Random matrix theory and modern machine learning

Michael W. Mahoney

(ICSI, LBNL, and Department of Statistics, UC Berkeley)

June 2025

Overview

Motivations:

- WeightWatcher, Weight Diagnostics for Analyzing ML Models
  (with Charles H. Martin)
- Randomized Numerical Linear Algebra for Modern ML
  (with Michal Derezinski)

Some Theory:

- RMT for NNs: Linear to Nonlinear; Shallow to Deep; etc.
  (with Zhenyu Liao)

Applications:

- Models of Heavy-Tailed Mechanistic Universality
  (with Zhichao Wang and Liam Hodgkinson)
- Spectral Estimation with Free Decompression
  (with Siavash Ameli, Chris van der Heide, and Liam Hodgkinson)
- Determinant Estimation under Memory Constraints and Neural Scaling Laws
  (with S. Ameli, C. van der Heide, L. Hodgkinson, and F. Roosta)

Overview

Motivations:

- WeightWatcher, Weight Diagnostics for Analyzing ML Models (with Charles H. Martin)
- Randomized Numerical Linear Algebra for Modern ML (with Michal Derezinski)

Some Theory:

- RMT for NNs: Linear to Nonlinear; Shallow to Deep; etc. (with Zhenyu Liao)

Applications:

- Models of Heavy-Tailed Mechanistic Universality (with Zhichao Wang and Liam Hodgkinson)
- Spectral Estimation with Free Decompression (with Siavash Ameli, Chris van der Heide, and Liam Hodgkinson)
- Determinant Estimation under Memory Constraints and Neural Scaling Laws (with S. Ameli, C. van der Heide, L. Hodgkinson, and F. Roosta)

WeightWatcher, an Open-Source Diagnostic Tool for Analyzing Deep Neural Nets

Michael W. Mahoney

ICSI and Dept of Statistics, UC Berkeley

http://www.stat.berkeley.edu/∼mmahoney/

April 2022

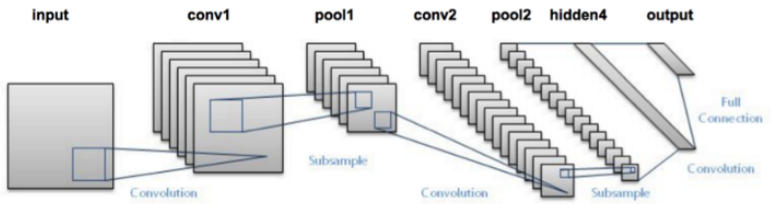
(Joint work with Charles H. Martin,

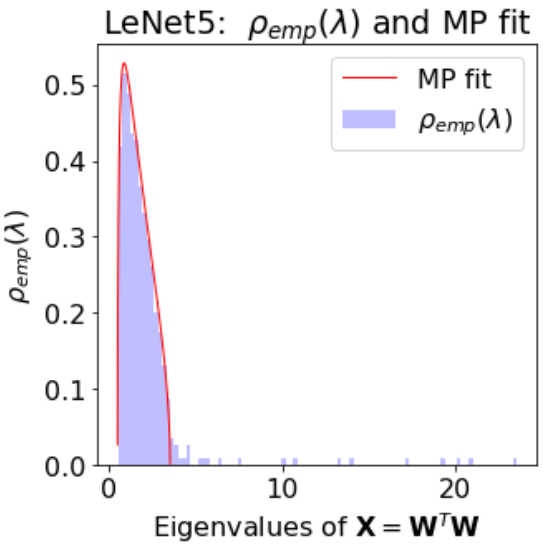
Calculation Consulting, charles@calculationconsulting.com)

Lots of DNNs analyzed: Look at nearly every publicly-available SOTA model in CV and NLP

- *Don't evaluate your method on one/two/three NNs, evaluate it on:*
  - *dozens (2017)*
  - *hundreds (2019)*
  - *thousands (2021)*
- *Don't use bad/toy models, use SOTA models.*
  - *If you do, don't be surprised if low-quality/toy models are different than high-quality/SOTA models.*
- *Don't train models, instead validate pre-trained models.*
  - *Validating models is harder than training models.*

Results: LeNet5 (an old/small NN example)





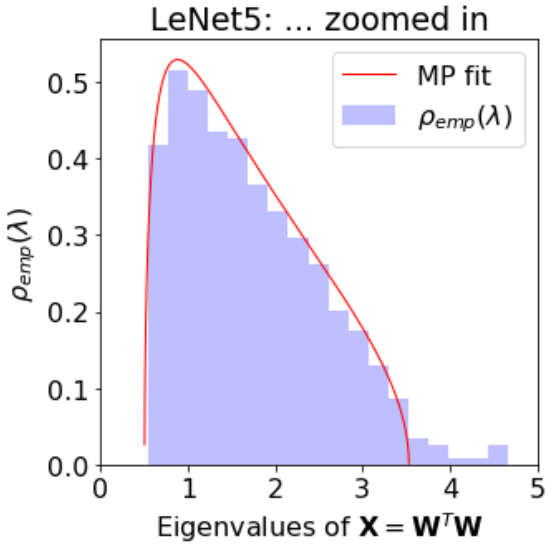
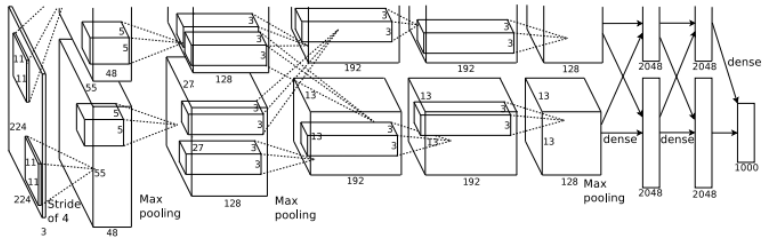
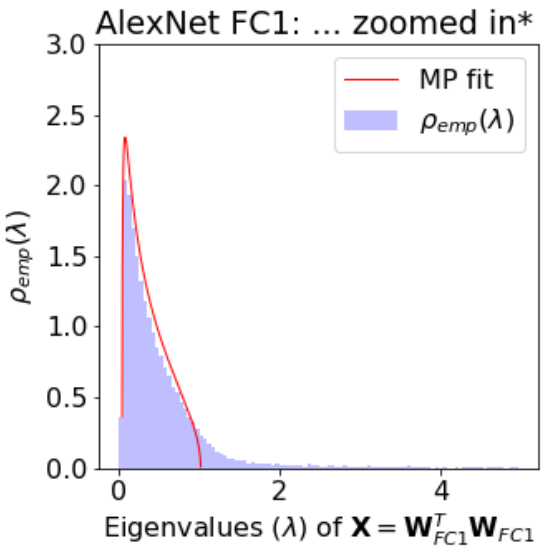


Figure: Full and zoomed-in ESD for LeNet5, Layer FC1.

Older and/or smaller and/or less well-trained models look like bulk+spike.

Results: AlexNet (a typical modern/large DNN example)





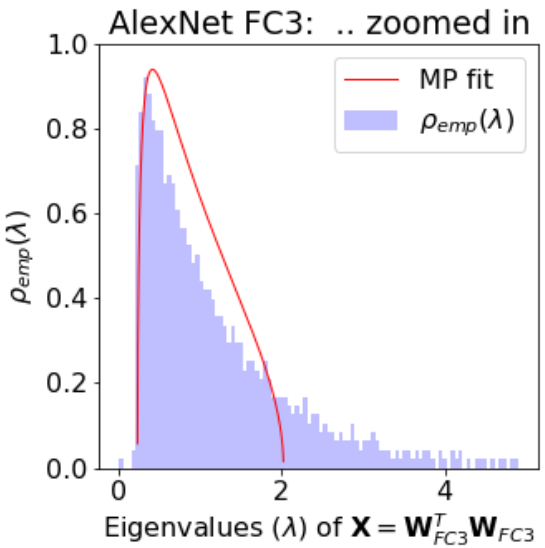
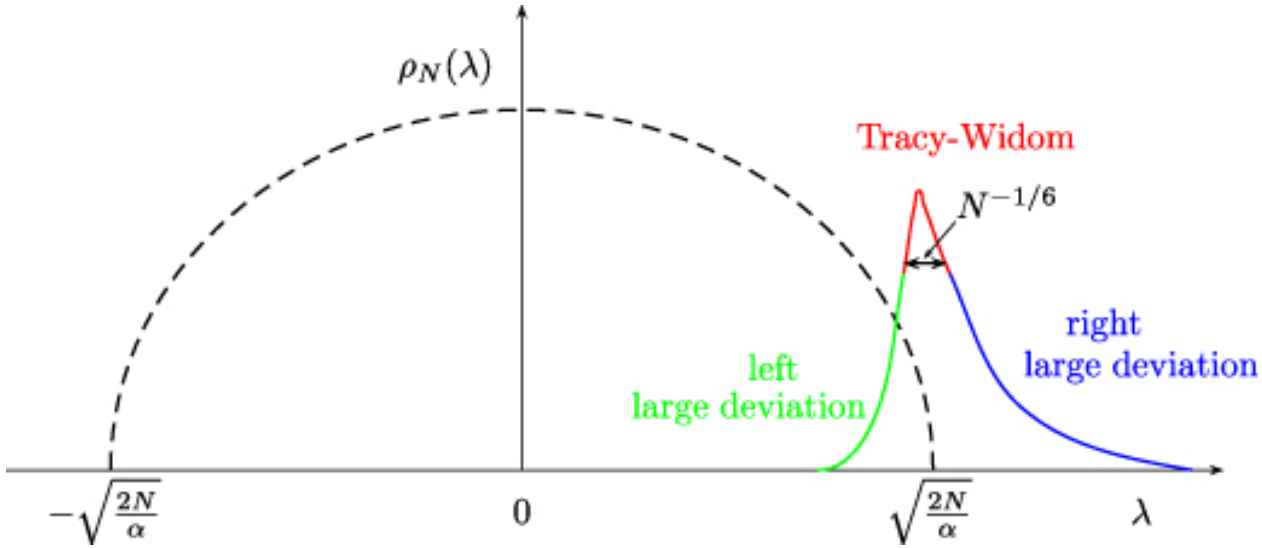


Figure: Zoomed-in ESD for Layer FC1 and FC3 of AlexNet.

Newer SOTA models have heavy-tail structure in their weight matrix correlations (i.e., not elements but eigenvalues).

Random Matrix Theory 101: Wigner and Tracy-Widom

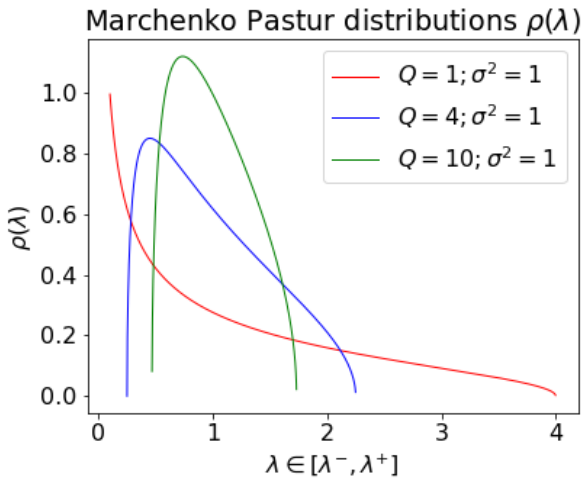
- Wigner: *global bulk statistics* approach universal semi-circular form
- Tracy-Widom: *local edge statistics* fluctuate in universal way



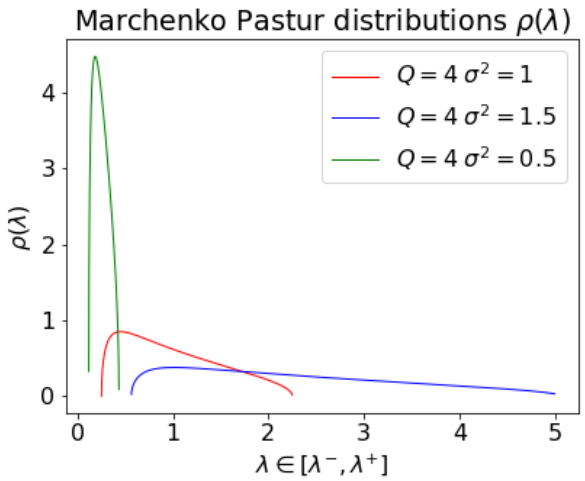
Problems with Wigner and Tracy-Widom:

- Weight matrices usually not square
- Typically do only a single training run

Random Matrix Theory 102': Marchenko-Pastur



(c) Vary aspect ratios



(d) Vary variance parameters

Figure: Marchenko-Pastur (MP) distributions.

Important points:

- *Global bulk stats*: The overall shape is deterministic, fixed by Q and σ.
- *Local edge stats*: The edge λ^+ is very crisp, i.e., $\Delta\lambda_M = |\lambda_{max} - \lambda^+| \sim O(M^{-2/3})$, plus Tracy-Widom fluctuations.

We use both *global bulk statistics* as well as *local edge statistics* in our theory.

Random Matrix Theory 103: Heavy-tailed RMT

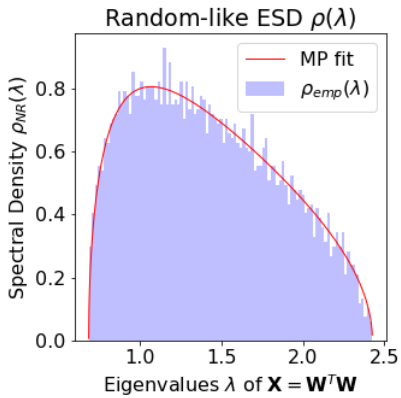
Go beyond the (relatively easy) Gaussian Universality class:

- *model* strongly-correlated systems ("signal") with heavy-tailed random matrices.

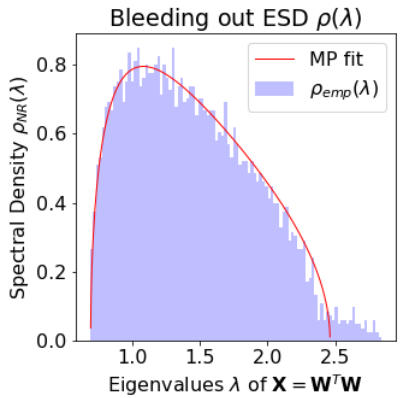
	Generative Model w/ elements from Universality class	Finite-N Global shape $\rho_N(\lambda)$	Limiting Global shape $\rho(\lambda),\ N \to \infty$	Bulk edge Local stats $\lambda \approx \lambda^+$	(far) Tail Local stats $\lambda \approx \lambda_{max}$
Basic MP	Gaussian	MP distribution	MP	TW	No tail.
Spiked-Covariance	Gaussian, + low-rank perturbations	MP + Gaussian spikes	MP	TW	Gaussian
Heavy tail, $4 < \mu$	(Weakly) Heavy-Tailed	MP + PL tail	MP	Heavy-Tailed*	Heavy-Tailed*
Heavy tail, $2 < \mu < 4$	(Moderately) Heavy-Tailed (or "fat tailed")	PL** $\sim \lambda^{-(a\mu+b)}$	PL $\sim \lambda^{-(\frac{1}{2}\mu+1)}$	No edge.	Frechet
Heavy tail, $0 < \mu < 2$	(Very) Heavy-Tailed	PL** $\sim \lambda^{-(\frac{1}{2}\mu+1)}$	PL $\sim \lambda^{-(\frac{1}{2}\mu+1)}$	No edge.	Frechet

Basic MP theory, and the spiked and Heavy-Tailed extensions we use, including known, empirically-observed, and conjectured relations between them. Boxes marked "*" are best described as following "TW with large finite size corrections" that are likely Heavy-Tailed, leading to bulk edge statistics and far tail statistics that are indistinguishable. Boxes marked "**" are phenomenological fits, describing large ($2 < \mu < 4$) or small ($0 < \mu < 2$) finite-size corrections on $N \to \infty$ behavior.

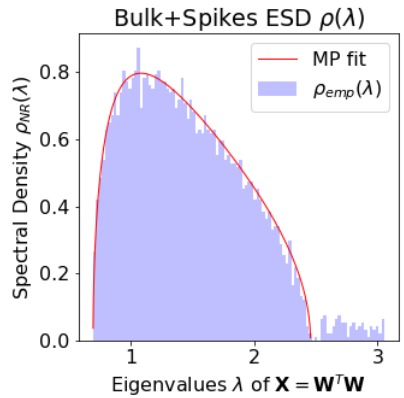
RMT-based 5+1 Phases of Training (in pictures)



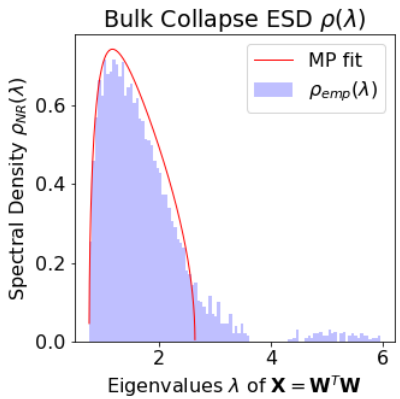
(a) RANDOM-LIKE.



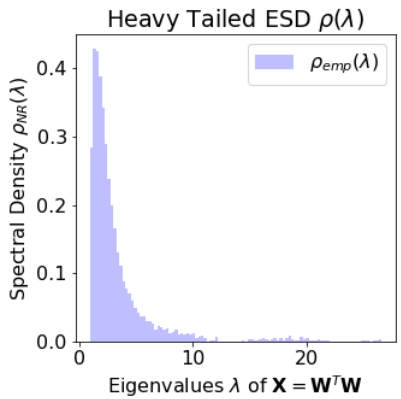
(b) BLEEDING-OUT.



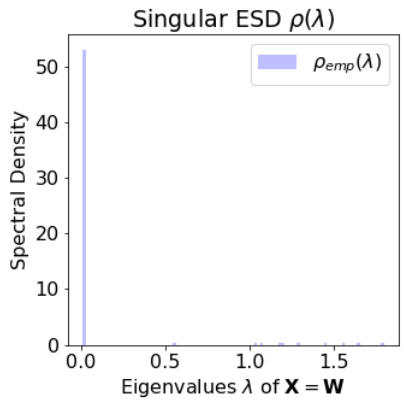
(c) BULK+SPIKES.



(d) BULK-DECAY.



(e) HEAVY-TAILED.



(f) RANK-COLLAPSE.

Figure: The 5+1 phases of learning we identified in DNN training.

Bulk+Spikes: Small Models ∼ Tikhonov regularization

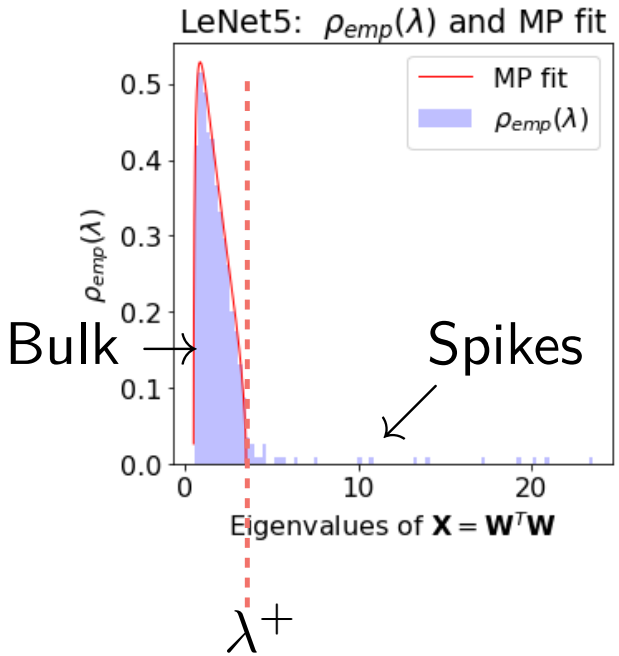
Low-rank perturbation

$$\mathbf{W}_l \simeq \mathbf{W}_l^{rand} + \Delta^{large}$$

Perturbative correction

$$\lambda_{max} = \sigma^2 \left(\frac{1}{Q} + \frac{|\Delta|^2}{N}\right)\left(1 + \frac{N}{|\Delta|^2}\right)$$

$$|\Delta| > (Q)^{-\frac{1}{4}}$$



simple scale threshold

$$\mathbf{x} = \left(\hat{\mathbf{X}} + \alpha\mathbf{I}\right)^{-1} \hat{\mathbf{W}}^T\mathbf{y}$$

eigenvalues $> \alpha$ (Spikes) carry most of the signal/information

Smaller, older models like LeNet5 exhibit traditional regularization and can be described perturbatively with Gaussian RMT

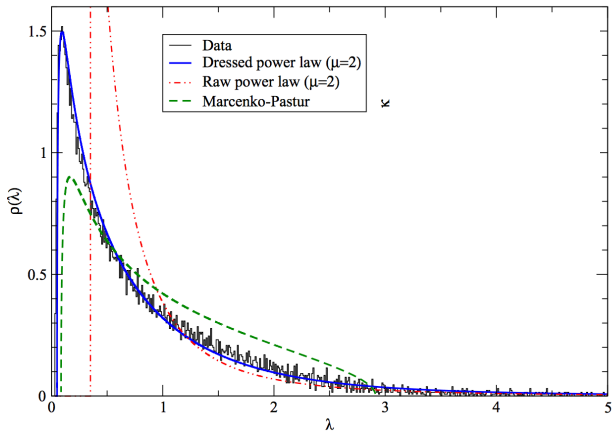
Heavy-tailed Self-regularization

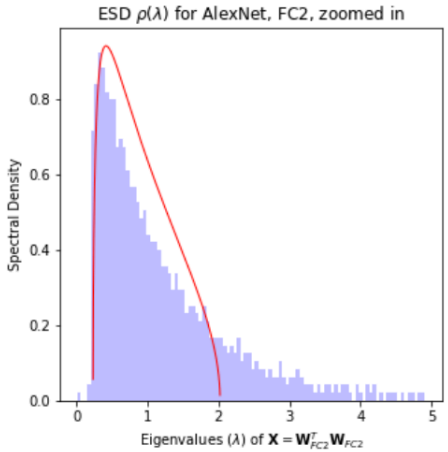
W is *strongly-correlated* and highly non-random

- We *model* strongly-correlated systems by heavy-tailed random matrices
- We *model* signal (not noise) by heavy-tailed random matrices

Then RMT/MP ESD will also have heavy tails.

- The eigenvalues are heavy-tailed; the weights are NOT.



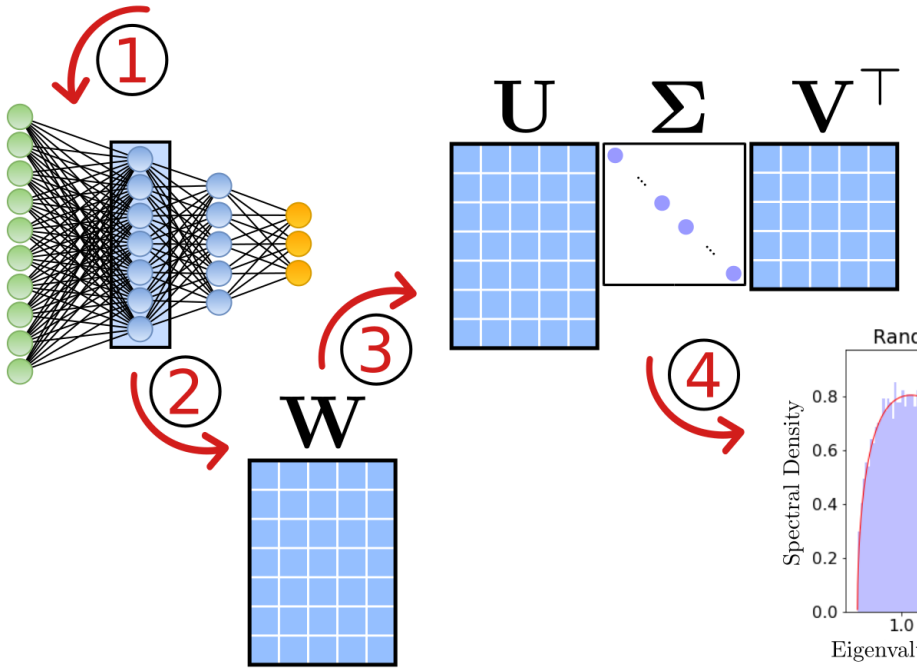


"All" larger, modern DNNs exhibit novel Heavy-tailed self-regularization

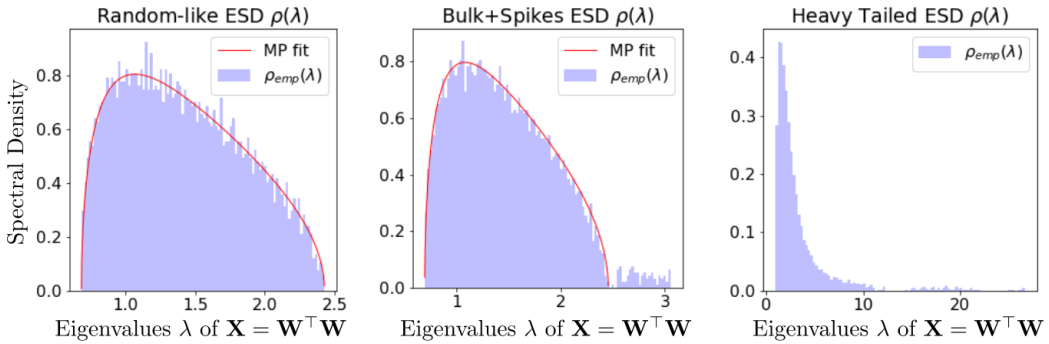
Watching weights with WeightWatcher

https://github.com/CalculatedContent/WeightWatcher

Analyzing DNN Weight matrices with WeightWatcher



1. Take a model
   - 2. Take a weight matrix
     - 3. Do Spectral analysis
       - 4. Histogram of eigenvalues



- Analyze one layer of pre-trained model
- Compare multiple layers of pre-trained model
- Monitor NN properties as you train your own model

"pip install weightwatcher"

Using the theory

Different ways one could *use* a theory.

