al. [ASD20] shows that normalizing flows compose nicely with other advances in black-box VI (e.g., stick the landing, importance weighting). However, none of these works treat the issue of fat-tailed targets and inappropriate tail decay. To our knowledge, only TAFs [Jai+20] explicitly consider flows with tails heavier than Gaussians. Our work here can be viewed as a direct improvement of Jaini et al. [Jai+20], and we make extensive comparison to this work throughout the body of this paper. At a high level, we provide a theory for fat-tails which is sensitive to the rate of tail decay and develop a framework to characterize and address the tail-isotropic limitations plaguing TAFs.

6.2 Flow-Based Methods for Fat-Tailed Variational Inference

Flow-Based VI Methods

The objective of VI is to approximate a target distribution $\pi(x)$ by searching over a *variational family* $\mathcal{Q} = \{q_\phi : \phi \in \Phi\}$ of probability distributions q_ϕ. While alternatives exist [LT16; WLL18], VI typically seeks to find q_ϕ "close" to π, as measured by Kullback-Leibler divergence $D(q_\phi \parallel \pi)$. To ensure tractability without sacrificing generality, in practice [WW13; RGB14] a Monte-Carlo approximation of the evidence lower bound (ELBO) is maximized:

$$\begin{aligned}\mathrm{ELBO}(\phi) &= \int q_\phi(x) \log \frac{\bar{\pi}(x)}{q_\phi(x)} dx \\ &\approx \frac{1}{n} \sum_{i=1}^{n} \log \frac{\bar{\pi}(x_i)}{q_\phi(x_i)},\ x_i \overset{\text{i.i.d.}}{\sim} q_\phi,\ \bar{\pi} \propto \pi.\end{aligned}$$

To summarize, this procedure enables tractable black-box VI by replacing π with $\bar{\pi} \propto \pi$ and approximating expectations with respect to q_ϕ (which are tractable only in simple variational families) through Monte-Carlo approximation. In Bayesian inference and probabilistic programming applications, the target posterior $\pi(x) = p(x \mid y) = \frac{p(x,y)}{p(y)}$ is typically intractable but $\bar{\pi}(x) = p(x, y)$ is computable (i.e., represented by the probabilistic program's generative / forward execution).

While it is possible to construct a variational family $\mathcal{Q}$ tailored to a specific task, we are interested in VI methods which are more broadly applicable and convenient to use: $\mathcal{Q}$ should be automatically constructed from introspection of a given probabilistic model/program. Automatic differentiation variational inference (ADVI, Kucukelbir et al. [Kuc+17]) is an early implementation of automatic VI and it is still the default in certain probabilistic programming languages [Car+17]. ADVI uses a Gaussian base distribution μ and a transport map $f_\theta = f \circ \Phi_{\text{Affine}}$ comprised of an invertible affine transform composed with a deterministic transformation f from $\mathbb{R}$ to the target distribution's support (e.g., $\exp : \mathbb{R} \to \mathbb{R}_{\geq 0}$, sigmoid : $\mathbb{R} \to [0, 1]$). As Gaussians are closed under affine transformations, ADVI's representational capacity is limited to deterministic transformations of Gaussians. Hence it cannot represent

Model	Autoregressive transform	Suff. cond. for Lipschitz-continuity
NICE[DKB15]	$z_j + \mu_j \cdot \mathbb{1}_{k \notin [j]}$	μ_j Lipschitz
MAF[PPM17]	$\sigma_j z_j + (1 - \sigma_j)\mu_j$	σ_j bounded
IAF[Kin+16]	$z_j \cdot \exp(\lambda_j) + \mu_j$	λ_j bounded, μ_j Lipschitz
Real-NVP[DSB17]	$\exp(\lambda_j \cdot \mathbb{1}_{k \notin [j]}) \cdot z_j + \mu_j \cdot \mathbb{1}_{k \notin [j]}$	λ_j bounded, μ_j Lipschitz
Glow[KD18]	$\sigma_j \cdot z_j + \mu_j \cdot \mathbb{1}_{k \notin [j]}$	σ_j bounded, μ_j Lipschitz
NAF[Hua+18]	$\sigma^{-1}(w^\top \cdot \sigma(\sigma_j z_j + \mu_j))$	Always (logistic mixture CDF)
NSF[Dur+19]	$z_j \mathbb{1}_{z_j \notin [-B,B]} + M_j(z_j; z_{<j}) \mathbb{1}_{x_j \in [-B,B]}$	Always (linear outside $[-B, B]$)
FFJORD[Gra+19]	n/a (not autoregressive)	Always (required for invertibility)
ResFlow[Che+19]	n/a (not autoregressive)	Always (required for invertibility)

Table 6.1: Some popular / recently developed flows, the autoregressive transform used in the flow (if applicable), and sufficient conditions conditions for Lipschitz-continuity. A subset of this table was first presented in Jaini et al. [Jai+20]. $M(\cdot)$ denotes monotonic rational quadratic splines [Dur+19].

complex multi-modal distributions. To address this, more recent work [Kin+16; Web+19a] replaces the affine map Φ_{Affine} with a flow Φ_{Flow} typically parameterized by an invertible neural network:

Definition 6.1 *ADVI (with normalizing flows) comprise the variational family*

$$\mathcal{Q}_{ADVI} \coloneqq \{(f \circ \Phi_{Flow})_* \mu\}$$

where $\mu = Normal(0_d, I_d)$, Φ_{Flow} is an invertible flow transform (e.g., Table 6.1) and f is a deterministic bijection between constrained supports [Kuc+17].

As first noted in Jaini et al. [Jai+20], the pushforward of a light-tailed Gaussian base distribution under a Lipschitz-continuous flow will remain light-tailed and provide poor approximation to fat-tailed targets. Despite this, many major probabilistic programming packages still make a default choice of Gaussian base distribution (`AutoNormalizingFlow`/`AutoIAFNormal` in Pyro [Bin+19], `method=variational` in Stan [Car+17], `NormalizingFlowGroup` in PyMC [PHF10]). To address this issue, tail-adaptive flows [Jai+20] use a base distribution $\mu_\nu = \prod_{i=1}^d \text{StudentT}(\nu)$, where a single degrees-of-freedom $\nu \in \mathbb{R}$ is used across all d dimensions. Here is a more precise definition.

Definition 6.2 *Tail adaptive flows (TAF) comprise the variational family $\mathcal{Q}_{TAF} \coloneqq \{(f \circ \Phi_{Flow})_* \mu_\nu\}$, where $\mu_\nu = \prod_{i=1}^d StudentT(\nu)$ with ν shared across all d dimensions, Φ_{Flow} is an invertible flow, and f is a bijection between constrained supports [Kuc+17]. During training, the shared degrees of freedom ν is treated as an additional variational parameter.*

Fat-Tailed Variational Inference

Fat-tailed variational inference (FTVI) considers the setting where the target $\pi(x)$ is fat-tailed. Such distributions commonly arise during a standard "robustification" approach

where light-tailed noise distributions are replaced with fat-tailed ones [TL05]. They also appear when weakly informative prior distributions are used in Bayesian hierarchical models [Gel+06].

To formalize these notions of fat-tailed versus light-tailed distributions, a quantitative classification for tails is required. While prior work classified distribution tails according to quantiles and the existence of moment generating functions [Jai+20, Section 3], here we propose a more natural and finer-grained classification based upon the theory of concentration functions [Led01, Chapter 1.2], which is sensitive to the rate of tail decay.

Definition 6.3 (Classification of tails) *For each $\alpha, p > 0$, we let*

- $\mathcal{E}_\alpha^p$ *denote the set of* exponential-type *random variables X with $\mathbb{P}(|X| \geq x) = \Theta(e^{-\alpha x^p})$;*
- $\mathcal{L}_\alpha^p$ *denote the set of* logarithmic-type *random variables X with $\mathbb{P}(|X| \geq x) = \Theta(e^{-\alpha(\log x)^p})$.*

In both cases, we call p the class index *and α the* tail parameter *for X. Note that every $\mathcal{E}_\alpha^p$ and $\mathcal{L}_\beta^q$ are disjoint, that is, $\mathcal{E}_\alpha^p \cap \mathcal{L}_\beta^q = \emptyset$ for all $\alpha, \beta, p, q > 0$. For brevity, we define the ascending families $\overline{\mathcal{E}_\alpha^p}$ and $\overline{\mathcal{L}_\alpha^p}$ analogously as before except with $\Theta(\cdot)$ replaced by $\mathcal{O}(\cdot)$. Similarly, we denote the class of distributions with exponential-type tails with class index at least p by $\overline{\mathcal{E}^p} = \cup_{\alpha \in \mathbb{R}_+} \overline{\mathcal{E}_\alpha^p}$, and similarly for $\overline{\mathcal{L}^p}$.*

For example, $\overline{\mathcal{E}_\alpha^2}$ corresponds to $\alpha^{-1/2}$-sub-Gaussian random variables, $\overline{\mathcal{E}_\alpha^1}$ corresponds to sub-exponentials, and (of particular relevance to this paper) $\mathcal{L}_\alpha^1$ corresponds to the class of power-law distributions.

6.3 Tail Behavior of Lipschitz Flows

This section contains our main theoretical contributions and proofs. We sharpen previous impossibility results approximating fat-tailed targets using light-tailed base distributions [Jai+20, Theorem 4] by characterizing the effects of Lipschitz-continuous transport maps on not only the tail class but also the class index and tail parameter (Definition 6.3). Furthermore, we extend the theory to include polynomial flows [JSY19]. For the multivariate setting, we define the tail-parameter function (Definition 6.4) to help formalize the notion of tail-isotropic distributions and prove a fundamental limitation that tail-isotropic pushforwards remain tail-isotropic (Proposition 6.1).

As an initial but nevertheless crucial result, we first bound the tail parameters of the sum of two power law random variables.

Lemma 6.1 *Suppose $X \in \mathcal{L}_\alpha^1$ and $Y \in \mathcal{L}_\beta^1$. Then $X + Y \in \mathcal{L}_{\min\{\alpha,\beta\}}^1$.*

Proof First, let $\gamma = \min\{\alpha, \beta\}$. It will suffice to show that (I) $\mathbb{P}(|X + Y| \geq r) = \mathcal{O}(r^{-\gamma})$, and (II) $\mathbb{P}(|X + Y| \geq r) \geq \Theta(r^{-\gamma})$. Since $(X, Y) \mapsto |X + Y|$ is a 1-Lipschitz function on

$\mathbb{R}^2$ and $\mathbb{P}(|X| \geq r) + \mathbb{P}(|Y| \geq r) = \mathcal{O}(r^{-\gamma})$, (I) follows directly from the hypotheses and Proposition 1.11 of Ledoux [Led01]. To show (II), note that for any $M > 0$, conditioning on the event $|Y| \leq M$,

$$\mathbb{P}\left(|X| + |Y| \geq r \,|\, |Y| \leq M\right) \geq \mathbb{P}\left(|X| \geq r - M\right).$$

Therefore, by taking M to be sufficiently large so that $\mathbb{P}(|Y| \leq M) \geq \frac{1}{2}$,

$$\begin{aligned}\mathbb{P}\left(|X+Y| \geq r\right) &\geq \mathbb{P}\left(|X| + |Y| \geq r\right) \\ &\geq \mathbb{P}\left(|X| + |Y| \geq r \,|\, |Y| \leq M\right) \mathbb{P}\left(|Y| \leq M\right) \\ &\geq \frac{1}{2}\mathbb{P}\left(|X| \geq r - M\right) = \Theta(r^{-\alpha}).\end{aligned}$$

The same process with X and Y reversed implies $\mathbb{P}(|X+Y| \geq r) \geq \Theta(r^{-\beta})$ as well. Both (II) and the claim follow. ■

Most of our following results are developed within the context of Lipschitz-continuous transport maps f_θ. In practice, many flow-based methods exhibit Lipschitz-continuity in their transport map, either by design [Gra+19; Che+19], or as a consequence of choice of architecture and activation function (Table 6.1). The following assumption encapsulates this premise.

Assumption 6.1 *f_θ is invertible, and both f_θ and f_θ^{-1} are L-Lipschitz continuous (e.g., sufficient conditions in Table 6.1 are satisfied).*

It is worth noting that domains other than $\mathbb{R}^d$ may require an additional bijection between supports (e.g. $\exp : \mathbb{R} \to \mathbb{R}_+$) which could violate assumption 6.1.

Closure of Tail Classes

Our first set of results pertains to the closure of the tail classes in Definition 6.3 under Lipschitz-continuous transport maps. While earlier work [Jai+20] demonstrated closure of exponential-type distributions $\cup_{p>0}\overline{\mathcal{E}^p}$ under flows satisfying Assumption 6.1, our results in Theorem 6.1 and Corollaries 6.1 and 6.2 sharpen these observations, showing that: (1) Lipschitz transport maps cannot decrease the class index p for exponential-type random variables, but they can alter the tail parameter α; and (2) under additional assumptions, they cannot change either class index p or the tail parameter α for logarithmic-type random variables.

Theorem 6.1 (Lipschitz maps of tail classes) *Under Assumption 6.1, the distribution classes $\overline{\mathcal{E}^p}$ and $\overline{\mathcal{L}^p_\alpha}$ (with $p, \alpha > 0$) are closed under every flow transformation in Table 6.1.*

Informally, Theorem 6.1 asserts that light-tailed base distributions cannot be transformed via Lipschitz transport maps into fat-tailed target distributions. Note this does not violate universality theorems for certain flows [Hua+18] as these results only apply in the infinite-dimensional limit. Indeed, certain exponential-type families (such as Gaussian mixtures) are dense in the class of *all* distributions, including those that are fat-tailed.

Proof [Proof of Theorem 6.1] Let X be a random variable from either $\mathcal{E}^p_\alpha$ or $\mathcal{L}^p_\alpha$. Its concentration function (Equation 1.6 Ledoux [Led01] is given by

$$\alpha_X(r) \coloneqq \sup\{\mu\{x : d(x, A) \geq r\}; A \subset \text{supp } X, \mu(A) \geq 1/2\} = \mathbb{P}(|X - m_X| \geq r).$$

Under Assumption 1, f_θ is Lipschitz (say with Lipschitz constant L) so by Proposition 1.3 of Ledoux [Led01],

$$\mathbb{P}(|f_\theta(X) - m_{f_\theta(X)}| \geq r) \leq 2\alpha_X(r/L) = \mathcal{O}(\alpha_X(r/L)),$$

where $m_{f_\theta(X)}$ is a median of $f_\theta(X)$. Furthermore, by the triangle inequality

$$\begin{aligned}
\mathbb{P}(|f_\theta(X)| \geq r) &= \mathbb{P}(|f_\theta(X) - m_{f_\theta(X)} + m_{f_\theta(X)}| \geq r) \\
&\leq \mathbb{P}(|f_\theta(X) - m_{f_\theta(X)}| \geq r - |m_{f_\theta(X)}|) \\
&= \mathcal{O}(\mathbb{P}(|f_\theta(X) - m_{f_\theta(X)}| \geq r)) \\
&= \mathcal{O}(\alpha_X(r/L)), \qquad (6.2)
\end{aligned}$$

where the asymptotic equivalence holds because $|m_{f_\theta(X)}|$ is independent of r. When $X \in \mathcal{E}^p_\alpha$, Equation (6.2) implies

$$\mathbb{P}(|f_\theta(X)| \geq r) = \mathcal{O}(e^{-\frac{\alpha}{L}r^p}) \implies f_\theta(X) \in \overline{\mathcal{E}}^p_{\alpha/L},$$

from whence we find that the Lipschitz transform of exponential-type tails continues to possess exponential-type tails with the same class index p, although the tail parameter may have changed. Hence, $\overline{\mathcal{E}^p}$ is closed under Lipschitz maps for each $p \in \mathbb{R}_{>0}$. On the other hand, when $X \in \mathcal{L}^p_\alpha$, Equation (6.2) also implies that

$$\mathbb{P}(|f_\theta(X)| \geq r) = \mathcal{O}(e^{-\alpha(\log(r/L))^p}) = \mathcal{O}(e^{-\alpha(\log r)^p}),$$

and therefore, $f_\theta(X) \in \overline{\mathcal{L}^p_\alpha}$. Unlike exponential-type tails, Lipschitz transforms of logarithmic-type tails not only remain logarithmic, but their tails decay no slower than a logarithmic-type tail of the same class index with the *same* tail parameter α. This upper bound suffices to show closure under Lipschitz maps for the ascending family $\overline{\mathcal{L}^p_\alpha}$. ■

Note that $\overline{\mathcal{L}^p_\alpha} \supset \mathcal{E}^q_\beta$ for all p, q, α, β, so Theorem 6.1 by itself does not preclude transformations of fat-tailed base distributions to light-tailed targets. Under additional assumptions on f_θ, we further establish a partial converse that a fat-tailed base distribution's tail parameter is unaffected after pushfoward, hence heavy-to-light transformations are impossible. Note here there is no ascending union over tail parameters (i.e., $\mathcal{L}^p_\alpha$ instead of $\overline{\mathcal{L}^p_\alpha}$).

Corollary 6.1 (Closure of $\mathcal{L}^p_\alpha$) *If in addition f_θ is smooth with no critical points on the interior or boundary of its domain, then $\mathcal{L}^p_\alpha$ is closed.*

This implies that simply fixing a fat-tailed base distribution *a priori* is insufficient; the tail-parameter(s) of the base distribution must be explicitly optimized alongside the other variational parameters during training. While these additional assumptions may seem restrictive, note that many flow transforms explicitly enforce smoothness and monotonicity [WL19; Hua+18; Dur+19] and hence satisfy the premises.

Proof [Proof of Corollary 6.1] Let f_θ be as before with the additional assumptions. Since f_θ is a smooth continuous bijection, it is a diffeomorphism. Furthermore, by assumption f_θ has invertible Jacobian on the closure of its domain hence $\sup_{x\in\text{dom } f_\theta}|(f_\theta)'(x)| \geq M > 0$. By the inverse function theorem, $(f_\theta)^{-1}$ exists and is a diffeomorphism with

$$\frac{d}{dx}(f_\theta)^{-1}(x) = \frac{1}{(f_\theta)'((f_\theta)^{-1}(x))} \leq \frac{1}{M}.$$

Therefore, $(f_\theta)^{-1}$ is M^{-1}-Lipschitz and we may apply Theorem 6.1 to conclude the desired result. ■

In fact, we can show a version of Theorem 6.1 ensuring closure of exponential-type distributions under polynomial transport maps which do not satisfy Assumption 6.1. This is significant because it extends the closure results to include polynomial flows such as sum-of-squares flows [JSY19].

Corollary 6.2 (Closure under polynomial maps) *For any $\alpha, \beta, p, q \in \mathbb{R}_+$, there does not exist a finite-degree polynomial map from $\mathcal{E}^p_\alpha$ into $\mathcal{L}^q_\beta$.*

Proof [Proof of Corollary 6.2] Let $X \in \mathcal{E}^p_\alpha$. By considering sufficiently large X such that leading powers dominate, it suffices to consider monomials $Y = X^k$. Notice $\mathbb{P}(Y \geq x) = \mathbb{P}(X \geq x^{1/k}) = \Theta(e^{-\alpha x^{p/k}})$, and so $Y \in \mathcal{E}^{p/k}_\alpha$. The result follows by disjointness of $\mathcal{E}$ and $\mathcal{L}$. ■

Multivariate Fat-Tails and Anisotropic Tail Adaptive Flows

Next, we restrict attention to power-law tails $\mathcal{L}^1_\alpha$, and we develop a multivariate fat-tailed theory and notions of isotropic/anisotropic tail indices. Using our theory, we prove that both ADVI and TAF are fundamentally limited because they are only capable of fitting tail-isotropic target measures (Proposition 6.1). We consider anisotropic tail adaptive flows (ATAF): a density modeling method which can represent tail-anisotropic distributions (Remark 6.1).

For example, consider the target distribution shown earlier in Figure 6.1 formed as the product of $\mathcal{N}(0,1)$ and StudentT($\nu = 1$) distributions. The marginal/conditional distribution along a horizontal slice (e.g., the distribution of $\langle X, e_0\rangle$) is fat-tailed, while along a vertical

slice (e.g., $\langle X, e_1 \rangle$) it is Gaussian. Another extreme example of tail-anisotropy where the tail parameter for $\langle X, v \rangle$ is different in every direction $v \in \mathcal{S}^1$ is given in Figure 6.2. Here $\mathcal{S}^{d-1}$ denotes the $(d-1)$-sphere in d dimensions. Noting that the tail parameter depends on the choice of direction, we are motivated to consider the following direction-dependent definition of multivariate tail parameters.

Definition 6.4 *For a d-dimensional random vector X, its* tail parameter function $\alpha_X : \mathcal{S}^{d-1} \to \bar{\mathbb{R}}_+$ *is defined as $\alpha_X(v) = -\lim_{x\to\infty} \log \mathbb{P}(\langle v, X \rangle \geq x)/\log x$ when the limit exists, and $\alpha_X(v) = +\infty$ otherwise. In other words, $\alpha_X(v)$ maps directions v into the tail parameter of the corresponding one-dimensional projection $\langle v, X \rangle$. The random vector X is* tail-isotropic *if $\alpha_X(v) \equiv c$ is constant and* tail-anisotropic *if $\alpha_X(v)$ is not constant but bounded.*

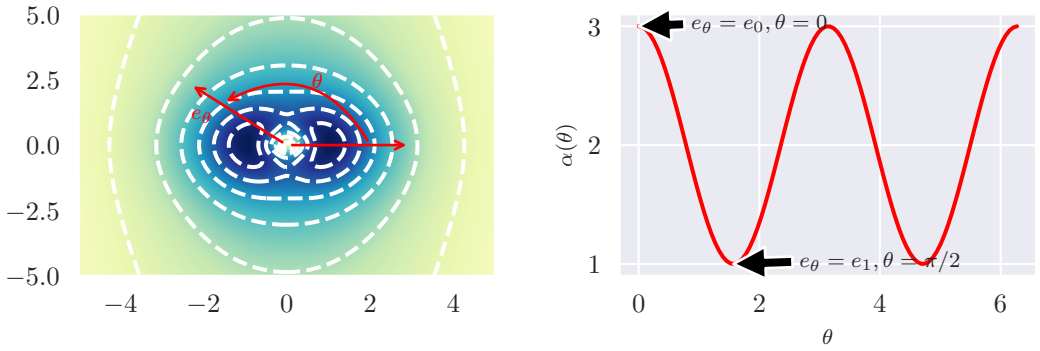


Figure 6.2: Illustration of the direction-dependent tail-parameter function (right) on a tail-anisotropic distribution (left) with PDF $\mathrm{d}P(r, \theta) = r^{-\alpha(\theta)} r \mathrm{d}r\mathrm{d}\theta$ and tail parameter $\alpha(\theta) = 2 + \cos(2\theta)$. While prior fat-tailed theory based on $\|X\|_2 = \sup_{\|v\|_2=1} \langle X, v \rangle$ is only sensitive to the largest tail parameter $\max_{\theta\in[0,2\pi]} \alpha(\theta) = 3.0$, our direction-dependent tail parameter function (bottom, red line) and its values along the standard basis axes ($\alpha(0)$ and $\alpha(\pi/2)$) capture *tail-anisotropy*.

Example of Non-existence of Tail Parameter Due to Oscillations

Of course, one can construct pathological densities where this definition is not effective. As a degenerate example, consider $\mathrm{StudentT}(\nu = 1) \otimes \mathrm{StudentT}(\nu = 2)$ and "spin" it using the radial transformation $(r, \theta) \mapsto (r, r + \theta)$ (Figure 6.3). Due to oscillations, $\alpha_X(v)$ is not well defined for all $v \in \mathcal{S}^1$.