- Perform diagnostics for model validation, to develop hypotheses, etc.*
- Make predictions about model quality, generalization, transferability, etc.*
- Did post-training modifications damage my model?*
- Will buying more data help?*
- Will training longer help?*
- Will quantizing or distilling help?*
- Construct a regularizer to do model training.**

*Ideally, by peeking at very little or no data.

**If you have lots of data, lots of GPUs, etc.

Predicting test accuracies ... lots of metrics ...

- Average log norm (a VC-like data-dependent capacity metric):

$$\langle \log \|\mathbf{W}\| \rangle = \frac{1}{N} \sum_{l,i} \log \|\mathbf{W}_{l,i}\| = \frac{1}{N} \sum_{l,i} \log(\lambda_{l,i}^{max})$$

- Average alpha (also data-dependent, from HT-SR theory):

$$\alpha = \frac{1}{N} \sum_{l,i} \alpha_{l,i}$$

- Combine the two into a weighted average (weighted to compensate for different size and scale of feature maps):

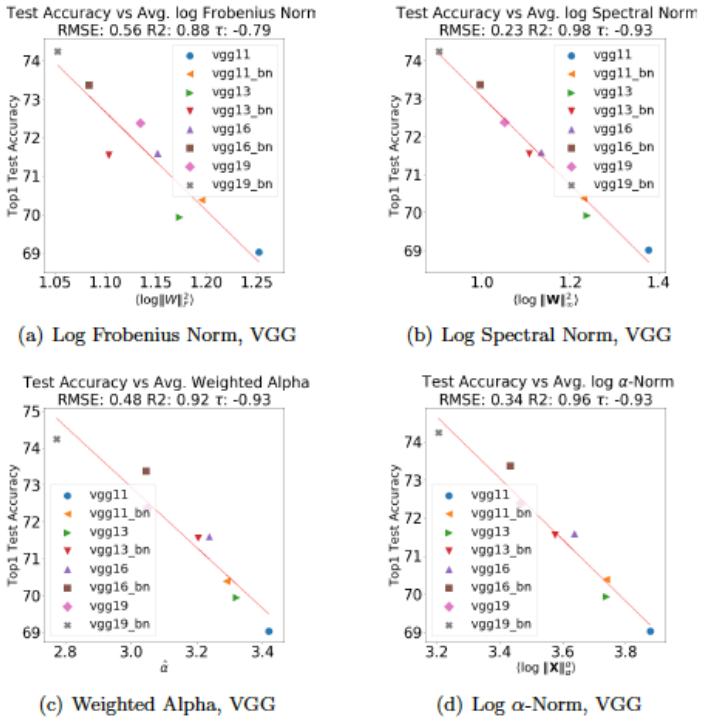
$$\hat{\alpha} = \frac{1}{N} \sum_{l,i} \log(\lambda_{l,i}^{max}) \alpha_{l,i}$$

- In a special case ($\alpha \approx 2$), for each layer:

PL–Norm Relation: $\alpha \log \lambda^{max} \approx \log \|\mathbf{W}\|_F^2$.

"pip install weightwatcher"

(The first) large-scale study (meta-analysis) of hundreds of SOTA pretrained models [‡]



(a) Log Frobenius Norm, VGG (b) Log Spectral Norm, VGG

(c) Weighted Alpha, VGG (d) Log α-Norm, VGG

Figure 2: Comparison of Average Log Norm and Weighted Alpha quality metrics versus re test accuracy for pretrained VGG models: VGG11, VGG13, VGG16, and VGG19, with and

Different metrics on pre-trained VGG.

Series	#	Metric	$\langle \log \|\mathbf{W}\|_F^2 \rangle$	$\langle \log \|\mathbf{W}\|_\infty^2 \rangle$	$\hat{\alpha}$	$\langle \log \|\mathbf{X}\|_\alpha^\alpha \rangle$
VGG	6	RMSE	0.56	**0.23**	0.48	0.34
		R^2	0.88	**0.98**	0.92	0.96
		Kendall-τ	-0.79	**-0.93**	**-0.93**	**-0.93**
ResNet	5	RMSE	0.9	0.97	**0.61**	0.66
		R^2	0.92	0.9	**0.96**	0.9
		Kendall-τ	-1.0	-1.0	-1.0	-1.0
ResNet-1K	19	RMSE	2.4	2.8	**1.8**	1.9
		R^2	0.81	0.74	**0.89**	0.88
		Kendall-τ	-0.79	-0.79	**-0.89**	-0.88
DenseNet	4	RMSE	0.3	**0.11**	0.16	0.21
		R^2	0.93	**0.99**	0.98	0.97
		Kendall-τ	-1.0	-1.0	-1.0	-1.0

Table 1: Quality metrics (for RMSE, smaller is better; for R^2, larger is better; and for Kendall-τ rank correlation, larger magnitude is better) for reported Top1 test error for pretrained models in each architecture series. Column # refers to number of models. VGG, ResNet, and DenseNet were pretrained on ImageNet. ResNet-1K was pretrained on ImageNet-1K.

Summary statistics: VGG; ResNet; DenseNet.

	$\log \|\cdot\|_F^2$	$\log \|\cdot\|_\infty^2$	$\hat{\alpha}$	$\log \|\cdot\|_\alpha^\alpha$
RMSE (mean)	4.84	5.57	4.58	4.55
RMSE (std)	9.14	9.16	9.16	9.17
R2 (mean)	3.9	3.85	3.89	3.89
R2 (std)	9.34	9.36	9.34	9.34
Kendal-tau (mean)	3.84	3.77	3.86	3.85
Kendal-tau (std)	9.37	9.4	9.36	9.36

Table 3: Comparison of linear regression fits for different average Log Norm and Weighted Alpha metrics across 5 CV datasets, 17 architectures, covering 108 (out of over 400) different pretrained

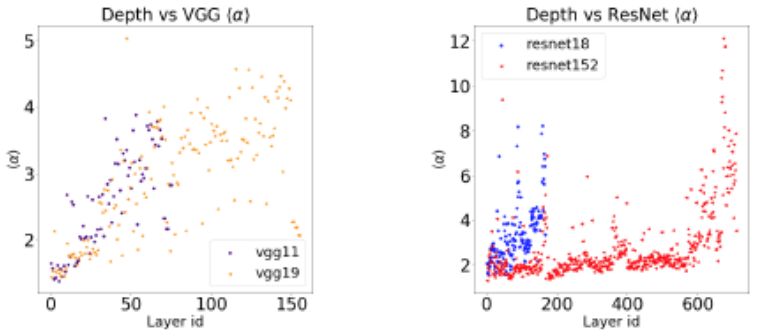
Summary statistics: hundreds of models.

Lots more plots to prove we can "predict trends . . . without access . . . "

[‡] "Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data," Martin, Peng, and Mahoney, arXiv:2002.06716, *Nature Communications*, 2021.

Using a theory: on SOTA models

Analyzing pre-trained models: properties of VGG vs ResNet vs DenseNet leads to the idea of *correlation flow*.



(a) VGG (b) ResNet

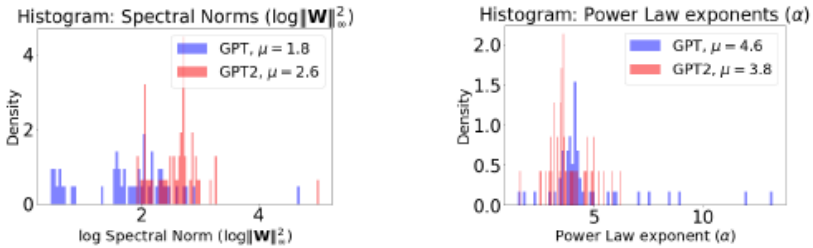
(c) DenseNet (d) ResNet (overlaid)

Figure 4: PL exponent (α) versus layer id, for the least and the most accurate models in VGG (a), ResNet (b), and DenseNet (c) series. (VGG is without BN; and note that the Y axes on

Alpha versus depth: VGG, ResNet, DenseNet.

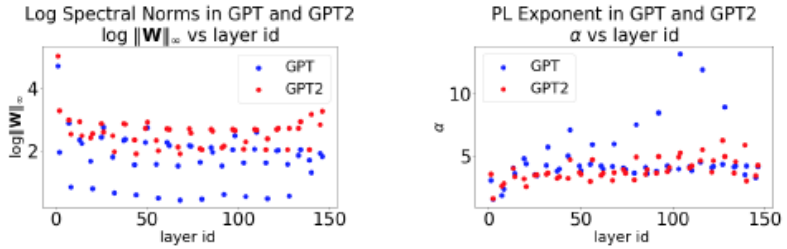
Using a theory: on SOTA models

Analyzing pre-trained models: properties of GPTx series leads to the idea of *scale collapse*.



(a) Log Spectral Norm ($\log\|\mathbf{W}\|_\infty$) (b) PL exponent ($\alpha$)

Figure 6: Histogram of PL exponents and Log Spectral Norms for weight matrices from the OpenAI GPT and GPT2-small pretrained models.



(a) Log Spectral Norm ($\log\|\mathbf{W}\|_\infty$) (b) PL exponent ($\alpha$)

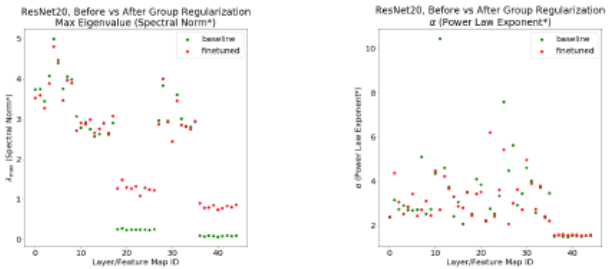
Figure 7: Log Spectral Norms (in (a)) and PL exponents (in (b)) for weight matrices from the OpenAI GPT and GPT2-small pretrained models. (Note that the quantities shown on each Y axis are different.) In the text, this is interpreted in terms of Scale Collapse and Correlation Flow.

Histogram and depth plots of $\alpha_{l,i}$ and $\lambda_{l,i}^{max}$.

Using a theory: easy to break popular SLT metrics

Easy to "break" popular SLT metrics:

- they are *not* validated counterfactually
- (but they drive the development of models)



(a) λ_{max} for ResNet20 layers (b) α for ResNet20 layers

Figure 5: ResNet20, distilled with Group Regularization, as implemented in the `distiller` (4D_regularized_5Lremoved) pretrained models. Log Spectral Norm ($\log \lambda_{max}$) and PL exponent (α) for individual layers, versus layer id, for both baseline (before distillation, green) and fine-tuned (after distillation, red) pretrained models.

Intel's distillation "broke" their models.

Series	#	$\langle \log \|\mathbf{W}\|_F \rangle$	$\langle \log \|\mathbf{W}\|_\infty \rangle$	$\hat{\alpha}$	$\langle \log \|\mathbf{X}\|_\alpha^\alpha \rangle$
GPT	49	1.64	1.72	7.01	7.28
GPT2-small	49	2.04	2.54	9.62	9.87
GPT2-medium	98	2.08	2.58	9.74	10.01
GPT2-large	146	1.85	1.99	7.67	7.94
GPT2-xl	194	1.86	1.92	7.17	7.51

Table 2: Average value for the average Log Norm and Weighted Alpha metrics for pretrained OpenAI GPT and GPT2 models. Column # refers to number of layers treated. Averages do

GPTx series: how does a model trained to "bad" data differ from one trained to "good" data?

Using a theory: leads to predictions

Based on analyzing hundreds of pre-trained SOTA models:

- "**Correlation flow**":
  - "Shape" of ESD of adjacent layers, as well as overlap between eigenvectors of adjecent layers, should be well-aligned.
- "**Scale collapse**":
  - "Size" of ESD of one or more layers changes dramatically, while the size of other layers changes very little, as a function of some perturbation of a model, during training (or post-training modification).
- "**Correlation traps**":
  - Spuriously large eigenvalues[§] may appear, and they may even be important for model convergence.

We can measure these quantities with Weightwatcher—so can you!

[§] Eigenvalues not due to signal in the data—we have theorems-style theory for Hessians ("Hessian Eigenspectra of More Realistic Nonlinear Models," Liao and Mahoney, `https://arxiv.org/abs/2103.01519`), but it's still open for Weights.

Overview

Motivations:

- WeightWatcher, Weight Diagnostics for Analyzing ML Models (with Charles H. Martin)
- Randomized Numerical Linear Algebra for Modern ML (with Michal Derezinski)

Some Theory:

- RMT for NNs: Linear to Nonlinear; Shallow to Deep; etc. (with Zhenyu Liao)

Applications:

- Models of Heavy-Tailed Mechanistic Universality (with Zhichao Wang and Liam Hodgkinson)
- Spectral Estimation with Free Decompression (with Siavash Ameli, Chris van der Heide, and Liam Hodgkinson)
- Determinant Estimation under Memory Constraints and Neural Scaling Laws (with S. Ameli, C. van der Heide, L. Hodgkinson, and F. Roosta)

Recent and Upcoming Developments in Randomized Numerical Linear Algebra for ML

Michał Dereziński and Michael W. Mahoney

University of Michigan and ICSI / LBNL / UC Berkeley

December 11, 2023

Outline

Part I
Foundations of RandNLA

1. Initial thoughts
   - Overview
2. Foundations of "classical" RandNLA
   - Matrix Multiplication
   - Least-squares Approximation
   - Low-rank Approximation
3. Foundations of "modern" RandNLA
   - Algorithmic Gaussianization via Random Matrix Theory
   - RMT for Sampling via DPPs

Part II
Recent and Upcoming Advances

4. Advances in RandNLA for Optimization
   - Gradient Sketch
   - Hessian Sketch
   - Sketch-and-Project
5. Advances in RandNLA for ML
   - Statistical Learning Approaches
   - Statistical Inference Approaches
   - Random Matrix Theory Approaches
6. Putting Randomness into LAPACK
   - RandBLAS/RandLAPACK
7. Concluding thoughts

RandNLA: Randomized Numerical Linear Algebra

- **"Classical" RandNLA**:
  - Sample/project and then solve subproblem or construct preconditioner
  - Theory from TCS/NLA, typically based on JL / subspace embeddings
  - Lots of data/ML and scientific computing applications
  - Initial proof-of-principle implementations (low-rank approximation, least-squares, optimization, etc.)
  - Relatively large theory-practice gap (esp. when used in ML pipelines)
- **"Modern" RandNLA**:
  - More sophisticated theory going beyond worst-case JL / subspace embeddings, with stronger connections to RMT
  - Improved statistical analysis and improved optimization algorithms
  - Implementations in **RandBLAS/RandLAPACK**, and more demands from GPU-based ML model training and scientific computing
  - Smaller theory-practice gap
- Opens up door to new theory, new implementations, new applications, ...

Basic Principles of "Classical" RandNLA [DM16]

Basic RandNLA method: given an input matrix:

- Construct a "sketch" (a smaller or sparser matrix that represents the essential information in the original matrix) by random sampling.
- Use that sketch as a surrogate to compute quantities of interest.

Basic design principles[1] underlying RandNLA:

- Randomly sample (in a careful data-dependent manner) a small number of elements to create a much sparser sketch of the original matrix.
- Randomly sample (in a careful data-dependent manner) a small number of columns and/or rows to create a much smaller sketch of the original matrix.
- Preprocess an input matrix with a random-projection-type matrix and then do uniform sampling of rows/columns/elements in order to create a sketch.

[1] First two principles deal with identifying nonuniformity structure. Third principle deals with preconditioning input (*i.e.*, uniformizing nonuniformity structure) s.t. uniform random sampling performs well.

Part I: Foundations of RandNLA

1. Initial thoughts
   - Overview
2. Foundations of "classical" RandNLA
   - Matrix Multiplication
   - Least-squares Approximation
   - Low-rank Approximation
3. Foundations of "modern" RandNLA
   - Algorithmic Gaussianization via Random Matrix Theory
   - RMT for Sampling via DPPs

Subspace Embeddings [Mah11, Woo14]

Definition

Let U be an $m \times n$ orthogonal matrix, and let S be any $n \times m$ matrix. Then, S is a *subspace embedding* if

$$\|U^T U - (SU)^T SU\|_2 = \|I - (SU)^T SU\|_2 \leq \epsilon.$$

Things to note:

- Many constructions (random sampling and projection methods, deterministic constructions, hashing functions, etc.) satisfy this condition.
- First used in data-aware context with leverage score sampling [DMM06, DMM08]
- Used in data-oblivious context with Hadamard-based projections [Sar06, DMMS10]
- For NLA, this is an acute perturbation.
- For TCS, this is a subspace analogue of JL lemma.

This is a "must must have" for TCS; for everyone else, it's optional.

- Numerical implementations: loosing rank still gives a good preconditioner.
- Statistics and machine learning: loosing rank introduces a bit of bias.

Least-squares approximation

Least-squares (LS) : given $m \times n$ matrix A and m-dimensional vector b, solve

$$x_{opt} = \arg \min_{x \in \mathbb{R}^n} \|Ax - b\|_2.$$

- If $m \gg n$, it is overdetermined/overconstrained.
- Compute solution in $O(mn^2)$ time (in RAM model) with one of several methods: normal equations; QR decompositions; or SVD.
- RandNLA provides faster algorithms for this ubiquitous problem.
  - **TCS**: faster in terms of low-precision asymptotic worst-case theory.
  - **NLA**: faster in terms of high-precision wall-clock time.
  - **Implementations**: can compute (in Spark/MPI/etc.) low, medium, and high precision solutions on up to terabyte-sized data.
  - **Data Applications**: faster algorithms and/or implicit regularization for many machine learning and data science problems.
- *The basic RandNLA approach extends to many other matrix problems.*

Least-squares approximation: basic structural result

Consider the over-determined least-squares approximation problem:

$$\mathcal{Z}_2^2 = \min_{x \in \mathbb{R}^n} \|b - Ax\|_2^2 = \|b - Ax_{opt}\|_2^2$$

as well as the "preconditioned " the least-squares approximation problem:

$$\tilde{\mathcal{Z}}_2^2 = \min_{x \in \mathbb{R}^n} \|\Omega(b - Ax)\|_2^2 = \|b - A\tilde{x}_{opt}\|_2^2$$

where Ω is *any* matrix.

Theorem (Fundamental Structural Result for Least-Squares)

If Ω satisfies the two basic conditions (constants are somewhat arbitrary):

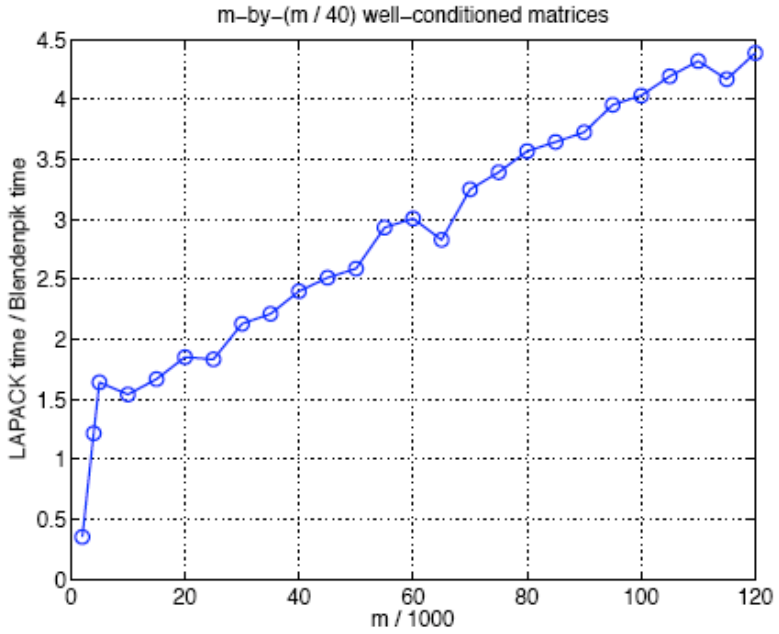
$$\begin{aligned} \sigma_{min}^2(\Omega U_A) &\geq 1/\sqrt{2} \\ \left\| U_A^T \Omega^T \Omega b^\perp \right\|_2^2 &\leq \epsilon \mathcal{Z}_2^2/2, \quad \text{where } b^\perp = b - U_A U_A^T A, \end{aligned}$$

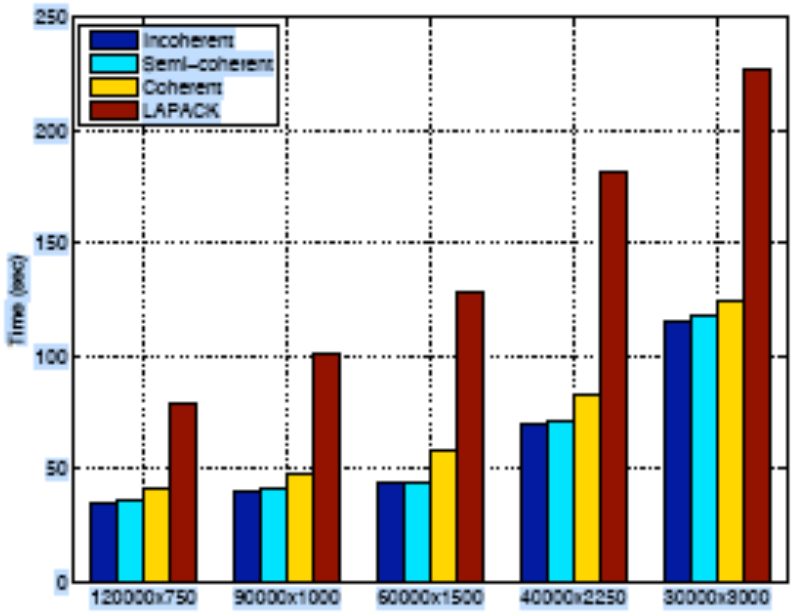
then:

$$\begin{aligned} \|A\tilde{x}_{opt} - b\|_2 &\leq (1+\epsilon)\mathcal{Z}_2 \\ \|x_{opt} - \tilde{x}_{opt}\|_2 &\leq \frac{1}{\sigma_{min}(A)} \sqrt{\epsilon} \mathcal{Z}_2. \end{aligned}$$

DMMS [DMMS10, Mah11]

Least-squares approximation: RAM implementations





Conclusions:

- *Randomized algorithms "beats Lapack's direct dense least-squares solver by a large margin on essentially any dense tall matrix."*
- *These results "suggest that random projection algorithms should be incorporated into future versions of Lapack."*

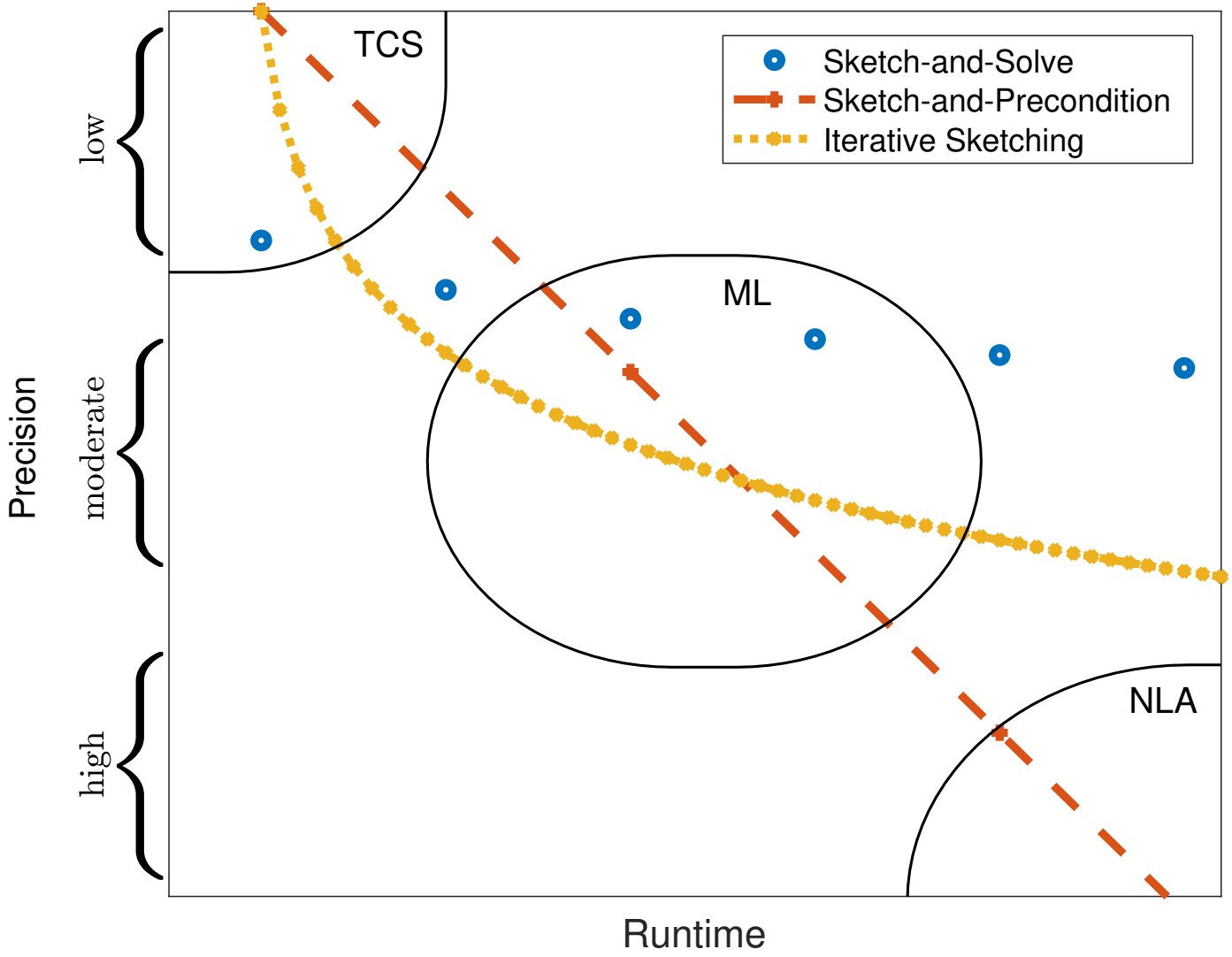
Avron, Maymounkov, and Toledo [AMT10]

Using RandNLA methods more generally ...