Comparison versus TAF

It is illustrative to contrast with the theory presented for TAF [Jai+20], where only the tail exponent of $\|X\|_2$ is considered. For $X = (X_1, \ldots, X_d)$ with $X_i \in \mathcal{L}^1_{\alpha_i}$, by Fatou-Lebesgue

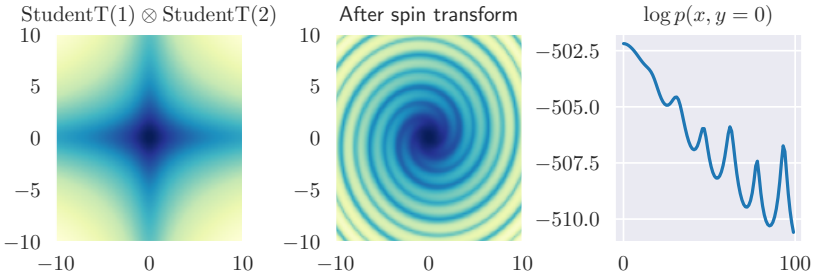


Figure 6.3: Taking a tail-anisotropic distribution (left) and "spinning" it (middle) results in one-dimensional projections which oscillate between tail parameters (as seen in $\log p(\langle X, e_0 \rangle)$ in right panel) and result in an ill-defined direction-dependent tail parameter function $\alpha_X(\cdot)$ due to a divergent limit.

and Lemma 6.1

$$\begin{aligned}\mathbb{P}[\|X\|_2 \geq t] &= \mathbb{P}\left[\sup_{z \in \mathcal{S}^{d-1}} \langle X, z \rangle \geq t\right] \\ &\geq \sup_{z \in \mathcal{S}^{d-1}} \mathbb{P}[\langle X, z \rangle \geq t] = \max_{1 \leq i \leq d} \nu_i = \max_{0 \leq i \leq d-1} \alpha_X(e_i).\end{aligned}$$

Therefore, considering only the tail exponent of $\|X\|_2$ is equivalent to summarizing $\alpha_X(\cdot)$ by an upper bound. Given the absence of the tail parameters for other directions (i.e., $\alpha_X(v) \neq \sup_{\|v\|=1} \alpha_X(v)$) in the theory for TAF [Jai+20], it should be unsurprising that both their multivariate theory as well as their experiments only consider tail-isotropic distributions obtained either as an elliptically-contoured distribution with fat-tailed radial distribution or $\prod_{i=1}^{d} \text{StudentT}(\nu)$ (tail-isotropic by Lemma 6.1). Our next proposition shows that this presents a significant limitation when the target distribution is tail-anisotropic.

Proposition 6.1 (Pushforwards of tail-isotropic distributions) *Let μ be tail isotropic with non-integer parameter ν and suppose f_θ satisfies Assumption 6.1. Then $(f_\theta)_* \mu$ is tail isotropic with parameter ν.*

To show Proposition 6.1, we will require a few extra assumptions to rule out pathological cases. The full content of Proposition 6.1 is contained in the following theorem.

Theorem 6.2 *Suppose there exists $\nu > 0$ such that $C : \mathcal{S}^{d-1} \to (0, \infty)$ satisfies $C(v) := \lim_{x \to \infty} x^\nu \mathbb{P}(|\langle v, X \rangle| > x)$ for all $v \in \mathcal{S}^{d-1}$. If ν is not an integer and f is a bilipschitz function, then $f(X)$ is tail-isotropic with tail index ν.*

Proof Since $x \mapsto \langle v, f(x) \rangle$ is Lipschitz continuous for any $v \in \mathcal{S}^{d-1}$, Theorem 6.1 implies $\langle v, f(X) \rangle \in \overline{\mathcal{L}^1_\nu}$. Let $\theta \in (0, \pi/2)$ (say, $\theta = \pi/4$), and let $S_v = \{x : \cos^{-1}(\langle x/\|x\|, v \rangle) \leq \theta\}$

for each $v \in \mathcal{S}^{d-1}$. Then

$$H_v \coloneqq \{x : \langle v, x\rangle > 1\} \supset \{x : \|x\| > (1-\cos\theta)^{-1}\} \cap S_v.$$

From Theorem C.2.1 of Buraczewski et al. [BDM16], since $\nu \notin \mathbb{Z}$, there exists a non-zero measure μ such that

$$\mu(E) = \lim_{x\to\infty} \frac{\mathbb{P}(x^{-1}X \in E)}{\mathbb{P}(\|X\| > x)},$$

for any Borel set E. Consequently, μ is regularly varying, and so by the spectral representation of regularly varying random vectors (see p. 281 Buraczewski et al. [BDM16]), there exists a measure P such that

$$\lim_{x\to\infty} \frac{\mathbb{P}(\|X\| > tx, X/\|X\| \in E)}{\mathbb{P}(\|X\| > x)} = t^{-\nu}P(E),$$

for any Borel set E on $\mathcal{S}^{d-1}$ and any $t > 0$. Letting $F_v = \{y/\|y\| : f(y) \in S_v\} \subset \mathcal{S}^{d-1}$ (noting that $P(F_v) > 0$ by assumption), since $m\|x-y\| \le \|f(x)-f(y)\| \le M\|x-y\|$ for all x, y,

$$\begin{aligned}
\liminf_{x\to\infty} \frac{\mathbb{P}(f(X) \in xH_v)}{\mathbb{P}(\|f(X)\| > x)} &\ge \liminf_{x\to\infty} \frac{\mathbb{P}(\|f(X)\| > x(1-\cos\theta)^{-1}, f(X) \in S_v)}{\mathbb{P}(\|f(X)\| > x)} \\
&\ge \liminf_{x\to\infty} \frac{\mathbb{P}(\|X\| > x(m(1-\cos\theta))^{-1}, X/\|X\| \in F_v)}{\|X\| > x/M} \\
&\ge P(F_v)\left(\frac{M}{m(1-\cos\theta)}\right)^{-\nu} > 0, yaB
\end{aligned}$$

where $P(F_v) > 0$ follows from the bilipschitz condition for f. Therefore, we have shown that $\mathbb{P}(\langle v, f(X)\rangle > x) = \Theta(\mathbb{P}(\|f(X)\| > x))$ for every $v \in \mathcal{S}^{d-1}$. Since $\mathbb{P}(\|f(X)\| > x)$ obeys a power law with exponent ν by Corollary 6.1, $f(X)$ is tail-isotropic with exponent ν. ■

Anisotropic tail adaptive flows

To work around this limitation without relaxing Assumption 6.1, it is evident that tail-anisotropic base distributions μ must be considered. Perhaps the most straightforward modification to incorporate a tail-anisotropic base distribution replaces TAF's isotropic base distribution $\prod_{i=1}^{d} \text{StudentT}(\nu)$ with $\prod_{i=1}^{d} \text{StudentT}(\nu_i)$. Note that ν is no longer shared across dimensions, enabling d different tail parameters to be represented:

Definition 6.5 *Anisotropic Tail-Adaptive Flows (ATAF) comprise the variational family* $\mathcal{Q}_{ATAF} \coloneqq \{(f \circ \Phi_{Flow})_* \mu_\nu\}$, *where* $\mu_\nu = \prod_{i=1}^{d} StudentT(\nu_i)$, *each* ν_i *is* distinct, *and* f *is a bijection between constrained supports [Kuc+17]. Analogous to Jaini et al. [Jai+20], ATAF's implementation treats* ν_i *identically to the other parameters in the flow and jointly optimizes over them.*

Remark 6.1 *Anisotropic tail-adaptive flows can represent tail-anisotropic distributions with up to d different tail parameters while simultaneously satisfying Assumption 6.1. For example, if* $\Phi_{Flow} = Identity$ *and* $\mu_\nu = \prod_{i=1}^{d} StudentT(i)$ *then the pushforward* $(\Phi_{Flow})_* \mu_\nu = \mu_\nu$ *is tail-anisotropic.*

Naturally, there are other parameterizations of the tail parameters ν_i that may be more effective depending on the application. For example, in high dimensions, one might prefer not to allow for d unique indices, but perhaps only fewer. On the other hand, by using only d tail parameters, an approximation error will necessarily be incurred when more than d different tail parameters are present. Figure 6.2 presents a worst-case scenario where the target distribution has a continuum of tail parameters. In theory, this density could itself be used as an underlying base distribution, although we have not found this to be a good option in practice. The key takeaway is that to capture several different tails in the target density, one must consider a base distribution that incorporates sufficiently many *distinct* tail parameters.

Concerning the choice of StudentT families, we remark that since $\text{StudentT}(\nu) \Rightarrow \mathcal{N}(0, 1)$ as $\nu \to \infty$, ATAF should still provide reasonably good approximations to target distributions in $\overline{\mathcal{E}^2}$ by taking ν sufficiently large. This can be seen in practice in Section 6.4.

6.4 Experiments

Here we validate ATAF's ability to improve a range of probabilistic modeling tasks. Prior work [Jai+20] demonstrated improved density modelling when fat tails are considered, and our experiments are complementary by evaluating TAFs and ATAFs for variational inference tasks as well as by demonstrating the effect of tail-anisotropy for modelling real-world financial returns and insurance claims datasets. We implement using the `beanmachine` probabilistic programming language [Teh+20a] and the `flowtorch` library for normalizing flows [Flo21], and we have open-sourced code for reproducing experiments in Supplementary Materials.

All experiments were performed on an Intel i8700K with 32GB RAM and a NVIDIA GTX 1080 running PyTorch 1.9.0 / Python 3.8.5 / CUDA 11.2 / Ubuntu Linux 20.04 via Windows Subsystem for Linux.

Toy Examples

Normal-normal Conjugate Model

We consider a Normal-Normal conjugate inference problem where the posterior is known to be a Normal distribution as well. Here, we aim to show that ATAF performs no worse than ADVI because $\text{StudentT}(\nu) \to N(0, 1)$ as $\nu \to \infty$. Figure 6.4 shows the resulting density approximation, which can be seen to be reasonable for both a Normal base distribution (the "correct" one) and a StudentT base distribution. This suggests that mis-specification (i.e., heavier tails in the base distribution than the target) may not be too problematic.

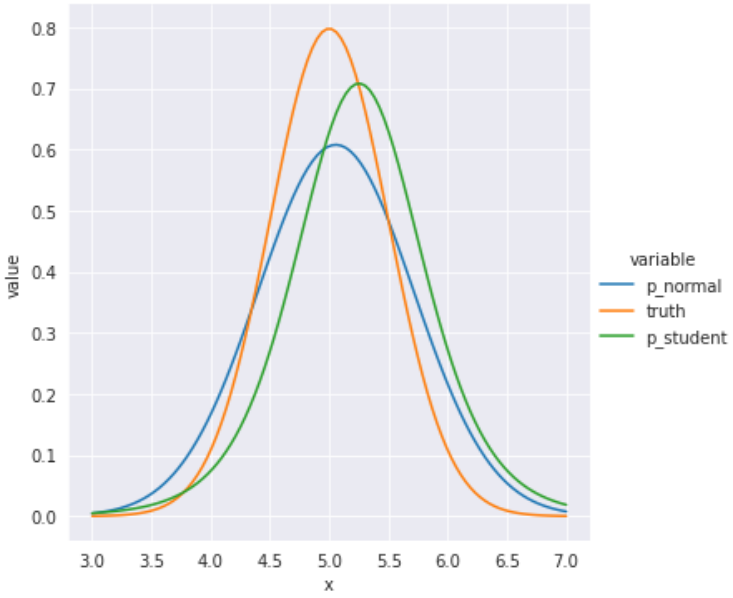


Figure 6.4: Variational inference against a light tailed Normal posterior. Both light and heavy tail variational families yield similar results.

Experiments Performing VI Against a Fat-tailed Cauchy Target

The motivation for the fat-tailed variational families used in TAF/ATAF is easily illustrated on a toy example consisting of $X \sim \text{Cauchy}(x_0 = 0, \gamma = 1) \in \mathcal{L}_1^1$. As seen in Figure 6.5, while ADVI with normalizing flows [Kin+16; Web+19a] appears to provide a reasonable fit to the bulk of the target distribution (left panel), the improper imposition of sub-Gaussian tails results in an exponentially bad tail approximation (middle panel). As a result, samples drawn from the variational approximation fail a Kolmogorov-Smirnov goodness-of-fit test against the true target distribution much more often (right panel, smaller p-values imply more rejections) than a variational approximation which permits fat-tails. This example is a special case of Theorem 6.1.

Real-World Datasets

For all flow-transforms Φ_{Flow}, we used inverse autoregressive flows [Kin+16] with a dense autoregressive conditioner consisting of two layers of either 32 or 256 hidden units depending on problem (see code for details) and ELU activation functions. As described in Jaini et al. [Jai+20], TAF is trained by including ν within the Adam optimizer alongside other flow parameters. For ATAF, we include all ν_i within the optimizer. Models were trained using the Adam optimizer with 10^{-3} learning rate for 10000 iterations, which we found empirically in all our experiments to result in negligible change in ELBO at the end of training.

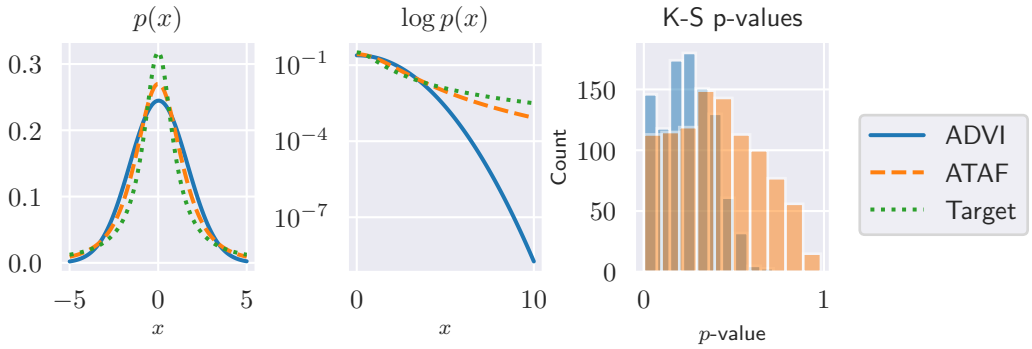


Figure 6.5: When performing FTVI to approximate a $X \sim \text{Cauchy}(x_0 = 0, \gamma = 1)$ target (left panel, green dotted line), the use of a Gaussian variational family (ADVI, solid blue line) can incur exponentially bad tail approximations (middle panel) compared to methods such as ATAF which permit heavier tails (orange dashed line). As a consequence, ADVI samples (blue, right panel) are rejected by the Kolmogorov-Smirnov test more often than ATAF samples (orange, right panel).

For table 6.2a and table 6.2b, the flow transform Φ_{Flow} used for ADVI, TAF, and ATAF is comprised of two hidden layers of 32 units each. NUTS uses no such flow transform. Variational parameters for each normalizing flow were initialized using `torch`'s default Kaiming initialization [He+15] Additionally, the tail parameters ν_i used in ATAF were initialized to all be equal to the tail parameters learned from training TAF. We empkkirically observed this resulted in more stable results (less variation in ELBO / $\log p(y)$ across trials), which may be due to the absence of outliers when using a Gaussian base distribution resulting in more stable ELBO gradients. This suggests other techniques for handling outliers such as winsorization may also be helpful, and we leave further investigation for future work.

For fig. 6.6, the closed-form posterior was computed over a finite element grid to produce the "Target" row. A similar progressive training scheme used for table 6.2a was also used here, with the TAF flow transform Φ_{Flow} initialized from the result of ADVI and ATAF additionally initialized all tail parameters ν_i based on the final shared tail parameter obtained from TAF training. Tails are computed along the $\beta = 1$ or $\sigma = 1$ axes because the posterior is identically zero for $\sigma = 0$, hence it reveals no information about the tails.

Bayesian Linear Regression

Consider one-dimensional Bayesian linear regression (BLR) with conjugate priors, defined by priors and likelihood

$$\sigma^2 \sim \text{Inv-Gamma}(a_0, b_0)$$
$$\beta \mid \sigma^2 \sim \mathcal{N}(0, \sigma^2), \qquad y \mid X, \beta, \sigma \sim \mathcal{N}(X\beta, \sigma^2),$$

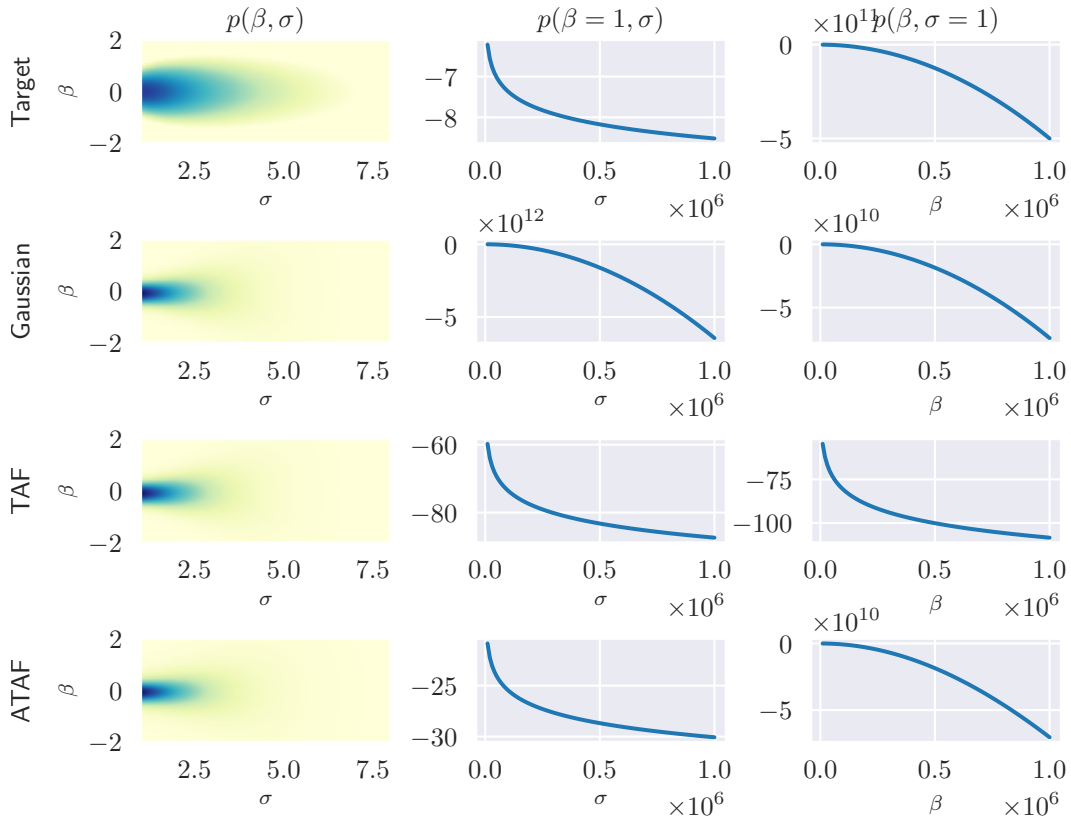


Figure 6.6: Bayesian linear regression's tail-anisotropic posterior (top left) exhibits a fat-tailed conditional in σ (as evidenced by the convex power-law decay in the top middle panel) and a Gaussian conditional in β (concave graph in top right panel). While all methods appear to provide a good approximation of the bulk (left column), Proposition 6.1 implies Gaussian (Gaussian, second row) or isotropic StudentT product (TAF, third row) base distributions yield Gaussian or power-law tails, respectively, for *both* σ and β. In contrast, ATAF (bottom row) illustrates Remark 6.1 by modeling simultaneously a power-law tail on σ and Gaussian tail on β.

	ELBO	$\log p(y)$
ADVI	$\mathbf{2873.90} \pm 6.95$	2969.73 ± 1.73
TAF	2839.64 ± 9.10	2973.85 ± 0.87
ATAF	2842.75 ± 8.83	$\mathbf{2976.75} \pm 0.66$
NUTS	n/a	3724.59 ± 0.036

(a) diamonds

	ELBO	$\log p(y)$
ADVI	-72.13 ± 6.89	-53.25 ± 3.44
TAF	-64.64 ± 4.88	-52.51 ± 4.41
ATAF	$\mathbf{-58.63} \pm 4.75$	$\mathbf{-51.01} \pm 3.71$
NUTS	n/a	-47.78 ± 0.093

(b) Eight schools

Table 6.2: Monte-Carlo ELBO and importance weighted Monte-Carlo marginal likelihood $p(y) = \mathbb{E}_{x \sim q_\theta} \frac{p(x,y)}{q_\theta(x)}$ (higher is better, $\pm$ standard errors) estimates from VI on real-world datasets. To understand the variational approximation gap, we include marginal likelihoods based on "golden samples" from `posteriordb` [The21] computed using No-U-Turn-Sampling (NUTS, Hoffman et al. [HG14] and Carpenter et al. [Car+17]).

	Fama-French 5 Industry Daily	CMS 2008-2010 DE-SynPUF
ADVI	-5.018 ± 0.056	-1.883 ± 0.012
TAF	-4.703 ± 0.023	-1.659 ± 0.004
ATAF	$\mathbf{-4.699} \pm 0.024$	$\mathbf{-1.603} \pm 0.034$

Table 6.3: Log-likelihoods (higher is better, $\pm$ standard errors) achieved on density modeling tasks involving financial returns [FF15] and insurance claims [Cen10] data.

where a_0, b_0 are hyperparameters and the task is to approximate the posterior distribution $p(\beta, \sigma^2 \mid X, y)$. Owing to conjugacy, the posterior distribution can be explicitly computed. Indeed, $p(\beta, \sigma^2 \mid X, y) = \rho(\sigma^2)\rho(\beta \mid \sigma)$ where $\rho(\beta \mid \sigma) = \mathcal{N}(\Sigma_n(X^\top X\hat{\beta}), \sigma^2\Sigma_n)$, $\Sigma_n = (X^\top X + \sigma^{-2})^{-1}$, $\hat{\beta} = (X^\top X)^{-1}X^\top y$, and

$$\rho(\sigma^2) = \text{Inv-Gamma}\left(a_0 + \frac{n}{2}, b_0 + \frac{1}{2}(y^\top y - \mu_n^\top \Sigma_n \mu_n)\right).$$

This calculation reveals that the posterior distribution is tail-anisotropic: for fixed c we have that $p(\sigma^2, \beta = c \mid X, y) \propto \rho(\sigma^2) \in \mathcal{L}^1_{\alpha_n}$ as a function of σ (with α_n a function of n) and $p(\sigma^2 = c, \beta \mid X, y) \propto \rho(\beta \mid c) \in \overline{\mathcal{E}^2}$ as a function of β. As a result of Proposition 6.1, we expect ADVI and TAF to erroneously impose Gaussian and power-law tails respectively for both β and σ^2 as neither method can produce a tail-anisotropic pushforward. This intuition is confirmed in Figure 6.6, where we see that only ATAF is the only method capable of modeling the tail-anisotropy present in the data.

Conducting Bayesian linear regression is among the standard tasks requested of a probabilistic programming language, yet it still displays tail-anisotropy. To accurately capture

large quantiles, this tail-anisotropy should not be ignored, necessitating a method such as ATAF.