Three paradigms that apply more broadly than least squares:

1. Sketch-and-solve: Construct a *smaller* least squares problem; then solve it using a direct method.
   - Low-precision estimate, e.g., $\epsilon = 0.1$
   - Simplest to highlight structure of the theory
2. Iterative sketching: Repeatedly sketch/sub-sample the problem; and iteratively refine the estimate.
   - Medium (to high, depending on method) precision estimate, e.g., $\epsilon = 10^{-3}$
   - SGD, SGD++, sketch-and-project, preconditioned weighted SGD
3. Sketch-and-precondition: Construct an *equivalent* but well-conditioned problem; then use a deterministic iterative method.
   - High-precision solution, e.g., $\epsilon = 10^{-10}$
   - Best (usually) for high-quality numerical solutions

Using RandNLA methods more generally ...



Part I: Foundations of RandNLA

1. Initial thoughts
   - Overview
2. Foundations of "classical" RandNLA
   - Matrix Multiplication
   - Least-squares Approximation
   - Low-rank Approximation
3. Foundations of "modern" RandNLA
   - Algorithmic Gaussianization via Random Matrix Theory
   - RMT for Sampling via DPPs

The proportional limit

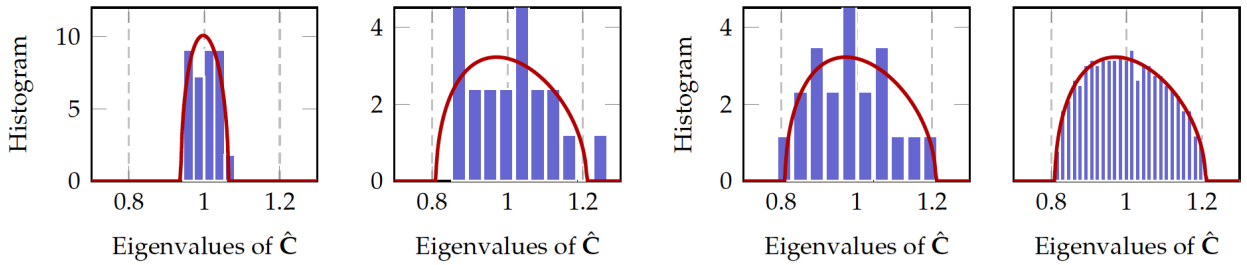


Figure: Histogram of the eigenvalues of $\hat{\mathbf{C}}$ (**blue**) versus the Marčenko-Pastur law (**red**), for $\mathbf{X}$ having standard Gaussian entries in different settings: (**left**: small versus large dimensional intuition) $p = 20, n = 1\,000p$ versus $p = 20, n = 100p$; and (**right**: non-asymptotic versus asymptotic MP law) $p = 20, n = 100p$ versus $p = 500, n = 100p$.

Consider $A \in \mathbb{R}^{n \times d}$ and iid Gaussian sketching matrix $S \in \mathbb{R}^{l \times d}$

Quality of $\tilde{A} = SA$ is often measured by $\mathrm{cond}(SU)$ for $U = \texttt{orth}(A)$ (e.g., subspace embedding, quality of a preconditioner, etc.)

Thanks to the rotation invariance of Gaussian distribution, SU is also Gaussian, so we can use the Marchenko-Pastur law:

$$\sigma_{\min}(SU) \sim 1 - \sqrt{\frac{d}{l}}, \qquad \sigma_{\max}(SU) \sim 1 + \sqrt{\frac{d}{l}}$$

Question: Can we obtain similar results with non-Gaussian sketches?

RMT analysis in RandNLA

Consider sketching matrix $S \in \mathbb{R}^{l \times n}$ with iid Gaussian entries.

- Sketch-and-precondition: Construct R^{-1} from the QR of SA

$$\mathrm{cond}(AR^{-1}) \leq 6 \quad \text{with high probability for} \quad l \geq 2d.$$

- Sketch-and-solve: $\hat{x} = \mathrm{argmin}_x \, \|S(Ax - b)\|_2^2$

$$\mathbb{E}\|A(\hat{x} - x^*)\|_2^2 = \frac{d}{l - d - 1}\|Ax^* - b\|_2^2 \quad \text{for} \quad l \geq d + 2.$$

- Low-rank approximation: Compute $Q = \texttt{orth}(AS)$

$$\mathbb{E}\|A - QQ^\top A\|_F^2 \leq \left(1 + \frac{k}{l - k - 1}\right) \cdot \|A - A_k\|_F^2 \quad \text{for} \quad l \geq k + 2.$$

These are all easy to show for iid Gaussian matrices.

Inversion bias: the key challenge [DM19, DLDM21]

Given $n \times d$ data matrix A of rank d, where $n \geq d$,

approximate $F((A^\top A)^{-1})$, where $F(\cdot)$ is a linear functional.

- $(A^\top A)^{-1}b$, for a vector b:
  - Is the OLS solution (multivariate statistical analysis, Newton's method in numerical optimization, etc.)
- $x^\top (A^\top A)^{-1}x$, for a vector x:
  - If $x = a_i$ is one of the rows of A, then it is leverage scores
  - If $x = \mathbf{e}_i$ is a standard basis vector, then this is the squared length of the confidence interval for the i-th coefficient in OLS
- $\operatorname{tr} C(A^\top A)^{-1}$ for a matrix C:
  - Used to quantify uncertainty
  - Used for experimental design criteria, e.g., A-designs and V-designs

Inversion bias: $\mathbb{E}[(\tilde{A}^\top \tilde{A})^{-1}] \neq (A^\top A)^{-1}$, even though $\mathbb{E}[\tilde{A}^\top \tilde{A}] = A^\top A$

Why focus on the inverse?

- Consider $S \in \mathbb{R}^{l \times n}$ having i.i.d. zero-mean rows statistically.
- $A^\top S^\top S A$ is a *sample covariance estimator* of the "population covariance matrix" $A^\top A \in \mathbb{R}^{d \times d}$.
- How does the spectrum differ between *sample* and *population* covariance?
- RMT answers this by looking at the *resolvent matrix*:

$$(A^\top S^\top S A - zI)^{-1} \quad \text{for} \quad z \in \mathbb{C} \setminus \mathbb{R}_+.$$

- The Stieltjes transform (normalized trace of the resolvent) exhibits *inversion bias*, leading to discrepancy between sample and population.
- Traditional RMT studies limiting eigenvalue distribution as $l, n, d \to \infty$.
- Our goal: *precise* and *non-asymptotic* results on resolvent matrices for sketching, e.g., $(A^\top S^\top S A)^{-1}$, leading to RMT analysis for RandNLA.

Correcting the bias (for Gaussian sketching matrices)

Consider $\hat{H} = \tilde{A}^\top \tilde{A} \approx A^\top A = H$,

(where $\tilde{A} = SA$ is an $l \times d$ sketch of an $n \times d$ matrix A)

Simple correction for a Gaussian sketching matrix S:

- Rescale by a dimensional factor: $\mathbb{E}\left[(\gamma \hat{H})^{-1}\right] = H^{-1}$ for $\gamma = \frac{l}{l-d-1}$

This is **not** true for other sketching methods. Other sketches:

- are *not perfectly rotationally symmetric*, etc.
- could *lose rank*, with very small probability
- suffer from *"coupon collector" problems*

In general, the *bias occurs differently in each direction*,

(so you cannot correct it with a single rescaling)

Q: Can we quickly correct the inversion bias, exactly or approximately?

Near-unbiasedness: an (ϵ, δ)-unbiased estimator

This motivates the following definition.

Definition

A random p.s.d. matrix $\tilde{C}$ is an (ϵ, δ)-unbiased estimator of C if there is an event $\mathcal{E}$ that holds with probability $1 - \delta$ such that

$$\mathbb{E}_{\mathcal{E}}[\tilde{C}] \approx_{1+\epsilon} C, \quad \text{and} \quad \tilde{C} \preceq O(1) \cdot C \quad \text{when conditioned on } \mathcal{E}.$$

Sub-gaussian sketches have small inversion bias

Consider a full rank $n \times d$ matrix A with $n \gg d$.

Proposition (Near-unbiasedness of sub-gaussian sketches)

Let S be an $m \times n$ random matrix such that $\sqrt{m}\,S$ has i.i.d. $O(1)$-sub-gaussian entries with mean zero and unit variance.

$$\textit{If} \qquad m \geq C\big(d + \sqrt{d}/\epsilon + \log(1/\delta)\big), \qquad \textit{then}$$

$(\frac{m}{m-d} A^\top S^\top S A)^{-1}$ is an (ϵ, δ)-unbiased estimator of $(A^\top A)^{-1}$.

So, there is an event $\mathcal{E}$ that holds with probability $1 - \mathrm{e}^{-cm}$, s.t.

$$\mathbb{E}_{\mathcal{E}}\left[(\tfrac{m}{m-d} A^\top S^\top S A)^{-1}\right] \approx_\epsilon (A^\top A)^{-1}, \quad \text{for} \quad \epsilon = O\left(\frac{\sqrt{d}}{m}\right).$$

[DLDM21]

Comparison with JL / subspace embeddings

Condition: Subspace embedding

Sketching matrix S with probability $1 - \delta$ satisfies

$$A^\top S^\top S A \approx_\eta A^\top A \qquad \text{for} \quad \eta = O(1).$$

Subspace embedding: w.h.p. $(A^\top S^\top S A)^{-1} \approx_\eta (A^\top A)^{-1}$

Near-unbiasedness: $\mathbb{E}_{\mathcal{S}}\left[(\frac{m}{m-d} A^\top S^\top S A)^{-1}\right] \approx_\epsilon (A^\top A)^{-1}$

For sub-gaussian sketches, we have:

$$\eta = \Theta\left(\sqrt{\frac{d}{m}}\right) \quad \text{and} \quad \epsilon = O\left(\frac{\sqrt{d}}{m}\right)$$

Subspace embedding is not enough to show near-unbiasedness!

Corollary for model averaging

Effectively, we showed that for sub-gaussian sketches:

$$\text{Bias}^2 \quad \ll \quad \text{Variance}$$

Corollary (Model averaging)

For $q = \tilde{O}(m)$ sub-gaussian sketches of size $m = O(d + \sqrt{d}/\epsilon)$,

$$\frac{1}{q}\sum_{i=1}^{q}\left(\tfrac{m}{m-d}A^\top S_i^\top S_i A\right)^{-1} \approx_\epsilon (A^\top A)^{-1}.$$

Applies to distributed averaging of linear functionals, e.g.:

$$\operatorname{tr} C\left(\tfrac{m}{m-d}A^\top S_i^\top S_i A\right)^{-1}.$$

Extending RMT-style analysis to fast sketching

- Most RMT for sketching requires:
  - different "gaussianization" assumptions
  - and different parameter regimes (e.g., proportional regime)

  compared to classical JL or subspace embedding approaches.
- Most out-of-the-box theory applies only to expensive dense Gaussian or sub-gaussian sketching matrices.
- Question: Can we extend this line of work to fast sketches, e.g., sparse or structured?
- Answer: **Yes!**

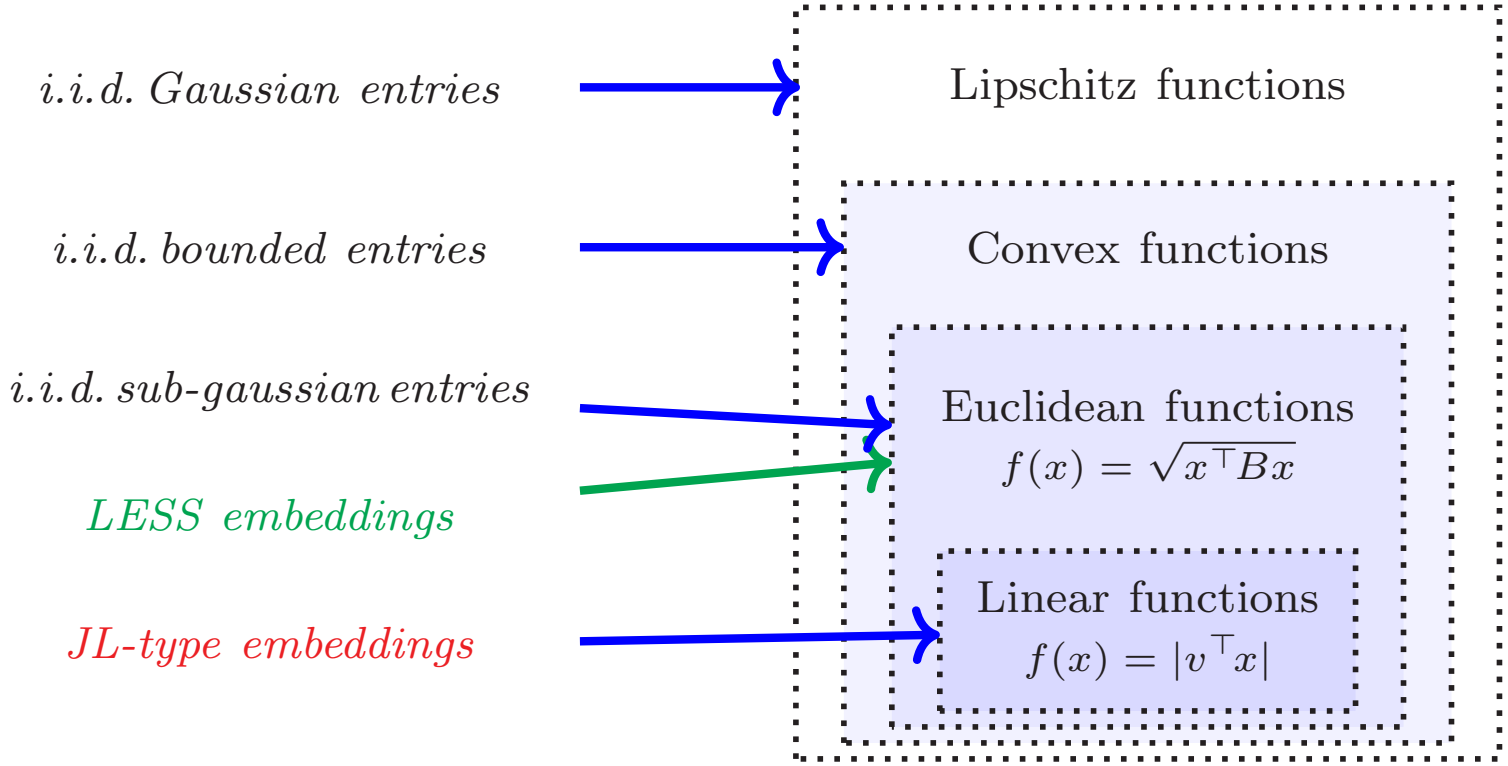
Landscape of Algorithmic Gaussianization

Sub-gaussian concentration of $x \in \mathbb{R}^d$ w.r.t. a set of functions $\mathcal{F}$

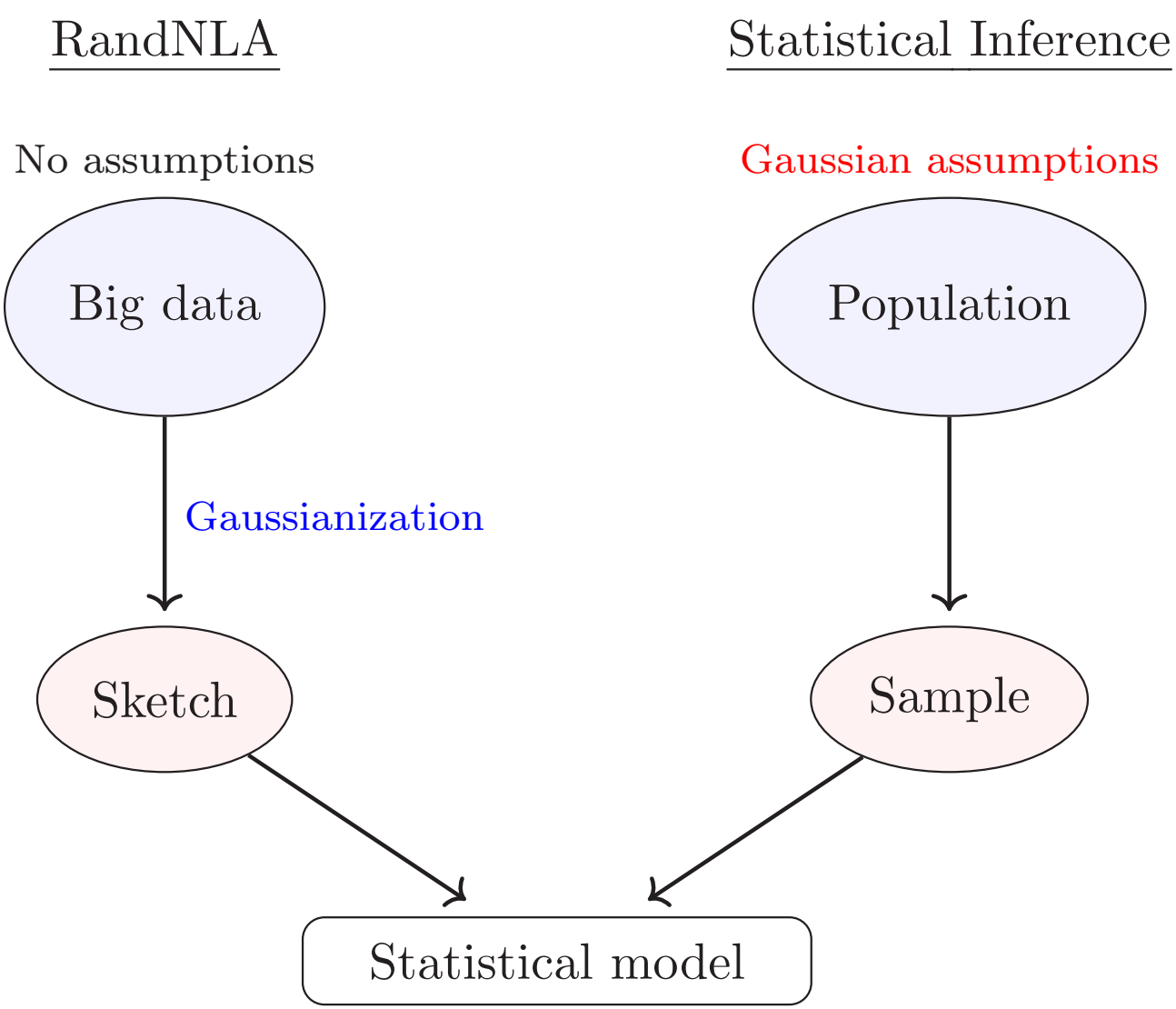
$$\forall f \in \mathcal{F}: \qquad X = f(x) - \mathbb{E}\, f(x) \quad \text{is} \quad \underbrace{O(\|f\|_{\text{Lip}})\text{-sub-gaussian}}_{\mathbb{E}\, \exp(cX^2/\|f\|_{\text{Lip}}) \leq 2}$$

Examples
$x \in \mathbb{R}^d$

Concentration
$\mathcal{F} \subseteq \{\mathbb{R}^d \to \mathbb{R}\}$

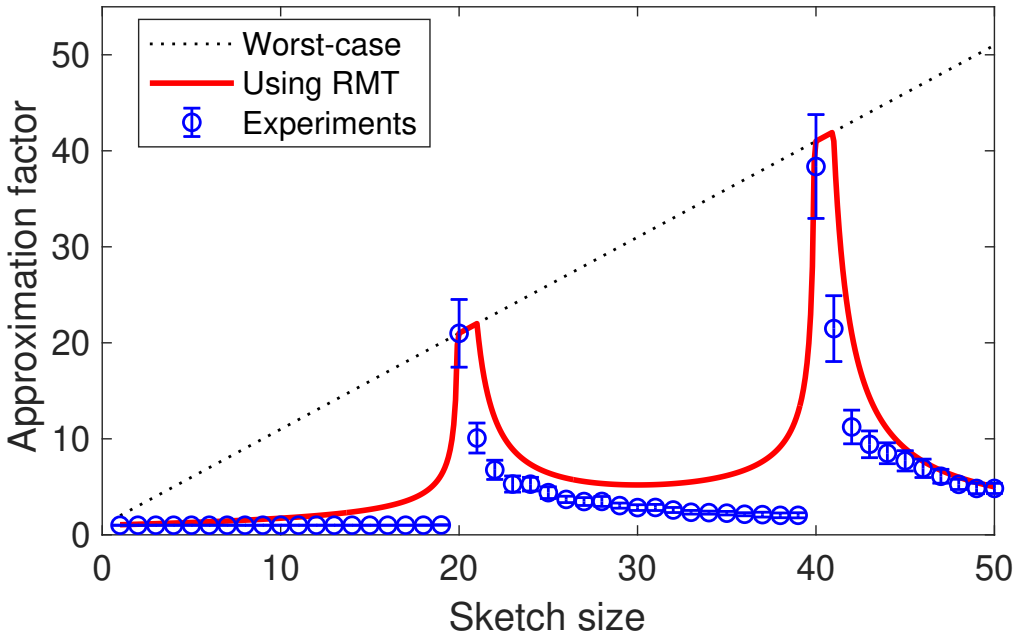


Gaussianization in RandNLA vs Statistical Inference



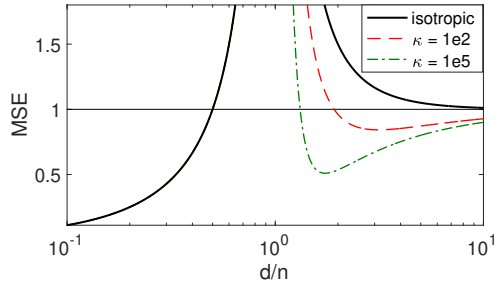
Multiple-descent in low-rank approximation

Theory: Characterizing the approximation factor using RMT [DKM20]



Connection: double descent in over-parameterized ML models [DLM20b]

"Classical" ML:	$\textit{parameters} \ll \textit{data}$
"Modern" ML:	$\textit{parameters} \gg \textit{data}$
Phase transition:	$\textit{parameters} \sim \textit{data}$



Developing standard libraries for RandNLA

RandBLAS

- Library that concerns basic sketching for dense data matrices.
- Reference implementation in C++.
- Hope: it grows to become a community standard for RandNLA, in the sense that its API would see wider adoption than any single implementation.

RandLAPACK

- Library that concerns algorithms for solving traditional linear algebra problems and advanced sketching functionality.
- To be written in C++, build on BLAS++/LAPACK++ portability layer
- Main drivers:
  - Least squares and optimization.
  - Low-rank approximation.
  - Full-rank decompositions.

"The RandLAPACK book"

Mathematics > Numerical Analysis

[Submitted on 22 Feb 2023]

Randomized Numerical Linear Algebra : A Perspective on the Field With an Eye to Software

Riley Murray, James Demmel, Michael W. Mahoney, N. Benjamin Erichson, Maksim Melnichenko, Osman Asif Malik, Laura Grigori, Piotr Luszczek, Michał Dereziński, Miles E. Lopes, Tianyu Liang, Hengrui Luo, Jack Dongarra

Randomized numerical linear algebra – RandNLA, for short – concerns the use of randomization as a resource to develop improved algorithms for large-scale linear algebra computations.

The origins of contemporary RandNLA lay in theoretical computer science, where it blossomed from a simple idea: randomization provides an avenue for computing approximate solutions to linear algebra problems more efficiently than deterministic algorithms. This idea proved fruitful in the development of scalable algorithms for machine learning and statistical data analysis applications. However, RandNLA's true potential only came into focus upon integration with the fields of numerical analysis and "classical" numerical linear algebra. Through the efforts of many individuals, randomized algorithms have been developed that provide full control over the accuracy of their solutions and that can be every bit as reliable as algorithms that might be found in libraries such as LAPACK. Recent years have even seen the incorporation of certain RandNLA methods into MATLAB, the NAG Library, NVIDIA's cuSOLVER, and SciPy.

For all its success, we believe that RandNLA has yet to realize its full potential. In particular, we believe the scientific community stands to benefit significantly from suitably defined "RandBLAS" and "RandLAPACK" libraries, to serve as standards conceptually analogous to BLAS and LAPACK. This 200-page monograph represents a step toward defining such standards. In it, we cover topics spanning basic sketching, least squares and optimization, low-rank approximation, full matrix decompositions, leverage score sampling, and sketching data with tensor product structures (among others). Much of the provided pseudo-code has been tested via publicly available Matlab and Python implementations.