Diamond Price Prediction Using Non-Conjugate Bayesian Regression

Without conjugacy, the BLR posterior is intractable and there is no reason *a priori* to expect tail-anisotropy. Regardless, this presents a realistic and practical scenario for evaluating ATAF's ability to improve VI. For this experiment, we consider BLR on the `diamonds` dataset [Wic11] included in `posteriordb` [The21]. This dataset contains a covariate matrix $X \in \mathbb{R}^{5000\times 24}$ consisting of 5000 diamonds each with 24 features as well as an outcome variable $y \in \mathbb{R}^{5000}$ representing each diamond's price. The probabilistic model for this inference task is specified in Stan code provided by The Stan Developers [The21] and is reproduced here for convenience:

$$\begin{gathered}\alpha \sim \text{StudentT}(\nu = 3, \text{loc} = 8, \text{scale} = 10)\\ \sigma \sim \text{HalfStudentT}(\nu = 3, \text{loc} = 0, \text{scale} = 10)\\ \beta \sim \mathcal{N}(0, \boldsymbol{I}_{24}), \qquad y \sim \mathcal{N}(\alpha + X\beta, \sigma).\end{gathered}$$

For each VI method, we performed 100 trials each consisting of 5000 descent steps on the Monte-Carlo ELBO estimated using 1000 samples and report the results in Table 6.2a. We report both the final Monte-Carlo ELBO as well as a Monte-Carlo importance-weighted approximation to the log marginal likelihood $\log p(y) = \log \mathbb{E}_{x\sim q_\theta} \frac{p(x,y)}{q_\theta(y)}$ both estimated using 1000 samples.

Eight Schools SAT Score Modelling with Fat-tailed Scale Mixtures

The eight-schools model [Rub81; Gel+13] is a classical Bayesian hierarchical model used originally to consider the relationship between standardized test scores and coaching programs in place at eight schools. A variation using half Cauchy non-informative priors [Gel+06] provides a real-world inference problem involving fat-tailed distributions, and is formally specified by the probabilistic model

$$\begin{gathered}\tau \sim \text{HalfCauchy}(\text{loc} = 0, \text{scale} = 5)\\ \mu \sim \mathcal{N}(0, 5), \qquad \theta \sim \mathcal{N}(\mu, \tau), \qquad y \sim \mathcal{N}(\theta, \sigma).\end{gathered}$$

Given test scores and standard errors $\{(y_i, \sigma_i)\}_{i=1}^8$, we are interested in the posterior distribution over treatment effects $\theta_1, \ldots, \theta_d$. The experimental parameters are identical to Section 6.4, and results are reported in Table 6.2b.

Financial and Actuarial Applications

To examine the advantage of tail-anisotropic modelling in practice, we considered two benchmark datasets from financial (daily log returns for five industry indices during 1926–2021

[FF15]) and actuarial (per-patient inpatient and outpatient cumulative Medicare/Medicid (CMS) claims during 2008–2010 [Cen10]) applications where practitioners actively seek to model fat-tails and account for black-swan events. Identical flow architectures and optimizers were used in both cases, with log-likelihoods presented in Table 6.3. Both datasets exhibited superior fits after allowing for heavier tails, with a further improved fit using ATAF for the CMS claims dataset.

6.5 Conclusion

In this work, we have sharpened existing theory for approximating fat-tailed distributions with normalizing flows, and we formalized tail-(an)isotropy through a direction-dependent tail parameter. With this, we have shown that many prior flow-based methods are inherently limited by tail-isotropy. With this in mind, we proposed a simple flow-based method capable of modeling tail-anisotropic targets. As we have seen, anisotropic FTVI is already applicable in fairly elementary examples such as Bayesian linear regression; and ATAFs provide one of the first methods for using the representational capacity of flow-based methods, while simultaneously producing tail-anisotropic distributions. A number of open problems still remain, including the study of other parameterizations of the tail behaviour of the base distribution. Even so, going forward, it seems prudent that density estimators, especially those used in black-box settings, consider accounting for tail-anisotropy using a method such as ATAF.

Chapter 7

The generalized gamma tail algebra

Whereas previous chapters considered adaptive methods which learn an approximation's bulk (chapter 5) and tail (chapter 6) from samples, in this chapter we develop a systematic approach for analyzing the tails of random variables during the static analysis (before drawing samples) pass of a probabilistic programming language (PPL) compiler. To characterize how the tails change under algebraic operations, we develop an algebra acting on a three-parameter family of tail asymptotics based on the generalized Gamma distribution. Our algebraic operations are closed under addition and multiplication, capable of distinguishing sub-Gaussians with differing scales, and handle ratios sufficiently well to reproduce the tails of most important statistical distributions directly from their definitions. Our experiments confirm that inference algorithms leveraging generalized Gamma algebra metadata attain superior performance across a number of density modeling and variational inference tasks. Parts of this chapter have been submitted for peer review as Feynman Liang, Liam Hodgkinson, and Michael Mahoney. "Static Analysis of Tail Behaviour with a Generalized Gamma Algebra". In: *Submitted to AISTATS 2023* (2023).

7.1 Introduction

To facilitate efficient probabilistic modelling and inference, modern probabilistic programming languages (PPLs) draw upon recent developments in functional programming [Tol+16], programming languages [Ber19], and deep variational inference [Bin+19]. Despite their broadening appeal, common pitfalls such as mismatched distribution supports [Lee+19] and non-integrable expectations [WLL18; Veh+15; Yao+18a] remain uncomfortably commonplace and challenging to debug. Recent innovations aiming to improve PPLs have automated verification of distribution constraints [Lee+19], tamed noisy gradient estimates [Esl+16] and unruly density ratios [Veh+15; WLL18], and approximated high-dimensional distributions with non-trivial bulks [Pap+21] and non-Gaussian tails [Jai+20].

Continuing this line of work, here we we consider how to statically analyze a probabilistic program in order to automate the inference of tail behavior for any random variables present.

At present, correct inference of tail behaviour for target distributions remains an outstanding issue [Yao+18a; WLL18], which causes challenges for downstream Monte Carlo tasks. For example, importance sampling estimators can exhibit infinite variance if the tail of the approximating density is lighter than the target. Most prominant black-box variational inference methods are incapable of changing their tail behaviour from an initial proposal distribution [Jai+20; LHM22]. MCMC algorithms may also lose ergodicity when the tail of the target density falls outside of a particular family [RT96]. All of these issues could be avoided if the tail of the target is known before runtime.

To classify tail asymptotics and define calibration, we propose a three-parameter family based on the generalized Gamma distribution (eq. (7.2)) which interpolates between established asymptotics on sub-Gaussian [Led01] and regularly varying [Mik99] random variables. Algebraic operations on random variables can be lifted to computations on the tail parameters resulting in what we call the *generalized Gamma algebra (GGA)*. Through analyzing operations like $X + Y$, X^2, and X/Y at the level of densities (e.g. additive convolution $p_X \oplus p_Y$), the tail parameters of a target density can be estimated from the parameters of any input distributions using Table 7.1.

Operationalizing the GGA, we propose *tail inferential* static analysis analogous to traditional *type inference* and provide a reference implementation using the `beanmachine graph` [Teh+20a] PPL compiler. GGA tail metadata can be used to diagnose and address tail-related problems in downstream tasks, such as employing Riemannian-manifold methods [GC11] to sample heavy tails or pre-emptively detect unbounded expectations. Here, we consider density estimation and variational inference where we use the GGA-computed tail of the target density to calibrate our density approximation. When composed with a learnable Lipschitz pushforward map (Section 7.4), the resulting combination is a flexible density approximator with provably calibrated tails.

Contributions

- The GGA is introduced, generalizing prior work on classifying tail asymptotics while including both sub-Gaussian / sub-exponentials [Led01] as well as power-law / Pareto-based tail indices [CSN09]. Composing operations outlined in table 7.1, one can compute the GGA tail class for downstream random variables of interest.
- The GGA is implemented in the static analysis phase of a PPL compiler. This unlocks the ability to leverage GGA metadata in order to better tailor the emitted inference algorithm.
- Finally, we propose and evaluate a density estimator which combines GGA tails with normalizing flows in order to simultaneously achieve good bulk approximation as well as correct tails.

7.2 Related Work

Heavy tails and probabilistic machine learning

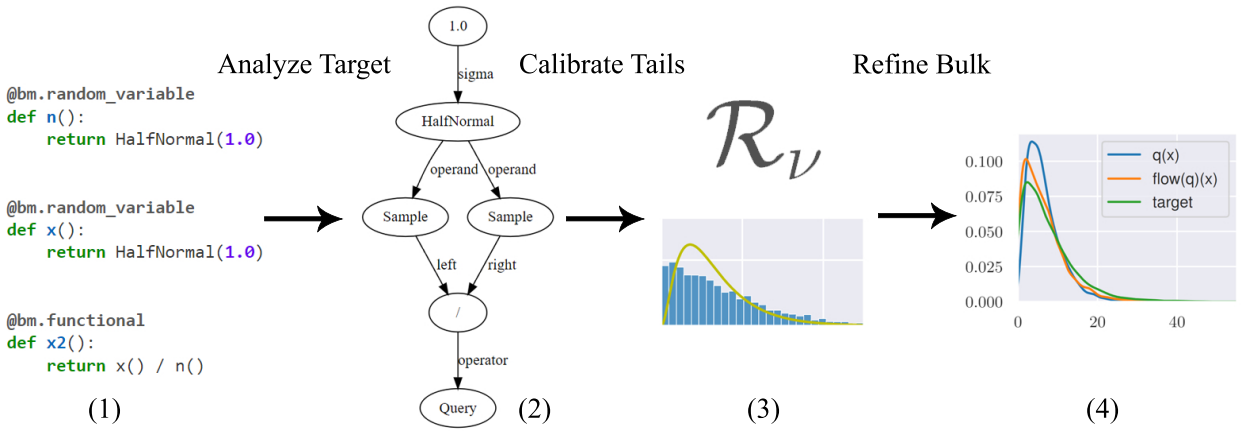


Figure 7.1: Our overall approach for density approximations with calibrated tails. A generative model expressed in a PPL (1) and analyzed using the GGA (2) to compute the tail parameters of the target. A representative distribution with calibrated tails is chosen for the initial approximation (3) and a learnable Lipschitz pushforward (see Lemma 7.2) is optimized (4) to correct the bulk approximation.

For studying heavy tails, methods based on subexponential distributions [GK98] and generalized Pareto distributions (GPD) or equivalently regularly varying distributions [Taj03] have recieved attention historically. Mikosch [Mik99] presents closure theorems for regularly varying which are special cases of Proposition 7.1 and Lemma 7.2. Heavy tails can impact probabilistic machine learning methods in a number of ways. The observation that density ratios $\frac{p(x)}{q(x)}$ tend to be heavy tailed has resulted in new methods for smoothing importance sampling [Veh+15], adaptively modifying divergences [WLL18], and diagnosing variational inference through the Pareto $\hat{k}$ diagnostic [Yao+18a]. These works are complementary to our paper and our reported results include $\hat{k}$ diagnostics for VI and $\hat{\alpha}$ tail index estimates based on GPD.

Our work considers heavy-tailed targets $p(x)$ which is the same setting as Jaini et al. [Jai+20] and Liang et al. [LHM22]. Whereas those respective works lump the tail parameter in as another variational parameter and may be more generally applicable, the GGA may be applied before samples are drawn and leads to perfectly calibrated tails when applicable.

Probabilistic programming

PPLs can be characterized by the primary use case optimized for, whether that's Gibbs sampling over Bayes nets [Spi+96; Val+17], stochastic control flow [Goo+12; WSG11], deep stochastic variational inference [Tra+18; Bin+19], or Hamiltonian Monte-Carlo [Car+17; Xu+20]. Our implementation target `beanmachine` [Teh+20a] is a declarative PPL selected due to availability of a PPL compiler and support for static analysis plugins. Similar to Bingham et al. [Bin+19] and Siddharth et al. [Sid+17], it uses PyTorch [Pas+19] for GPU tensors and automatic differentiation. Synthesizing an approximating distribution during

PPL compilation (Section 7.4) is also performed in the Stan language by Kucukelbir et al. [Kuc+17] and normalizing flow extensions in Webb et al. [Web+19b]. We compare directly against these related density approximators in Section 7.5.

Static analysis

There is a long history of formal methods and probabilistic programming [Koz79; JP89]. While much of the research [Cla+13] is concerned with defining formal semantics and establishing invariants [WHR18] See [Ber19] for a recent review. Static analysis utilizes the abstract syntax tree (AST) representation of a program in order to compute invariants (e.g. the return type of a function, the number of classes implementing a trait) without executing the underlying program. As dynamic analysis in PPs is less reliable due to non-determinism, static analysis methods for PPs become increasingly important.

Within PPLs, static analysis has traditionally been applied in the context of formalizing semantics [Koz79] and has been used to verify probabilistic programs by ensuring termination, bounding random values values [SCG13]. [Lee+19] proposes a static analyzer for the Pyro PPL [Bin+19] to verify distribution supports and avoid $-\texttt{Inf}$ log probabilities.

More relevant to our work are applications of static analysis to improve inference. Nori et al. [Nor+14] statically analyzes a probabilistic program and computes pre-images of observations in order to better adapt MCMC proposal distributions. While we also perform static analysis over abstract syntax tree (AST) representations of a probabilistic program, applying GGA yields an upper bound on the tails of all random variables so that calibrated tails can be imposed on distribution estimates.

7.3 The Generalized Gamma Algebra

Here we formulate an algebra of random variables that is closed under most standard elementary operations (addition, multiplication, powers) which forms the foundation for our static analysis.

Definition 7.1 *A random variable X is said to have a* generalized Gamma tail *if the Lebesgue density of $|X|$ satisfies*

$$p_{|X|}(x) \sim cx^{\nu}e^{-\sigma x^{\rho}}, \qquad \text{as } x \to \infty, \tag{7.1}$$

for some $c > 0$, $\nu \in \mathbb{R}$, $\sigma > 0$ and $\rho \in \mathbb{R}$. Denote the set of all such random variables by $\mathcal{G}$.

Consider the following equivalence relation on $\mathcal{G}$: $X \equiv Y$ if and only if $0 < p_{|X|}(x)/p_{|Y|}(x) < +\infty$ for all sufficiently large x. The resulting equivalence classes can be represented by their corresponding parameters ν, σ, ρ, and hence, we denote the class of random variables X satisfying eq. (7.1) by (ν, σ, ρ). In the special case where $\rho = 0$, for a fixed $\nu < -1$, each class $(\nu, \sigma, 0)$ for $\sigma > 0$ is equivalent, and is denoted by $\mathcal{R}_{|\nu|}$, representing *regularly varying* tails. Our algebra operates on these equivalence classes of $\mathcal{G}$, characterizing the change in tail behaviour under various operations.

The form of eq. (7.1) and the name of the algebra is derived from the generalized Gamma distribution.

Ordering	$\max\{(\nu_1,\sigma_1,\rho_1),(\nu_2,\sigma_2,\rho_2)\}$ $\equiv \begin{cases}(\nu_1,\sigma_1,\rho_1) & \text{if } \limsup_{x\to\infty}\frac{x_1^{\nu}e^{-\sigma_1 x^{\rho_1}}}{x_2^{\nu}e^{-\sigma_2 x^{\rho_2}}}<+\infty\\(\nu_2,\sigma_2,\rho_2) & \text{otherwise.}\end{cases}$
Addition	$(\nu_1,\sigma_1,\rho_1)\oplus(\nu_2,\sigma_2,\rho_2)$ $\equiv \begin{cases}\max\{(\nu_1,\sigma_1,\rho_1),(\nu_2,\sigma_2,\rho_2)\} & \text{if } \rho_1\neq\rho_2 \text{ or } \rho_1,\rho_2<1\\(\nu_1+\nu_2+1,\min\{\sigma_1,\sigma_2\},1) & \text{if } \rho_1=\rho_2=1\\(\nu_1+\nu_2+\frac{2-\rho}{2},(\sigma_1^{-\frac{1}{\rho-1}}+\sigma_2^{-\frac{1}{\rho-1}})^{1-\rho},\rho) & \text{if } \rho=\rho_1=\rho_2>1.\end{cases}$
Powers	$(\nu,\sigma,\rho)^{\beta}\equiv(\frac{\nu+1}{\beta}-1,\sigma,\frac{\rho}{\beta})$ for $\beta>0$
Reciprocal*	$(\nu,\sigma,\rho)^{-1}\equiv\begin{cases}(-\nu-2,\sigma,-\rho) & \text{if } (\nu+1)/\rho>0 \text{ and } \rho\neq 0\\\mathcal{R}_2 & \text{otherwise}\end{cases}$
Scalar Multiplication	$c(\nu,\sigma,\rho)\equiv(\nu,\sigma/\lvert c\rvert^{\rho},\rho)$
Multiplication	$(\nu_1,\sigma_1,\rho_1)\otimes(\nu_2,\sigma_2,\rho_2)$ $\equiv\begin{cases}\left(\frac{1}{\mu}\left(\frac{\nu_1}{\lvert\rho_1\rvert}+\frac{\nu_2}{\lvert\rho_2\rvert}+\frac{1}{2}\right),\sigma,-\frac{1}{\mu}\right) & \text{if } \rho_1,\rho_2<0\\\left(\frac{1}{\mu}\left(\frac{\nu_1}{\rho_1}+\frac{\nu_2}{\rho_2}-\frac{1}{2}\right),\sigma,\frac{1}{\mu}\right) & \text{if } \rho_1,\rho_2>0\\\mathcal{R}_{\lvert\nu_1\rvert} & \text{if } \rho_1\leq 0,\rho_2>0\\\mathcal{R}_{\min\{\lvert\nu_1\rvert,\lvert\nu_2\rvert\}} & \text{if } \rho_1=0,\rho_2=0\end{cases}$ where $\mu=\frac{1}{\lvert\rho_1\rvert}+\frac{1}{\lvert\rho_2\rvert}=\frac{\lvert\rho_1\rvert+\lvert\rho_2\rvert}{\lvert\rho_1\rho_2\rvert}$, $\sigma=\mu(\sigma_1\lvert\rho_1\rvert)^{\frac{1}{\mu\lvert\rho_1\rvert}}(\sigma_2\lvert\rho_2\rvert)^{\frac{1}{\mu\lvert\rho_2\rvert}}$.
Product of Densities	$(\nu_1,\sigma_1,\rho_1)\&(\nu_2,\sigma_2,\rho_2)\equiv\begin{cases}(\nu_1+\nu_2,\sigma_1,\rho_1) & \text{if } \rho_1<\rho_2\\(\nu_1+\nu_2,\sigma_1+\sigma_2,\rho) & \text{if } \rho=\rho_1=\rho_2\\(\nu_1+\nu_2,\sigma_2,\rho_2) & \text{otherwise.}\end{cases}$
Functions (L-Lipschitz)	$f(X_1,\ldots,X_n)\equiv L\max\{X_1,\ldots,X_n\}$

Table 7.1: Operations on random variables (e.g.X_1+X_2) are viewed as actions on density functions (e.g. convolution $(\nu_1,\sigma_1,\rho_1)\oplus(\nu_2,\sigma_2,\rho_2)$) and the tail parameters of the result are analyzed and reported.

Definition 7.2 *Let $\nu\in\mathbb{R}$, $\sigma>0$, and $\rho\in\mathbb{R}\backslash\{0\}$ be such that $(\nu+1)/\rho>0$. A non-negative random variable X is* generalized Gamma distributed *with parameters ν,σ,ρ if it has Lebesgue density*

$$p_{\nu,\sigma,\rho}(x)=c_{\nu,\sigma,\rho}x^{\nu}e^{-\sigma x^{\rho}},\qquad x>0, \tag{7.2}$$

where $c_{\nu,\sigma,\rho}=\rho\sigma^{(\nu+1)/\rho}/\Gamma((\nu+1)/\rho)$ is the normalizing constant.

The importance of the generalized Gamma form arises due to a combination of two factors:

(i) The majority of interesting continuous univariate distributions with infinite support satisfy eq. (7.1), including Gaussians ($\nu = 0$, $\rho = 2$), gamma/exponential/chi-squared ($\nu > -1$, $\rho = 1$), Weibull/Frechet ($\rho = \nu + 1$), and Student T/Cauchy/Pareto ($\mathcal{R}_\nu$). However, some notable exceptions include the log-normal distributions.

(ii) The set $\mathcal{G}$ is known to be closed under additive convolution, positive powers, and Lipschitz functions — we will show it is closed under multiplicative convolution as well. This covers the majority of elementary operations on independent random variables, with reciprocals, exponentials and logarithms the only exceptions. However, we will introduce a few "tricks" to handle these cases as well.

The full list of operations in GGA is compiled in table 7.1. All operations in the GGA can be proven to exhibit identical behaviour with their corresponding operations on random variables, with the sole exception of reciprocals (marked by asterisk), where additional assumptions are required.

Illustrative examples

To further illustrate the GGA through example, in this section we work out explicit GGA computations using distributions from table 7.2 and operations in table 7.1 and recover some common probability identities.

Example 7.1 (Chi-squared random variables) *Let $X_1, \ldots, X_k$ be k independent standard normal random variables. The variable $Z = \sum_{i=1}^k X_i^2$ is* chi-squared distributed *with k degrees of freedom. Using the generalized Gamma algebra, we can accurately determine the tail behaviour of this random variable directly from its construction. Recall that each $X_i \equiv (0, 1/2, 2)$, and by the power operation, $X_i^2 \equiv (-1/2, 1/2, 1)$. Applying the addition operation k times reveals that $Z \equiv (k/2-1, 1/2, 1)$ and implies that the density of Z is asymptotically $cx^{k/2-1}e^{-x/2}$ as $x \to \infty$. In fact, the density of Z is exactly $p_Z(x) = c_k x^{k/2-1}e^{-x/2}$ where $c_k = 2^{-k/2}/\Gamma(k/2)$.*

Example 7.2 (Products of random variables) *To demonstrate the efficacy of the multiplication operation in our algebra, we consider the product of two exponential, Gaussian, and reciprocal Gaussian random variables. In section 7.3, we manually prove the following.*

Lemma 7.1 *Let $X_1, X_2 \sim Exp(\lambda)$ and $Z_1, Z_2 \sim \mathcal{N}(0, 1)$ be independent. The densities of X_1X_2, Z_1Z_2 and $Z = 1/Z_1 \cdot 1/Z_2$ satisfy as $x \to \infty$,*

$$p_{X_1X_2}(x) \sim \frac{\lambda^{3/2}\sqrt{\pi}}{x^{1/4}} e^{-2\lambda\sqrt{x}}, \quad p_{Z_1Z_2}(x) \sim \frac{1}{\sqrt{2\pi x}} e^{-x}, \quad p_Z(x) \sim \frac{1}{\sqrt{2\pi}|z|^{3/2}} e^{-1/|z|}.$$

With ease, our algebra correctly determines that $X_1X_2 \equiv (-\frac{1}{4}, 2\lambda, \frac{1}{2})$, $Z_1Z_2 \equiv (-\frac{1}{2}, 1, 1)$ and $Z \equiv (-\frac{3}{2}, 1, -1)$.

Example 7.3 (Reciprocal distributions) *Perhaps the most significant challenge with a tail algebra is correctly identifying the tail behaviour of reciprocal distributions. Here, we test the efficacy of our formulation with known reciprocal distributions.*

- Reciprocal normal: $X \sim \mathcal{N}(0,1) \equiv (0, 1/2, 2)$, *and* $X^{-1} \equiv (-2, 1/2, -2)$.
- Inverse exponential: $X \sim Exp(\lambda) \equiv (0, \lambda, 1)$, *and* $X^{-1} \equiv (-2, \lambda, -1)$.
- Inverse t-distribution: $X \equiv \mathcal{R}_\nu$, *and* $X^{-1} \equiv \mathcal{R}_2$.
- Inverse Cauchy: $X \equiv \mathcal{R}_2$, *it is known* X^{-1} *has the same distribution and our theory predicts* $X^{-1} \equiv \mathcal{R}_2$.