Comments: v1: this is the first arXiv release of LAPACK Working Note 299

Subjects: **Numerical Analysis (math.NA)**; Mathematical Software (cs.MS); Optimization and Control (math.OC)

Cite as: arXiv:2302.11474 **[math.NA]**

(or arXiv:2302.11474v1 **[math.NA]** for this version)

https://doi.org/10.48550/arXiv.2302.11474

Overview

Motivations:

- WeightWatcher, Weight Diagnostics for Analyzing ML Models
  (with Charles H. Martin)
- Randomized Numerical Linear Algebra for Modern ML
  (with Michal Derezinski)

Some Theory:

- RMT for NNs: Linear to Nonlinear; Shallow to Deep; etc.
  (with Zhenyu Liao)

Applications:

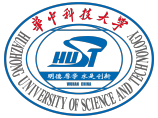
- Models of Heavy-Tailed Mechanistic Universality
  (with Zhichao Wang and Liam Hodgkinson)
- Spectral Estimation with Free Decompression
  (with Siavash Ameli, Chris van der Heide, and Liam Hodgkinson)
- Determinant Estimation under Memory Constraints and Neural Scaling Laws
  (with S. Ameli, C. van der Heide, L. Hodgkinson, and F. Roosta)

A Random Matrix Approach to Neural Networks: From Linear to Nonlinear, and from Shallow to Deep

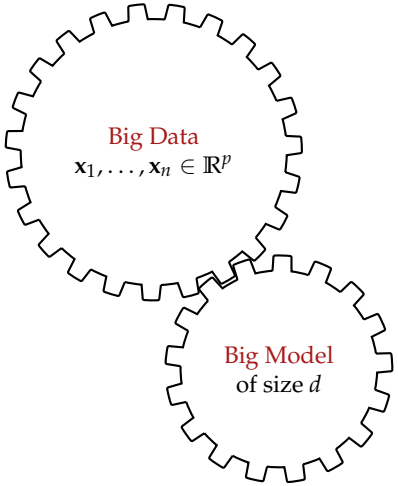
Michael W. Mahoney

joint work with Z. Liao (HUST, China) and R. Couillet (UGA, France)

June 15, 2025



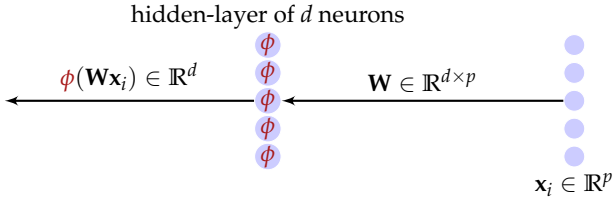
Motivation: understanding large-dimensional machine learning



- **Big Data era**: exploit large n, p, d
- counterintuitive phenomena different from classical asymptotics statistics
- change of understanding of many methods in statistics and machine learning
- **Random Matrix Theory (RMT)** provides the tools!
- In this talk, a review of some recent progress on RMT analysis of **neural networks** models, from linear to nonlinear, and from shallow to deep

Outline

A deep neural network model



- **linear transformation** with first-layer weight matrix $\mathbf{W} \in \mathbb{R}^{d\times p}$
- **nonlinear transformation**: activation function $\phi\colon \mathbb{R} \to \mathbb{R}$ acting entry-wise on $\mathbf{W}\mathbf{x}_i$
- **data representation** at the output of first-layer $\boxed{\mathbf{x}_i \mapsto \phi(\mathbf{W}\mathbf{x}_i)}$
- do the same thing in a layer-by-layer fashion:

$$\frac{1}{\sqrt{d_L}}\mathbf{w}^{\mathsf{T}}\phi_L\left(\frac{1}{\sqrt{d_{L-1}}}\mathbf{W}_L\phi_{L-1}\left(\cdots\frac{1}{\sqrt{d_2}}\phi_2\left(\frac{1}{\sqrt{d_1}}\mathbf{W}_2\phi_1(\mathbf{W}_1\mathbf{x}_i)\right)\right)\right), \tag{1}$$

for a large number n of input data points $\mathbf{x}_1, \ldots, \mathbf{x}_n$

Technical challenges and key ideas

Analyze and Optimize Large-scale ML model $\mathcal{M}_\phi(\mathbf{X};\Theta)$

Objective: Evaluation of $\mathcal{M}_\phi(\mathbf{X};\Theta)$ via Performance Metric $f(\cdot)$

Technical Challenge 1
High-dimensionality in $\mathbf{X}, \Theta$

→

Key Idea 1
Concentration of $f\left(\mathcal{M}_\phi(\mathbf{X};\Theta)\right) \simeq \mathbb{E}[f\left(\mathcal{M}_\phi(\mathbf{X};\Theta)\right)]$

Technical Challenge 2
Analysis of Eigen-functional

→

Key Idea 2
Deterministic Equivalent for Resolvent

Technical Challenge 3
Non-linearity in ML model

→

Key Idea 3
High-dimensional linearization of $\mathcal{M}_\phi(\mathbf{X};\Theta)$

High-dimensional Equivalent

Definition (High-dimensional Equivalent)

Let $\mathcal{M}_\phi(\mathbf{X}) \in \mathbb{R}^{p\times n}$ be a (nonlinear) random matrix model that depends on a random matrix $\mathbf{X} \in \mathbb{R}^{p\times n}$ and function $\phi\colon \mathbb{R} \to \mathbb{R}$ (typically applied entrywise). Let $f\left(\mathcal{M}_\phi(\mathbf{X})\right)$ be a scalar observation of $\mathcal{M}_\phi(\mathbf{X})$ for some $f\colon \mathbb{R}^{p\times n} \to \mathbb{R}$. We say that $\tilde{\mathcal{M}}_\phi(\mathbf{X})$ (random or deterministic) is a High-dimensional Equivalent of $\mathcal{M}_\phi(\mathbf{X})$ with respect to $f(\cdot)$ if

$$f(\mathcal{M}_\phi(\mathbf{X})) - f(\tilde{\mathcal{M}}_\phi(\mathbf{X})) \to 0, \tag{2}$$

in probability or almost surely as $n, p \to \infty$ with $p/n \to c \in (0, \infty)$. We denote this relation as

$$\mathcal{M}_\phi(\mathbf{X}) \overset{f}{\leftrightarrow} \tilde{\mathcal{M}}_\phi(\mathbf{X}) \text{ or simply } \mathcal{M}_\phi(\mathbf{X}) \leftrightarrow \tilde{\mathcal{M}}_\phi(\mathbf{X}), \tag{3}$$

when f is clear from context.

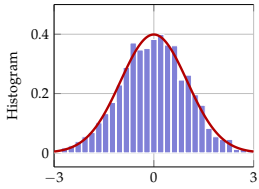
- without (entrywise) nonlinearities, $f(\mathbf{X})$ concentrates around expectation $f(\mathbf{X}) \simeq \mathbb{E}[f(\mathbf{X})]$, and can be assessed through **Deterministic Equivalent** $f(\bar{\mathbf{X}})$;
- for scalar eigenspectral functionals, Deterministic Equivalent for Resolvent framework provides a unified approach to eigenspectral functionals of random matrices;
- for nonlinear models in two different scaling regimes (LLN versus CLT), $\phi(\mathbf{X})$ can be linearized to yield a **Linear Equivalent**.

Concentration versus non-concentration behavior

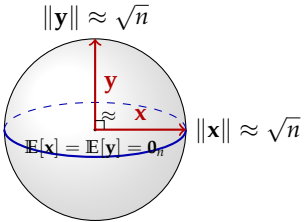
"Concentration" versus "non-concentration" around the mean

Consider two independent random vectors $\mathbf{x} = [x_1, \ldots, x_n]^\top$ and $\mathbf{y} = [y_1, \ldots, y_n]^\top \in \mathbb{R}^n$, with i.i.d. entries of zero mean and unit variance. We have the following observations.

1. In the one-dimensional case with $n = 1$, we have $\Pr(|x - 0| > t) \leq t^{-2}$ and $\Pr(|y - 0| > t) \leq t^{-2}$ by Markov's inequality, so that one-dimensional random variables "concentrate" around their means.
2. In the multi-dimensional case with $n \geq 1$, we have $\mathbb{E}[\|\mathbf{x} - \mathbf{0}\|_2^2] = \mathbb{E}[\mathbf{x}^\top \mathbf{x}] = \mathrm{tr}(\mathbb{E}[\mathbf{x}\mathbf{x}^\top]) = n$ and $\mathbb{E}[\|\mathbf{x} - \mathbf{y}\|_2^2] = \mathbb{E}[\mathbf{x}^\top \mathbf{x} + \mathbf{y}^\top \mathbf{y}] = 2n$. Thus, for $n \gg 1$, the expected Euclidean distance between $\mathbf{x}$ and its mean $\mathbf{0}$ is large: high-dimensional random vectors do not "concentrate" around their means.



(a) "Concentration" around the mean



(b) "Non-concentration" around the mean

High-dimensional concentration of scalar observation

- while large random vectors do not "concentrate" round their means, their scalar functionals (often) do
- for a scalar observation map $f\colon \mathbb{R}^n \to \mathbb{R}$ and random vector $\mathbf{x} \in \mathbb{R}^n$, we typically have

$$f(\mathbf{x}) - \mathbb{E}[f(\mathbf{x})] \to 0, \tag{4}$$

  with high probability for n large.
- a basic example is the linear function $f(\mathbf{x}) = \mathbf{1}_n^\top \mathbf{x}/n = \frac{1}{n}\sum_{i=1}^n x_i$: By the Large of Large Numbers (LLN) and the Central Limit Theorem (CLT), we have $f(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})] + O(n^{-1/2})$ with high probability
- For a random matrix $\mathbf{X} \in \mathbb{R}^{p\times n}$ in the proportional regime with n, p both large, similar holds:

1. just as for vectors, $\mathbf{X}$ does not concentrate, e.g., in a spectral norm sense; e.g., $\|\mathbf{X} - \mathbb{E}[\mathbf{X}]\| \not\to 0$ as $n, p \to \infty$.
2. at the same time, scalar (e.g., eigenspectral) functionals $f\colon \mathbb{R}^{p\times n} \to \mathbb{R}$ of the random matrix $\mathbf{X}$ do concentrate; i.e., $f(\mathbf{X}) - \mathbb{E}[f(\mathbf{X})] \to 0$ as $n, p \to \infty$. This is the key idea of Deterministic Equivalent.

Definition (Deterministic Equivalent)

A Deterministic Equivalent is a special case of the High-Dimensional Equivalent, applied to a linear model $\mathcal{M}_\phi(\mathbf{X}) = \mathbf{X}$. We denote

$$f(\mathbf{X}) - f(\tilde{\mathbf{X}}) \to 0 \text{ as } n, p \to \infty \quad \Leftrightarrow \quad \mathbf{X} \overset{f}{\leftrightarrow} \tilde{\mathbf{X}} \text{ or simply } \mathbf{X} \leftrightarrow \tilde{\mathbf{X}}. \tag{5}$$

Nonlinear objects in two different scaling regimes

Definition (Two scaling regimes)

Consider a scalar functional $f(\mathbf{x})$ of $\mathbf{x} \in \mathbb{R}^n$, via an observation map $f: \mathbb{R}^n \to \mathbb{R}$:

1. **LLN regime**: this holds when $f(\mathbf{x})$ exhibits a LLN-type concentration, strongly concentrating around its mean $\mathbb{E}[f(\mathbf{x})]$, and its distribution function becomes degenerate; that is, it holds when $f(\mathbf{x}) - \mathbb{E}[f(\mathbf{x})] \to 0$ in probability or almost surely, as $n \to \infty$.
2. **CLT regime**: this holds when $f(\mathbf{x})$ exhibits a CLT-type concentration, remaining random and maintaining a non-degenerate distribution function; that is, it holds when $\sqrt{n}\,(f(\mathbf{x}) - \mathbb{E}[f(\mathbf{x})]) \to \mathcal{N}(0,1)$ in distribution, as $n \to \infty$.

Nonlinear objects in two scaling regimes

Let $\mathbf{x} \in \mathbb{R}^n$ be a random vector such that $\sqrt{n}\mathbf{x}$ has i.i.d. Gaussian entries $\mathcal{N}(0,1)$ (the $\sqrt{n}$ scaling ensures $\mathbb{E}[\|\mathbf{x}\|^2] = 1$). Let $\mathbf{y} \in \mathbb{R}^n$ be a deterministic vector of unit norm $\|\mathbf{y}\| = 1$. Consider two nonlinear objects:

1. **LLN regime**: random variables $f_{\mathrm{LLN}}(\mathbf{x}) = \|\mathbf{x}\|_2^2$ or $f_{\mathrm{LLN}}(\mathbf{x}) = \mathbf{x}^\top \mathbf{y}$ that both exhibit LLN-type concentration (i.e., nearly deterministic for n large), and we are interested in $\phi(f_{\mathrm{LLN}}(\mathbf{x}))$; and
2. **CLT regime**: random variables $f_{\mathrm{CLT}}(\mathbf{x}) = \sqrt{n}(\|\mathbf{x}\|_2^2 - 1)$ or $f_{\mathrm{CLT}}(\mathbf{x}) = \sqrt{n} \cdot \mathbf{x}^\top \mathbf{y}$ that both exhibit CLT-type concentration (they remain inherently random and have non-degenerate distributions for n large), and we are interested in $\phi(f_{\mathrm{CLT}}(\mathbf{x}))$.

Linearization in the two scaling regimes

Theorem (Taylor's theorem)

Let $\phi\colon \mathbb{R} \to \mathbb{R}$ be a function that is at least k times continuously differentiable in a neighborhood of some point $\tau \in \mathbb{R}$. Then, there exists $h_k\colon \mathbb{R} \to \mathbb{R}$ such that

$\phi(x) = \phi(\tau) + \phi'(\tau)(x-\tau) + \frac{\phi''(\tau)}{2}(x-\tau)^2 + \ldots + \frac{\phi^{(k)}(\tau)}{k!}(x-\tau)^k + h_k(x)(x-\tau)^k$, *with* $\lim_{x\to\tau} h_k(x) = 0$.
Consequently, $h_k(x)(x-\tau)^k = o(|x-\tau|^k)$ *as* $x \to \tau$.

Theorem (Hermite polynomial expansion)

The i^{th} normalized Hermite polynomial, $\mathrm{He}_i(t)$, *is given by* $\mathrm{He}_0(t) = 1, \mathrm{He}_i(t) = \frac{(-1)^i}{\sqrt{i!}} e^{\frac{t^2}{2}} \frac{d^i}{dt^i}\left(e^{-\frac{t^2}{2}}\right), i \geq 1$. *The normalized Hermite polynomials*

1. *are orthogonal with respect to Gaussian measure, i.e.,* $\int \mathrm{He}_m(t)\mathrm{He}_n(t)\mu(dt) = \delta_{mn}$ *for* $\mu(dt) = \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$; *and*
2. *can be used to formally expand any square-integrable function* $\phi \in L^2(\mu)$ *as* $\phi(\xi) \sim \sum_{i=0}^{\infty} a_{\phi;i}\mathrm{He}_i(\xi)$, $\quad a_{\phi;i} = \int \phi(t)\mathrm{He}_i(t)\mu(dt) = \mathbb{E}[\phi(\xi)\mathrm{He}_i(\xi)]$, *for* $\xi \sim \mathcal{N}(0,1)$. *The coefficients* $a_{\phi;i}$s *are the Hermite coefficients of* ϕ:

$$a_{\phi;0} = \mathbb{E}[\phi(\xi)],\ a_{\phi;1} = \mathbb{E}[\xi\phi(\xi)],\ \sqrt{2}a_{\phi;2} = \mathbb{E}[\xi^2\phi(\xi)] - a_{\phi;0},\ \nu_\phi = \mathbb{E}[\phi^2(\xi)] = \sum_{i=0} a_{\phi;i}^2. \tag{6}$$

Linearization in the two scaling regimes: an example

Example (Distinct linearizations of tanh in two scaling regimes)

Consider $\phi(t) = \tanh(t)$. By Taylor and Hermite polynomial expansion, this nonlinear function is "close" to different quadratic functions, depending on the scaling regime.
Consider $\mathbf{x} \in \mathbb{R}^n$ be a random vector such that $\sqrt{n}\mathbf{x}$ has i.i.d. standard Gaussian entries, and let $\mathbf{y} \in \mathbb{R}^n$ be a deterministic vector of unit norm ($\|\mathbf{y}\| = 1$). Then:

1. **In the LLN regime**, we have for $f_{\text{LLN}}(\mathbf{x}) = \mathbf{x}^\top \mathbf{y}$ that

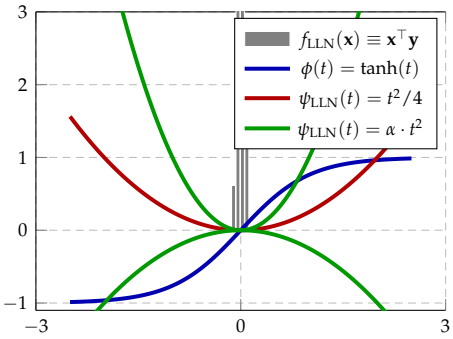
$$\tanh(f_{\text{LLN}}(\mathbf{x})) - \psi_{\text{LLN}}(f_{\text{LLN}}(\mathbf{x})) \to 0, \tag{7}$$

   as $n \to \infty$, with $\psi_{\text{LLN}}(t) = t^2/4$. This is as a consequence of $\tanh(t = 0) = \psi_{\text{LLN}}(t = 0) = 0$. In particular, we also have $\mathbb{E}[\tanh(f_{\text{LLN}}(\mathbf{x}))] \simeq \mathbb{E}[\psi(f_{\text{LLN}}(\mathbf{x}))]$ as a result.

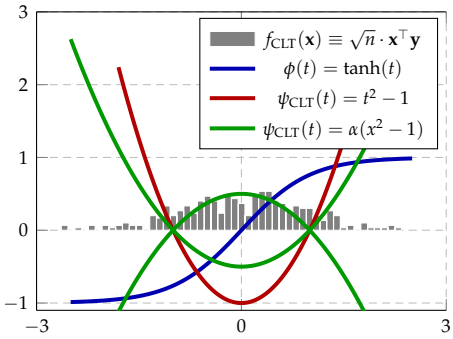
2. **In the CLT regime**, we have for $f_{\text{CLT}}(\mathbf{x}) = \sqrt{n} \cdot \mathbf{x}^\top \mathbf{y}$ that

$$\mathbb{E}[\tanh(f_{\text{CLT}}(\mathbf{x}))] = \mathbb{E}[\psi_{\text{CLT}}(f_{\text{CLT}}(\mathbf{x}))], \tag{8}$$

   in expectation, where the corresponding quadratic function is $\psi_{\text{CLT}}(t) = t^2 - 1$. This follows from the fact that both functions have the same zeroth-order Hermite coefficient, $a_{\tanh;0} = a_{\psi;0} = 0$.



(a) LLN regime



(b) CLT regime

Figure: Different behavior of nonlinear $\phi(f_{\mathrm{LLN}}(\mathbf{x}))$ and $\phi(f_{\mathrm{CLT}}(\mathbf{x}))$ for $\phi(t) = \tanh(t)$ (in **blue**) in the LLN and CLT regime, with $n = 500$. We have $\phi(f_{\mathrm{LLN}}(\mathbf{x})) \simeq \psi_{\mathrm{LLN}}(f_{\mathrm{LLN}}(\mathbf{x}))$ in the LLN regime (as a consequence of $\phi(0) = \psi_{\mathrm{LLN}}(0) = 0$) and $\mathbb{E}[\phi(f_{\mathrm{CLT}}(\mathbf{x}))] = \mathbb{E}[\psi_{\mathrm{CLT}}(f_{\mathrm{CLT}}(\mathbf{x}))]$ in the CLT regime (as a consequence of $a_{\phi;0} = a_{\psi_{\mathrm{CLT}};0} = 0$), with different quadratic functions $\psi_{\mathrm{LLN}}(t) = t^2/4$ and $\psi_{\mathrm{CLT}}(t) = t^2 - 1 = \sqrt{2}\mathrm{He}_2(t)$ in **red**. Note that the these linearizations (in the two different regimes respectively) are not unique and all functions in dashed **green** are also valid linearizations.

Four ways to characterize sample covariance matrices

Definition (Sample Covariance Matrix, SCM)

The SCM $\hat{\mathbf{C}} \in \mathbb{R}^{p \times p}$ of data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p \times n}$ composed of n independent data samples $\mathbf{x}_i \in \mathbb{R}^p$ of zero mean is given by

$$\hat{\mathbf{C}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^{\mathsf{T}} = \frac{1}{n} \mathbf{X}\mathbf{X}^{\mathsf{T}}. \tag{9}$$

Definition (Classical versus proportional regimes)

For SCM $\hat{\mathbf{C}} \in \mathbb{R}^{p \times p}$ from n samples of dimension p, consider the following two regimes.

1. **Classical regime** with $n \gg p$, this includes both asymptotic ($n \to \infty$ with p fixed) and non-asymptotic characterizations ($n \gg p$ for large but finite n).
2. **Proportional regime** with $n \sim p$, this includes both asymptotic ($n, p \to \infty$ with $p/n \to c \in (0, \infty)$, also known as thermodynamic limit in the statistical physics literature) and non-asymptotic characterizations ($n \sim p \gg 1$ both large but finite).

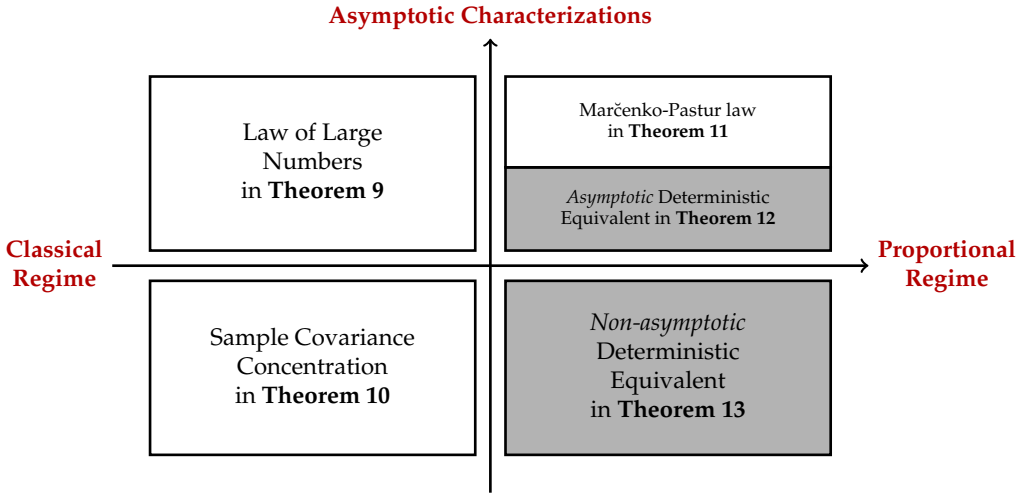


Figure: Taxonomy of four different ways to characterize the sample covariance matrix $\hat{\mathbf{C}} = \frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$.

Asymptotic behavior of SCM in the classical regime via law of large numbers

Theorem (Asymptotic Law of Large Numbers for SCM)

Let p be fixed, and let $\mathbf{X} \in \mathbb{R}^{p \times n}$ be a random matrix with independent sub-gaussian columns $\mathbf{x}_i \in \mathbb{R}^p$ such that $\mathbb{E}[\mathbf{x}_i] = \mathbf{0}$ and $\mathbb{E}[\mathbf{x}_i \mathbf{x}_i^{\mathsf{T}}] = \mathbf{I}_p$. Then one has,

$$\|\hat{\mathbf{C}} - \mathbf{I}_p\|_2 \to 0, \tag{10}$$

almost surely, as $n \to \infty$.

- LLN is "parameterized" to hold only in the **classical limit**, not the **proportional limit**
- many variants and extensions of the LLN exist, but become vacuous when applied to the **proportional regime** $n, p \to \infty$ and $p/n \to c \in (0, \infty)$, see below for an example

Non-asymptotic behavior of SCM in the classical regime via matrix concentration

Theorem (Non-asymptotic matrix concentration for SCM, [Ver18, Theorem 4.6.1])

Let $\mathbf{X} \in \mathbb{R}^{p\times n}$ *be a random matrix with independent sub-gaussian columns* $\mathbf{x}_i \in \mathbb{R}^p$ *such that* $\mathbb{E}[\mathbf{x}_i] = \mathbf{0}$ *and* $\mathbb{E}[\mathbf{x}_i\mathbf{x}_i^\mathsf{T}] = \mathbf{I}_p$. *Then, one has, with probability at least* $1 - 2\exp(-t^2)$, *for any* $t \geq 0$, *that*

$$\|\hat{\mathbf{C}} - \mathbf{I}_p\|_2 \leq C_1 \max(\delta, \delta^2), \quad \delta = C_2(\sqrt{p/n} + t/\sqrt{n}), \tag{11}$$

for some constants $C_1, C_2 > 0$, *independent of* n, p.

Proof: combines Bernstein's concentration inequality with ϵ-net argument, see [Ver18] for details.

1. can reproduce the LLN asymptotic result by taking $n \to \infty$ with Borel–Cantelli lemma
2. **Classical regime.** Here, $n \gg p$, **say that** $n \sim p^2$. Then with high probability, that $\|\hat{\mathbf{C}} - \mathbf{I}_p\|_2 = O(n^{-1/4})$ and conveys a similar intuition to the asymptotic LLN result
3. **Proportional regime.** Here, n, p **are both large and** $n \sim p$. Then, with high probability, that $\|\hat{\mathbf{C}} - \mathbf{I}_p\|_2 = O(\sqrt{p/n}) = O(1)$, and qualitatively different LLN with a vacuous $\sim 100\%$ relative error, e.g., as $n, p \to \infty$ with $p/n \to c \in (0, \infty)$.

Proportional regime: eigenvalues via traditional RMT and the Marčenko-Pastur law

Theorem (Limiting spectral distribution for SCM: Marčenko-Pastur law, [MP67])

Let $\mathbf{X} \in \mathbb{R}^{p\times n}$ be a random matrix with i.i.d. sub-gaussian columns $\mathbf{x}_i \in \mathbb{R}^p$ such that $\mathbb{E}[\mathbf{x}_i] = \mathbf{0}$ and $\mathbb{E}[\mathbf{x}_i\mathbf{x}_i^\mathsf{T}] = \mathbf{I}_p$. Then, as $n, p \to \infty$ with $p/n \to c \in (0,\infty)$, with probability one, the empirical spectral measure (ESD) $\mu_{\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}}$ of $\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$ converges weakly to a probability measure μ given explicitly by

$$\boxed{\mu(dx) = (1-c^{-1})^+\delta_0(x) + \frac{1}{2\pi c x}\sqrt{(x-E_-)^+(E_+-x)^+}\,dx,} \tag{12}$$

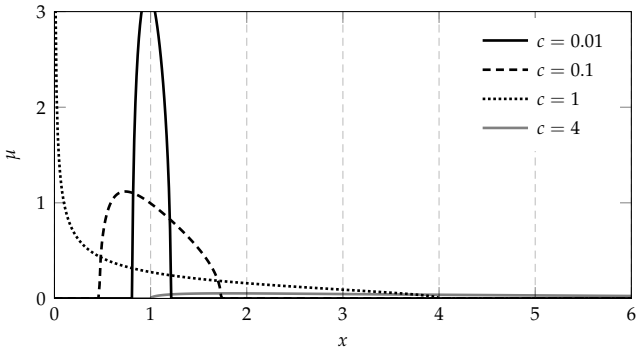
where $E_\pm = (1\pm\sqrt{c})^2$ and $(x)^+ = \max(0,x)$, which is known as the Marčenko-Pastur distribution.

- provides a more refined characterization of the eigenspectrum of $\hat{\mathbf{C}}$ (than, e.g., matrix concentration):

(i) **Classical regime.** Here, $n \gg p$ **so that** $c = p/n \to 0$, the Marčenko-Pastur law in Equation (12) shrinks to a Dirac mass, in agreement with $\|\hat{\mathbf{C}} - \mathbf{I}_p\|_2 \sim 0$

(ii) **Proportional regime.** Here, $n \sim p \gg 1$, and by the (true but vacuous) matrix concentration result $\|\hat{\mathbf{C}} - \mathbf{I}_p\|_2 = O(p/n) = O(1)$, and, depending on the ratio $c = p/n$, the eigenvalues of $\hat{\mathbf{C}}$ can be very different from one, and takes the form of the Marčenko-Pastur law

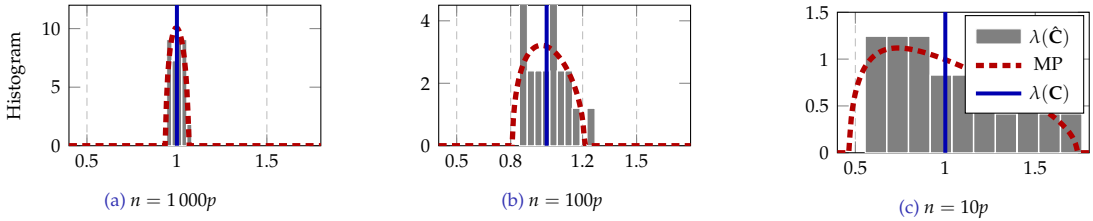
- we have in fact $\|\hat{\mathbf{C}} - \mathbf{I}_p\|_2 \simeq c + 2\sqrt{c}$ as $n, p \to \infty$ with $p/n \to c \in (0,\infty)$



- averaged amount of eigenvalues of $\hat{\mathbf{C}}$ lying within the interval $[1-\delta, 1+\delta]$, for $\delta \ll 1$, as

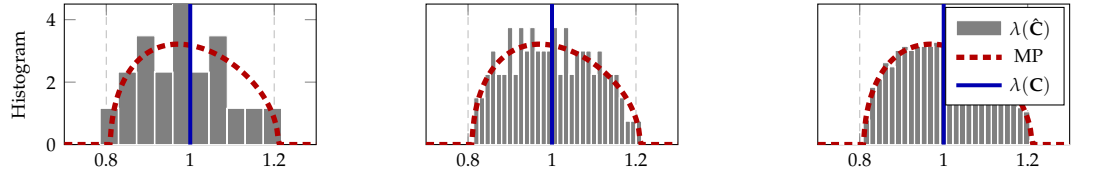
$$\mu([1-\delta,1+\delta]) = \int_{1-\delta}^{1+\delta} \frac{1}{2\pi c x} \sqrt{\left(x-(1-\sqrt{c})^2\right)^+ \left((1+\sqrt{c})^2 - x\right)^+}\, dx$$
$$= \frac{1}{2\pi c} \int_{-\delta}^{\delta} \left(\sqrt{4c-c^2} + O(\varepsilon)\right) d\varepsilon = \frac{\sqrt{4c^{-1}-1}}{\pi}\delta + O(\delta^2).$$

- for $p \approx 4n$ there is asymptotically no eigenvalue of $\hat{\mathbf{C}}$ close to one!
- in accordance with the shape of the limiting Marčenko-Pastur law with $c = 4$ above



(a) $n = 1\,000p$ (b) $n = 100p$ (c) $n = 10p$

Figure: **Varying n and $c = p/n$ for fixed p.** Histogram of the eigenvalues of $\hat{\mathbf{C}}$ versus the limiting Marčenko-Pastur law in Theorem 11, for $\mathbf{X}$ having standard Gaussian entries with $p = 20$ and different $n = 1\,000p, 100p, 10p$ from left to right.



(a) $p = 20$ (b) $p = 100$ (c) $p = 500$

Figure: **Varying n and p for fixed $c = p/n$.** Histogram of the eigenvalues of $\hat{\mathbf{C}}$ versus the Marčenko-Pastur law, for $\mathbf{X}$ having standard Gaussian entries with $n = 100p$ and different $p = 20, 100, 500$ from left to right.

An asymptotic Deterministic Equivalent for resolvent

Theorem (An asymptotic Deterministic Equivalent for resolvent, [CL22, Theorem 2.4])

Let $\mathbf{X} \in \mathbb{R}^{p\times n}$ *be a random matrix having i.i.d. sub-gaussian entries of zero mean and unit variance, and denote* $\mathbf{Q}(z) = (\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T} - z\mathbf{I}_p)^{-1}$ *the resolvent of* $\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$ *for* $z \in \mathbb{C}$ *not an eigenvalue of* $\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$*. Then, as* $n, p \to \infty$ *with* $p/n \to c \in (0, \infty)$*, the deterministic matrix* $\bar{\mathbf{Q}}(z)$ *is a Deterministic Equivalent of the random resolvent matrix* $\mathbf{Q}(z)$ *with*

$$\mathbf{Q}(z) \leftrightarrow \bar{\mathbf{Q}}(z), \quad \bar{\mathbf{Q}}(z) = m(z)\mathbf{I}_p, \tag{13}$$

with $m(z)$ *the unique valid Stieltjes transform as solution to*

$$czm^2(z) - (1 - c - z)m(z) + 1 = 0. \tag{14}$$

- The equation of $m(z)$ is quadratic and has two solutions defined via the complex square root
- **only one** satisfies $\Im[z] \cdot \Im[m(z)] > 0$ as a "valid" Stieltjes transform, and leads to the Marčenko-Pastur law

$$\boxed{\mu(dx) = (1 - c^{-1})^+\delta_0(x) + \frac{1}{2\pi cx}\sqrt{(x - E_-)^+ (E_+ - x)^+}\, dx,} \tag{15}$$

for $E_\pm = (1 \pm \sqrt{c})^2$ and $(x)^+ = \max(0, x)$.

A non-asymptotic Deterministic Equivalent for resolvent

Theorem (A non-asymptotic Deterministic Equivalent for resolvent)

Let $\mathbf{X} \in \mathbb{R}^{p\times n}$ be a random matrix having i.i.d. sub-gaussian entries with zero mean and unit variance, and denote $\mathbf{Q}(z) = (\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T} - z\mathbf{I}_p)^{-1}$ the resolvent of $\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$ for $z < 0$. Then, there exists universal constants $C_1, C_2 > 0$ depending only on the sub-gaussian norm of the entries of $\mathbf{X}$ and $|z|$, such that for any $\varepsilon \in (0,1)$, if $n \geq (C_1+\varepsilon)p$, one has

$$\|\mathbb{E}[\mathbf{Q}(z)] - \bar{\mathbf{Q}}(z)\|_2 \leq \frac{C_2}{\varepsilon}\cdot n^{-\frac{1}{2}}, \quad \bar{\mathbf{Q}}(z) = m(z)\mathbf{I}_p, \tag{16}$$

for $m(z)$ the unique positive solution to the Marčenko-Pastur equation $czm^2(z) - (1-c-z)m(z) + 1 = 0, c = p/n$.

- this is a deterministic characterization of the **expected resolvent**
- to get DE, it remains to show **concentration** results for trace and bilinear forms: more or less standard

Remark: as extensions to results in the classical regime

(i) In the "easy" **classical regime**, with $n \gg p$ (and thus $p/n \to c = 0$), one has that $\hat{\mathbf{C}} \equiv \frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T} \to \mathbb{E}[\hat{\mathbf{C}}] = \mathbf{I}_p$ as $n \to \infty$, so that

$$(\hat{\mathbf{C}} - z\mathbf{I}_p)^{-1} \simeq (\mathbb{E}[\hat{\mathbf{C}}] - z\mathbf{I}_p)^{-1} = (1-z)^{-1}\mathbf{I}_p = \bar{\mathbf{Q}}(z). \tag{17}$$

(ii) In the "harder" and more general **proportional regime**, for $n \sim p$ with $p/n \to c \in (0, \infty)$, one has instead

$$\bar{\mathbf{Q}}(z) \simeq \mathbb{E}[\mathbf{Q}(z)] \equiv \mathbb{E}[(\hat{\mathbf{C}} - z\mathbf{I}_p)^{-1}] \not\simeq (\mathbb{E}[\hat{\mathbf{C}}] - z\mathbf{I}_p)^{-1}. \tag{18}$$

In this case, a Deterministic Equivalent $\bar{\mathbf{Q}}(z)$ can be very different from $(\mathbb{E}[\hat{\mathbf{C}}] - z\mathbf{I}_p)^{-1}$.

- this is not surprising, consider the scalar case where $\mathbb{E}[1/x] \neq 1/\mathbb{E}[x]$ in general, unless $x \simeq C$ for some constant C

Remark: Deterministic Equivalents for Gaussian inverse SCM

- consider the sample covariance matrix $\hat{\mathbf{C}} = \frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$ for $\mathbf{X} = \mathbf{C}^{\frac{1}{2}}\mathbf{Z}$ and positive definite $\mathbf{C} \in \mathbb{R}^{p\times p}$ and $\mathbf{Z} \in \mathbb{R}^{p\times n}$ having i.i.d. standard Gaussian entries
- the inverse $\hat{\mathbf{C}}^{-1}$ is known to follow the inverse-Wishart distribution [MKB79] with p degrees of freedom and scale matrix $\mathbf{C}^{-1}$, such that

$$\mathbb{E}[\hat{\mathbf{C}}^{-1}] = \frac{n}{n-p-1}\mathbf{C}^{-1} \tag{19}$$

  for $n \geq p+2$.
- On the other hand, it follows from our non-asymptotic result above by taking $z = 0$ that

$$\mathbb{E}[\mathbf{Q}(z)] \leftrightarrow \bar{\mathbf{Q}}(z) = m(z)\mathbf{I}_p = \frac{n}{n-p}\mathbf{I}_p \tag{20}$$

  with $m(z) = \frac{1}{1-c} = \frac{n}{n-p}$.
- **note**: Deterministic Equivalents **are not unique**: could replace the "-1" in denominator by any constant $C' \ll n, p$ to propose another equally correct Deterministic Equivalent.

[1] Kanti Mardia, J. Kent, and J. Bibby. *Multivariate Analysis*. 1st ed. Probability and Mathematical Statistics. Academic Press, Dec. 1979

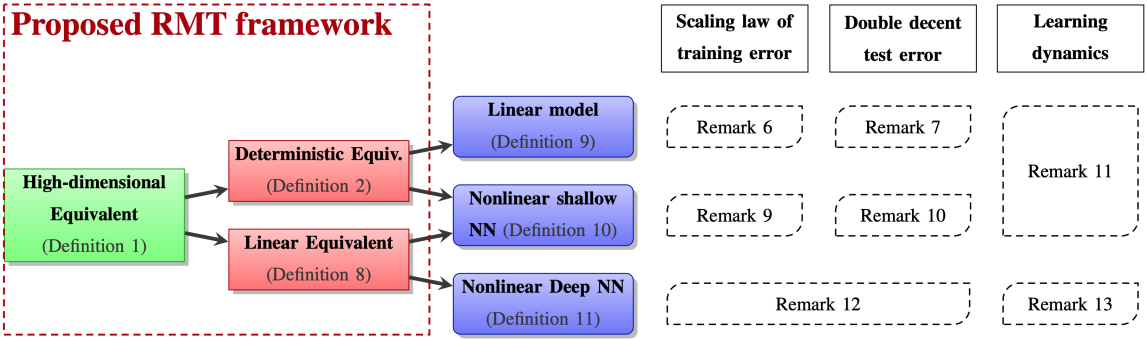
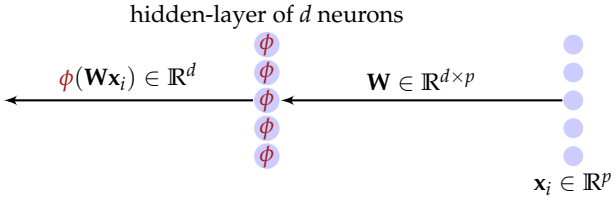


Figure: Overview of [LM25], summarizing major concepts and results and where to find them.

Two-layer network with random first layer



Definition (Single-hidden-layer NN model)

Consider a single-hidden-layer NN model with first-layer weights $\mathbf{W} \in \mathbb{R}^{d\times p}$ and second-layer weights $\boldsymbol{\beta} \in \mathbb{R}^d$. For an input vector $\mathbf{x} \in \mathbb{R}^p$, the network output is given by $\hat{y}(\mathbf{x}) = \boldsymbol{\beta}^\top \phi(\mathbf{W}\mathbf{x})$, where $\phi(\cdot)$ is an entrywise activation function. We are interested in the NN performance measured by

1. its **training MSE** $E_{\text{train}} = \frac{1}{n}\sum_{i=1}^n (y_i - \hat{y}(\mathbf{x}_i))^2 = \frac{1}{n}\|\mathbf{y} - \boldsymbol{\Phi}^\top\boldsymbol{\beta}\|^2$ with $\boldsymbol{\Phi} \equiv \phi(\mathbf{W}\mathbf{X})$ for a training set $(\mathbf{X}, \mathbf{y})$ of size n, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p\times n}$, $\mathbf{y} = [y_1, \ldots, y_n]^\top \in \mathbb{R}^n$; and
2. its **test MSE** $E_{\text{test}} = \frac{1}{n}\sum_{i=1}^{n'} (y'_i - \hat{y}(\mathbf{x}'_i))^2 = \frac{1}{n'}\|\mathbf{y}' - \phi(\mathbf{W}\mathbf{X}')^\top\boldsymbol{\beta}\|^2$ on a test set $(\mathbf{X}', \mathbf{y}')$ of size n', with $\mathbf{X}' = [\mathbf{x}'_1, \ldots, \mathbf{x}'_{n'}] \in \mathbb{R}^{p\times n'}$ and $\mathbf{y}' = [y'_1, \ldots, y'_{n'}]^\top \in \mathbb{R}^{n'}$.

Single-hidden-layer NN model and a Deterministic Equivalent for nonlinear resolvent

- Given first-layer $\mathbf{W}$ and training data $\mathbf{X} \in \mathbb{R}^{p\times n}$, consider the random feature matrix $\mathbf{\Phi} \equiv \phi(\mathbf{WX}) \in \mathbb{R}^{d\times n}$ and regress against the target $\mathbf{y}$ by minimizing the following ridge-regularized MSE

$$L(\boldsymbol{\beta}) = \frac{1}{2n}\sum_{i=1}^{n}(y_i - \hat{y}(\mathbf{x}_i))^2 + \frac{\gamma}{2}\|\boldsymbol{\beta}\|_2^2 = \frac{1}{2n}\|\mathbf{y} - \mathbf{\Phi}^\top\boldsymbol{\beta}\|_2^2 + \frac{\gamma}{2}\|\boldsymbol{\beta}\|_2^2, \quad \gamma \geq 0, \tag{21}$$

- solution is uniquely given by $\boldsymbol{\beta}_\gamma = \frac{1}{n}\mathbf{\Phi}\left(\frac{1}{n}\mathbf{\Phi}^\top\mathbf{\Phi} + \gamma\mathbf{I}_n\right)^{-1}\mathbf{y} = \left(\frac{1}{n}\mathbf{\Phi}\mathbf{\Phi}^\top + \gamma\mathbf{I}_d\right)^{-1}\frac{1}{n}\mathbf{\Phi}\mathbf{y}$, for $\gamma > 0$.
- Training MSE is $E_{\text{train}} = \frac{1}{n}\|\mathbf{y} - \mathbf{\Phi}^\top\boldsymbol{\beta}_\gamma\|_2^2 = \frac{\gamma^2}{n}\frac{\partial \mathbf{y}^\top\mathbf{Q}^2(-\gamma)\mathbf{y}}{\partial\gamma}$, with **resolvent** of nonlinear Gram $\mathbf{\Phi}^\top\mathbf{\Phi}$.

$$\mathbf{Q}(-\gamma) \equiv \left(\frac{1}{n}\mathbf{\Phi}^\top\mathbf{\Phi} + \gamma\mathbf{I}_n\right)^{-1}, \quad \mathbf{\Phi}^\top\mathbf{\Phi} = \phi(\mathbf{X}^\top\mathbf{W}^\top)\phi(\mathbf{WX}). \tag{22}$$

Theorem (Deterministic Equivalent for nonlinear resolvent, [LLC18, Theorem 1])

Let $\mathbf{W} \in \mathbb{R}^{d\times p}$ be a random matrix with i.i.d. sub-gaussian entries of zero mean and unit variance, and let $\mathbf{X} \in \mathbb{R}^{p\times n}$ be independent of $\mathbf{W}$ with $\|\mathbf{X}\|_2 \leq 1$. Then, as $n, p, d \to \infty$ together and for Lipschitz $\phi\colon \mathbb{R} \to \mathbb{R}$,

$$\mathbf{Q}(z) \leftrightarrow \bar{\mathbf{Q}}(z), \quad \bar{\mathbf{Q}}(z) = \left(\frac{d}{n}\frac{\mathbf{K}}{1+\delta(z)} - z\mathbf{I}_n\right)^{-1}, \quad \delta(z) = \frac{1}{n}\operatorname{tr}\mathbf{K}\bar{\mathbf{Q}}(z), \quad \mathbf{K} \equiv \mathbb{E}_{\mathbf{w}}[\phi(\mathbf{X}^\top\mathbf{w})\phi(\mathbf{w}^\top\mathbf{X})], \tag{23}$$

where $\delta(z)$ is the unique Stieltjes transform solution, and $\mathbf{K}$ the kernel matrix.

Implications of the Deterministic Equivalent

Scaling law of training MSE

Consider the ridgeless setting with $\gamma = 0$ and the under-parameterized regime with n, p, d all large but $d < n$

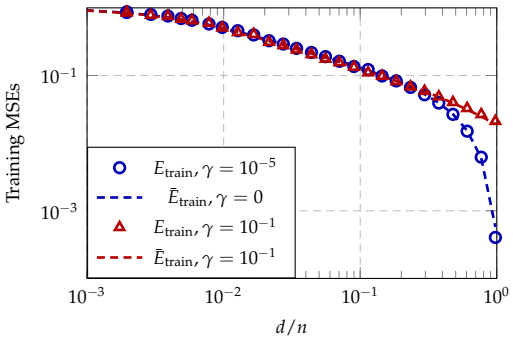
- δ diverges as $\gamma \to 0$, however, $\gamma\delta = \frac{1}{n} \operatorname{tr} \mathbf{K} \left(\frac{d}{n} \frac{\mathbf{K}}{\gamma + \gamma\delta} + \mathbf{I}_n \right)^{-1} \xrightarrow{\gamma \to 0} \theta = \frac{1}{n} \operatorname{tr} \mathbf{K} \left(\frac{d}{n} \frac{\mathbf{K}}{\theta} + \mathbf{I}_n \right)^{-1}$
- explicit scaling laws for the training MSEs that depend on the **eigenspectrum** of $\mathbf{K}$

1. **exponential eigendecay** (e.g., RBF kernel related to cosine activation [RW05]) yields an error decay rate of $\log(n)/n$ (which is slightly slower than the n^{-1} rate of linear models);
2. **polynomial decay** (e.g., Matérn kernel associated with to ReLU activation [Gei+20]) yields an error decay rate of $n^{-1-\beta}$ (with $\beta > 0$), which is faster than the linear case.

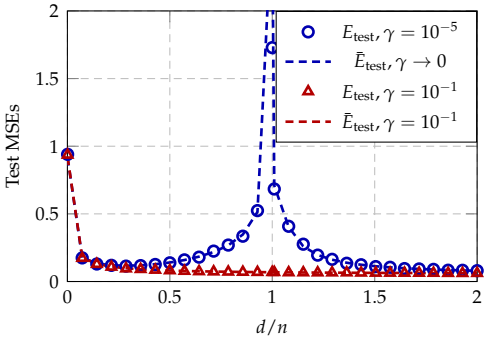
Double descent behavior for test MSE

- it can be checked that both θ and δ diverge as $\gamma \to 0$ at $n/d = 1$.
- thus, the test risk likewise exhibits a singularity at $d/n = 1$.
- mirrors the **double descent** phenomenon for linear models, but applies here to nonlinear NN model, regardless of the activation function or the training/test data.

Numerical results



(a) Training MSE



(b) Test MSE

Figure: Empirical and theoretical training and test MSEs of single-hidden-layer NN model, as a function of d/n, for $\gamma = 10^{-1}$ and $\gamma = 10^{-5}$, with Gaussian $\mathbf{W}$ and ReLU activation $\phi(t) = \max(t, 0)$, $n = 1\,024$ training samples and $n' = 1\,024$ test samples from the MNIST dataset (number 1 and 2).**Figure 7a**: log-log plot of training MSEs averaged over 30 runs. **Figure 7b**: test MSEs averaged over 30 runs on independent test sets of size $\hat{n} = 2\,048$.

Theorem (High-dimensional linearization of kernel matrix)

Let $\mathbf{w} \sim \mathbb{R}^p$ *be standard Gaussian* $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_p)$ *and let* $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ *be independently drawn from the unit sphere* $\mathbb{S}^{p-1} \subset \mathbb{R}^p$*. Then, as* $n, p \to \infty$ *with* $p/n \in (0, \infty)$*, the kernel matrix* $\mathbf{K} = \mathbb{E}_{\mathbf{w}}[\phi(\mathbf{X}^\top \mathbf{w})\phi(\mathbf{w}^\top \mathbf{X})]$ *admits the following* ***Linear Equivalent****:*

$$\mathbf{K} \leftrightarrow \tilde{\mathbf{K}}_\phi, \quad \tilde{\mathbf{K}}_\phi = a_{\phi;0}^2 \mathbf{1}_n \mathbf{1}_n^\top + a_{\phi;1}^2 \mathbf{X}^\top \mathbf{X} + a_{\phi;2}^2 \cdot \frac{1}{p} \mathbf{1}_n \mathbf{1}_n^\top + \left(\nu_\phi - a_{\phi;0}^2 - a_{\phi;1}^2 \right) \mathbf{I}_n, \tag{24}$$

with high probability, up to a spectral norm error $\|\mathbf{K} - \tilde{\mathbf{K}}\|_2 = O(n^{-1/2})$*, where* $a_{\phi;0}, a_{\phi;1}, a_{\phi;2}, \nu_\phi$ *are the Hermite coefficients of* ϕ*.*

- a striking (and perhaps counterintuitive) consequence is that, in the proportional regime with n, p both large and comparable, the eigenvalue distribution of $\mathbf{K}$ becomes independent of the activation function ϕ, up to a scaling and shift
- the eigenspectrum of $\mathbf{K}$ coincides with that of $\mathbf{X}^\top \mathbf{X}$ (which approximates the Marčenko-Pastur law), and depends only on the dimension ratio p/n—provided the data are unstructured and uniformly distributed on the unit sphere.

CK of fully-connected random deep neural networks

- everyone cares **more** about deep neural networks
- with some additional efforts, extension to fully-connected **deep** neural networks of depth L,

$$\frac{1}{\sqrt{d_L}}\mathbf{w}^{\mathsf{T}}\phi_L\left(\frac{1}{\sqrt{d_{L-1}}}\mathbf{W}_L\phi_{L-1}\left(\cdots\frac{1}{\sqrt{d_2}}\phi_2\left(\frac{1}{\sqrt{d_1}}\mathbf{W}_2\phi_1(\mathbf{W}_1\mathbf{x})\right)\right)\right), \tag{25}$$

again for random $\mathbf{W}_1,\ldots,\mathbf{W}_L$ and activations $\phi_1(\cdot),\ldots,\phi_L(\cdot)$.