Example 7.4 (Cauchy distribution) *A simple special case of the Student T distribution is the Cauchy distribution, which arises as the ratio of two standard normal random variables. For* $X \sim \mathcal{N}(0,1)$, $X \equiv (0, 1/2, 2)$ *and* $X^{-1} \equiv (-2, 1/2, -2)$. *Hence, the multiplication operation correctly predicts that the ratio of two standard normal random variables is in* $\mathcal{R}_2$.

Example 7.5 (Student T distribution) *Let* X *be a standard normal random variable, and* V *a chi-squared random variable with* ν *degrees of freedom. The random variable* $T = X/\sqrt{V/\nu}$ *is* t-distributed *with* ν *degrees of freedom. Since* $V \equiv (\nu/2 - 1, 1/2, 1)$, *multiplying by the constant* $1/\nu$ *reveals* $V/\nu \equiv (\nu/2 - 1, 1/(2\nu), 1)$. *Applying the square root operation,* $\sqrt{V/\nu} \equiv (\nu - 1, 1/(2\nu), 2)$. *To compute the division operation, we first take the reciprocal to find* $(V/\nu)^{-1/2} \equiv (-\nu - 1, 1/(2\nu), -2)$. *Finally, since* $\rho = -2 < 1$ *for this random variable, the multiplication operation with* $X \equiv (0, 1/2, 2)$ *yields* $T \equiv \mathcal{R}_{\nu+1}$, *and so the density of* T *is asymptotically* $cx^{-\nu-1}$ *as* $x \to \infty$. *Indeed, the density of* T *satisfies* $p_T(x) = c_\nu(1 + x^2/\nu)^{-(\nu+1)/2}$ *where* $c_\nu = \Gamma(\frac{\nu+1}{2})/\Gamma(\frac{\nu}{2})(\nu\pi)^{-1/2}$, *which exhibits the predicted tail behaviour.*

Example 7.6 *Log-normal distribution Although the log-normal distribution does not lie in* $\mathcal{G}$, *the existence of log-normal tails arising from the multiplicative central limit theorem is suggested by our algebra. Let* $X_1, X_2, \ldots$ *be independent standard normal random variables and let* $Z_k = X_1 \cdots X_{2^k}$ *for each* $k = 1, 2, \ldots$. *By the multiplicative central limit theorem, letting* $\tau = \exp(\mathbb{E}\log|X_i|) \approx 1.13$, $\left(\frac{X_1 \cdots X_n}{\tau}\right)^{1/\sqrt{n}}$ *converges in distribution as* $n \to \infty$ *to a log-normal random variable* Z *with density*

$$p_Z(x) = \frac{1}{x\sqrt{2\pi}} \exp(-\tfrac{1}{2}(\log x)^2).$$

Therefore, the same is true for $V_k = (Z_k/\tau)^{2^{-k/2}}$. *Using our algebra, we will attempt to reproduce the tail of this density. Letting* $\tilde{Z}_k = X_{2^k} \cdots X_{2^{k+1}}$, *we see that* $Z_{k+1} = Z_k\tilde{Z}_k$, *and* $Z_k, \tilde{Z}_k$ *are iid. Let* $Z_k \equiv (\nu_k, \sigma_k, \rho_k)$, *by induction using the multiplication operation, we find that* $\nu_{k+1} = \frac{1}{\mu}\left(\frac{2\nu_k}{\rho_k} - \frac{1}{2}\right) = \nu_k - \frac{\rho_k}{4}$, $\sigma_{k+1} = \mu\,(\sigma_k\rho_k)^{\frac{2}{\mu\rho_k}} = \frac{2}{\rho_k}\,(\sigma_k\rho_k) = 2\sigma_k$, *and* $\rho_{k+1} = \frac{1}{\mu} = \frac{\rho_k}{2}$. *Since* $\rho_0 = 2$, $\sigma_0 = 1/2$, *and* $\nu_0 = 0$, *we find that* $\rho_k = 2^{1-k}$ *and* $\sigma_k = 2^{k-1}$.

Furthermore, $\nu_{k+1} = \nu_k - 2^{-k-1}$ *and so* $\nu_k = -1 + 2^{-k}$. *Therefore* $Z_k \equiv (-1+2^{-k}, 2^{k-1}, 2^{1-k})$, *and*

$$V_k \equiv (-1 + 2^{-k/2}, 2^{k-1}\tau^{-2^{1-k}}, 2^{1-k/2}),$$

and letting $\epsilon_k = 2^{-k/2}$, *the tail behaviour of the density of* V_k *satisfies*

$$\begin{aligned} p_k(x) &\sim c_k x^{-1+\epsilon_k} \exp\left(-\frac{\epsilon_k^{-2}}{2\tau^{-2\epsilon_k^2}} x^{2\epsilon_k}\right) \\ &\sim c_k x^{-1+\epsilon_k} \exp\left(-\frac{1}{2\tau^{-2\epsilon_k^2}} \left(\frac{x^{\epsilon_k}-1}{\epsilon_k}\right)^2\right) \approx c_k x^{-1} \exp\left(-\frac{1}{2}(\log x)^2\right), \end{aligned}$$

as $x \to \infty$, *where the approximation improves as* k *gets larger. The quality of this approximation is shown in fig. 7.2.*

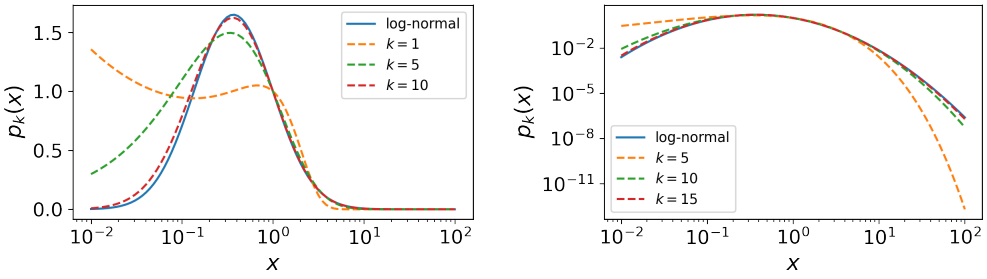


Figure 7.2: Estimation of the log-normal density by tail algebra applied to V_k.

Operations in the Generalized Gamma Algebra

Whereas table 7.1 provides a summary of our theory useful for referencing, in this section we provide additional explanation and references for how operations on random variables affect their GGA tails.

Ordering. A total ordering is imposed on the equivalence classes of $\mathcal{G}$ according to the heaviness of tails. In particular, we say that $(\nu_1, \sigma_1, \rho_1) \leq (\nu_2, \sigma_2, \rho_2)$ if $(x^{\nu_1} e^{-\sigma_1 x^{\rho_1}})/(x^{\nu_2} e^{-\sigma_2 x^{\rho_2}})$ is bounded as $x \to \infty$. As usual, we say $(\nu_1, \sigma_1, \rho_1) < (\nu_2, \sigma_2, \rho_2)$ if $(\nu_1, \sigma_1, \rho_1) \leq (\nu_2, \sigma_2, \rho_2)$ but $(\nu_1, \sigma_1, \rho_1) \not\equiv (\nu_2, \sigma_2, \rho_2)$.

Addition. Tails of this form are closed under addition. Combining subexponentiality for $\rho < 1$ [AA10, Chapter X.1], with [Asm+17, Thm 3.1 & eqn. (8.3)],

Proposition 7.1 *Denoting the addition of random variables (additive convolution of densities) by* $\oplus$*,*

$$(\nu_1,\sigma_1,\rho_1)\oplus(\nu_2,\sigma_2,\rho_2)$$
$$\equiv\begin{cases}\max\{(\nu_1,\sigma_1,\rho_1),(\nu_2,\sigma_2,\rho_2)\} & \text{if } \rho_1\neq\rho_2 \text{ or } \rho_1,\rho_2<1\\ (\nu_1+\nu_2+1,\min\{\sigma_1,\sigma_2\},1) & \text{if } \rho_1=\rho_2=1\\ (\nu_1+\nu_2+1-\frac{\rho}{2},(\sigma_1^{-\frac{1}{\rho-1}}+\sigma_2^{-\frac{1}{\rho-1}})^{1-\rho},\rho) & \text{if } \rho=\rho_1=\rho_2>1.\end{cases}\tag{7.3}$$

Powers. For all exponents $\beta>0$, by invoking a change of variables $x\mapsto x^\beta$, it is easy to show that $(\nu,\sigma,\rho)^\beta\equiv\left(\frac{\nu+1}{\beta}-1,\sigma,\frac{\rho}{\beta}\right)$. We *define* negative powers and reciprocals equivalently to positive powers in the case $\beta<0$. This equivalence cannot be proven to hold in general since we cannot determine tail asymptotics of the reciprocal without knowledge of its behaviour around zero. Therefore, we implicitly assume that the behaviour around zero mimics the tail behaviour, that is, eq. (7.1) holds as $x\to 0^+$. However, this can only hold provided $(\nu+1)/\rho>0$ and $\rho\neq 0$. In all other cases, including $\mathcal{R}_\nu$, we assume that the density of X approaches a nonzero value near zero, and define the reciprocal to be $\mathcal{R}_2$.

Multiplication. For any $c\in\mathbb{R}\backslash\{0\}$, it can be readily seen from a change of variables $x\mapsto cx$ that $c(\nu,\sigma,\rho)=(\nu,\sigma/|c|^\rho,\rho)$. The class $\mathcal{G}$ is also closed under multiplication (assuming independence of random variables), as we show in the following result — the proof is delayed to Appendix C.

Proposition 7.2 *Denoting the multiplication of independent random variables (multiplicative convolution) by* $\otimes$*,*

$$(\nu_1,\sigma_1,\rho_1)\otimes(\nu_2,\sigma_2,\rho_2)\equiv\begin{cases}\left(\frac{1}{\mu}\left(\frac{\nu_1}{|\rho_1|}+\frac{\nu_2}{|\rho_2|}+\frac{1}{2}\right),\sigma,-\frac{1}{\mu}\right) & \text{if } \rho_1,\rho_2<0\\ \left(\frac{1}{\mu}\left(\frac{\nu_1}{\rho_1}+\frac{\nu_2}{\rho_2}-\frac{1}{2}\right),\sigma,\frac{1}{\mu}\right) & \text{if } \rho_1,\rho_2>0\\ \mathcal{R}_{|\nu_1|} & \text{if } \rho_1\leq 0,\rho_2>0\\ \mathcal{R}_{\min\{|\nu_1|,|\nu_2|\}} & \text{if } \rho_1=0,\rho_2=0\end{cases}$$

where $\mu=\frac{1}{|\rho_1|}+\frac{1}{|\rho_2|}=\frac{|\rho_1|+|\rho_2|}{|\rho_1\rho_2|}$ *and* $\sigma=\mu(\sigma_1|\rho_1|)^{\frac{1}{\mu|\rho_1|}}(\sigma_2|\rho_2|)^{\frac{1}{\mu|\rho_2|}}$.

Product of Densities. We can also consider a product of densities operation acting on two random variables X,Y, denoted $X\&Y$, by $p_{X\&Y}(x)=cp_X(x)p_Y(x)$, where $c>0$ is an appropriate normalizing constant and $p_X,p_Y,p_{X\&Y}$ are the densities of X, Y, and $X\&Y$, respectively. In terms of the equivalence classes:

$$(\nu_1,\sigma_1,\rho_1)\&(\nu_2,\sigma_2,\rho_2)\equiv\begin{cases}(\nu_1+\nu_2,\sigma_1,\rho_1) & \text{if } \rho_1<\rho_2\\ (\nu_1+\nu_2,\sigma_1+\sigma_2,\rho) & \text{if } \rho=\rho_1=\rho_2\\ (\nu_1+\nu_2,\sigma_2,\rho_2) & \text{otherwise.}\end{cases}$$

Note that this particular operation does not require either p_X or p_Y to be normalized — only the tail behaviour is needed. We may also use this to work out the tail behaviour of a posterior density, provided the tail behaviour of the likelihood in the parameters is known.

Lipschitz Functions. There are many multivariate functions that cannot be readily represented in terms of the operations covered thus far. For these, it is important to specify the tail behaviour of pushforward measures under Lipschitz-continuous functions. Fortunately, this is covered by lemma 7.2 below, presented in [Led01, Proposition 1.3]. Hölder-continuous functions can also be represented as a composition of a power operation and a Lipschitz-continuous function.

Lemma 7.2 *For any Lipschitz continuous function $f : \mathbb{R}^d \to \mathbb{R}$ satisfying $\|f(x) - f(y)\| \leq L\|x - y\|$ for $x, y \in \mathbb{R}^d$, there is $f(X_1, \ldots, X_d) \equiv L \max\{X_1, \ldots, X_d\}$.*

Power Law Approximation. Note that as $x \to \infty$, $p_{|X|}(x) \sim cx^\nu e^{-\sigma x^\rho} = \tilde{c}x^\nu e^{-\sigma\rho\frac{x^\rho-1}{\rho}} \approx \tilde{c}x^\nu e^{-\sigma\rho\log x}$

$$p_{|X|}(x) \sim cx^\nu e^{-\sigma x^\rho} = \tilde{c}x^\nu e^{-\sigma(x^\rho-0)} = \tilde{c}x^\nu e^{-\sigma\rho\frac{x^\rho-1}{\rho}} \approx \tilde{c}x^\nu e^{-\sigma\rho\log x} = \tilde{c}x^{\nu-\sigma\rho},$$

where we have used the approximation $\log x = \rho^{-2}(x^\rho - 1) + \mathcal{O}(\rho^2)$. Consequently, we can represent tails of this form by the Student t distribution with $|\nu - \sigma\rho| - 1$ degrees of freedom. In practice, we find this approximation tends to *overestimate* the heaviness of the tail. Alternatively, the generalized Gamma density (7.2) satisfies $\mathbb{E}X^r = \sigma^{-r/\rho}\Gamma(\frac{\nu+1+r}{\rho})/\Gamma(\frac{\nu+1}{\rho})$ for $r > 0$. Let $\alpha > 0$ be such that $\mathbb{E}X^\alpha = 2$. By Markov's inequality, the tail of X satisfies $\mathbb{P}(X > x) \leq 2x^{-\alpha}$. Therefore, we can represent tails of this form by the Student t distribution with $\alpha + 1$ degrees of freedom (generate $X \sim t_\alpha$). In practice, we find this approximation to be more accurate, and is hence used in Section 4.1.

List of univariate distributions

Here we provide an enumeration of common parametric distributions and their corresponding GGA parameterizations.

Table 7.2: List of univariate distributions

Name	Support	Density $p(x)$	Class
Benktander Type II	$(0, \infty)$	$e^{\frac{a}{b}(1-x^b)}x^{b-2}(ax^b - b + 1)$	$(2b-2, \frac{a}{b}, b)$
Beta prime distribution	$(0, \infty)$	$\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}x^{\alpha-1}(1+x)^{-\alpha-\beta}$	$\mathcal{R}_{\beta+1}$

Burr distribution	$(0,\infty)$	$ckx^{c-1}(1+x^c)^{-k-1}$	$\mathcal{R}_{ck+1}$
Cauchy distribution	$(-\infty,\infty)$	$(\pi\gamma)^{-1}\left[1+\left(\frac{x-x_0}{\gamma}\right)^2\right]^{-1}$	$\mathcal{R}_2$
Chi distribution	$(0,\infty)$	$\frac{1}{2^{k/2-1}\Gamma(k/2)}x^{k-1}e^{-x^2/2}$	$(k-1,\frac{1}{2},2)$
Chi-squared distribution	$(0,\infty)$	$\frac{1}{2^{k/2}\Gamma(k/2)}x^{\frac{k}{2}-1}e^{-x/2}$	$(\frac{k}{2}-1,\frac{1}{2},1)$
Dagum distribution	$(0,\infty)$	$\frac{ap}{x}\left(\frac{x}{b}\right)^{ap}\left(\left(\frac{x}{b}\right)^a+1\right)^{-p-1}$	$\mathcal{R}_{a+1}$
Davis distribution	$(0,\infty)$	$\propto (x-\mu)^{-1-n}/\left(e^{\frac{b}{x-\mu}}-1\right)$	$(-1-n,b,-1)$
Exponential distribution	$(0,\infty)$	$\lambda e^{-\lambda x}$	$(0,\lambda,1)$
F distribution	$(0,\infty)$	$\propto x^{d_1/2-1}(d_1x+d_2)^{-(d_1+d_2)/2}$	$\mathcal{R}_{d_2/2+1}$
Fisher z-distribution	$(-\infty,\infty)$	$\propto \frac{e^{d_1x}}{(d_1e^{2x}+d_2)^{(d_1+d_2)/2}}$	$(0,d_2,1)$
Frechet distribution	$(0,\infty)$	$\frac{\alpha}{\lambda}\left(\frac{x-m}{\lambda}\right)^{-1-\alpha}e^{-\left(\frac{x-m}{\lambda}\right)^{-\alpha}}$	$(-1-\alpha,\lambda^\alpha,-\alpha)$
Gamma distribution	$(0,\infty)$	$\frac{\beta^\alpha}{\Gamma(\alpha)}x^{\alpha-1}e^{-\beta x}$	$(\alpha-1,\beta,1)$
Gamma/Gompertz distribution	$(0,\infty)$	$bse^{bx}\beta^s/(\beta-1+e^{bx})^{s+1}$	$(0,bs,1)$
Gen. hyperbolic distribution	$(-\infty,\infty)$	$\propto e^{\beta(x-\mu)}\frac{K_{\lambda-1/2}(\alpha\sqrt{\delta^2+(x-\mu)^2})}{(\delta^2+(x-\mu)^2)^{1/4-\lambda/2}}$	$(\lambda-1,\alpha-\beta,1)$
Gen. Normal distribution	$(-\infty,\infty)$	$\frac{\beta}{2\alpha\Gamma(1/\beta)}\exp\left(-\left(\frac{\lvert x-\mu\rvert}{\alpha}\right)^\beta\right)$	$(0,\alpha^{-\beta},\beta)$
Geometric stable distribution	$(-\infty,\infty)$	no closed form	$\mathcal{R}_{\alpha+1}$
Gompertz distribution	$(0,\infty)$	$\sigma\eta\exp(\eta+\sigma x-\eta e^{\sigma x})$	$\mathcal{L}$
Gumbel distribution	$(0,\infty)$	$\beta^{-1}e^{-(\beta^{-1}(x-\mu)+e^{-\beta^{-1}(x-\mu)})}$	$(0,\frac{1}{\beta},1)$
Gumbel Type II distribution	$(0,\infty)$	$\alpha\beta x^{-\alpha-1}e^{-\beta x^{-\alpha}}$	$(-\alpha-1,\beta,-\alpha)$
Holtsmark distribution	$(-\infty,\infty)$	no closed form	$\mathcal{R}_{5/2}$
Hyperbolic secant distribution	$(-\infty,\infty)$	$\frac{1}{2}\text{sech}\left(\frac{\pi x}{2}\right)$	$(0,\frac{\pi}{2},1)$
Inv. chi-squared distribution	$(0,\infty)$	$\frac{2^{-k/2}}{\Gamma(k/2)}x^{-k/2-1}e^{-1/(2x)}$	$(-\frac{k}{2}-1,\frac{1}{2},-1)$

Inv. gamma distribution	$(0,\infty)$	$\frac{\beta^\alpha}{\Gamma(\alpha)}x^{-\alpha-1}e^{-\beta/x}$	$(-\alpha-1,\beta,-1)$
Levy distribution	$(0,\infty)$	$\sqrt{\frac{c}{2\pi}}(x-\mu)^{-3/2}e^{-\frac{c}{2(x-\mu)}}$	$(-\frac{3}{2},\frac{c}{2},-1)$
Laplace distribution	$(-\infty,\infty)$	$\frac{1}{2\lambda}\exp\left(-\frac{\lvert x-\mu\rvert}{\lambda}\right)$	$(0,\frac{1}{\lambda},1)$
Logistic distribution	$(-\infty,\infty)$	$\frac{e^{-(x-\mu)/\lambda}}{\lambda(1+e^{-(x-\mu)/\lambda})^2}$	$(0,\frac{1}{\lambda},1)$
Log-Cauchy distribution	$(0,\infty)$	$\frac{\sigma}{x\pi}((\log x-\mu)^2+\sigma^2)^{-1}$	$\mathcal{R}_1$
Log-Laplace distribution	$(0,\infty)$	$\frac{1}{2\lambda x}\exp\left(-\frac{\lvert\log x-\mu\rvert}{\lambda}\right)$	$\mathcal{R}_{1/\lambda+1}$
Log-logistic distribution	$(0,\infty)$	$\frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1}\left(1+\left(\frac{x}{\alpha}\right)^\beta\right)^{-2}$	$\mathcal{R}_{\beta+1}$
Log-t distribution	$(0,\infty)$	$\propto x^{-1}(1+\frac{1}{\nu}(\log x-\mu)^2)^{-\frac{\nu+1}{2}}$	$\mathcal{R}_1$
Lomax distribution	$(0,\infty)$	$\frac{\alpha}{\lambda}\left(1+\frac{x}{\lambda}\right)^{-\alpha-1}$	$\mathcal{R}_{\alpha+1}$
Maxwell-Boltzmann distribution	$(0,\infty)$	$\sqrt{\frac{2}{\pi}}\frac{x^2e^{-x^2/(2\sigma^2)}}{\sigma^3}$	$(2,\frac{1}{2\sigma^2},2)$
Normal distribution	$(-\infty,\infty)$	$\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$	$(0,\frac{1}{2\sigma^2},2)$
Pareto distribution	(x_0,∞)	$\alpha x_0^\alpha x^{-\alpha-1}$	$\mathcal{R}_{\alpha+1}$
Rayleigh distribution	$(0,\infty)$	$\frac{x}{\sigma^2}e^{-x^2/(2\sigma^2)}$	$(1,\frac{1}{2\sigma^2},2)$
Rice distribution	$(0,\infty)$	$\frac{x}{\sigma^2}\exp\left(-\frac{(x^2+\nu^2)}{2\sigma^2}\right)I_0\left(\frac{x\nu}{\sigma^2}\right)$	$(\frac{1}{2},\frac{1}{2\sigma^2},2)$
Skew normal distribution	$(-\infty,\infty)$	no closed form	$(0,\frac{1}{2\sigma^2},2)$
Slash distribution	$(-\infty,\infty)$	$\frac{1-e^{-\frac{1}{2}x^2}}{\sqrt{2\pi}x^2}$	$(-2,\frac{1}{2},2)$
Stable distribution	$(-\infty,\infty)$	no closed form	$\mathcal{R}_{\alpha+1}$
Student's t-distribution	$(-\infty,\infty)$	$\frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})}\left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}}$	$\mathcal{R}_{\nu+1}$
Tracy-Widom distribution	$(-\infty,\infty)$	no closed form	$(-\frac{3\beta}{4}-1,\frac{2\beta}{3},\frac{3}{2})$
Voigt distribution	$(-\infty,\infty)$	no closed form	$\mathcal{R}_2$
Weibull distribution	$(0,\infty)$	$\frac{\rho}{\lambda}\left(\frac{x}{\lambda}\right)^{\rho-1}e^{-(x/\lambda)^\rho}$	$(\rho-1,\lambda^{-\rho},\rho)$

The following densities are not supported by our algebra: Benini distribution; Benktander Type I distribution; Johnson's S_U-distribution; and the log-normal distribution. All of these densities exhibit log-normal tails.

Proofs of new results

Proof [Proof of Lemma 7.1] The proof relies on the following integral definition [Wat95, pg. 183] and asymptotic relation as $z \to \infty$ [Wat95, pg. 202] of the modified Bessel function $K_\nu(z)$ for $z > 0$ and $\nu \geq 0$,

$$K_\nu(z) = \frac{1}{2}\left(\frac{z}{2}\right)^\nu \int_0^\infty u^{-\nu-1} \exp\left(-u - \frac{z^2}{4u}\right) \mathrm{d}u \sim \sqrt{\frac{\pi}{2z}} e^{-z}. \tag{7.4}$$

We also make use of the known density for the product of two independent continuous random variables: if X and Y have densities p_X and p_Y respectively, then $Z = XY$ has density

$$p_Z(z) = \int_{\mathbb{R}} p_X(x) p_Y(z/x) |x|^{-1} \mathrm{d}x.$$

Exponentials. Recalling that the density of $X \sim \mathrm{Exp}(\lambda)$ is $p_X(x) = \lambda e^{-\lambda x}$ for $x \geq 0$, for $Z = XY$ where $X \sim \mathrm{Exp}(\lambda_1)$ and $Y \sim \mathrm{Exp}(\lambda_2)$ are independent,

$$p_Z(z) = \int_0^\infty x^{-1} \lambda_1 e^{-\lambda_1 x} \lambda_2 e^{-\lambda_2 z/x} \mathrm{d}x = \lambda_1 \lambda_2 \int_0^\infty x^{-1} e^{-\lambda_1 x - \lambda_2 z/x} \mathrm{d}x.$$

Since $2K_0(2\sqrt{z}) = \int_0^\infty u^{-1} \exp(-u - \frac{z}{u}) \mathrm{d}u$, let $u = \lambda_1 v$, so that $\mathrm{d}u = \lambda_1 \mathrm{d}v$,

$$2K_0(2\sqrt{\lambda_1 \lambda_2 z}) = \int_0^\infty u^{-1} \exp\left(-\lambda_1 v - \lambda_2 \frac{z}{v}\right) \mathrm{d}v.$$

Therefore, letting $\lambda = \sqrt{\lambda_1 \lambda_2}$,

$$p_Z(z) = 2\lambda^2 K_0(2\lambda\sqrt{z}) \sim \sqrt{\pi} \lambda^{3/2} z^{-1/4} e^{-2\lambda z^{1/2}}.$$

Normals. Recalling that the density of $X \sim \mathcal{N}(0,1)$ is $p_X(x) = (2\pi)^{-1/2} \exp(-\frac{1}{2}x^2)$, for $Z = XY$ where $X, Y \sim \mathcal{N}(0,1)$ are independent,

$$\begin{aligned} p_Z(z) &= \frac{1}{2\pi} \int_{\mathbb{R}} |x|^{-1} e^{-\frac{1}{2}x^2} e^{-\frac{1}{2}z^2/x^2} \mathrm{d}x \\ &= \frac{1}{\pi} \int_0^\infty x^{-1} e^{-\frac{1}{2}x^2 - \frac{1}{2}z^2/x^2} \mathrm{d}x \\ &= \frac{1}{\pi} \int_0^\infty x^{-1} e^{-\frac{1}{2}x^2 - \frac{1}{2}z^2/x^2} \mathrm{d}x. \end{aligned}$$

Let $u = \frac{1}{2}x^2$ so that $\mathrm{d}u = x\mathrm{d}x$ and

$$K_\nu(z) = z^\nu \int_0^\infty x^{-2\nu-1} \exp\left(-\frac{1}{2}x^2 - \frac{z^2}{2x^2}\right) \mathrm{d}x.$$

In particular, for any $z \in \mathbb{R}$,

$$K_0(|z|) = \int_0^\infty x^{-1} \exp\left(-\frac{1}{2}x^2 - \frac{z^2}{2x^2}\right) \mathrm{d}x, \tag{7.5}$$

and so

$$p_Z(z) = \frac{1}{\pi}K_0(|z|) \sim \frac{1}{\sqrt{2\pi|z|}}e^{-|z|}.$$

Reciprocal Normals. Finally, by a change of variables, we note that the density of X^{-1} where $X \sim \mathcal{N}(0,1)$ is $p_{X^{-1}}(x) = (2\pi)^{-1/2}x^{-2}\exp(-\frac{1}{2x^2})$. Therefore, the density of $Z = 1/(XY)$ where $X, Y \sim \mathcal{N}(0,1)$ are independent is given by

$$\begin{aligned}
p_Z(z) &= \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}x^2}e^{-\frac{1}{2x^2}} \frac{x^2}{\sqrt{2\pi}z^2}e^{-\frac{x^2}{2z^2}} \frac{1}{|x|}\mathrm{d}x \\
&= \frac{1}{2\pi z^2}\int_{\mathbb{R}} e^{-\frac{1}{2x^2}-\frac{x^2}{2z^2}} \frac{1}{|x|}\mathrm{d}x \\
&= \frac{1}{\pi z^2}\int_0^\infty e^{-\frac{1}{2x^2}-\frac{x^2}{2z^2}} \frac{1}{x}\mathrm{d}x \\
&= \frac{1}{\pi z^2}K_0(|z|^{-1}) \sim \sqrt{\frac{1}{2\pi}}|z|^{-3/2}e^{-|z|^{-1}},
\end{aligned}$$

where we have once again used (7.5). ■

Recall that the Mellin transform of a function f on $(0,\infty)$ is given by

$$\mathcal{M}_s[f] = \int_0^\infty x^{s-1}f(x)\mathrm{d}x.$$

Letting p_{XY} denote the density of the product of independent random variables X, Y with respective densities p_X and p_Y, $\mathcal{M}_s[p_{XY}] = \mathcal{M}_s[p_X]\mathcal{M}_s[p_Y]$. There is

$$\mathcal{M}_s[cx^\nu e^{-\sigma x^\rho}] = \frac{c\sigma^{-\nu/\rho}}{\rho}\sigma^{-s/\rho}\Gamma\left(\frac{\nu}{\rho} + \frac{s}{\rho}\right).$$

To facilitate the proof of Proposition 7.2, we define the Fox H-function

$$H_{p,q}^{m,n}\left[z \,\middle|\, \begin{matrix}(a_1,A_1),\ldots,(a_p,A_p)\\(b_1,B_1),\ldots,(b_q,B_q)\end{matrix}\right]$$

as the inverse Mellin transform of

$$\Theta(s) = z^{-s} \frac{\prod_{j=1}^{m} \Gamma(b_j + B_j s) \cdots \prod_{j=1}^{n} \Gamma(1 - a_j - A_j s)}{\prod_{j=m+1}^{q} \Gamma(1 - b_j - B_j s) \prod_{j=n+1}^{p} \Gamma(a_j + A_j s)}.$$

An important property of the Fox H-function is its asymptotic behaviour as $z \to \infty$. From [MSH09, Theorem 1.3],

$$H_{p,q}^{q,0}\left[z \left| \begin{matrix} (a_1, A_1), \ldots, (a_p, A_p) \\ (b_1, B_1), \ldots, (b_q, B_q) \end{matrix} \right.\right] \sim c x^{(\delta + \frac{1}{2})/\mu} \exp(-\mu \beta^{-1/\mu} x^{1/\mu}), \qquad \text{as } x \to \infty,$$

for some constant $c > 0$, where $\beta = \prod_{j=1}^{p} (A_j)^{-A_j} \prod_{j=1}^{q} B_j^{B_j}$, $\mu = \sum_{j=1}^{q} B_j - \sum_{j=1}^{p} A_j$, and $\delta = \sum_{j=1}^{q} b_j - \sum_{j=1}^{p} a_j + \frac{p-q}{2}$.

Proof [Proof of Proposition 7.2] The $\rho_1 \leq 0, \rho_2 > 0$ and $\rho_1 = \rho_2 = 0$ cases follow from Breiman's lemma [BDM16, Lemma B.5.1]. Our argument proceeds similar to [Asm+17]. Assume that $\rho_1, \rho_2 > 0$ and let $0 < \epsilon < 1$ be such that $0 < a_- < a_+ < 1$, where

$$a_+ = \frac{(1+\epsilon)\rho_2}{\rho_1 + \rho_2}, \qquad a_- = 1 - \frac{(1+\epsilon)\rho_1}{\rho_1 + \rho_2}.$$

Then for $\rho = \frac{\rho_1 \rho_2}{\rho_1 + \rho_2}$, if $X \equiv (\nu_1, \sigma_1, \rho_1)$ and $Y \equiv (\nu_2, \sigma_2, \rho_2)$, then

$$\begin{aligned} \mathbb{P}(XY > x, X \notin [x^{a_-}, x^{a_+}]) &\leq \mathbb{P}(X > x^{a_+}) + \mathbb{P}(Y > x^{1-a_-}) \\ &\sim c_1 x^{\nu_1 a_+} e^{-\sigma_1 x^{\rho_1 a_+}} + c_2 x^{\nu_2 (1-a_-)} e^{-\sigma_2 x^{\rho_2 (1-a_-)}} \\ &\leq \left(c_1 x^{\nu_1 a_+} + c_2 x^{\nu_2 (1-a_-)}\right) e^{-\min\{\sigma_1, \sigma_2\} x^{(1+\epsilon)\rho}} = o(x^{\nu} e^{-\sigma x^{\rho}}), \end{aligned}$$

for any $\nu, \sigma > 0$. Hence, it will suffice to show the claimed tail asymptotics for the generalized Gamma distribution. In this case, since $a_- > 0$ and $a_+ < 1$, the tail of the distribution for the product of X, Y depends only on the tail of the distributions for X and Y.

Therefore, assume without loss of generality that $p_X(x) = c_X x^{\nu_1} e^{-\sigma_1 x^{\rho_1}}$ and $p_Y(x) = c_Y x^{\nu_2} e^{-\sigma_2 x^{\rho_2}}$. Then

$$\mathcal{M}_s[p_{XY}] = c_X c_Y \frac{\sigma_1^{-\nu_1/\rho_1}}{\rho_1} \frac{\sigma_2^{-\nu_2/\rho_2}}{\rho_2} \left(\sigma_1^{1/\rho_1} \sigma_2^{1/\rho_2}\right)^{-s} \Gamma\left(\frac{\nu_1}{\rho_1} + \frac{s}{\rho_1}\right) \Gamma\left(\frac{\nu_2}{\rho_2} + \frac{s}{\rho_2}\right).$$

Consequently,

$$p_{XY}(z) = c_X c_Y \frac{\sigma_1^{-\nu_1/\rho_1}}{\rho_1} \frac{\sigma_2^{-\nu_2/\rho_2}}{\rho_2} H_{p,q}^{m,n}\left[\sigma_1^{1/\rho_1} \sigma_2^{1/\rho_2} z \left| \begin{matrix} - \\ (\frac{\nu_1}{\rho_1}, \frac{1}{\rho_1}), (\frac{\nu_2}{\rho_2}, \frac{1}{\rho_2}) \end{matrix} \right.\right]$$

Computing the corresponding β, δ, μ for the asymptotic expansion, we find that

$$\mu = \frac{1}{\rho_1} + \frac{1}{\rho_2}, \qquad \delta = \frac{\nu_1}{\rho_1} + \frac{\nu_2}{\rho_2} - 1, \qquad \beta = \rho_1^{-1/\rho_1} \rho_2^{-1/\rho_2}.$$

Consequently, for some $c > 0$,

$$p_{XY}(z) \sim cz^{\frac{1}{\mu}(\frac{1}{2}+\delta)} \exp\left(-\mu\beta^{-\frac{1}{\mu}}(\sigma_1^{1/\rho_1}\sigma_2^{1/\rho_2})^{\frac{1}{\mu}} z^{\frac{1}{\mu}}\right),$$

which completes the $\rho_1, \rho_2 > 0$ case. The final case follows by composing the multiplication and reciprocal operations. Note that

$$\begin{aligned}
(\nu_1, \sigma_1, -\rho_1)^{-1} \otimes (\nu_2, \sigma_2, -\rho_2)^{-1} &\equiv (-\nu_1 - 2, \sigma_1, \rho_1) \otimes (-\nu_2 - 2, \sigma_2, \rho_2) \\
&\equiv \left(\frac{1}{\mu}\left(\frac{-\nu_1 - 2}{\rho_1} + \frac{-\nu_2 - 2}{\rho_2} - \frac{1}{2}\right), \sigma, \frac{1}{\mu}\right) \\
&\equiv \left(\frac{1}{\mu}\left(\frac{-\nu_1}{\rho_1} + \frac{-\nu_2}{\rho_2} - 2\mu - \frac{1}{2}\right), \sigma, \frac{1}{\mu}\right) \\
&\equiv \left(\frac{1}{\mu}\left(\frac{-\nu_1}{\rho_1} + \frac{-\nu_2}{\rho_2} - \frac{1}{2}\right) - 2, \sigma, \frac{1}{\mu}\right),
\end{aligned}$$

and therefore

$$(\nu_1, \sigma_1, -\rho_1) \otimes (\nu_2, \sigma_2, -\rho_2) \equiv \left(\frac{1}{\mu}\left(\frac{\nu_1}{\rho_1} + \frac{\nu_2}{\rho_2} + \frac{1}{2}\right), \sigma, -\frac{1}{\mu}\right).$$

■

7.4 Implementation

Compile-time static analysis

To illustrate an implementation of GGA for static analysis, we sketch the operation of the PPL compiler at a high-level and defer to the supplementary code for details. A probabilistic program is first inspected using Python's built-in `ast` module and transformed to static single assignment (SSA) form [RWZ88]. Next, standard compiler optimizations (e.g. dead code elimination, constant propagation) are applied and an execution of the optimized program is traced [WSG11; Bin+19] and accumulated in a directed acyclic graph representation. A breadth-first type checking pass, as seen in Algorithm 2, completes in linear time, and GGA results may be applied to implement `computeGGA()` using the following steps:

- If a node has no parents, then it is an atomic distribution and its tail parameters are known (Table 7.2)
- Otherwise, the node is an operation taking its potentially stochastic inputs (parents) to its output. Consult Table 7.1 for the output GGA tails.

Algorithm 2 Pseudocode for a GGA tails static analysis pass

Require: Abstract syntax tree for a PPL program
  frontier $\leftarrow$ [rv : Parents(rv) $= \emptyset$]
  GGAs $\leftarrow$ {}
  while frontier $\neq \emptyset$ **do**
    next $\leftarrow$ frontier.popLeft()
    GGAs[next] $\leftarrow$ computeGGA(next.op, next.parent)
    frontier $\leftarrow$ frontier + next.children()
  end while
  return GGA parameter estimates for all random variables

Representative distributions

For each (ν, σ, ρ) we make a carefully defined choice of p on $\mathbb{R}$ such that if $X \sim p$, then $X \equiv (\nu, \sigma, \rho)$. This way, any random variable $f(X)$, where f is 1-Lipschitz, will exhibit the correct tail, and so approximations of this form may be used for variational inference or density estimation. Let $X \equiv (\nu, \sigma, \rho)$ and $0 < \epsilon \ll 1$ denote a small parameter such that tails e^{-x^ϵ} are deemed to be "very heavy" (we chose $\epsilon = 0.1$).

($\rho \leq 0$) If $\rho \leq -1$, then $p_X(x) \sim cx^{-|\nu|}$. A prominent distribution on $\mathbb{R}$ with power law tails is the *Student t distribution*, in this case, with $|\nu| - 1$ degrees of freedom if $\nu < -1$ (generate $X \sim t_{|\nu|-1}$).

($\rho > \epsilon$) For moderately sized $\rho > 0$, we consider a symmetrized variant of the generalized Gamma density (Equation (7.2)).

($\rho \leq \epsilon$) If $X \equiv (\nu, \sigma, \rho)$ where ρ is small, then X will exhibit much heavier tails, and the generalized Gamma distribution in Case 1 will become challenging to sample from. In these cases, we expect that the tail of X should be well represented by a power law. The generalized Gamma density (Equation (7.2)) satisfies $\mathbb{E}X^r = \sigma^{-r/\rho}\Gamma(\frac{\nu+1+r}{\rho})/\Gamma(\frac{\nu+1}{\rho})$ for $r > 0$. Let $\alpha > 0$ be such that $\mathbb{E}X^\alpha = 2$. By Markov's inequality, the tail of X satisfies $\mathbb{P}(X > x) \leq 2x^{-\alpha}$. Therefore, we can represent tails of this form by the Student t distribution with $\alpha + 1$ degrees of freedom (generate $X \sim t_\alpha$).

Bulk correction by Lipschitz mapping

While a representative distribution will exhibit the desired tails, the target distribution's bulk may be very different from a generalized Gamma and result in poor distributional approximation. To address this, we propose splicing together the tails from a generalized Gamma with a flexible density approximation for the bulk. While many combinations are possible, in this work we rely on Lemma 7.2 and post-compose neural spline flows [Dur+19] (which are identity functions outside of a bounded interval) after properly initialized

generalized Gamma distributions. Optimizing the parameters of the flow results in good bulk approximation while simultaneously preserving the tail correctness guarantees attained by the GGA.

Example 7.7 *Let $A \in \mathbb{R}^{k\times k}$, $x, y \in \mathbb{R}^k$, with $x_i, y_i, A_{ij} \overset{iid}{\sim} \mathcal{N}(-1, 1)$. The distribution of $x^\top Ay = \sum_{i,j} x_i A_{ij} y_j$ is convolution of normal-powers [GG08] and has no convenient closed form expression. Using GGA's closure theorems (table 7.1), one can compute its tail parameters to be $(\frac{k}{2} - 1, \frac{3}{2}, \frac{2}{3})$.*

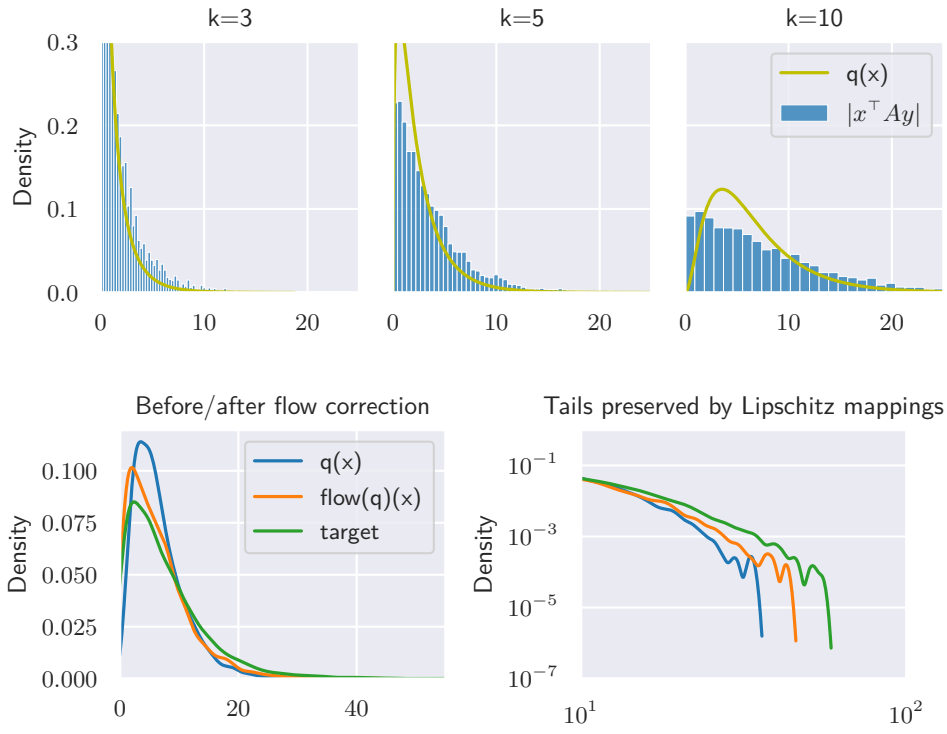


Figure 7.3: (Top) 5000 samples of $|x^\top Ay|$ vs the calibrated GGA density $q(x)$. While calibrated tails are provably guarantees, the target distribution's bulk differs from the assumed generalized Gamma representative distribution (section 7.4) for all k. To fix the bulk approximation, a normalizing flow is composed with the GGA representative to form $\text{flow}(q)(x)$. The bulk approximation is improved (bottom) while the tails continue to exhibit the same behavior (bottom right).

The GGA representative is a gamma distribution with the correct tails, but there is non-negligible error in the bulk where x is small. To address this, a learnable bijector can be optimized as in Figure 7.3 bottom left to correct the bulk approximation. Guaranteed by Lemma 7.2 and visualized in Figure 7.3 bottom right, the tails of the overall composition remain calibrated.

7.5 Experiments

In this section we demonstrate that GGA-based density estimation yields improvements across a variety of metrics. We consider the parametric family defined in Section 7.4 and compare against pushfowards of Normal distributions. To understand the individual effect of using a GGA base distribution over standard normals versus more expressive pushforward maps [Dur+19], we also report ablation results where normalizing flows are replaced by affine transforms as originally proposed in [Kuc+17]. All experiments are repeated for 50 trials, trained to convergence using the Adam optimizer with manually tuned learning rate, and conducted on i7-8700K CPU and GTX 1080 GPU hardware.

All target distributions in this section are expressed as generative PPL programs: Cauchy using a reciprocal normal, Chi2 using a sum of squared normals, IG (Inverse Gamma) using a reciprocal exponential, Normal using a sum of normals, and StudentT using a normal and Cauchy ratio. Doing so tasks the static analyzer to infer the target's tails and makes the analysis non-trivial. See supplementary for full details.

Our results in the following tables share a consistent narrative where a GGA base distribution rarely hurts and can significantly help with heavy tailed targets. Except for when targets are truly light tailed ($\alpha = \infty$ in Chi2 and Normal), GGA-based approximations are the only ones to reproduce appropriate GPD tail index $\hat{\alpha}$ in density estimation and achieve a passing Pareto $\hat{k}$ diagnostic [Yao+18a] below 0.2 in variational inference. When viewed through traditional evaluation metrics such as negative cross-entropy $H(p, q) = E_p \log p(X)$, ELBO $E_q \log \frac{q(X)}{p(X)}$, and importance-weighted autoencoder bound [BGS15] $E_q \log \sum_i^{1000} \frac{p(X)}{q(X)}$, GGA-based approximations remain favorable on almost all heavy-tailed targets and have negligible difference for light tailed targets. Less surprising is the result that adding a flow improved approximation metrics, as we expect the additional representation flexibility to be beneficial.

Density Estimation We minimize a Monte-Carlo estimate of the cross entropy $H(p, q) = -E_p[\log q(X)] \approx -\frac{1}{N} \sum_{i=1}^{N} \log q(x_i), \quad x_i \sim p$. The results are shown in Table 7.3 along with power-law tail index estimates [CSN09] $\hat{\alpha}$. Overall, we see that GGA performs better (lower NLL, $\hat{\alpha}$ closer to target) when the target has heavier tails (lower $\hat{\alpha}$ target/theory) and that the difference is smaller but still non-negligible for distributions such as Chi1 which possess tails heavier than Gaussian.

Variational Inference The optimization objective is the ELBO

$$E_q \log \frac{p(X)}{q(X)} \approx \frac{1}{N} \sum_{i=1}^{N} \log \frac{p(x_i)}{q(x_i)}, \quad x_i \sim q$$

Here, the density p must also be evaluated so for simplicity experiments in table 7.4 use closed-form marginalized densities for targets. The overall trends also show that GGA yields consistent improvements as measured by both ELBO and importance-weighted estimates of marginal likelihood and that the difference was greater when the tails of $p(z)$ were heavier. The $\hat{k}$ diagnostics [Yao+18a] corroborate our findings that variational inference succeeds

Table 7.3: Density estimation metrics attained (mean, standard deviation in parenthesis) on targets of varying tail index (smaller α = heavier tails). Higher negative cross entropy $-H(p,q) = E_p \log q(X)$ implies a better overall approximation (row maxes bolded) while close agreement between the target Pareto tail index α [CSN09] and its estimate $\hat{\alpha}$ in $q(x)$ suggest calibrated tails (closest in row bolded).