Theorem (Asymptotic approximation for conjugate kernels, informal)

Under the same condition, define output features of layer $\ell \in \{1,\ldots,L\}$, as

$$\mathbf{\Sigma}_\ell = \frac{1}{\sqrt{d_\ell}}\phi_\ell\left(\frac{1}{\sqrt{d_{\ell-1}}}\mathbf{W}_\ell\phi_{\ell-1}\left(\cdots\frac{1}{\sqrt{d_2}}\phi_2\left(\frac{1}{\sqrt{d_1}}\mathbf{W}_2\phi_1(\mathbf{W}_1\mathbf{X})\right)\right)\right). \tag{26}$$

we have for the Conjugate Kernel $\mathbf{K}_{\mathrm{CK},\ell}$ at layer ℓ defined as

$$\mathbf{K}_{\mathrm{CK},\ell} = \mathbb{E}[\mathbf{\Sigma}_\ell^{\mathsf{T}}\mathbf{\Sigma}_\ell] \in \mathbb{R}^{n\times n}, \tag{27}$$

that $\|\mathbf{K}_{\mathrm{CK},\ell} - \tilde{\mathbf{K}}_{\mathrm{CK},\ell}\| \to 0$, some random matrix $\tilde{\mathbf{K}}_{\mathrm{CK},\ell}$ dependent of data, of activation ϕ_ℓ but only via a few parameters, and independent of the distribution of $\mathbf{W}$, as long as of normalized to have zero mean and unit variance.

Theorem (High-dimensional linearization of CK matrices for DNN)

Consider a DNN as in Equation (26)*, with weights* $\mathbf{W}_\ell \in \mathbb{R}^{d_\ell \times d_{\ell-1}}$ *having i.i.d.* $\mathcal{N}(0, 1/d_{\ell-1})$ *entries for* $\ell = 1, \ldots, L$*. Assume each activation* ϕ_ℓ *has Hermite coefficients satisfying* $a_{\phi_\ell;0} = 0$ *and* $\nu_{\phi_\ell} = 1$*. Let* $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ *be independently drawn from the unit sphere* $\mathbb{S}^{p-1} \subset \mathbb{R}^p$*. Then, as* $n, p \to \infty$ *with* $p/n \in (0, \infty)$*, the CK matrix* $\mathbf{K}_{\mathrm{CK},\ell} = \mathbb{E}[\mathbf{\Phi}_\ell^\top \mathbf{\Phi}_\ell]$ *defined in* (27) *admits the following* ***Linear Equivalent****:*

$$\mathbf{K}_{\mathrm{CK},\ell} \overset{f}{\leftrightarrow} \tilde{\mathbf{K}}_{\phi,\ell}, \quad \tilde{\mathbf{K}}_\phi = \alpha_{\ell,1}^2 \mathbf{X}^\top \mathbf{X} + \alpha_{\ell,2}^2 \cdot \frac{1}{p} \mathbf{1}_n \mathbf{1}_n^\top + \left(1 - \alpha_{\ell,1}^2\right) \mathbf{I}_n, \tag{28}$$

for Lipschitz function $f\colon \mathbb{R}^{n\times n} \to \mathbb{R}$ *of bounded Lipschitz constant with respect to matrix spectral norm, i.e.,* $|f(\mathbf{A}) - f(\mathbf{B})| \le C\|\mathbf{A} - \mathbf{B}\|_2, \forall \mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$ *and some* $C \in (0, \infty)$*, for* $\alpha_{\ell,1}, \alpha_{\ell,2}$ *satisfying*

$$\alpha_{\ell,1} = a_{\phi_\ell;1} \cdot \alpha_{\ell-1,1}, \quad \alpha_{\ell,2} = \sqrt{a_{\phi_\ell;1}^2 \cdot \alpha_{\ell-1,2}^2 + a_{\phi_\ell;2}^2 \cdot \alpha_{\ell-1,1}^2}, \tag{29}$$

where $a_{\phi_\ell;1}, a_{\phi_\ell;2}$ *are the Hermite coefficients of* ϕ_ℓ *at layer* ℓ*.*

Implications

- Comparing the result for DNNs to that for single-hidden-layer NNs, observe a "curse of depth" for random, untrained DNNs.
- Specifically, since $\nu_{\phi_\ell} = \sum_{i=0}^{\infty} a_{\phi_\ell;i}^2 = 1$, we have $\max(a_{\phi_\ell;1}, a_{\phi_\ell;2}) \leq 1$ for each $\ell \in \{1, \ldots, L\}$: both $\alpha_{\ell,1}$ and $\alpha_{\ell,2}$ tend to decrease with growing depth ℓ.
- In particular, if $a_{\phi_\ell;1} < 1, \forall \ell \in \{1, \ldots, L\}$, then in the limit of $L \to \infty$, we obtain a degenerate DNN with $\mathbf{K}_L \to \mathbf{I}_n$. This negative "curse of depth" result arises from:
  1. the unstructured input $\mathbf{x}$s (uniformly distributed on the high-dimensional unit sphere); and
  2. the "normalization" of all activations ($a_{\phi_\ell;0} = 0$ and $\nu_{\phi_\ell} = 1, \forall \ell \in \{1, \ldots, L\}$); and
  3. the random untrained weights.
- In contrast with this, [Gu+22] showed that for structured Gaussian mixture inputs (which contain richer statistical information than the unstructured inputs considered above), deeper (but only infinitely so as considered in [Gu+22]) NNs with appropriately chosen activation functions can more effectively separate the input mixture, thereby outperforming their shallow counterparts.

Fully-connected deep nets: CK, NTK, and beyond

- happy with the study of (limiting) CK for random DNN models
- extension to NTK via intrinsic **connection** between CK and neural tangent kernel (NTK) [JGH18]

$$\mathbf{K}_{\mathrm{NTK},\ell}(\mathbf{X}) = \mathbf{K}_{\mathrm{CK},\ell}(\mathbf{X}) + \mathbf{K}_{\mathrm{NTK},\ell-1}(\mathbf{X}) \circ \mathbf{K}'_{\mathrm{CK},\ell}(\mathbf{X}), \quad \mathbf{K}_{\mathrm{NTK},0}(\mathbf{X}) = \mathbf{K}_{\mathrm{CK},0}(\mathbf{X}) = \mathbf{X}^{\mathsf{T}}\mathbf{X}, \tag{30}$$

  and some additional efforts
- **convergence** and **generalization** theory via NTK [JGH18]: for
  1. sufficiently wide nets
  2. trained with gradient descent of sufficiently small step size
- NTK is **determined** at random initialization and remains unchanged during training, and applies to **explicitly** characterize DNN convergence and generalization properties

[2] Arthur Jacot, Franck Gabriel, and Clément Hongler. "Neural Tangent Kernel: Convergence and Generalization in Neural Networks". In: *Advances in Neural Information Processing Systems*. Vol. 31. NIPS'18. Curran Associates, Inc., 2018, pp. 8571–8580

References

- Zhenyu Liao and Michael W. Mahoney. *Random Matrix Theory for Deep Learning: Beyond Eigenvalues of Linear Models*. 2025. arXiv: 2201.04753 [cs, math]
- Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018
- Romain Couillet and Zhenyu Liao. *Random Matrix Methods for Machine Learning*. https://zhenyu-liao.github.io/book/. Cambridge University Press, 2022
- Lingyu Gu, Yongqi Du, Yuan Zhang, Di Xie, Shiliang Pu, Robert Qiu, and Zhenyu Liao. ""Lossless" Compression of Deep Neural Networks: A High-dimensional Neural Tangent Kernel Approach". In: *Advances in Neural Information Processing Systems*. Vol. 35. Curran Associates, Inc., 2022, pp. 3774–3787 (**Please** refer to the ArXiv version on https://arxiv.org/abs/2403.00258 that fixed typos in Theorems 1 and 2 from the NeurIPS 2022 proceeding version.)

Overview

Motivations:
- WeightWatcher, Weight Diagnostics for Analyzing ML Models
  (with Charles H. Martin)
- Randomized Numerical Linear Algebra for Modern ML
  (with Michal Derezinski)

Some Theory:
- RMT for NNs: Linear to Nonlinear; Shallow to Deep; etc.
  (with Zhenyu Liao)

Applications:
- Models of Heavy-Tailed Mechanistic Universality
  (with Zhichao Wang and Liam Hodgkinson)
- Spectral Estimation with Free Decompression
  (with Siavash Ameli, Chris van der Heide, and Liam Hodgkinson)
- Determinant Estimation under Memory Constraints and Neural Scaling Laws
  (with S. Ameli, C. van der Heide, L. Hodgkinson, and F. Roosta)

Models of Heavy-Tailed Mechanistic Universality

Michael W. Mahoney

ICSI, LBNL, and Department of Statistics at UC Berkeley

Joint work with

Zhichao Wang and Liam Hodgkinson

Outline

Motivation and Introduction

Motivation: Heavy-Tailed Phenomena in Modern Models

- Gradient norms (Simsekli et al., 2019) and loss curves (Hestness et al., 2017; Kaplan et al., 2020; Hoffmann et al., 2022).
- Eigenvalues of Gram matrices in neural nets: data covariance (Sorscher et al., 2022; Zhang et al., 2023), activation (conjugate kernel) (Pillaud-Vivien et al., 2018; Agrawal et al., 2022; Wang et al., 2023), Hessian (Xie et al., 2023), Jacobian (Wang et al., 2023).
- Strong correlation between heavy-tailed trained weight matrices & model performance: Heavy-Tailed Self-Regularization (HT-SR) Theory (Martin and Mahoney, 2021b) and Layer-wise Diagnostics (Zhou et al., 2023; Lu et al., 2024).
- Power law appears in neural scaling laws (Kaplan et al., 2020; Wei et al., 2022; Defilippis et al., 2024; Paquette et al., 2024; Lin et al., 2024).

Need new RMT for **Heavy-Tailed Mechanistic Universality (HT-MU)**.

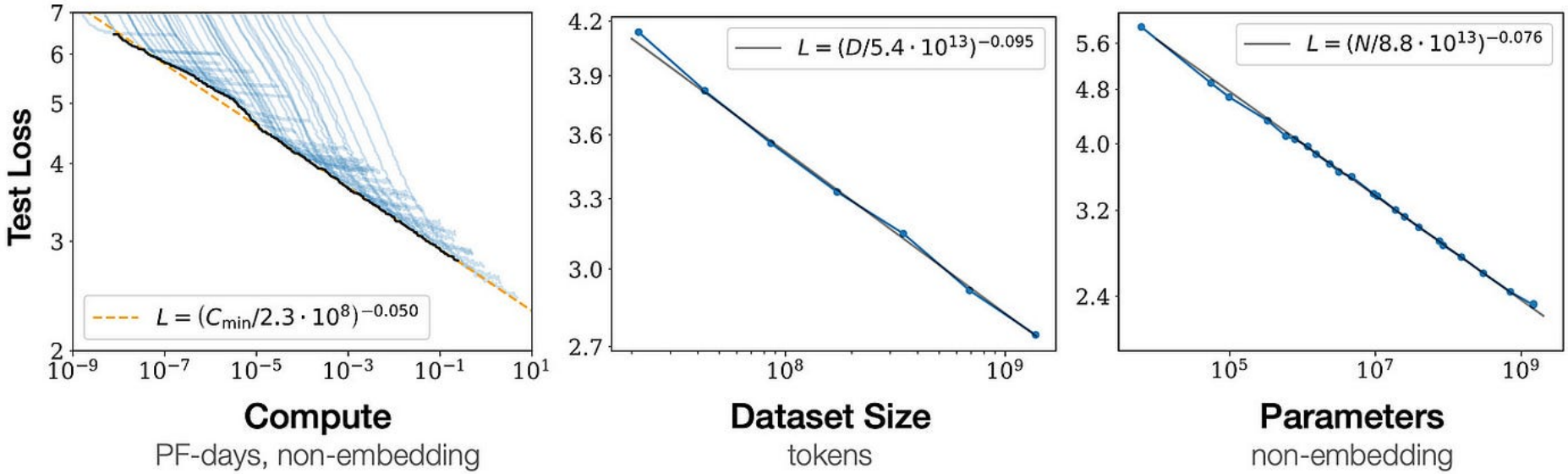
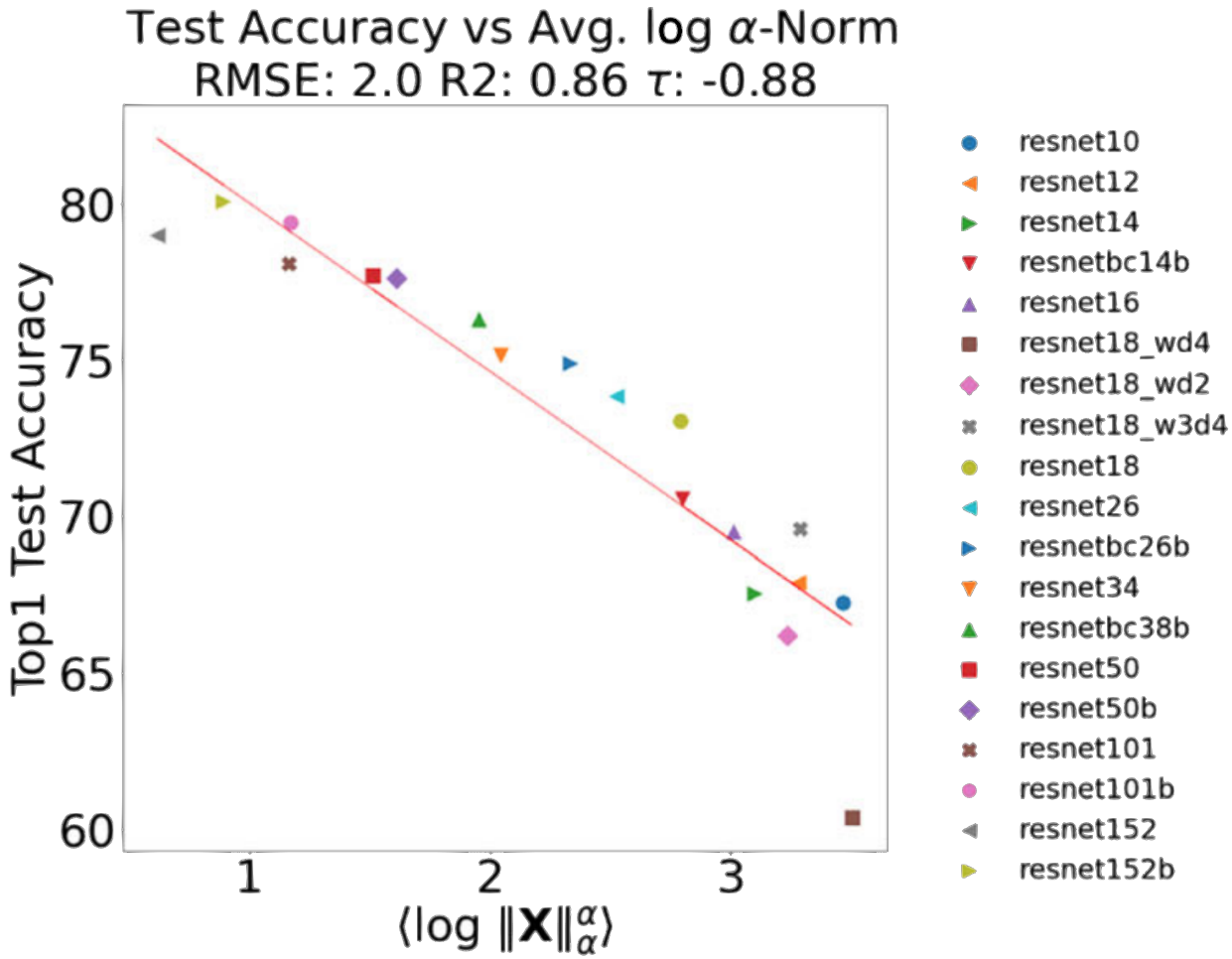


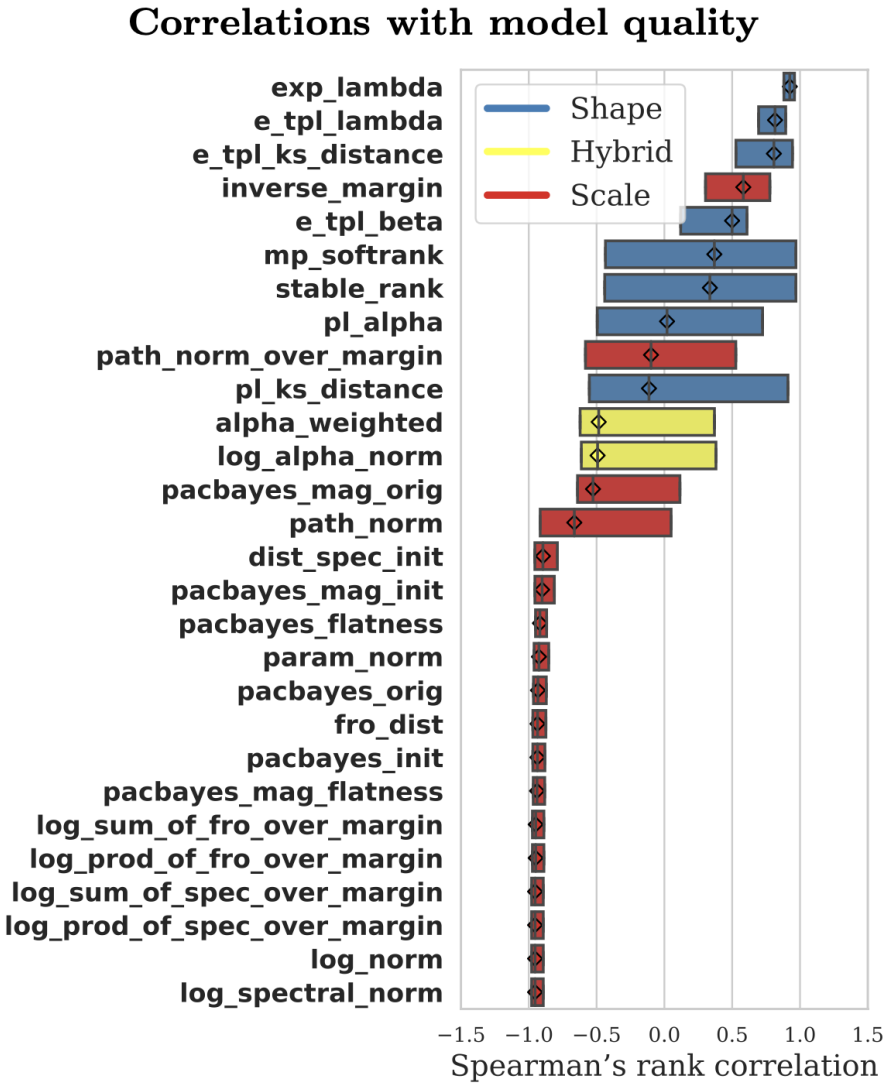
Figure 1 Language modeling performance improves smoothly as we increase the model size, datasetset size, and amount of compute[2] used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

Kaplan et al. (2020). Scaling laws for neural language models.

Hoffmann et al. (2022). Training compute-optimal large language models.



Martin, C. H., Peng, T., & Mahoney, M. W. (2021). Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. Nature Communications, 12(1), 4122.



Yang, Y., Theisen, R., Hodgkinson, L., Gonzalez, J. E., Ramchandran, K., Martin, C. H., & Mahoney, M. W. (2023). *Test accuracy vs. generalization gap: model selection in NLP without accessing training or testing data.*

Open Questions:

- Why do spectral densities of trained feature and weight matrices exhibit heavy-tailed behavior?
- How do data structure, training dynamics, and implicit model bias interplay to produce heavy tails?

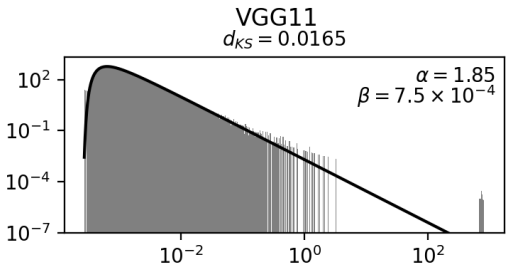
Heavy-Tailed Mechanistic Universality

What might constitute "universality" in neural network weights?

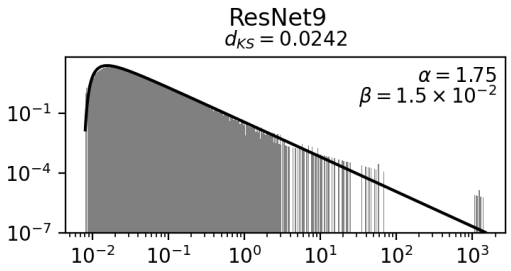
- In RMT:
  - it denotes the emergence of system-independent properties derivable from a few global parameters defining an ensemble.
- In statistical physics:
  - it arises in systems with very strong correlations, at or near a critical point or phase transition;
  - it is characterized by measuring experimentally "observables" that display heavy-tailed behavior, with (universal) power law exponents.

Although trained weight matrices are *not* random, but rather strongly correlated through training, RMT provides a useful descriptive framework.

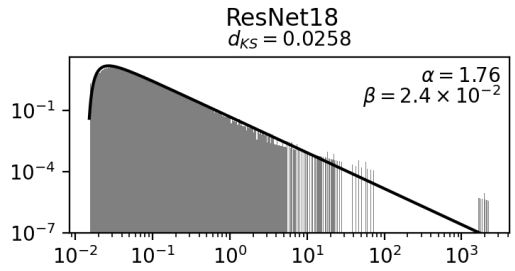
NTK Spectra at Initialization vs. Post-Training



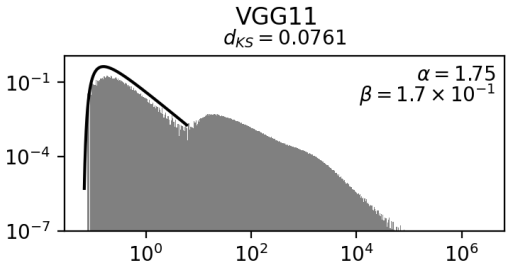
(a) VGG11 Init



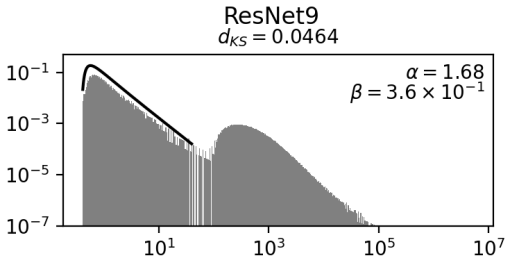
(b) ResNet9 Init



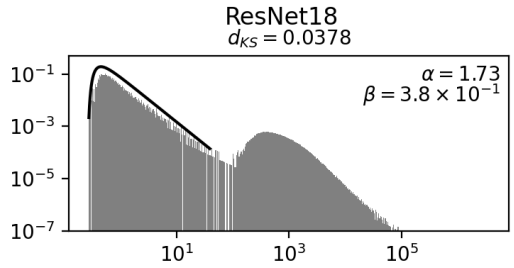
(c) ResNet18 Init



(d) VGG11 Trained



(e) ResNet9 Trained



(f) ResNet18 Trained

Figure: NTK eigenvalue histograms and inverse-Gamma fits near zero. Initialization: mild inverse-Gamma behavior. Post-Training: pronounced heavy-tail

Heavy-Tailed Mechanistic Universality

Definition

Heavy-tailed distributions (informally): densities decaying slower than exponential, often exhibiting power-law tails

$$f(x) \sim c\, x^{-\alpha}, \quad x \to \infty,$$

or inverse-Gamma behavior near zero $f(x) \sim c\, x^{\alpha} e^{-\beta/x}, \quad x \to 0^{+}$.

Possible Approaches for Describing HT-MU:

- iid Heavy-Tailed Elements: (Arous and Guionnet, 2008) Elements of feature matrices are not independent and heavy-tailed.
- Kesten Phenomenon: (Hodgkinson and Mahoney, 2021; Vladimirova et al., 2018; Hanin and Nica, 2020) a mechanism discovered by Kesten (1973) for recursive systems.
- Population Covariance: power-law in, power-law out (PIPO) principle.

Comparison of Possible Mechanisms

Mechanism	Power Law Elements	Power Law Spectrum	Inverse Gamma
iid Heavy-Tailed Elements	✓	✓	×
Kesten Phenomenon	✓	✓	✓/×
Population Covariance	✓/×	✓	✓/×
Structured Matrices (Ours)	×	✓	✓
Empirical Observations (Features)	×	✓	✓
Empirical Observations (Weights)	×	✓	×

Table: Comparison of various mechanisms: capacity to yield power laws, in feature matrix **elements** and feature matrix **spectral densities**; capacity to yield an inverse Gamma law for the spectral density in a neighborhood of zero.

Modeling Framework

Entropic Regularization Setup

- **Stochastic Minimization Operator**

$$\operatorname*{smin}_{\Theta}^{\pi_\Theta,\tau} f(\Theta) \;:=\; \min_{q\in\mathcal{P}} \left[\mathbb{E}_{q(\Theta)}[f(\Theta)] \;+\; \tau\,\mathrm{KL}(q\,\|\,\pi_\Theta)\right],$$

  where $\mathcal{P}$ is the set of probability densities on the support of π_Θ, and

  - π_Θ is the initial prior (Θ = model coefficients).
  - $\tau > 0$ is the "temperature" (controls early stopping).

- Stochastic optimization models (Mandt et al., 2016; Chaudhari and Soatto, 2018) have strong links to Bayesian inference (Germain et al., 2016) and statistical physics of generalization (Mezard and Montanari, 2009).
- Applying to the training loss optimizes a PAC-Bayes bound on the test error (Xie et al., 2023). As τ decreases during training, optimizer smoothly interpolates between π_Θ and the final optimal density.

Entropic Regularization Setup

Feature Learning Setup: Stochastic minimization in *two stages*

$$\operatorname*{smin}_{\Theta}^{\pi_\Theta,\tau} L(\Theta,\Phi) \quad \text{and} \quad q(\Phi) \;=\; \operatorname*{argsmin}_{\Phi}^{\pi_\Phi,\eta}\Big[\operatorname*{smin}_{\Theta}^{\pi_\Theta,\tau} L(\Theta,\Phi)\Big].$$

- π_Θ, π_Φ: initial densities of model coefficients Θ and features Φ.
- $\tau, \eta > 0$: "temperatures" control coefficient vs. feature learning rates.

Proposition (Optimal Feature Density)

$$q(\Phi) \;\propto\; \big[\mathcal{Z}_\tau(\Phi)\big]^{\tau/\eta}\,\pi_\Phi(\Phi), \quad \mathcal{Z}_\tau(\Phi) \;=\; \mathbb{E}_{\Theta\sim\pi_\Theta} \exp\big(-L(\Theta,\Phi)/\tau\big).$$

Of particular interest: late stage of training, $\tau, \eta \to 0^+$ with $\tau/\eta \to \rho > 0$.

Examples of Feature Matrices: Activation Matrix

Activation Matrix (Last Layer).
Neural Network with m output $f(x) = W^\top \varphi(x)$, with $\Phi_{ij} = \varphi_j(x_i)$, $W \in \mathbb{R}^{d \times m}$ trained by ridge regression

$$L(W, \Phi) = \|\Phi W - Y\|_F^2 + \mu \|W\|_F^2.$$

where $\mu > 0$, $\Phi_{ij} = \varphi_j(x_i)$ and $Y = (y_i)_{i=1}^n \in \mathbb{R}^{n \times m}$. For $\pi_W = \mathcal{N}(0, \sigma^2 I)$ and $\tilde{\sigma}^2 = \frac{\sigma^2}{1 + \frac{2\mu\sigma^2}{\tau}}$, the marginal likelihood for optimal feature density:

$$\mathcal{Z}_\tau(\Phi) \propto \frac{\exp\left(-\frac{1}{2}\mathrm{tr}(Y^\top(\tilde{\sigma}^2 \Phi\Phi^\top + \frac{\tau}{2} I)^{-1} Y)\right)}{\det(\tilde{\sigma}^2 \Phi\Phi^\top + \frac{\tau}{2} I)^{m/2}}.$$

$\Sigma = YY^\top$ and $M = (1 + \frac{2\mu\sigma^2}{\tau})^{-1} \Phi\Phi^\top + \frac{\tau}{2\sigma^2} I$. Applying Proposition,

$$q(M) \propto (\det M)^{-\rho m/2} \exp(-\tfrac{1}{2}\rho\sigma^2 \mathrm{tr}(\Sigma M^{-1}))\pi(M)$$

Examples of Feature Matrices: NTK & Hessian

- **Neural Tangent Kernel (NTK)** $J(\Phi) \in \mathbb{R}^{mn \times mn}$.
  Consider $J(\Phi)_{ij} = Df_{\Theta,\Phi}(x_i)^\top Df_{\Theta,\Phi}(x_j)$. Use linearization approximation (Jacot et al., 2018; Rudner et al., 2023; Wilson et al., 2025) to get $f(\Theta) \approx f(\Theta^*) + Df(\Theta^*)(\Theta - \Theta^*)$ with square loss. Then

$$\mathcal{Z}_\tau(\Phi) \propto \frac{\exp(-\frac{1}{2}\mathrm{tr}(\bar{Y}^\top(\sigma^2 J(\Phi) + \frac{\tau}{2} I)^{-1}\bar{Y}))}{\det(\sigma^2 J(\Phi) + \frac{\tau}{2} I)}.$$

  Applying Proposition for $M = J(\Phi)$,

$$q(M) \propto (\det M)^{-\rho/2} \exp\left(-\frac{\rho\sigma^2}{2}\mathrm{tr}(\Sigma M^{-1}) \right) \pi(M)$$

- **Hessian Matrix** $H(\Theta, \Phi) = \nabla^2_\Theta L(\Theta, \Phi)$.
  $\nabla^2_\Theta L(\Theta^*, \Phi) = \sum_{i=1}^n Df(x_i) Df(x_i)^\top$, when $L(\Theta, \Phi) = 0$, and so the spectrum of the Hessian is equivalent (up to zeros) to that of the NTK. Thus, the same $q(M)$ applies for the Hessian for small training loss.

Master Model Ansatz

- **Ansatz:** for trained feature matrices, with parameters $\alpha, \beta > 0$ and initial density π:

$$q(M) \;\propto\; (\det M)^{-\alpha}\, \exp\!\big(-\beta\, \mathrm{tr}(\Sigma\, M^{-1})\big)\, \pi(M)$$

  - $\alpha, \beta > 0$ depend on model/optimizer hyperparameters.
  - Σ is label/covariance-related (e.g., $Y\,Y^\top$).
  - $\pi(M)$ is the prior "initialization" density of the feature matrix.

- **Key Observation:** The trained feature matrix M generally follows an *inverse-Wishart-type density* (Mardia et al., 2024).
  1. First consider $\Sigma = I$ to remove the effect of Σ, the density π of feature matrices M at initialization completely determines the density $q(M)$. Change of variables $M \mapsto Q\Lambda Q^\top$ for orthogonal Q and diagonal Λ; so we only need to study the spectral distribution Λ.
  2. Second, we will consider a general Σ to get spectral densities of trained feature/weight matrices.

RMT for Heavy-Tailed Spectral Behavior

Eigenvector Structure and Beta-Ensembles

- To derive a *spectral density* from the Master Model Ansatz, diagonalize $M = Q\,\mathrm{diag}(\lambda)\,Q^\top$ and set $\Sigma = I$.
- **Key Assumption:** *Distribution of eigenvectors Q is not uniform!* (non-Haar) due to implicit model biases.
- Use **Beta-Ensemble** (Dumitriu and Edelman, 2002; Forrester, 2010) with parameter $\kappa \in [0, \infty]$ to capture the Master Model Ansatz:

$$q_\kappa(\lambda_1, \ldots, \lambda_N) \;\propto\; \prod_{i=1}^{N} V(\lambda_i) \prod_{i<j} |\lambda_i - \lambda_j|^{\kappa/N}$$

- Take $V(\lambda) = \lambda^{-\alpha} \exp(-\beta\,\lambda^{-1})$ to match *Master Model Ansatz*.
- The $1/N$ "high temperature" scaling has also been examined (Forrester and Mazzuca, 2021), but with a different application.
- Although $\pi(M)$ could be complicated, we argue that much of the behavior of π is captured by the extent of the eigenvalue repulsions. κ controls *eigenvalue repulsion*.

Interpreting κ: Structured Feature Matrices

We consider π is uniform over different structured matrix classes with different block Structures ($N \times N$ matrix comprised of $n \times n$ blocks, each of size $m \times m$):

1. **Diagonal:** $\kappa = 0$ (no eigenvector randomness).
2. **Commuting Block-Diagonal:** $\kappa \sim \frac{m}{n}$.
3. **Symmetric Block-Diagonal:** $\kappa \sim (m-1)\frac{mn}{mn-1}$.
4. **Kronecker-Like** $Q_1 \otimes Q_2$, where $Q_1 \in \mathbb{R}^{m \times m}$ and $Q_2 \in \mathbb{R}^{n \times n}$: $\kappa \sim \frac{n}{m} + \frac{m}{n}$.
5. **Fully Symmetric (no structure):** $\kappa = mn$ (Haar eigenvectors).

- As model architecture induces *more structure* (fewer free eigenvector degrees of freedom), κ *decreases* $\Rightarrow$ heavier tail in spectrum.
- We provide a numerical algorithm to efficiently estimate κ.

Main Theorem: HTMP Distribution

Theorem (Generalized Marchenko–Pastur)

Let M_N *follow* $q_\kappa(\lambda_1, \ldots, \lambda_N) \propto \prod_{i=1}^N \lambda_i^{-\alpha} e^{-\beta \lambda_i^{-1}} \prod_{i<j} |\lambda_i - \lambda_j|^{\frac{\kappa(N)}{N}}$ *with parameter* $\kappa(N)$. *Define*

$$\gamma(N) = \frac{\kappa(N)/2}{\alpha - \kappa(N)/2 - 1} \to \gamma \in (0,1) \quad \text{as } N \to \infty.$$

Then the empirical spectral distribution of $\frac{2\gamma(N)\beta}{\kappa(N)} M_N^{-1}$ *converges to:*

1. $\mathbf{MP}_\gamma$ *(Marchenko-Pastur distribution) if* $\kappa(N) \to \infty$;
2. $\mathbf{HTMP}_{\gamma,\kappa}$ *(High-Temperature MP) if* $\kappa(N) \to \kappa \in (0, \infty)$.

This beta-ensemble result is derived from a sequence of random matrix theory from Dumitriu and Edelman (2006); Dung and Duy (2021).

MP v.s. HTMP

- The Marchenko-Pastur distribution $\mathbf{MP}_\gamma$ with parameter $\gamma \in (0,1)$ is absolutely continuous on $(0,\infty)$ with finite support only on the interval $I_\gamma = [\gamma_-, \gamma_+]$ where $\gamma_\pm = (1 \pm \sqrt{\gamma})^{1/2}$. The corresponding probability density function is given by

$$\rho_\gamma(x) = \frac{1}{2\pi} \frac{\sqrt{(\gamma_+ - x)(x - \gamma_-)}}{\gamma x}, \qquad x \in I_\gamma.$$

- The high-temperature Marchenko-Pastur distribution $\mathbf{HTMP}_{\gamma,\kappa}$ is a probability distribution on $(0,\infty)$ with a probability density function

$$\rho_{\gamma,\kappa}(x) = \frac{\kappa}{2\gamma} \frac{1}{\Gamma(\kappa/2+1)\Gamma(\kappa/2\gamma)} \frac{\left(\frac{\kappa x}{2\gamma}\right)^{\frac{\kappa}{2\gamma}-1-\frac{\kappa}{2}} e^{-\frac{\kappa x}{2\gamma}}}{|U(\kappa/2, -\frac{\kappa}{2\gamma}+1+\frac{\kappa}{2}; -\kappa x/2\gamma)|^2}$$

where $U(a,b;z)$ denote the Tricomi confluent hypergeometric function.

Main Theorem: Tail Behavior for Trained Features

Theorem (Spectral Density of Trained Feature Matrix)

Let ρ_N be the ESD of a trained feature matrix M_N, and μ_Σ the spectral measure of label covariance Σ. Then

$$\rho_N(\lambda) \xrightarrow[N\to\infty]{} \big(\mu_\Sigma \boxtimes \rho\big)(\lambda),$$

where $\boxtimes$ is multiplicative free convolution, ρ is either $\lambda^{-2}\,\rho_{\mathrm{MP}}(\lambda^{-1})$ (if $\kappa = \infty$) or $\lambda^{-2}\,\rho_{\mathrm{HTMP}}(\lambda^{-1})$ (if $\kappa < \infty$). Additionally,

- *Bounded vs. Heavy-Tailed: $\kappa = \infty \implies$ bounded support; $\kappa < \infty \implies$ power-law tail.*
- *Inverse-Gamma near zero: If $\kappa < \infty$, density $\rho(x) \sim x^{-\frac{\kappa}{2\gamma}-1-\frac{\kappa}{2}} \exp\left(-\frac{\beta_-}{x}\right)$ as $x \to 0^+$.*
- *Power-law Tail: $\rho(x) \sim x^{-\frac{\kappa}{2\gamma}-1+\frac{\kappa}{2}}$ for $x \to \infty$.*

Remarks

- The power law for the limiting density ρ contains a tail exponent that gets heavier as κ decreases: i.e., as the structure of the underlying matrix becomes more rigid.
- Decreasing κ increases implicit model bias, consistent with Martin and Mahoney (2021b) and Simsekli et al. (2019), who claim heavier tails imply stronger model biases and better model quality and generalization ability.[1]
- HTMP model represents the first RMT ensemble that captures key empirical properties of (strongly-correlated) modern state-of-the-art neural networks (Martin and Mahoney, 2020, 2021a,b; Yang et al., 2023).

[1] Very important: these models' elements need *not* have heavy-tailed behavior.

Applications

Application 1: Neural Scaling Laws

- **Setup:** Ridge regression on activation matrix $\Phi \in \mathbb{R}^{n \times d}$, $m = 1$:

$$\hat{w} = \underset{w}{\operatorname{argmin}}\ L(w) \ = \ \frac{1}{n}\|\Phi w - Y\|^2 + \frac{\mu}{n}\|w\|^2.$$

  Assume $y_i = w_*^\top \varphi(x_i)$, and $\mathbb{E}_x[\varphi(x)\varphi(x)^\top] = I$.

- **Spectral Assumption:** $\Phi\Phi^\top$ follows $\mathbf{HTMP}_{\gamma,\kappa}$ (Master Model).

- *Data-Free Scaling Law:* Predicts test loss decay solely from spectral tail; no access to held-out data required. Previous scaling law works focus on power laws in the dataset (e.g., Wei et al., 2022; Defilippis et al., 2024; Paquette et al., 2024; Lin et al., 2024)

Proposition

Let $\mu = n^{-\ell}$ with $\ell \in (0,1)$. Then, the **Generalization Error** *satisfies*

$$\mathcal{L} := \mathbb{E}_{x,w_*}[(\varphi(x)^\top \hat{w} - y)^2] \ \asymp \ n^{-\ell\left(2+\frac{\kappa}{2\gamma}-\frac{\kappa}{2}\right)}, \quad n \to \infty$$

with high probability.

Application 2: Optimizer Trajectories

- Empirical observation (Mandt et al., 2016; Simsekli et al., 2019; Hodgkinson et al., 2022): *Lower* and *Upper* power-law tails in the distribution of stochastic gradient norms $\|\widehat{\nabla} L_N\|$ during training:

$$\Pr(\|\widehat{\nabla} L_N\| \leq x) \sim C_- \, x^{\alpha}, \quad x \to 0^+,$$
$$\Pr(\|\widehat{\nabla} L_N\| > x) \sim C_+ \, x^{-\beta}, \quad x \to \infty.$$

- **Model:** Assume residuals $\bar{Y}$ are Gaussian, NTK matrix $J \sim$ inverse-Wishart (or **HTMP**) independent of $\bar{Y}$.
- **Application:** Under these assumptions, $\|\widehat{\nabla} L_N\|$ exhibits both lower and upper power-law tails.
- There has been significant theoretical justification for the upper power law in terms of the Kesten mechanism (Hodgkinson and Mahoney, 2021; Gurbuzbalaban et al., 2021, 2022), but there has been little justification for the lower power law before.

Application 3: 5+1 Phases of Trained Weight Matrices

- **Empirical Observation** (Martin and Mahoney, 2019, 2020, 2021b; Yang et al., 2023; Zhou et al., 2023): Trained weight matrices can exhibit 5+1 Phases of Training:
  1. Random-Like (MP bulk, no outliers).
  2. Bleeding-Out (MP bulk with emerging spikes).
  3. Bulk+Spikes (distinct spikes outside bulk).
  4. Bulk-Decay (bulk extends, no finite support).
  5. Heavy-Tailed (power-law tail).
  6. Rank-Collapse (mass at zero eigenvalue).
- **Application:** Consider $A = W^\top W$ with trained weight W, then $\frac{\beta}{\alpha - \kappa/2 - 1} A$ converges to $\textbf{HTMP}_{\gamma,\kappa}$.
- Decreasing κ across training $\Rightarrow$ transition from bounded support to heavy tail. Power law exponents in the spectrum of weight matrices are strongly predictive of model performance.

5+1 Phases for Trained Weight: HTMP Fits

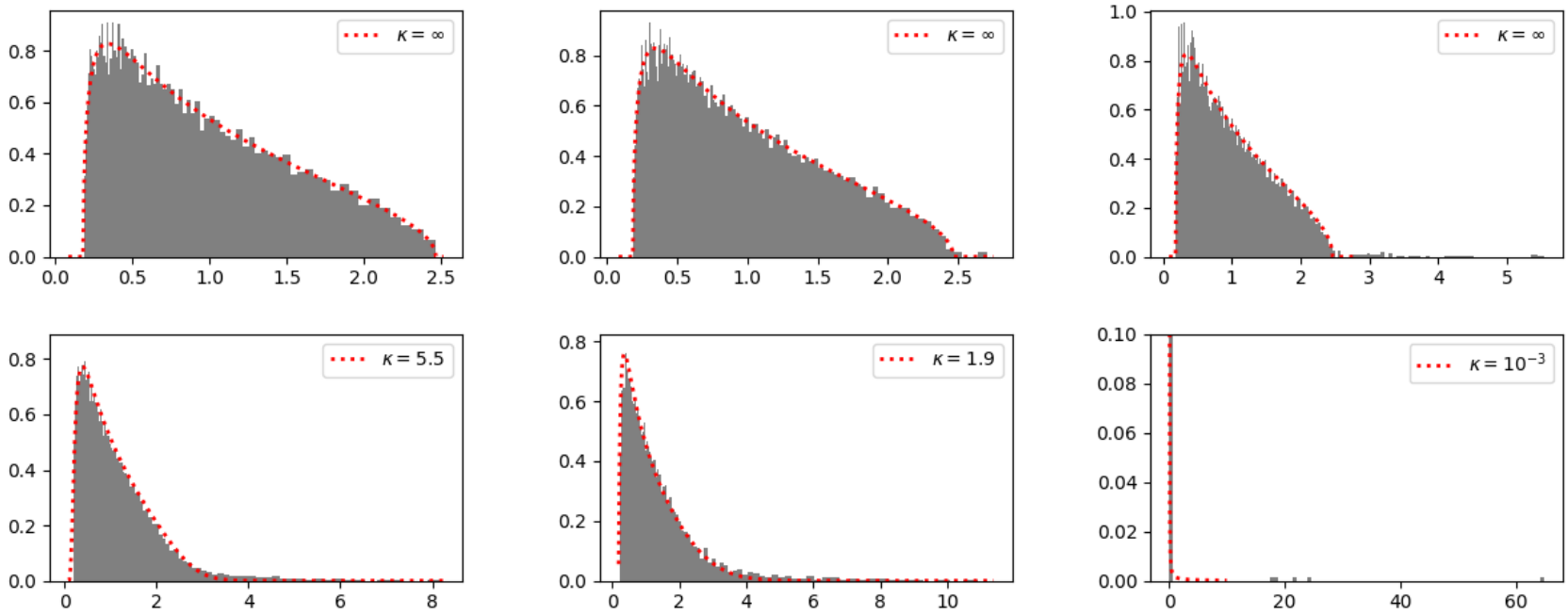


Figure: Weight spectral densities for MiniAlexNet trained on CIFAR-10 with batch sizes 1000, 800, 250, 100, 50, 5 (top to bottom). *Fitted MP/HTMP curves shown in red dashed* with different κ.

As batch size decreases, κ decreases $\Rightarrow$ heavier tail.
(a)–(c): $\kappa = \infty$ for MP or MP+spike behavior.
(d)–(f): Finite κ for heavy tail plus eventual rank collapse.

Summary of Simulations

- *Initialization:* NTK spectra show mild inverse-Gamma edge, no heavy-tail ($\kappa \approx \infty$).
- *During Training:*
  - NTK spectra develop a mixture: initial and trained components diverge.
  - Weight matrices transition MP $\Rightarrow$ MP+spike $\Rightarrow$ heavy tail.
- *Post-Training:*
  - NTK and Hessian spectra exhibit clear power-law tails at both edges ($\kappa < \infty$).
  - Final-layer weight spectra match $\mathbf{HTMP}_{\gamma,\kappa}$ fits.
- **Takeaway:** HTMP family $\{\mathbf{HTMP}_{\gamma,\kappa}\}$ successfully interpolates from MP-like to heavy-tailed regimes by tuning κ.

Conclusions

- **Master Model**: A unified RMT framework (Master Model Ansatz) that captures heavy-tailed spectral behavior of trained feature matrices from a Bayesian perspective.
- **HTMP Ensemble**: High-temperature MP (**HTMP**$_{\gamma,\kappa}$) arises when eigenvector entropy $\propto \kappa$ is finite; interpolates between MP ($\kappa \to \infty$) and heavy-tailed regimes ($\kappa \to 0^+$).
- **Key Insights**
  1. *Data Contribution*: Heavy-tailed population covariance $\Sigma \implies$ heavy-tailed trained spectra (PIPO).
  2. *Eigenvector Structure*: More architectural bias (smaller κ) $\implies$ heavier tails.
  3. *Training Dynamics*: As $\tau, \eta \to 0$, HTMP hyperparameters α, β, κ evolve, explaining transitions (5+1 phases).
- **Applications**
  - Neural scaling laws (ridge regression) predicted by HTMP exponents.
  - Lower/upper power-law tails in SGD trajectories explained.
  - 5+1 training phases fit by tuning κ for HTMP.

Thank You!

Liam Hodgkinson, Zhichao Wang, Michael W. Mahoney.
"Models of Heavy-Tailed Mechanistic Universality"
https://arxiv.org/abs/2506.03470.

References I

K. K. Agrawal, A. K. Mondal, A. Ghosh, and B. Richards. α-ReQ: Assessing representation quality in self-supervised learning by measuring eigenspectrum decay. *Advances in Neural Information Processing Systems*, 35:17626–17638, 2022.

G. B. Arous and A. Guionnet. The spectrum of heavy tailed random matrices. *Communications in Mathematical Physics*, 278 (3):715–751, 2008.

P. Chaudhari and S. Soatto. Stochastic gradient descent performs variational inference, converges to limit cycles for deep networks. In *2018 Information Theory and Applications Workshop (ITA)*, pages 1–10. IEEE, 2018.

L. Defilippis, B. Loureiro, and T. Misiakiewicz. Dimension-free deterministic equivalents and scaling laws for random feature regression. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

I. Dumitriu and A. Edelman. Matrix models for beta ensembles. *Journal of Mathematical Physics*, 43(11):5830–5847, 2002.

I. Dumitriu and A. Edelman. Global spectrum fluctuations for the β-hermite and β-laguerre ensembles via matrix models. *Journal of Mathematical Physics*, 47(6), 2006.

T. H. Dung and T. K. Duy. Beta laguerre ensembles in global regime. *Osaka Journal of Mathematics*, 58(2):435–450, 2021.

P. J. Forrester. *Log-gases and random matrices*. Princeton University Press, 2010.

P. J. Forrester and G. Mazzuca. The classical β-ensembles with β proportional to $1/N$: from loop equations to Dyson's disordered chain. *Journal of Mathematical Physics*, 62(7), 2021.

P. Germain, F. Bach, A. Lacoste, and S. Lacoste-Julien. PAC-Bayesian theory meets Bayesian inference. *Advances in Neural Information Processing Systems*, 29, 2016.

M. Gurbuzbalaban, U. Simsekli, and L. Zhu. The heavy-tail phenomenon in SGD. In *International Conference on Machine Learning*, pages 3964–3975, 2021.

M. Gurbuzbalaban, Y. Hu, U. Simsekli, K. Yuan, and L. Zhu. Heavy-tail phenomenon in decentralized sgd. *arXiv preprint arXiv:2205.06689*, 2022.

B. Hanin and M. Nica. Products of many large random matrices and gradients in deep neural networks. *Communications in Mathematical Physics*, 376(1):287–322, 2020.

J. Hestness, S. Narang, N. Ardalani, G. Diamos, H. Jun, H. Kianinejad, M. M. A. Patwary, Y. Yang, and Y. Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

L. Hodgkinson and M. W. Mahoney. Multiplicative noise and heavy tails in stochastic optimization. In *International Conference on Machine Learning*, pages 4262–4274, 2021.

L. Hodgkinson, U. Simsekli, R. Khanna, and M. W. Mahoney. Generalization bounds using lower tail exponents in stochastic optimizers. In *International Conference on Machine Learning*, pages 8774–8795, 2022.

References II

J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

A. Jacot, F. Gabriel, and C. Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in Neural Information Processing Systems*, 31, 2018.

J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

H. Kesten. Random difference equations and renewal theory for products of random matrices. *Acta Mathematica*, 131(1): 207–248, 1973.

L. Lin, J. Wu, S. M. Kakade, P. Bartlett, and J. D. Lee. Scaling laws in linear regression: Compute, parameters, and data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

H. Lu, Y. Zhou, S. Liu, Z. Wang, M. W. Mahoney, and Y. Yang. AlphaPruning: Using heavy-tailed self regularization theory for improved layer-wise pruning of large language models. In *Thirty-eighth Conference on Neural Information Processing Systems*, 2024.

S. Mandt, M. Hoffman, and D. Blei. A variational analysis of stochastic gradient algorithms. In *International Conference on Machine Learning*, pages 354–363, 2016.

K. V. Mardia, J. T. Kent, and C. C. Taylor. *Multivariate analysis*. John Wiley & Sons, 2024.

C. H. Martin and M. W. Mahoney. Traditional and heavy tailed self regularization in neural network models. In *International Conference on Machine Learning*, pages 4284–4293, 2019.