Target	Method Metric	Normal Affine	Normal Flow	GGA Affine	GGA Flow
Cauchy	$\hat{\alpha}$	7.7 (2.5)	7.1 (6.6)	2.1 (0.064)	**2 (0.067)**
($\alpha = 2$)	-H(p,q)	-1.4e7 (6.2e7)	-5.3e+10 (2.6e+11)	**-3.9e3 (56)**	-3.9e3 (55)
Chi2	$\hat{\alpha}$	**6.8 (2.4)**	6.4 (0.88)	5.5 (1.2)	5.2 (1.6)
($\alpha = \infty$)	-H(p,q)	**-2.8e3 (38)**	-2.9e3 (55)	-2.8e3 (26)	-2.8e3 (44)
IG	$\hat{\alpha}$	7.3 (1.7)	27 (39)	**1.9 (0.092)**	1.9 (0.092)
($\alpha = 2$)	-H(p,q)	-1.4e8 (6.2e8)	-4.3e9 (2.1e+10)	-4e3 (54)	**-3.9e3 (47)**
Normal	$\hat{\alpha}$	**8.4 (3.5)**	8.8 (4.6)	8.8 (2.8)	8.2 (4)
($\alpha = \infty$)	-H(p,q)	**-1.4e3 (19)**	-1.4e3 (19)	-1.4e3 (21)	-1.4e3 (24)
StudentT	$\hat{\alpha}$	7.7 (2.3)	13 (11)	**3.1 (0.16)**	3.3 (0.45)
($\alpha = 3$)	-H(p,q)	-3e3 (4.7e2)	**-2.7e3 (6.4e2)**	-3.6e3 (28)	-3.4e3 (42)

($\hat{k} < -1.2$) when a GGA with appropriately matched tails is used and fails ($\hat{k} > 1$) when Gaussian tails are erroneously imposed.

The targets in Table 7.3 and Table 7.4 are analyzed using the GGA. Note that Inverse Gamma ("IG") corresponds to the inverse exponential. We selected closed form targets so that the Pareto tail index α is known analytically and the quality of theoretical predictions as well as empirical results can be evaluated against. All experiments are repeated for 100 trials and $1,000$ samples from the model (as well as the approximation in VI) were used to compute each gradient estimate. Losses were trained until convergence, which all occurred in under $10,000$ iterations at a 0.05 learning rate and the Adam [KB14] optimizer.

SGD for least-squares linear regression

For inputs X and labels Y from a dataset $\mathcal{D}$, the least squares estimator for linear regression satisfies $\beta = \min_\beta \frac{1}{2}\mathbb{E}_{X,Y\sim\mathcal{D}}(Y - X\beta)^2$. To solve for this estimator, one can apply stochastic gradient descent (SGD) sampling over independent $X_k, Y_k \sim \mathcal{D}$ to obtain the sequence of iterations

$$\beta_{k+1} = (I - \delta X_k X_k^\top)\beta_k + \delta Y_k X_k$$

for a step size $\delta > 0$. For large δ, the iterates β_k typically exhibit heavy-tailed fluctuations; in this regard, this sequence of iterates has been used as a simple model for more general stochastic optimization dynamics [GSZ21; HM21]. In particular, generalization performance has been tied to the heaviness of the tails in the iterates [SSG19]. Here we use our algebra

Table 7.4: Variational inference metrics (mean, standard deviation in parenthesis) on targets of varying tail index (smaller α = heavier tails). Both the IWAE bound $E_q \log \sum_i^K \frac{p(X_i)}{q(X_i)}$ and the ELBO ($K = 1$) measure (a lower bound) on the marginal likelihood where larger is better (row maxes bolded). In Yao et al. [Yao+18a], a Pareto $\hat{k}$ diagnostic > 0.2 is interpreted as potentially problematic so only values below are bolded.

Target	Method Metric	Normal Affine	Normal Flow	GGA Affine	GGA Flow
Cauchy ($\alpha = 2$)	$\hat{k}$	0.46 (0.13)	0.35 (0.43)	**0.011 (0.0063)**	**0.034 (0.01)**
	ELBO	-0.19 (0.011)	-0.1 (0.028)	**1.4 (0.00027)**	1.4 (0.0015)
	IWAE	6.8 (0.031)	6.9 (0.15)	**8.3 (0.00028)**	8.3 (0.0015)
Chi2 ($\alpha = \infty$)	$\hat{k}$	0.26 (0.094)	0.23 (0.12)	**0.075 (0.07)**	**0.14 (0.1)**
	ELBO	-0.024 (0.0072)	-0.046 (0.034)	**-0.002 (0.003)**	-0.031 (0.031)
	IWAE	**6.9 (0.0066)**	6.9 (0.0098)	6.9 (0.0016)	6.9 (0.0067)
IG ($\alpha = 2$)	$\hat{k}$	13 (3.4)	0.63 (0.55)	11 (3.2)	5.7 (5.7)
	ELBO	-0.63 (6.5)	-1.5 (0.1)	**0.44 (4.2)**	-0.14 (0.9)
	IWAE	**2e3 (3.9e3)**	11 (23)	9.5e2 (1.6e3)	1.6e2 (1.6e2)
Normal ($\alpha = \infty$)	$\hat{k}$	**0.0055 (0.0082)**	**0.022 (0.017)**	**0.007 (0.007)**	**0.017 (0.014)**
	ELBO	**-0.000 (0.001)**	-0.00038 (0.0013)	-0.0002 (0.0006)	-0.00071 (0.001)
	IWAE	**6.9 (0.0005)**	6.9 (0.0013)	6.9 (0.00055)	6.9 (0.00094)
StudentT ($\alpha = 3$)	$\hat{k}$	0.53 (0.17)	0.21 (0.26)	**0.002 (0.003)**	**0.12 (0.064)**
	ELBO	-0.072 (0.0099)	-0.017 (0.0025)	**1.4 (0.00012)**	1.4 (0.0052)
	IWAE	6.9 (0.058)	6.9 (0.01)	**8.3 (0.00012)**	8.3 (0.0052)

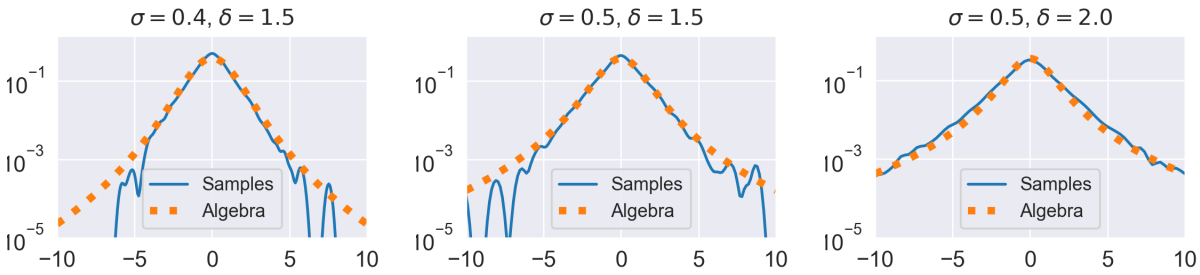


Figure 7.4: Density of iterates of SGD vs. predicted tail behaviour

to predict the tail behaviour in a simple one-dimensional setting where $X_k \sim \mathcal{N}(0, \sigma^2)$ and $Y_k \sim \mathcal{N}(0, 1)$. From classical theory [BDM16], it is known that X_k converges in distribution to a power law with tail exponent $\alpha > 0$ satisfying $\mathbb{E}|1 - \delta X_k^2|^\alpha = 1$. In fig. 7.4, we plot the density of the representative obtained using our algebra after 10^4 iterations against a kernel density estimate of the first 10^6 iterates when $\sigma \in \{0.4, 0.5\}$ and $\delta \in \{1.5, 2.0\}$. In all cases, the density obtained from the algebra provides a surprisingly close fit.

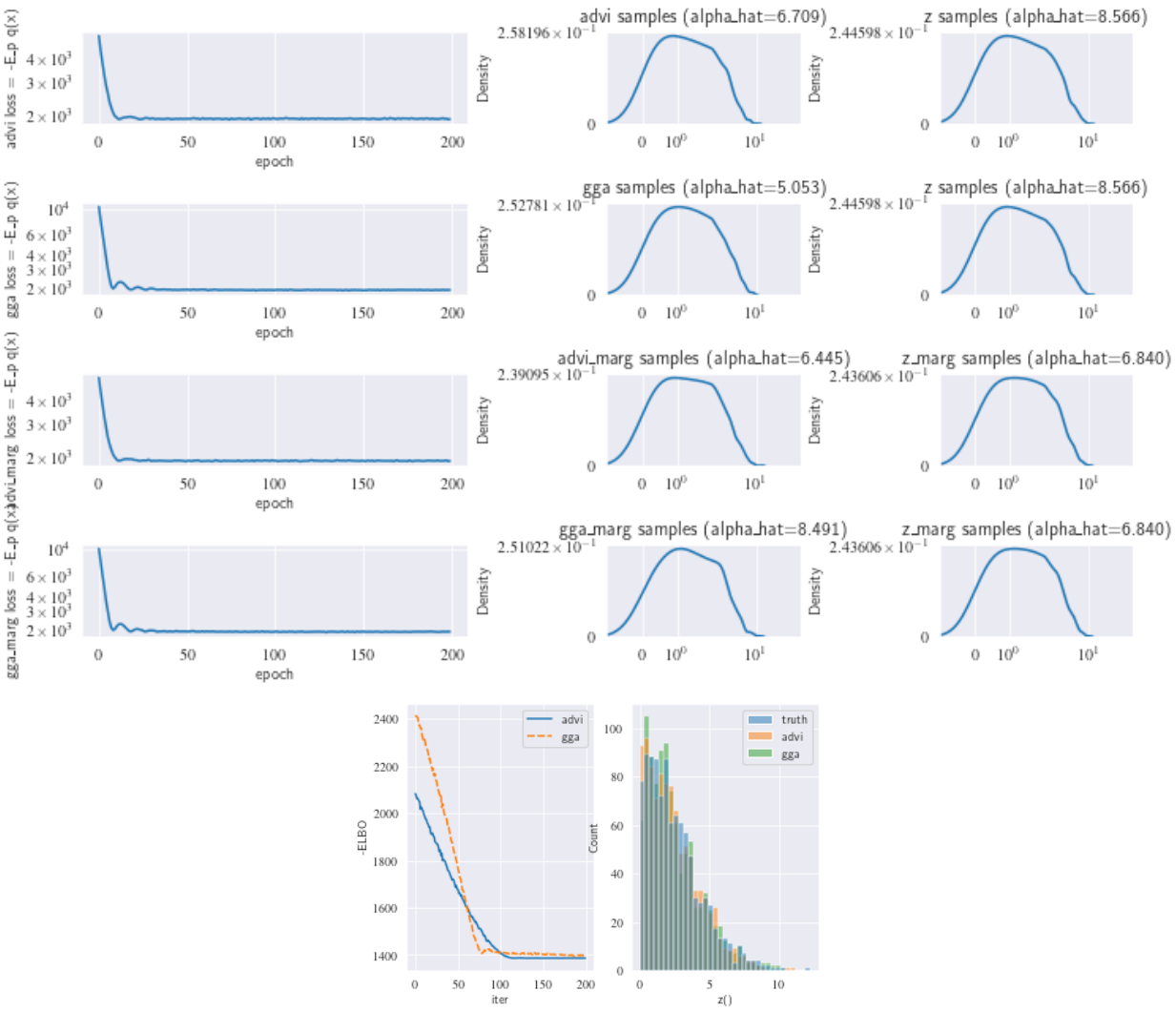


Figure 7.5: Density estimation and VI against a known normal target

Normal target

Consider the toy example of a Normal target. This case is trivial for Gaussian based methods and is oftentimes the initialization. This lack of approximation gap in ADVI is seen in Figure 7.5, where we also see that GGA achieves similar approximation quality. This is unsurprising as the GGA approximation in Table 7.2 is also a Normal distribution.

Chi-square

Now let $X_{ij} \sim N(0, 1)$ and consider $trX^\top X$. Such quantities arise in the analysis of random projections. It is important here to recognize that the power operation $X \mapsto X^2$ is not equiv-

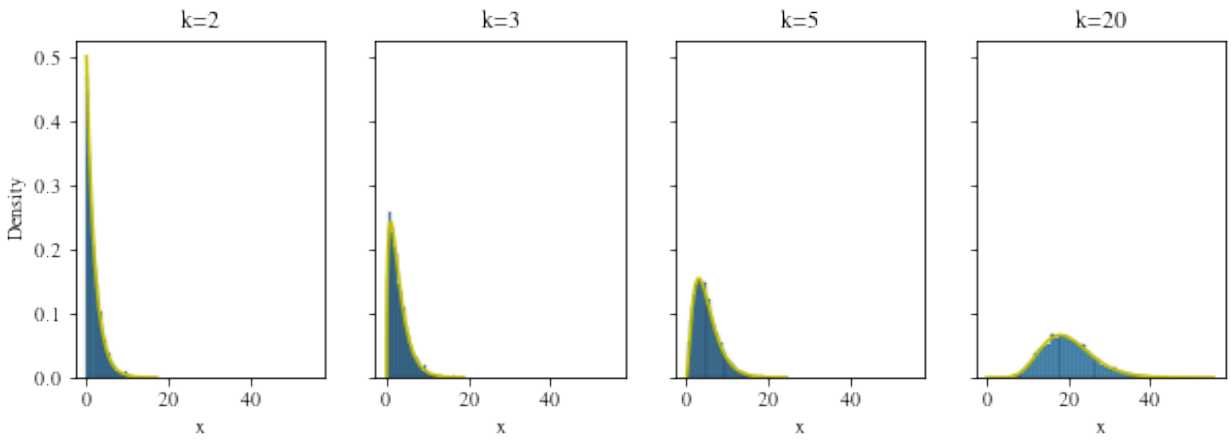


Figure 7.6: 5000 samples of JL matrix trace (blue) vs GGA prediction (yellow)

alent to the multiplication operation $X \mapsto X \otimes X$, as multiplication assumes independence.

7.6 Conclusion

In this work, we have proposed a novel systematic approach for conducting tail inferential static analysis by implementing a three-parameter generalized Gamma algebra into a PPL compiler. Initial results are promising, showing that improved inference with simpler approximation families is possible when combined with tail metadata. While already useful, the generalized Gamma algebra and its implementation currently has some notable limitations:

- Since the algebra assumes independence, handling of dependencies between defined random variables must be conducted externally. This will inevitably require interoperability with a symbolic package to decompose complex expressions into operations on independent random variables.
- The GGA is formulated for univariate distributions only. Suitably defining multivariate tails is an open problem with interesting alternatives [Jai+20; LHM22] all of which could extend GGA to higher dimensions.
- Conditioning is arguably the most important feature of a PPL and what distinguishes it from a glorified simulator. Exact marginalization in general is NP-hard [KF09], so treatment of conditional distributions using symbolic manipulations is a significant open problem, with some basic developments [SR17; CJ19]. Since only the tails are required in our setup, it may be possible to construct a dual algebra for operations under conditioning; this is left for future work.

- Compile-time static analysis only applicable to fixed model structure. While out of scope for our current work, open-universe models [MR10] and PPLs to support them [Bin+19] are an important research direction.
- The most significant omission to the algebra itself is classification of log-normal tails; while addition may be treated using [GT16] for example, multiplicative convolution with log-normal tails remains elusive.
- At present, reciprocals are approximated by assuming behaviour near zero. Reciprocals may be better treated by covering near-zero asymptotics separately.

The GGA provides a necessary first step into the static analysis of tails in a probabilistic program. As the above limitations are improved in future work and GGA becomes more broadly applicable, we are excited to see how improved tail modelling will improve downstream PPL applications as well as other researchers will utilize GGA metadata to develop novel PPL applications.

Bibliography

[AA10] Soren Asmussen and Hansjorg Albrecher. *Ruin probabilities*. Vol. 14. World scientific, 2010.

[AB13] Haim Avron and Christos Boutsidis. "Faster Subset Selection for Matrices and Applications". In: *SIAM Journal on Matrix Analysis and Applications* 34.4 (2013), pp. 1464–1499.

[Ach03] Dimitris Achlioptas. "Database-friendly random projections: Johnson-Lindenstrauss with binary coins". In: *Journal of computer and System Sciences* 66.4 (2003), pp. 671–687.

[All+17] Zeyuan Allen-Zhu, Yuanzhi Li, Aarti Singh, and Yining Wang. "Near-Optimal Design of Experiments via Regret Minimization". In: *Proceedings of the 34th International Conference on Machine Learning*. Vol. 70. Proceedings of Machine Learning Research. Sydney, Australia, Aug. 2017, pp. 126–135. URL: `http://proceedings.mlr.press/v70/allen-zhu17e.html`.

[AM15] Ahmed El Alaoui and Michael W. Mahoney. "Fast Randomized Kernel Ridge Regression with Statistical Guarantees". In: *Proceedings of the 28th International Conference on Neural Information Processing Systems*. 2015, pp. 775–783.

[AMT10] Haim Avron, Petar Maymounkov, and Sivan Toledo. "Blendenpik: Supercharging LAPACK's Least-Squares Solver". In: *SIAM Journal on Scientific Computing* 32.3 (2010), pp. 1217–1236.

[AO20] Najmeh Abiri and Mattias Ohlsson. "Variational auto-encoders with Student's t-prior". In: *arXiv preprint arXiv:2004.02581* (2020).

[Aro+19] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. "Implicit Regularization in Deep Matrix Factorization". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 7411–7422.

[Aro+20] Nimar S Arora, Nazanin Khosravani Tehrani, Kinjal Divesh Shah, Michael Tingley, Yucen Lily Li, Narjes Torabi, David Noursi, Sepehr Akhavan Masouleh, Eric Lippert, and Erik Meijer. "Newtonian Monte Carlo: single-site MCMC meets second-order gradient methods". In: *arXiv preprint arXiv:2001.05567* (2020).

[ASD20] Abhinav Agrawal, Daniel R Sheldon, and Justin Domke. "Advances in black-box VI: Normalizing flows, importance weighting, and optimization". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 17358–17369.

[Asm+17] Søren Asmussen, Enkelejd Hashorva, Patrick J Laub, and Thomas Taimre. "Tail asymptotics of light-tailed Weibull-like sums". In: *Probability and Mathematical Statistics* 37.2 (2017), pp. 235–256.

[Bar+19] P. L. Bartlett, P. M. Long, G. Lugosi, and A. Tsigler. *Benign Overfitting in Linear Regression*. Tech. rep. Preprint: arXiv:1906.11300. 2019.

[BDM16] Dariusz Buraczewski, Ewa Damek, and Thomas Mikosch. "Stochastic models with power-law tails". In: *Springer Ser. Oper. Res. Financ. Eng., Springer, Cham* 10 (2016), pp. 978–3.

[Bel+19] M. Belkin, D. Hsu, S. Ma, and S. Mandal. "Reconciling modern machine-learning practice and the classical bias–variance trade-off". In: *Proc. Natl. Acad. Sci. USA* 116 (2019), pp. 15849–15854.

[Ber+02] Donald A Berry, Peter Mueller, Andy P Grieve, Michael Smith, Tom Parke, Richard Blazek, Neil Mitchard, and Michael Krams. "Adaptive Bayesian designs for dose-ranging drug trials". In: *Case studies in Bayesian statistics*. Springer, 2002, pp. 99–181.

[Ber11] Dennis S. Bernstein. *Matrix Mathematics: Theory, Facts, and Formulas*. Second. Princeton University Press, 2011.

[Ber19] Ryan Bernstein. "Static analysis for probabilistic programs". In: *arXiv preprint arXiv:1909.05076* (2019).

[BGS10] Mustapha Bouhtou, Stéphane Gaubert, and Guillaume Sagnol. "Submodularity and Randomized rounding techniques for Optimal Experimental Design". In: *Electronic Notes in Discrete Mathematics* 36 (Aug. 2010), pp. 679–686. DOI: 10.1016/j.endm.2010.05.086.

[BGS15] Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. "Importance weighted autoencoders". In: *arXiv preprint arXiv:1509.00519* (2015).

[BHM18] Mikhail Belkin, Daniel J Hsu, and Partha Mitra. "Overfitting or perfect fitting? Risk bounds for classification and regression rules that interpolate". In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Curran Associates, Inc., 2018, pp. 2300–2311.

[BHX19] Mikhail Belkin, Daniel Hsu, and Ji Xu. "Two models of double descent for weak features". In: *arXiv preprint arXiv:1903.07571* (2019).

[Bia+17] Andrew An Bian, Joachim M. Buhmann, Andreas Krause, and Sebastian Tschiatschek. "Guarantees for Greedy Maximization of Non-submodular Functions with Applications". In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. International Convention Centre, Sydney, Australia: PMLR, June 2017, pp. 498–507. URL: `http://proceedings.mlr.press/v70/bian17a.html`.

[Bin+19] Eli Bingham, Jonathan P Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D Goodman. "Pyro: Deep universal probabilistic programming". In: *The Journal of Machine Learning Research* 20.1 (2019), pp. 973–978.

[BJ03] Francis R. Bach and Michael I. Jordan. "Kernel Independent Component Analysis". In: *J. Mach. Learn. Res.* 3 (Mar. 2003), pp. 1–48. ISSN: 1532-4435.

[BMD08] Christos Boutsidis, Michael Mahoney, and Petros Drineas. "An Improved Approximation Algorithm for the Column Subset Selection Problem". In: *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms* (Dec. 2008).

[BMM18] M. Belkin, S. Ma, and S. Mandal. "To understand deep learning we need to understand kernel learning". In: *Proceedings of the 35st International Conference on Machine Learning*. Vol. 80. Proceedings of Machine Learning Research. Stockholm, Sweden: PMLR, 2018.

[Boe+20] Benedikt Boenninghoff, Steffen Zeiler, Robert M Nickel, and Dorothea Kolossa. "Variational Autoencoder with Embedded Student-t Mixture Model for Authorship Attribution". In: *arXiv preprint arXiv:2005.13930* (2020).

[BRT19] M. Belkin, A. Rakhlin, and A. B. Tsybakov. "Does data interpolation contradict statistical optimality?" In: *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*. Vol. 89. Proceedings of Machine Learning Research. Naha, Okinawa, Japan: PMLR, 2019.

[Bru+13] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. "Spectral networks and locally connected networks on graphs". In: *arXiv preprint arXiv:1312.6203* (2013).

[BRV19] David Burt, Carl Edward Rasmussen, and Mark Van Der Wilk. "Rates of Convergence for Sparse Variational Gaussian Process Regression". In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. Long Beach, California, USA: PMLR, Sept. 2019, pp. 862–871.

[BS+98] Zhi-Dong Bai, Jack W Silverstein, et al. "No eigenvalues outside the support of the limiting spectral distribution of large-dimensional sample covariance matrices". In: *The Annals of Probability* 26.1 (1998), pp. 316–345.

[BS10] Zhidong Bai and Jack W Silverstein. *Spectral analysis of large dimensional random matrices*. Vol. 20. Springer, 2010.

[Bur73] Donald L Burkholder. "Distribution function inequalities for martingales". In: *the Annals of Probability* (1973), pp. 19–42.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[BY+93] ZD Bai, YQ Yin, et al. "Limit of the Smallest Eigenvalue of a Large Dimensional Sample Covariance Matrix". In: *The Annals of Probability* 21.3 (1993), pp. 1275–1294.

[Cao+22] Shichen Cao, Jingjing Li, Kenric P Nelson, and Mark A Kon. "Coupled VAE: Improved accuracy and robustness of a variational autoencoder". In: *Entropy* 24.3 (2022), p. 423.

[Car+17] Bob Carpenter, Andrew Gelman, Matthew D Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. "Stan: A probabilistic programming language". In: *Journal of statistical software* 76.1 (2017).

[Cen10] Centers for Medicare and Medicaid Services. *CMS 2008-2010 Data Entrepreneurs' Synthetic Public Use File (DE-SynPUF)*. [Online; accessed 10-March-2020]. 2010. URL: `https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/SynPUFs/DE_Syn_PUF`.

[CF11] R. Dennis Cook and Liliana Forzani. "On the mean and variance of the generalized inverse of a singular Wishart matrix". In: *Electron. J. Statist.* 5 (2011), pp. 146–158.

[Che+19] Ricky TQ Chen, Jens Behrmann, David K Duvenaud, and Jörn-Henrik Jacobsen. "Residual flows for invertible generative modeling". In: *Advances in Neural Information Processing Systems* 32 (2019), pp. 9913–9923.

[Chi90] Yasuko Chikuse. "The matrix angular central Gaussian distribution". In: *Journal of Multivariate Analysis* 33.2 (1990), pp. 265–274.

[Chi91] Yasuko Chikuse. "High dimensional limit theorems and matrix decompositions on the Stiefel manifold". In: *Journal of Multivariate Analysis* 36.2 (1991), pp. 145–162.

[Chi98] Yasuko Chikuse. "Density Estimation on the Stiefel Manifold". In: *Journal of Multivariate Analysis* 66.2 (1998), pp. 188–206.

[CJ19] Kenta Cho and Bart Jacobs. "Disintegration and Bayesian inversion via string diagrams". In: *Mathematical Structures in Computer Science* 29.7 (2019), pp. 938–971.

[CL11] Chih-Chung Chang and Chih-Jen Lin. "LIBSVM: A library for support vector machines". In: *ACM Transactions on Intelligent Systems and Technology* 2 (3 2011), 27:1–27:27.

[Cla+13] Guillaume Claret, Sriram K Rajamani, Aditya V Nori, Andrew D Gordon, and Johannes Borgström. "Bayesian inference using data flow analysis". In: *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*. 2013, pp. 92–102.

[CN80] R Dennis Cook and Christopher J Nachtrheim. "A comparison of algorithms for constructing exact D-optimal designs". In: *Technometrics* 22.3 (1980), pp. 315–324.

[CNW16] Michael B. Cohen, Jelani Nelson, and David P. Woodruff. "Optimal Approximate Matrix Product in Terms of Stable Rank". In: *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*. 2016, 11:1–11:14.

[Coh+15] Michael B. Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu. "Dimensionality Reduction for k-Means Clustering and Low Rank Approximation". In: *Proceedings of the Forty-seventh Annual ACM Symposium on Theory of Computing*. STOC '15. Portland, Oregon, USA: ACM, 2015, pp. 163–172. ISBN: 978-1-4503-3536-2. DOI: 10.1145/2746539.2746569.

[CR17] Luiz Chamon and Alejandro Ribeiro. "Approximate supermodularity bounds for experimental design". In: *Advances in Neural Information Processing Systems*. 2017, pp. 5403–5412.

[CR18] L. F. O. Chamon and A. Ribeiro. "Greedy Sampling of Graph Signals". In: *IEEE Transactions on Signal Processing* 66.1 (Jan. 2018), pp. 34–47.

[CSN09] Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. "Power-law distributions in empirical data". In: *SIAM review* 51.4 (2009), pp. 661–703.

[CSN20] Kevin R Chen, Daniel Svoboda, and Kenric P Nelson. "Use of Student's t-Distribution for the Latent Layer in a Coupled Variational Autoencoder". In: *arXiv preprint arXiv:2011.10879* (2020).

[Cus+19] Marco F Cusumano-Towner, Feras A Saad, Alexander K Lew, and Vikash K Mansinghka. "Gen: a general-purpose probabilistic programming system with programmable inference". In: *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation*. 2019, pp. 221–236.

[CV95] Kathryn Chaloner and Isabella Verdinelli. "Bayesian Experimental Design: A Review". In: *Statist. Sci.* 10.3 (Aug. 1995), pp. 273–304. DOI: 10.1214/ss/1177009939. URL: https://doi.org/10.1214/ss/1177009939.

[CW17] Kenneth L. Clarkson and David P. Woodruff. "Low-Rank Approximation and Regression in Input Sparsity Time". In: *J. ACM* 63.6 (Jan. 2017), 54:1–54:45. ISSN: 0004-5411. DOI: 10.1145/3019134. URL: http://doi.acm.org/10.1145/3019134.

[DDS16] Hanjun Dai, Bo Dai, and Le Song. "Discriminative embeddings of latent variable models for structured data". In: *International conference on machine learning*. 2016, pp. 2702–2711.

[Der+19] Michał Dereziński, Kenneth L. Clarkson, Michael W. Mahoney, and Manfred K. Warmuth. "Minimax experimental design: Bridging the gap between statistical and worst-case approaches to least squares regression". In: *Proceedings of the Thirty-Second Conference on Learning Theory*. Ed. by Alina Beygelzimer and Daniel Hsu. Vol. 99. Proceedings of Machine Learning Research. Phoenix, USA, 25–28 Jun 2019, pp. 1050–1069.

[Der+20a] Michał Dereziński, Burak Bartan, Mert Pilanci, and Michael W Mahoney. "Debiasing Distributed Second Order Optimization with Surrogate Sketching and Scaled Regularization". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 6684–6695.

[Der+20b] Michał Dereziński, Feynman Liang, Zhenyu Liao, and Michael W Mahoney. "Precise expressions for random projections: Low-rank approximation and randomized Newton". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 18272–18283.

[Der19] Michał Dereziński. "Fast determinantal point processes via distortion-free intermediate sampling". In: *Proceedings of the Thirty-Second Conference on Learning Theory*. 2019, pp. 1029–1049.

[DKB15] L Dinh, D Krueger, and Y Bengio. "NICE: non-linear independent components estimation". In: *3rd International Conference on Learning Representations, Workshop Track Proceedings*. 2015.

[DKM20] Michał Dereziński, Rajiv Khanna, and Michael W Mahoney. "Improved guarantees and a multiple-descent curve for Column Subset Selection and the Nyström method". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 4953–4964.

[DL19] Edgar Dobriban and Sifan Liu. "Asymptotics for sketching in least squares regression". In: *Advances in Neural Information Processing Systems*. 2019, pp. 3675–3685.

[DLM20a] Michał Dereziński, Feynman Liang, and Michael Mahoney. "Bayesian experimental design using regularized determinantal point processes". In: *International Conference on Artificial Intelligence and Statistics*. 2020, pp. 3197–3207.

[DLM20b] Michał Dereziński, Feynman Liang, and Michael W Mahoney. "Exact expressions for double descent and implicit regularization via surrogate random design". In: *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 5152–5164.

[DM14] D. F. Gleich and M. W. Mahoney. "Anti-differentiating Approximation Algorithms: A case study with Min-cuts, Spectral, and Flow". In: *Proceedings of the 31st International Conference on Machine Learning*. 2014, pp. 1018–1025.

[DM16] Petros Drineas and Michael W. Mahoney. "RandNLA: Randomized Numerical Linear Algebra". In: *Communications of the ACM* 59 (2016), pp. 80–90.

[DM17] Petros Drineas and Michael W. Mahoney. *Lectures on Randomized Numerical Linear Algebra*. Tech. rep. Preprint: arXiv:1712.08880; To appear in: *Lectures of the 2016 PCMI Summer School on Mathematics of Data*. 2017.

[DM18] P. Drineas and M. W. Mahoney. "Lectures on Randomized Numerical Linear Algebra". In: *The Mathematics of Data*. Ed. by M. W. Mahoney, J. C. Duchi, and A. C. Gilbert. IAS/Park City Mathematics Series. AMS/IAS/SIAM, 2018, pp. 1–48.

[DM19] Michał Dereziński and Michael W Mahoney. "Distributed estimation of the inverse Hessian by determinantal averaging". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 11401–11411.

[DM21] Michał Dereziński and Michael W Mahoney. "Determinantal Point Processes in Randomized Numerical Linear Algebra". In: *Notices of the American Mathematical Society* 68.1 (2021), pp. 34–45.

[DQV11] Nan Ding, Yuan Qi, and Svn Vishwanathan. "t-divergence based approximate inference". In: *Advances in Neural Information Processing Systems* 24 (2011), pp. 1494–1502.

[DRM08] Meichun Ding, Gary L Rosner, and Peter Müller. "Bayesian optimal design for phase II screening trials". In: *Biometrics* 64.3 (2008), pp. 886–894.

[DSB17] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. "Density estimation using Real NVP". In: *5th International Conference on Learning Representations*. 2017.

[Dur+19] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. "Neural spline flows". In: *Advances in Neural Information Processing Systems* 32 (2019), pp. 7509–7520.

[DW17] Michał Dereziński and Manfred K. Warmuth. "Unbiased estimates for linear regression via volume sampling". In: *Advances in Neural Information Processing Systems 30*. Long Beach, CA, USA, 2017, pp. 3087–3096.

[DW18a] Michał Dereziński and Manfred K. Warmuth. "Reverse Iterative Volume Sampling for Linear Regression". In: *Journal of Machine Learning Research* 19.23 (2018), pp. 1–39.

[DW18b] Michał Dereziński and Manfred K. Warmuth. "Subsampling for Ridge Regression via Regularized Volume Sampling". In: *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*. Ed. by Amos Storkey and Fernando Perez-Cruz. Playa Blanca, Lanzarote, Canary Islands, Apr. 2018, pp. 716–725.

[DWH18] Michał Dereziński, Manfred K. Warmuth, and Daniel Hsu. "Leveraged volume sampling for linear regression". In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Curran Associates, Inc., 2018, pp. 2510–2519.

[DWH19a] Michał Dereziński, Manfred K. Warmuth, and Daniel Hsu. "Correcting the bias in least squares regression with volume-rescaled sampling". In: *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*. Ed. by Kamalika Chaudhuri and Masashi Sugiyama. Vol. 89. Proceedings of Machine Learning Research. PMLR, 16–18 Apr 2019, pp. 944–953.

[DWH19b] Michał Dereziński, Manfred K. Warmuth, and Daniel Hsu. "Unbiased estimators for random design regression". In: *arXiv e-prints*, arXiv:1907.03411 (July 2019), arXiv:1907.03411. arXiv: `1907.03411 [stat.ML]`.

[Esl+16] SM Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Geoffrey E Hinton, et al. "Attend, infer, repeat: Fast scene understanding with generative models". In: *Advances in Neural Information Processing Systems* 29 (2016).

[FF15] Eugene F Fama and Kenneth R French. "A five-factor asset pricing model". In: *Journal of Financial Economics* 116.1 (2015), pp. 1–22.

[FHT01] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Vol. 1. 10. Springer series in statistics New York, 2001.

[Flo21] FlowTorch Development Team. *Flowtorch*. [Online; accessed 15-May-2021]. 2021. URL: `https://flowtorch.ai/`.

[Flo93] Nancy Flournoy. "A clinical experiment in bone marrow transplantation: Estimating a percentage point of a quantal response curve". In: *case studies in Bayesian Statistics*. Springer, 1993, pp. 324–336.

[FSS17] Futoshi Futami, Issei Sato, and Masashi Sugiyama. "Expectation propagation for t-exponential family using q-algebra". In: *Advances in Neural Information Processing Systems* 30 (2017), pp. 2245–2254.

[FSS20] Michaël Fanuel, Joachim Schreurs, and Johan AK Suykens. "Diversity sampling is an implicit regularization for kernel methods". In: *arXiv:2002.08616* (2020).

[FW16] Peter I Frazier and Jialei Wang. "Bayesian optimization for materials design". In: *Information Science for Materials Discovery and Design*. Springer, 2016, pp. 45–75.

[GC11] Mark Girolami and Ben Calderhead. "Riemann manifold langevin and hamiltonian monte carlo methods". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73.2 (2011), pp. 123–214.

[Gei+19] M. Geiger, A. Jacot, S. Spigler, F. Gabriel, L. Sagun, S. d'Ascoli, G. Biroli, C. Hongler, and M. Wyart. *Scaling description of generalization with number of parameters in deep learning*. Tech. rep. Preprint: arXiv:1901.01608. 2019.

[Gel+06] Andrew Gelman et al. "Prior distributions for variance parameters in hierarchical models (comment on article by Browne and Draper)". In: *Bayesian analysis* 1.3 (2006), pp. 515–534.

[Gel+13] Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. CRC press, 2013.

[Gey11] Charles Geyer. "Introduction to markov chain monte carlo". In: *Handbook of markov chain monte carlo* 20116022 (2011), p. 45.

[GFS16] Paul H Garthwaite, Yanan Fan, and Scott A Sisson. "Adaptive optimal scaling of Metropolis–Hastings algorithms using the Robbins–Monro process". In: *Communications in Statistics-Theory and Methods* 45.17 (2016), pp. 5098–5111.

[GG08] Rameshwar D Gupta and Ramesh C Gupta. "Analyzing skewed data by power normal model". In: *Test* 17.1 (2008), pp. 197–210.

[GG14] Samuel Gershman and Noah Goodman. "Amortized inference in probabilistic reasoning". In: *Proceedings of the annual meeting of the cognitive science society*. Vol. 36. 36. 2014.

[GH06] Andrew Gelman and Jennifer Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press, 2006.

[Gha15] Zoubin Ghahramani. "Probabilistic machine learning and artificial intelligence". In: *Nature* 521.7553 (2015), pp. 452–459.

[GK17] Surbhi Goel and Adam Klivans. "Eigenvalue Decay Implies Polynomial-Time Learnability for Neural Networks". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Curran Associates, Inc., 2017, pp. 2192–2202. URL: `http://papers.nips.cc/paper/6814-eigenvalue-decay-implies-polynomial-time-learnability-for-neural-networks.pdf`.

[GK98] Charles M Goldie and Claudia Klüppelberg. "Subexponential distributions". In: *A practical guide to heavy tails: statistical techniques and applications* (1998), pp. 435–459.

[GM16] Alex Gittens and Michael W. Mahoney. "Revisiting the Nyström Method for Improved Large-scale Machine Learning". In: *J. Mach. Learn. Res.* 17.1 (Jan. 2016), pp. 3977–4041. ISSN: 1532-4435.

[Goo+12] Noah Goodman, Vikash Mansinghka, Daniel M Roy, Keith Bonawitz, and Joshua B Tenenbaum. "Church: a language for generative models". In: *arXiv preprint arXiv:1206.3255* (2012).

[Goo13] Noah D Goodman. "The principles and practice of probabilistic programming". In: *ACM SIGPLAN Notices* 48.1 (2013), pp. 399–402.

[Gor+20] Maria I Gorinova, Andrew D Gordon, Charles Sutton, and Matthijs Vakar. "Conditional independence by typing". In: *arXiv preprint arXiv:2010.11887* (2020).

[Gow+19] Robert Gower, Dmitry Koralev, Felix Lieder, and Peter Richtarik. "RSN: Randomized Subspace Newton". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 614–623. URL: `http://papers.nips.cc/paper/8351-rsn-randomized-subspace-newton.pdf`.

[GR15] Robert M. Gower and Peter Richtárik. "Randomized Iterative Methods for Linear Systems". In: *SIAM. J. Matrix Anal. & Appl., 36(4), 1660–1690, 2015* (2015).

[Gra+19] Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. "FFJORD: Free-Form Continuous Dynamics for Scalable Reversible Generative Models". In: *International Conference on Learning Representations*. 2019.

[GRB20] Robert Gower, Peter Richtárik, and Francis Bach. "Stochastic quasi-gradient methods: variance reduction via Jacobian sketching". In: *Mathematical Programming* (May 2020). DOI: `10.1007/s10107-020-01506-0`.

[GSZ21] Mert Gurbuzbalaban, Umut Simsekli, and Lingjiong Zhu. "The heavy-tail phenomenon in SGD". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 3964–3975.

[GT16] Archil Gulisashvili and Peter Tankov. "Tail behavior of sums and differences of log-normal random variables". In: *Bernoulli* 22.1 (2016), pp. 444–493.

[Gun+17] Suriya Gunasekar, Blake E Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. "Implicit Regularization in Matrix Factorization". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Curran Associates, Inc., 2017, pp. 6151–6159.

[Hac+13] Walid Hachem, Philippe Loubaton, Jamal Najim, and Pascal Vallet. "On bilinear forms based on the resolvent of large random matrices". In: *Annales de l'IHP Probabilités et statistiques* 49.1 (2013), pp. 36–63.

[Har+19] William Harvey, Andreas Munk, Atılım Güneş Baydin, Alexander Bergholm, and Frank Wood. "Attention for Inference Compilation". In: *arXiv preprint arXiv:1910.11961* (2019).

[Has+19] T. Hastie, A. Montanari, S. Rosset, and R. J. Tibshirani. *Surprises in High-Dimensional Ridgeless Least Squares Interpolation*. Tech. rep. Preprint: arXiv:1903.08560. 2019.

[He+15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034.

[HG14] Matthew D Hoffman and Andrew Gelman. "The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo." In: *J. Mach. Learn. Res.* 15.1 (2014), pp. 1593–1623.

[HLN+07] Walid Hachem, Philippe Loubaton, Jamal Najim, et al. "Deterministic equivalents for certain functionals of large random matrices". In: *The Annals of Applied Probability* 17.3 (2007), pp. 875–930.

[HM21] Liam Hodgkinson and Michael Mahoney. "Multiplicative noise and heavy tails in stochastic optimization". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 4262–4274.

[HMT11] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions". In: *SIAM review* 53.2 (2011), pp. 217–288.

[Hoc98] Sepp Hochreiter. "The vanishing gradient problem during learning recurrent neural nets and problem solutions". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6.02 (1998), pp. 107–116.

[Hou+06] J. Ben Hough, Manjunath Krishnapur, Yuval Peres, Bálint Virág, et al. "Determinantal processes and independence". In: *Probability surveys* 3 (2006), pp. 206–229.

[Hua+18] Chin-Wei Huang, David Krueger, Alexandre Lacoste, and Aaron Courville. "Neural autoregressive flows". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 2078–2087.

[Hus17] Ferenc Huszár. "Variational inference using implicit distributions". In: *arXiv preprint arXiv:1702.08235* (2017).

[Jai+20] Priyank Jaini, Ivan Kobyzev, Yaoliang Yu, and Marcus Brubaker. "Tails of Lipschitz Triangular Flows". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 4673–4681.

[JP89] Claire Jones and Gordon D Plotkin. "A probabilistic powerdomain of evaluations". In: *Proceedings. Fourth Annual Symposium on Logic in Computer Science*. IEEE Computer Society. 1989, pp. 186–187.

[JSY19] Priyank Jaini, Kira A Selby, and Yaoliang Yu. "Sum-of-squares polynomial flow". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 3009–3018.

[KB14] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[KD18] Diederik Kingma and Prafulla Dhariwal. "Glow: Generative Flow with Invertible 1x1 Convolutions". In: *Advances in Neural Information Processing Systems* 31 (2018), pp. 10236–10245.

[KF09] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[Kin+16] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. "Improved variational inference with inverse autoregressive flow". In: *Advances in neural information processing systems*. 2016, pp. 4743–4751.

[KLS18] D. Kobak, J. Lomond, and B. Sanchez. *Optimal ridge penalty for real-world high-dimensional data can be zero or negative due to the implicit ridge regularization*. Tech. rep. Preprint: arXiv:1805.10939. 2018.

[Koz79] Dexter Kozen. "Semantics of probabilistic programs". In: *20th Annual Symposium on Foundations of Computer Science (sfcs 1979)*. IEEE. 1979, pp. 101–114.

[KT12] Alex Kulesza and Ben Taskar. *Determinantal Point Processes for Machine Learning*. Hanover, MA, USA: Now Publishers Inc., 2012.

[Kub+19] M. Kubo, R. Banno, H. Manabe, and M. Minoji. *Implicit Regularization in Over-parameterized Neural Networks*. Tech. rep. Preprint: arXiv:1903.01997. 2019.

[Kuc+17] Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M Blei. "Automatic differentiation variational inference". In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 430–474.

[KW16] Thomas N Kipf and Max Welling. "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907* (2016).

[LBW17] Tuan Anh Le, Atilim Gunes Baydin, and Frank Wood. "Inference compilation and universal probabilistic programming". In: *Artificial Intelligence and Statistics*. PMLR. 2017, pp. 1338–1348.

[Led01] Michel Ledoux. *The concentration of measure phenomenon*. Mathematical surveys and monographs 89. American Mathematical Soc., 2001.

[Lee+19] Wonyeol Lee, Hangyeol Yu, Xavier Rival, and Hongseok Yang. "Towards verified stochastic variational inference for probabilistic programs". In: *Proceedings of the ACM on Programming Languages* 4.POPL (2019), pp. 1–33.

[LEM19] Miles E Lopes, N Benjamin Erichson, and Michael W Mahoney. "Bootstrapping the Operator Norm in High Dimensions: Error Estimation for Covariance Matrices and Sketching". In: *arXiv preprint arXiv:1909.06120* (2019).

[LHM22] Feynman Liang, Liam Hodgkinson, and Michael Mahoney. "Fat–Tailed Variational Inference with Anisotropic Tail Adaptive Flows". In: *Proceedings of the 39th International Conference on Machine Learning*. Vol. 162. 2022, p. 132.

[LHM23] Feynman Liang, Liam Hodgkinson, and Michael Mahoney. "Static Analysis of Tail Behaviour with a Generalized Gamma Algebra". In: *Submitted to AISTATS 2023* (2023).

[Lia+21] Feynman Liang, Nimar Arora, Nazanin Tehrani, Yucen Li, Michael Tingley, and Erik Meijer. "Accelerating Metropolis-Hastings with Lightweight Inference Compilation". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 181–189.

[LJB19] D. LeJeune, H. Javadi, and R. G. Baraniuk. *The Implicit Regularization of Ordinary Least Squares Ensembles*. Tech. rep. Preprint: arXiv:1910.04743. 2019.

[LP11] Olivier Ledoit and Sandrine Péché. "Eigenvectors of some large sample covariance matrix ensembles". In: *Probability Theory and Related Fields* 151.1-2 (2011), pp. 233–264.

[LP19] Jonathan Lacotte and Mert Pilanci. "Faster Least Squares Optimization". In: *arXiv preprint arXiv:1911.02675* (2019).

[LPP19] Jonathan Lacotte, Mert Pilanci, and Marco Pavone. "High-Dimensional Optimization in Adaptive Random Subspaces". In: *Advances in Neural Information Processing Systems*. 2019, pp. 10846–10856.

[LR19] T. Liang and A. Rakhlin. "Just Interpolate: Kernel "Ridgeless" Regression Can Generalize". In: *The Annals of Statistics, to appear* (2019).

[LT16] Yingzhen Li and Richard E Turner. "Rényi divergence variational inference". In: *Advances in Neural Information Processing Systems* 29 (2016), pp. 1073–1081.

[Lun+00] David J Lunn, Andrew Thomas, Nicky Best, and David Spiegelhalter. "WinBUGS-a Bayesian modelling framework: concepts, structure, and extensibility". In: *Statistics and computing* 10.4 (2000), pp. 325–337.

[M W12] M. W. Mahoney. "Approximate Computation and Implicit Regularization for Very Large-scale Data Analysis". In: *Proceedings of the 31st ACM Symposium on Principles of Database Systems*. 2012, pp. 143–154.

[MAM10] Iain Murray, Ryan Adams, and David MacKay. "Elliptical slice sampling". In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 2010, pp. 541–548.

[Mat+19] Emile Mathieu, Tom Rainforth, N Siddharth, and Yee Whye Teh. "Disentangling disentanglement in variational autoencoders". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 4402–4412.

[MDK20] Mojmir Mutny, Michał Dereziński, and Andreas Krause. "Convergence Analysis of Block Coordinate Algorithms with Determinantal Sampling". In: *International Conference on Artificial Intelligence and Statistics*. 2020, pp. 3110–3120.

[Mey73] Carl D. Meyer. "Generalized Inversion of Modified Matrices". In: *SIAM Journal on Applied Mathematics* 24.3 (1973), pp. 315–323. ISSN: 00361399. URL: `http://www.jstor.org/stable/2099767`.

[Mic11] Michael W. Mahoney. "Randomized algorithms for matrices and data". In: *Foundations and Trends in Machine Learning* 3.2 (2011). Also available at: arXiv:1104.5557, pp. 123–224.

[Mik99] T Mikosch. *Regular Variation Subexponentiality and Their Applications in Probability Theory*. 1999. URL: `https://www.eurandom.tue.nl/reports/1999/013-report.pdf`.

[Mil+07] Brian Milch, Bhaskara Marthi, Stuart Russell, David Sontag, Daniel L Ong, and Andrey Kolobov. "1 blog: Probabilistic models with unknown objects". In: *Statistical relational learning* (2007), p. 373.

[Mit19] P. P. Mitra. *Understanding overfitting peaks in generalization error: Analytical risk curves for l2 and l1 penalized interpolation*. Tech. rep. Preprint: arXiv:1906.03667. 2019.

[ML11] M. W. Mahoney and L. Orecchia. "Implementing regularization implicitly via approximate eigenvector computation". In: *Proceedings of the 28th International Conference on Machine Learning*. 2011, pp. 121–128.

[MM18] C. H. Martin and M. W. Mahoney. *Implicit Self-Regularization in Deep Neural Networks: Evidence from Random Matrix Theory and Implications for Learning*. Tech. rep. Preprint: arXiv:1810.01075. 2018.

[MM19a] C. H. Martin and M. W. Mahoney. "Traditional and Heavy-Tailed Self Regularization in Neural Network Models". In: *Proceedings of the 36th International Conference on Machine Learning*. 2019, pp. 4284–4293.

[MM19b] S. Mei and A. Montanari. *The generalization error of random features regression: Precise asymptotics and double descent curve*. Tech. rep. Preprint: arXiv:1908.05355. 2019.

[MR10] Brian Milch and Stuart Russell. "Extending Bayesian networks to the open-universe case". In: *Heuristics, Probability and Causality: A Tribute to Judea Pearl. College Publications* (2010).

[MSH09] Arakaparampil M Mathai, Ram Kishore Saxena, and Hans J Haubold. *The H-function: theory and applications*. Springer Science & Business Media, 2009.

[MSM14] X. Meng, M. A. Saunders, and M. W. Mahoney. "LSRN: A Parallel Iterative Solver for Strongly Over- or Under-Determined Systems". In: *SIAM Journal on Scientific Computing* 36.2 (2014), pp. C95–C118.

[MSP14] Vikash Mansinghka, Daniel Selsam, and Yura Perov. "Venture: a higher-order probabilistic programming platform with programmable inference". In: *arXiv preprint arXiv:1404.0099* (2014).

[Mut+19] V. Muthukumar, K. Vodrahalli, V. Subramanian, and A. Sahai. *Harmless interpolation of noisy data in regression*. Tech. rep. Preprint: arXiv:1903.09139. 2019.

[MYM18] Joseph Marino, Yisong Yue, and Stephan Mandt. "Iterative amortized inference". In: *arXiv preprint arXiv:1807.09356* (2018).

[Ney17] B. Neyshabur. *Implicit Regularization in Deep Learning*. Tech. rep. Preprint: arXiv:1709.01953. 2017.

[NN13] Jelani Nelson and Huy L. Nguyên. "OSNAP: Faster Numerical Linear Algebra Algorithms via Sparser Subspace Embeddings". In: *Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. FOCS '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 117–126. ISBN: 978-0-7695-5135-7. DOI: 10.1109/FOCS.2013.21. URL: http://dx.doi.org/10.1109/FOCS.2013.21.

[Nor+14] Aditya Nori, Chung-Kil Hur, Sriram Rajamani, and Selva Samuel. "R2: An efficient MCMC sampler for probabilistic programs". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 28. 2014.

[NST19] Aleksandar Nikolov, Mohit Singh, and Uthaipon Tao Tantipongpipat. "Proportional Volume Sampling and Approximation Algorithms for A -Optimal Design". In: *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*. Jan. 2019, pp. 1369–1386.

[NTS14] B. Neyshabur, R. Tomioka, and N. Srebro. *In search of the real inductive bias: on the role of implicit regularization in deep learning*. Tech. rep. Preprint: arXiv:1412.6614. 2014.

[ODo+16] Brendan O'Donoghue, Eric Chu, Neal Parikh, and Stephen Boyd. "Conic optimization via operator splitting and homogeneous self-dual embedding". In: *Journal of Optimization Theory and Applications* 169.3 (2016), pp. 1042–1068.

[Owe+16] David Owen, Andrew Melbourne, David Thomas, Enrico De Vita, Jonathan Rohrer, and Sebastien Ourselin. "Optimisation of arterial spin labelling using bayesian experimental design". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2016, pp. 511–518.

[Pap+21] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. "Normalizing flows for probabilistic modeling and inference". In: *Journal of Machine Learning Research* 22.57 (2021), pp. 1–64.

[Pas+19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. "Pytorch: An imperative style, high-performance deep learning library". In: *Advances in neural information processing systems* 32 (2019).

[Pea87] Judea Pearl. "Evidential reasoning using stochastic simulation of causal models". In: *Artificial Intelligence* 32.2 (1987), pp. 245–257.

[PHF10] Anand Patil, David Huard, and Christopher J Fonnesbeck. "PyMC: Bayesian stochastic modelling in Python". In: *Journal of Statistical Software* 35.4 (2010), p. 1.

[Plu+03] Martyn Plummer et al. "JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling". In: *Proceedings of the 3rd international workshop on distributed statistical computing*. Vol. 124. 125.10. Vienna, Austria. 2003, pp. 1–10.

[PM11] P. O. Perry and M. W. Mahoney. "Regularized Laplacian Estimation and Fast Eigenvector Approximation". In: *Annual Advances in Neural Information Processing Systems 24: Proceedings of the 2011 Conference*. 2011.

[PMB15] P. Ma, M. W. Mahoney, and B. Yu. "A Statistical Perspective on Algorithmic Leveraging". In: *Journal of Machine Learning Research* 16 (2015), pp. 861–911.

[Por+08] Ian Porteous, David Newman, Alexander Ihler, Arthur Asuncion, Padhraic Smyth, and Max Welling. "Fast collapsed gibbs sampling for latent dirichlet allocation". In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2008, pp. 569–577.

[PPM17] George Papamakarios, Theo Pavlakou, and Iain Murray. "Masked autoregressive flow for density estimation". In: *Advances in Neural Information Processing Systems*. 2017, pp. 2338–2347.

[Puk06] Friedrich Pukelsheim. *Optimal Design of Experiments*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2006. ISBN: 0898716047.

[PW16a] Brooks Paige and Frank Wood. "Inference networks for sequential Monte Carlo in graphical models". In: *International Conference on Machine Learning*. 2016, pp. 3040–3049.

[PW16b] Mert Pilanci and Martin J Wainwright. "Iterative Hessian sketch: Fast and accurate solution approximation for constrained least-squares". In: *The Journal of Machine Learning Research* 17.1 (2016), pp. 1842–1879.

[PyP20] PyProb. *PyProb*. `https://github.com/pyprob/pyprob`. 2020.

[QR16] Zheng Qu and Peter Richtárik. "Coordinate descent with arbitrary sampling II: Expected separable overapproximation". In: *Optimization Methods and Software* 31.5 (2016), pp. 858–884.

[Qu+16] Zheng Qu, Peter Richtárik, Martin Takác, and Olivier Fercoq. "SDNA: Stochastic Dual Newton Ascent for Empirical Risk Minimization". In: *Proceedings of The 33rd International Conference on Machine Learning* (Feb. 2016). eprint: `1502.02268`. URL: `https://arxiv.org/abs/1502.02268`.

[RDP+16] Caitriona M Ryan, Christopher C Drovandi, Anthony N Pettitt, et al. "Optimal Bayesian experimental design for models with intractable likelihoods using indirect inference applied to biological process models". In: *Bayesian Analysis* 11.3 (2016), pp. 857–883.

[RDP15] Elizabeth Ryan, Christopher Drovandi, and Anthony Pettitt. "Fully Bayesian experimental design for pharmacokinetic studies". In: *Entropy* 17.3 (2015), pp. 1063–1089.

[RGB14] Rajesh Ranganath, Sean Gerrish, and David Blei. "Black box variational inference". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2014, pp. 814–822.

[RHG16] Daniel Ritchie, Paul Horsfall, and Noah D Goodman. "Deep amortized inference for probabilistic programs". In: *arXiv preprint arXiv:1610.05735* (2016).

[RM15] Danilo Rezende and Shakir Mohamed. "Variational inference with normalizing flows". In: *International Conference on Machine Learning*. PMLR. 2015, pp. 1530–1538.

[RM16] G. Raskutti and M. W. Mahoney. "A Statistical Perspective on Randomized Sketching for Ordinary Least-Squares". In: *Journal of Machine Learning Research* 17.214 (2016), pp. 1–31.

[RM19] Farbod Roosta-Khorasani and Michael W Mahoney. "Sub-sampled Newton methods". In: *Mathematical Programming* 174.1-2 (2019), pp. 293–326.

[Roo+18] F. Roosta, Y. Liu, P. Xu, and M. W. Mahoney. *Newton-MR: Newton's Method Without Smoothness or Convexity*. Tech. rep. Preprint: arXiv:1810.00303. 2018.

[RSG16] Daniel Ritchie, Andreas Stuhlmüller, and Noah Goodman. "C3: Lightweight incrementalized MCMC for probabilistic programs using continuations and callsite caching". In: *Artificial Intelligence and Statistics*. 2016, pp. 28–37.

[RT96] Gareth O Roberts and Richard L Tweedie. "Exponential convergence of Langevin distributions and their discrete approximations". In: *Bernoulli* (1996), pp. 341–363.

[Rub81] Donald B Rubin. "Estimation in parallel randomized experiments". In: *Journal of Educational Statistics* 6.4 (1981), pp. 377–401.

[RV13] Mark Rudelson and Roman Vershynin. "Hanson-Wright inequality and sub-gaussian concentration". In: *Electronic Communications in Probability* 18 (2013).

[RW06] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[RWZ88] Barry K Rosen, Mark N Wegman, and F Kenneth Zadeck. "Global value numbers and redundant computations". In: *Proceedings of the 15th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*. 1988, pp. 12–27.

[SA01] Alex J Sutton and Keith R Abrams. "Bayesian methods in meta-analysis and evidence synthesis". In: *Statistical methods in medical research* 10.4 (2001), pp. 277–303.

[San+97] Huaiyu Zhu Santa, Huaiyu Zhu, Christopher K. I. Williams, Richard Rohwer, and Michal Morciniec. "Gaussian Regression and Optimal Finite Dimensional Linear Models". In: *Neural Networks and Machine Learning*. Springer-Verlag, 1997, pp. 167–184.

[Sar06] Tamas Sarlos. "Improved Approximation Algorithms for Large Matrices via Random Projections". In: *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*. FOCS '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 143–152.

[SB95] Jack W Silverstein and ZD Bai. "On the empirical distribution of eigenvalues of a class of large dimensional random matrices". In: *Journal of Multivariate analysis* 54.2 (1995), pp. 175–192.

[SB98] Dalene K Stangl and Donald A Berry. "Bayesian statistics in medicine: Where are we and where should we be going?" In: *Sankhyā: The Indian Journal of Statistics, Series B* (1998), pp. 176–195.

[Sca+08] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. "The graph neural network model". In: *IEEE Transactions on Neural Networks* 20.1 (2008), pp. 61–80.

[SCG13] Sriram Sankaranarayanan, Aleksandar Chakarov, and Sumit Gulwani. "Static analysis for probabilistic programs: inferring whole program properties from finitely many paths". In: *Proceedings of the 34th ACM SIGPLAN conference on Programming language design and implementation*. 2013, pp. 447–458.

[Ser10] D. Serre. *Matrices: Theory and Applications*. Graduate Texts in Mathematics. Springer, 2010. ISBN: 9781441930101. URL: https://books.google.to/books?id=IYWLcgAACAAJ.

[Sid+17] N. Siddharth, Brooks Paige, Jan-Willem van de Meent, Alban Desmaison, Noah D. Goodman, Pushmeet Kohli, Frank Wood, and Philip Torr. "Learning Disentangled Representations with Semi-Supervised Deep Generative Models". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Curran Associates, Inc., 2017, pp. 5927–5937. URL: http://papers.nips.cc/paper/7174-learning-disentangled-representations-with-semi-supervised-deep-generative-models.pdf.

[Sou+18] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. "The implicit bias of gradient descent on separable data". In: *The Journal of Machine Learning Research* 19.1 (2018), pp. 2822–2878.

[Spi+04] David J Spiegelhalter et al. "Incorporating Bayesian ideas into health-care evaluation". In: *Statistical Science* 19.1 (2004), pp. 156–174.

[Spi+96] David Spiegelhalter, Andrew Thomas, Nicky Best, and Wally Gilks. "BUGS 0.5: Bayesian inference using Gibbs sampling manual (version ii)". In: *MRC Biostatistics Unit, Institute of Public Health, Cambridge, UK* (1996), pp. 1–59.

[SR17] Chung-chieh Shan and Norman Ramsey. "Exact Bayesian inference by symbolic disintegration". In: *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages*. 2017, pp. 130–144.

[Sri03] M.S. Srivastava. "Singular Wishart and multivariate beta distributions". In: *Ann. Statist.* 31.5 (Oct. 2003), pp. 1537–1560.

[SSG19] Umut Simsekli, Levent Sagun, and Mert Gurbuzbalaban. "A tail-index analysis of stochastic gradient noise in deep neural networks". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 5827–5837.

[Taj03] Nader Tajvidi. "Confidence intervals and accuracy estimation for heavy-tailed generalized Pareto distributions". In: *Extremes* 6.2 (2003), pp. 111–123.

[Teh+20a] Nazanin Tehrani, Nimar S Arora, Yucen Lily Li, Kinjal Divesh Shah, David Noursi, Michael Tingley, Narjes Torabi, Eric Lippert, Erik Meijer, et al. "Bean machine: A declarative probabilistic programming language for efficient programmable inference". In: *International Conference on Probabilistic Graphical Models*. PMLR. 2020.

[Teh+20b] Nazanin Tehrani, Nimar S Arora, Yucen Lily Li, Kinjal Divesh Shah, David Noursi, Michael Tingley, Narjes Torabi, Sepehr Masouleh, Eric Lippert, Erik Meijer, and et al. "Bean Machine: A Declarative Probabilistic Programming Language For Efficient Programmable Inference". In: *The 10th International Conference on Probabilistic Graphical Models*. 2020.

[The21] The Stan Developers. *posteriordb: a database of Bayesian posterior inference*. `https://github.com/stan-dev/posteriordb`. 2021.

[TL05] Michael E Tipping and Neil D Lawrence. "Variational inference for Student-t models: Robust Bayesian interpolation and generalised component analysis". In: *Neurocomputing* 69.1-3 (2005), pp. 123–141.

[Tol+16] David Tolpin, Jan-Willem van de Meent, Hongseok Yang, and Frank Wood. "Design and implementation of probabilistic programming language anglican". In: *Proceedings of the 28th Symposium on the Implementation and Application of Functional programming Languages*. 2016, pp. 1–12.

[Tra+18] Dustin Tran, Matthew W Hoffman, Dave Moore, Christopher Suter, Srinivas Vasudevan, and Alexey Radul. "Simple, distributed, and accelerated probabilistic programming". In: *Advances in Neural Information Processing Systems* 31 (2018).

[TUM12] Gabriel Terejanu, Rochan R Upadhyay, and Kenji Miki. "Bayesian experimental design for the active nitridation of graphite by atomic nitrogen". In: *Experimental Thermal and Fluid Science* 36 (2012), pp. 178–193.

[Uen+16] Tsuyoshi Ueno, Trevor David Rhone, Zhufeng Hou, Teruyasu Mizoguchi, and Koji Tsuda. "COMBO: an efficient Bayesian optimization library for materials science". In: *Materials discovery* 4 (2016), pp. 18–21.

[Vaa65] H. Robert van der Vaart. "A Note on Wilks' Internal Scatter". In: *Ann. Math. Statist.* 36.4 (Aug. 1965), pp. 1308–1312.

[Val+17] Perry de Valpine, Daniel Turek, Christopher J Paciorek, Clifford Anderson-Bergman, Duncan Temple Lang, and Rastislav Bodik. "Programming with models: writing statistical algorithms for general model structures with NIMBLE". In: *Journal of Computational and Graphical Statistics* 26.2 (2017), pp. 403–413.

[Veh+15] Aki Vehtari, Daniel Simpson, Andrew Gelman, Yuling Yao, and Jonah Gabry. "Pareto smoothed importance sampling". In: *arXiv preprint arXiv:1507.02646* (2015).

[Veh+20] Aki Vehtari, Andrew Gelman, Daniel Simpson, Bob Carpenter, Paul-Christian Bürkner, et al. "Rank-normalization, folding, and localization: An improved $\widehat{R}$ for assessing convergence of MCMC". In: *Bayesian Analysis* (2020).

[Ver18] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*. Vol. 47. Cambridge university press, 2018.

[Wan+17a] Shusen Wang, Farbod Roosta-Khorasani, Peng Xu, and Michael W. Mahoney. "GIANT: Globally Improved Approximate Newton Method for Distributed Optimization". In: *CoRR* abs/1709.03528 (2017). arXiv: `1709.03528`. URL: `http://arxiv.org/abs/1709.03528`.

[Wan+17b] Tongzhou Wang, Yi Wu, David A Moore, and Stuart J Russell. "Meta-learning MCMC proposals". In: *arXiv preprint arXiv:1708.06040* (2017).

[Wat95] George Neville Watson. *A treatise on the theory of Bessel functions*. Cambridge university press, 1995.

[Web+18] Stefan Webb, Adam Golinski, Rob Zinkov, N Siddharth, Tom Rainforth, Yee Whye Teh, and Frank Wood. "Faithful inversion of generative models for effective amortized inference". In: *Advances in Neural Information Processing Systems*. 2018, pp. 3070–3080.

[Web+19a] Stefan Webb, J.P. Chen, Martin Jankowiak, and Noah Goodman. "Improving Automated Variational Inference with Normalizing Flows". In: *6th ICML Workshop on Automated Machine Learning (AutoML)* (2019).

[Web+19b] Stefan Webb, Jonathan P. Chen, Matrin Jankowiak, and Noah Goodman. "Improving automated variational inference with normalizing flows". In: *ICML Workshop on Automated Machine Learning*. 2019.

[Wei+19] Christian Weilbach, Boyan Beronov, William Harvey, and Frank Wood. "Efficient Inference Amortization in Graphical Models using Structured Continuous Conditional Normalizing Flows". In: (2019).

[WGM17] Shusen Wang, Alex Gittens, and Michael W. Mahoney. "Sketched Ridge Regression: Optimization Perspective, Statistical Perspective, and Model Averaging". In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. International Convention Centre, Sydney, Australia: PMLR, June 2017, pp. 3608–3616. URL: `http://proceedings.mlr.press/v70/wang17c.html`.

[WHR18] Di Wang, Jan Hoffmann, and Thomas Reps. "PMAF: an algebraic framework for static analysis of probabilistic programs". In: *ACM SIGPLAN Notices* 53.4 (2018), pp. 513–528.

[Wic11] Hadley Wickham. "ggplot2". In: *Wiley Interdisciplinary Reviews: Computational Statistics* 3.2 (2011), pp. 180–185.

[WL19] Antoine Wehenkel and Gilles Louppe. "Unconstrained monotonic neural networks". In: *Advances in Neural Information Processing Systems* 32 (2019), pp. 1543–1553.

[WLL18] Dilin Wang, Hao Liu, and Qiang Liu. "Variational inference with tail-adaptive f-divergence". In: *Advances in Neural Information Processing Systems* 31 (2018), pp. 5737–5747.

[WMM14] Frank Wood, Jan Willem Meent, and Vikash Mansinghka. "A new approach to probabilistic programming inference". In: *Artificial Intelligence and Statistics*. 2014, pp. 1024–1032.

[Woo14] David P. Woodruff. "Sketching as a tool for numerical linear algebra". In: *Foundations and Trends® in Theoretical Computer Science* 10.1–2 (2014), pp. 1–157.

[WS01] Christopher K. I. Williams and Matthias Seeger. "Using the Nyström Method to Speed Up Kernel Machines". In: *Advances in Neural Information Processing Systems 13*. Ed. by T. K. Leen, T. G. Dietterich, and V. Tresp. MIT Press, 2001, pp. 682–688.

[WSG11] David Wingate, Andreas Stuhlmüller, and Noah Goodman. "Lightweight implementations of probabilistic programming languages via transformational compilation". In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. 2011, pp. 770–778.

[WW13] David Wingate and Theophane Weber. "Automated variational inference in probabilistic programming". In: *arXiv preprint arXiv:1301.1299* (2013).

[WYS17] Yining Wang, Adams W. Yu, and Aarti Singh. "On Computationally Tractable Selection of Experiments in Measurement-constrained Regression Models". In: *J. Mach. Learn. Res.* 18.1 (Jan. 2017), pp. 5238–5278. ISSN: 1532-4435. URL: `http://dl.acm.org/citation.cfm?id=3122009.3208024`.

[XRM17] P. Xu, F. Roosta-Khorasani, and M. W. Mahoney. *Newton-Type Methods for Non-Convex Optimization Under Inexact Hessian Information*. Tech. rep. Preprint: arXiv:1708.07164. 2017.

[Xu+20] Kai Xu, Hong Ge, Will Tebbutt, Mohamed Tarek, Martin Trapp, and Zoubin Ghahramani. "AdvancedHMC. jl: A robust, modular and efficient implementation of advanced HMC algorithms". In: *Symposium on Advances in Approximate Bayesian Inference*. PMLR. 2020, pp. 1–10.

[Yan+20] Fan Yang, Sifan Liu, Edgar Dobriban, and David P Woodruff. "How to reduce dimension with PCA and random projections?" In: *arXiv preprint arXiv:2005.00511* (2020).

[Yao+18a] Yuling Yao, Aki Vehtari, Daniel Simpson, and Andrew Gelman. "Yes, but did it work?: Evaluating variational inference". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 5581–5590.

[Yao+18b] Z. Yao, P. Xu, F. Roosta-Khorasani, and M. W. Mahoney. *Inexact Non-Convex Newton-Type Methods*. Tech. rep. Preprint: arXiv:1802.06925. 2018.

[Zah+17] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan Salakhutdinov, and Alexander Smola. "Deep sets". In: *arXiv preprint arXiv:1703.06114* (2017).

[Zaj18] Krzysztof Zajkowski. "Bounds on tail probabilities for quadratic forms in dependent sub-gaussian random variables". In: *arXiv preprint arXiv:1809.08569* (2018).

[Zha+18] Cheng Zhang, Judith Bütepage, Hedvig Kjellström, and Stephan Mandt. "Advances in variational inference". In: *IEEE transactions on pattern analysis and machine intelligence* 41.8 (2018), pp. 2008–2026.

[Zhu+15] Rong Zhu, Ping Ma, Michael W Mahoney, and Bin Yu. "Optimal subsampling approaches for large sample linear regression". In: *arXiv preprint arXiv:1509.05111* (2015).