C. H. Martin and M. W. Mahoney. Heavy-tailed universality predicts trends in test accuracies for very large pre-trained deep neural networks. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, pages 505–513. SIAM, 2020.

C. H. Martin and M. W. Mahoney. Post-mortem on a deep learning contest: a simpson's paradox and the complementary roles of scale metrics versus shape metrics. *arXiv preprint arXiv:2106.00734*, 2021a.

C. H. Martin and M. W. Mahoney. Implicit self-regularization in deep neural networks: Evidence from random matrix theory and implications for learning. *The Journal of Machine Learning Research*, 22(1):7479–7551, 2021b.

M. Mezard and A. Montanari. *Information, physics, and computation*. Oxford University Press, 2009.

E. Paquette, C. Paquette, L. Xiao, and J. Pennington. 4+3 phases of compute-optimal neural scaling laws. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

L. Pillaud-Vivien, A. Rudi, and F. Bach. Statistical optimality of stochastic gradient descent on hard learning problems through multiple passes. *Advances in Neural Information Processing Systems*, 31, 2018.

References III

T. G. Rudner, S. Kapoor, S. Qiu, and A. G. Wilson. Function-space regularization in neural networks: A probabilistic perspective. In *International Conference on Machine Learning*, pages 29275–29290, 2023.

U. Simsekli, L. Sagun, and M. Gurbuzbalaban. A tail-index analysis of stochastic gradient noise in deep neural networks. In *International Conference on Machine Learning*, pages 5827–5837, 2019.

B. Sorscher, R. Geirhos, S. Shekhar, S. Ganguli, and A. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022.

M. Vladimirova, J. Arbel, and P. Mesejo. Bayesian neural networks become heavier-tailed with depth. In *NeurIPS 2018-Thirty-second Conference on Neural Information Processing Systems*, pages 1–7, 2018.

Z. Wang, A. Engel, A. D. Sarwate, I. Dumitriu, and T. Chiang. Spectral evolution and invariance in linear-width neural networks. *Advances in neural information processing systems*, 36:20695–20728, 2023.

A. Wei, W. Hu, and J. Steinhardt. More than a toy: Random matrix models predict how real-world neural representations generalize. In *International Conference on Machine Learning*, pages 23549–23588, 2022.

J. Wilson, C. van der Heide, L. Hodgkinson, and F. Roosta. Uncertainty quantification with the empirical neural tangent kernel. *arXiv preprint arXiv:2502.02870*, 2025.

Z. Xie, Q.-Y. Tang, Y. Cai, and M. Sun. On the power-law Hessian spectra in deep learning. *arXiv preprint arXiv:2201.13011*, 2023.

Y. Yang, R. Theisen, L. Hodgkinson, J. E. Gonzalez, K. Ramchandran, C. H. Martin, and M. W. Mahoney. Test accuracy vs. generalization gap: Model selection in NLP without accessing training or testing data. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3011–3021, 2023.

Y. Zhang, B. Kang, B. Hooi, S. Yan, and J. Feng. Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Y. Zhou, T. Pang, K. Liu, C. H. Martin, M. W. Mahoney, and Y. Yang. Temperature balancing, layer-wise weight analysis, and neural network training. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Overview

Motivations:

- WeightWatcher, Weight Diagnostics for Analyzing ML Models
  (with Charles H. Martin)
- Randomized Numerical Linear Algebra for Modern ML
  (with Michal Derezinski)

Some Theory:

- RMT for NNs: Linear to Nonlinear; Shallow to Deep; etc.
  (with Zhenyu Liao)

Applications:

- Models of Heavy-Tailed Mechanistic Universality
  (with Zhichao Wang and Liam Hodgkinson)
- Spectral Estimation with Free Decompression
  (with Siavash Ameli, Chris van der Heide, and Liam Hodgkinson)
- Determinant Estimation under Memory Constraints and Neural Scaling Laws
  (with S. Ameli, C. van der Heide, L. Hodgkinson, and F. Roosta)

Spectral Estimation with Free Decompression

—

Team Members

Siavash Ameli

International Computer Science Institute

University of California, Berkeley

Chris van der Heide

The University of Melbourne

Liam Hodgkinson

The University of Melbourne

Michael W. Mahoney

International Computer Science Institute

Lawrence Berkeley National Laboratory

University of California, Berkeley

Computing Eigenvalues of Large Matrices

- Eigenvalues encode essential matrix information; empirical spectral distribution is useful for diagnostics, e.g. is the spectrum heavy-tailed?
- Particularly useful for computing spectral functions, including

$$\log\det A = \sum_i \log \lambda_i(A) \qquad \operatorname{tr}(A^{-k}) = \sum_i \lambda_i(A)^{-k}$$

$$\operatorname{cond}(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}, \quad \text{if } A \text{ is positive-definite.}$$

- These quantities are important e.g. for Gaussian processes, but need the *entire range* of eigenvalues, not just largest/smallest
- Standard eigenvalue solvers are $\mathcal{O}(n^3)$ complexity; expensive for large matrices!

Tiers of Matrix Difficulty

Explicit: the whole matrix fits in memory

Implicit: can make use of matrix-vector products (e.g. CG, SLQ)

Out-of-core: parts of the matrix can be loaded into memory a piece at a time

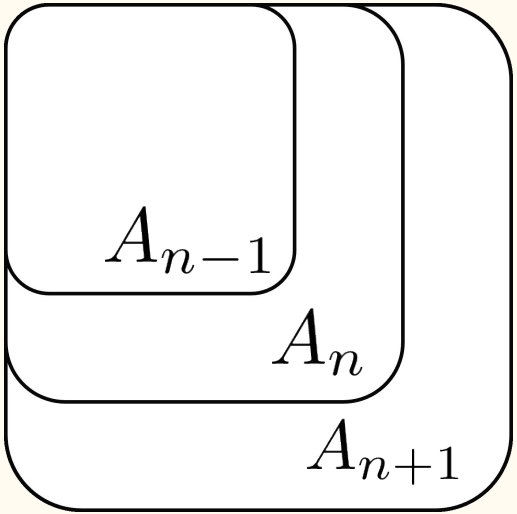
Impalpable: most matrix entries are inaccessible, matrix-vector products are unavailable (e.g. distributed or enormous datasets)

	Access in Memory		
Type	Matrix	Matrix–Vector Product	Any Subblock
Explicit	✓	✓	✓
Implicit	✗	✓	✗
Out-of-core	✗	∼	✓
Impalpable	✗	✗	✗

Extrapolating Matrices

Suppose our matrix of interest is embedded in an infinite sequence of nested matrices

$$A_1, A_2, A_3, \ldots \qquad A_n \in \mathbb{R}^{n \times n}$$



so that $(A_n)_{ij} = (A_{n+1})_{ij}$

Objective: Find eigenspectrum of A_n using eigenspectrum of A_{n_s} for $n_s \ll n$

Free Probability

How do we ensure the eigenvalues of submatrices represent the whole matrix?

An important topic in random matrix theory involving random matrices with uniformly random eigenvectors, so that probability distributions of matrix dependents (including submatrices) *depend only on the eigenspectra*.

Theorem (Nica, 1993): Any sequence of matrices can be turned into an (asymptotically) free sequence of random matrices by applying random permutations σ to the rows and columns:

$$\tilde{A}_{ij} = A_{\sigma(i)\sigma(j)}$$

Stieltjes Transform

The spectral density of a matrix A is encoded in its Stieltjes transform:

$$m_n(z) = \frac{1}{n}\mathrm{tr}(A - zI)^{-1} \qquad A \in \mathbb{R}^{n\times n}$$

In the large matrix limit, when the eigenvalues are drawn from a density ρ, there is a one-to-one correspondence between ρ and the Stieltjes transform m.

$$m(z) = \int_{-\infty}^{\infty} \frac{\rho(x)}{x - z}\,\mathrm{d}x \qquad \rho(x) = \frac{1}{\pi}\lim_{\varepsilon\to 0^+} \Im[m(x + i\varepsilon)]$$

Free Decompression

Let $m(t, \cdot)$ be the Stieltjes transform of the enlargement of A by a factor of e^t
Under the large matrix limit, $m(t, \cdot)$ satisfies the *partial differential equation:*

$$\frac{\partial m}{\partial t} = -m + \frac{1}{m}\frac{\partial m}{\partial z}$$

Proof: Random matrix theory arguments involving the R-transform and the celebrated theorem of (Nica & Speicher, 1996).

To our knowledge, this operation has always been considered in reverse (*free compression*), finding eigenspectra of submatrices, given the eigenspectrum of the full matrix. We are the first to attempt ***free decompression.***

Free decompression of a random submatrix $\mathbf{A}_n$ to a larger matrix $\mathbf{A}$ requires:

1. **estimation** of its Stieltjes transform $m_{\mathbf{A}_n}$;
2. **evolution** of $m_{\mathbf{A}_n}$ in n using PDE;
3. **evaluation** of the spectral distribution of $\mathbf{A}$.

An Engineering Challenge

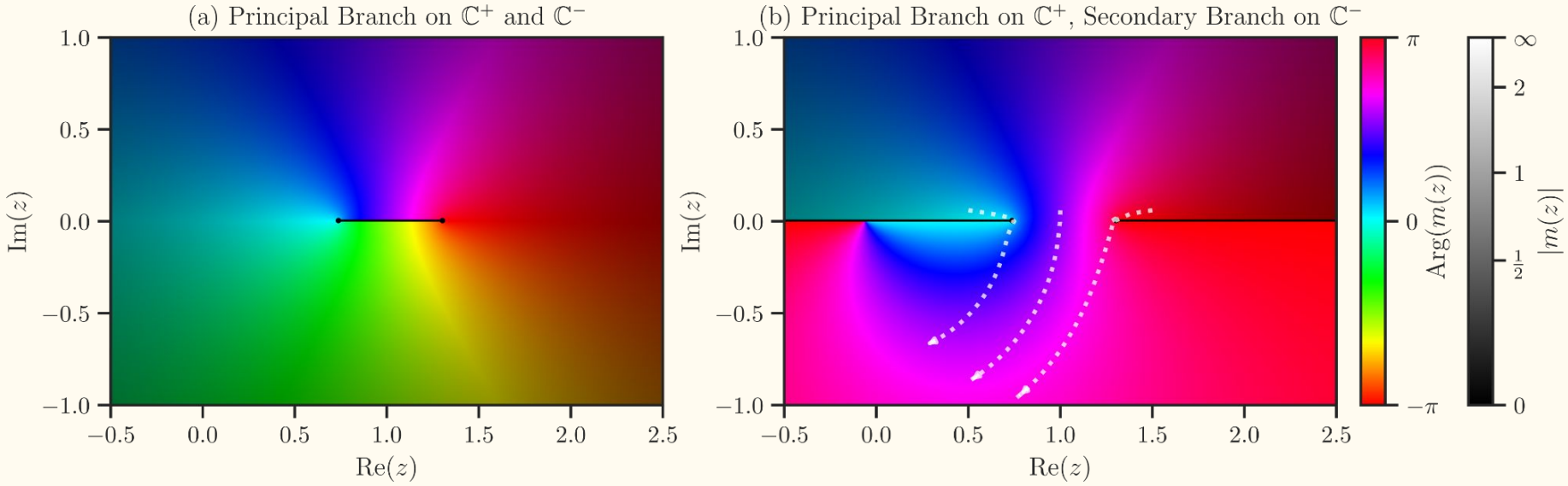
This is a very difficult equation to solve!

Solve the PDE using method of characteristics in the complex plane. But...

Proposition: All characteristic curves pass through the (discontinuous) branch cut for the principal branch of the Stieltjes transform.

- ➢ To solve the characteristic equations, a new *secondary* branch is required.
- ➢ Tantamount to (ill-posed) numerical analytic continuation.
- ➢ Naively solving the PDE fails: we need to directly tackle the analytic continuation problem.

Analytic Continuation of Stieltjes Transform



An Engineering Challenge

This is a very difficult equation to solve!

Theorem: The error grows **at most polynomially** in the matrix size.

Requires significantly more engineering than first glance:

- ➢ Multiple layers of polynomial approximation from eigenvalues (Lanczos iteration and Kernel Polynomial Method are not accurate enough)
- ➢ Construct a particular Padé approximant
- ➢ Solve characteristic curves using Newton iterations

Performed properly, in practice, error grows **at most logarithmically** in the matrix size.

Random Matrix Ensembles

Distribution	Free Corresp.	Abs. Cont. Density $\rho(x)$	Support $\lambda_\pm$	Number of Atoms
Wigner semicircle	Free Gaussian	$\frac{2\sqrt{r^2-x^2}}{\pi r^2}$	$\pm r$	None
Marchenko–Pastur	Free Poisson	$\frac{\sqrt{(\lambda_+-x)(x-\lambda_-)}}{2\pi\lambda x}$	$(1\pm\sqrt{\lambda})^2$	$(1-\frac{1}{\lambda})\delta(x)$ if $\lambda>1$
Kesten–McKay	Free Binomial	$\frac{d\sqrt{4(d-1)-x^2}}{2\pi(d^2-x^2)}$	$\pm 2\sqrt{d-1}$	None
Wachter	Free Jacobi	$\frac{(a+b)\sqrt{(\lambda_+-x)(x-\lambda_-)}}{2\pi x(1-x)}$	$\left(\frac{\sqrt{b}\pm\sqrt{a(a+b-1)}}{a+b}\right)^2$	$x=0,1$
Meixner	Free Meixner	$\frac{c\sqrt{4b-(x-a)^2}}{2\pi((1-c)x^2+acx+bc^2)}$	$a\pm 2\sqrt{b}$	At most two

Random Matrix Ensembles

Distribution	Matrix or Combinatorial Model	Parameters
Wigner semicircle	Gaussian orthogonal ensemble $\frac{1}{\sqrt{2}}(\mathbf{X} + \mathbf{X}^\intercal)$	$r = 2\sqrt{n}$
	Adjacency matrix of Erdős–Rényi graph $G(n, p)$	$pn = \mathcal{O}(\log(n))$
Marchenko–Pastur	Sample covariance (Wishart) $\frac{1}{d}\mathbf{X}\mathbf{X}^\intercal$, $\mathbf{X} \in \mathbb{R}^{n \times d}$	$\lambda = \frac{n}{d}$
Kesten–McKay	Haar–orthogonal Hermitian sum $\sum_{i=1}^{k}(\mathbf{O}_i + \mathbf{O}_i^\intercal)$	$d = 2k$
	Projection model $d\,\mathbf{PODO}^\intercal\mathbf{P}$ (Longoria & and, 2023)	$d \geq 2$
	Adjacency matrix of a random d-regular graph	$d \geq 2$
Wachter	Generalized eigenvalues of $(\mathbf{S}_1, \mathbf{S}_1 + \mathbf{S}_2)$, $\mathbf{S}_i = \frac{1}{d_i}\mathbf{X}_i\mathbf{X}_i^\intercal$	$a = \frac{d_1}{n}$, $b = \frac{d_2}{n}$
	Arises in MANOVA problems	
Meixner	Bordered Toeplitz tridiagonal with Jacobi coefficients α_1, β_1	$a = \alpha_1, b = \beta_1 - 1$
	Block–Gaussian ensembles (Lenczewski, 2015)	

Random Matrix Ensembles

$$\mathbf{J} = \left[\begin{array}{c|ccccc} \alpha_0 & \beta_0 & & & & \\ \hline \beta_0 & \alpha_1 & \beta_1 & & & \\ & \beta_1 & \alpha_1 & \beta_1 & & \\ & & \ddots & \ddots & \ddots & \end{array}\right]$$

For Meixner family, the Jacobi matrix of orthogonal polynomial recursion is periodic.

$$m(z) = \cfrac{1}{z - \alpha_0 - \cfrac{\beta_0^2}{z - \alpha_1 - \cfrac{\beta_1^2}{z - \alpha_1 - \cfrac{\beta_1^2}{\ddots}}}}$$

Stieltjes transform, as continued fraction of Jacobi coefficients, becomes periodic.

Stieltjes transform can be solved by quadratic equation:

$$m(z) = \frac{P(z) + \sqrt{P(z)^2 - 4Q(z)}}{2Q(z)}$$

Random Matrix Ensembles

	Stieltjes and Hilbert Transforms		
Distribution	$P(z)$	$Q(z)$	R-Transform
Wigner semicircle	$-z$	$\frac{r^2}{4}$	$\frac{r^2}{4}z$
Marchenko–Pastur	$1-\lambda-z$	λz	$\frac{1}{1-\lambda z}$
Kesten–McKay	$\frac{(2-d)z}{d-1}$	$\frac{d^2-z^2}{d-1}$	$\frac{-d+d\sqrt{1+4z^2}}{2z}$
Wachter	$\frac{a-1-(a+b-2)z}{a+b-1}$	$\frac{z(1-z)}{a+b-1}$	$\frac{-(a+b)+z+\sqrt{(a+b)^2+2(a-b)z+z^2}}{2z}$
Meixner	$\frac{ac-(c-2)z}{2}$	$\frac{(1-c)z^2+acz+bc^2}{4}$	$\left(\frac{c}{1-c}\right)\frac{1-az+\sqrt{(1-az)^2-4b(1-c)z^2}}{2z}$

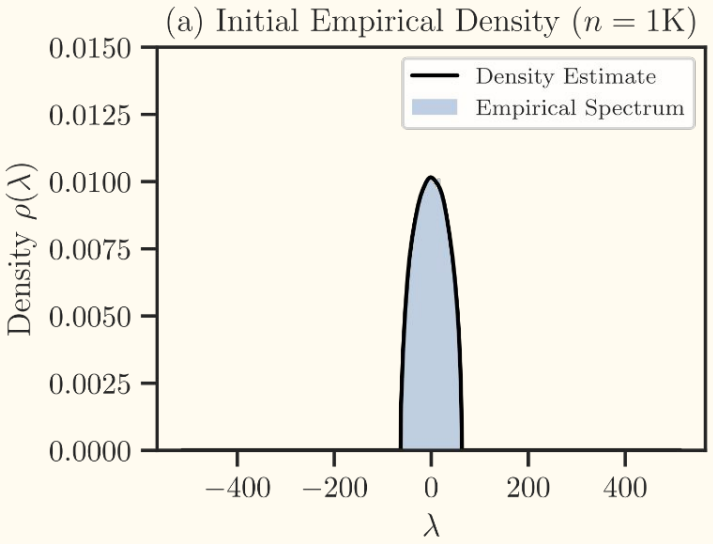
Experiments with Random Matrix Ensembles

These are convenient baselines, since we know the expected shape of the eigenspectrum in advance *for any matrix size* (computing eigenvalues is expensive!)

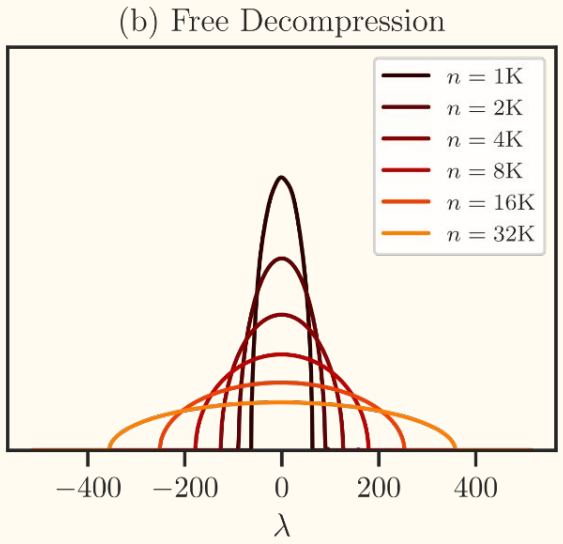
Under normally-distributed synthetic data, we expand

$$n = 1000 \xrightarrow{\text{free decompression}} n = 32,000$$

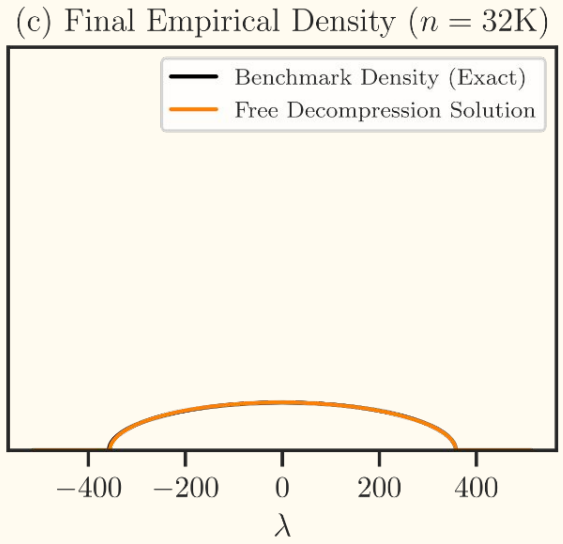
Matrices with iid Entries (Wigner Semicircle Law)



Histogram of eigenvalues of small matrix & density estimate

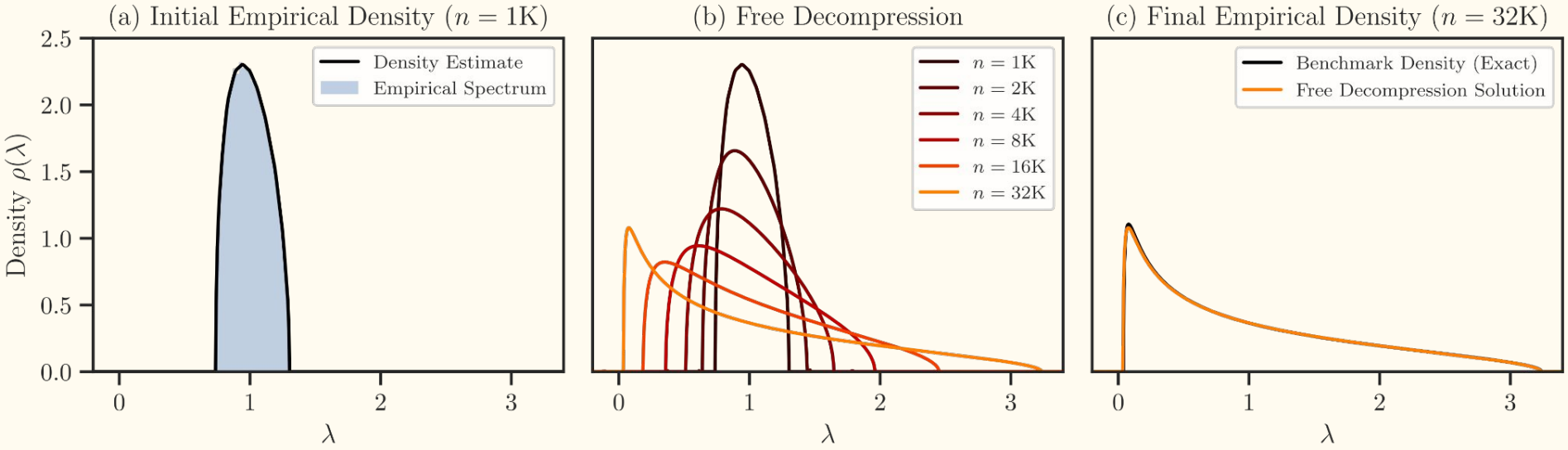


Densities under free decompression



Expected density & solution from free decompression

Wishart Matrices (Marchenko-Pastur Law)

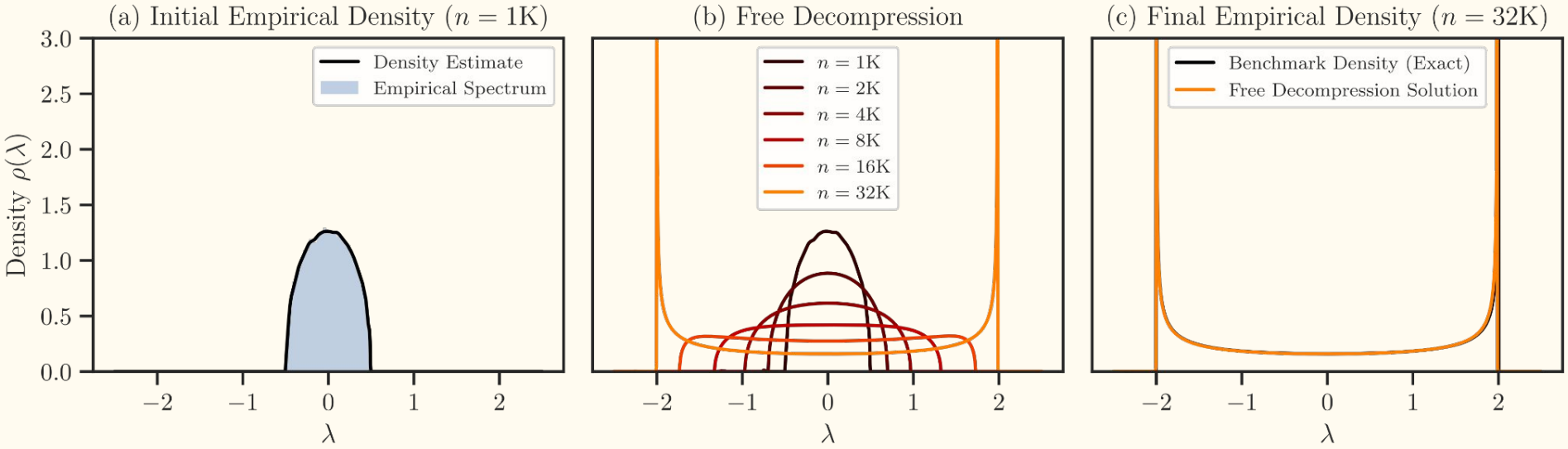


Histogram of eigenvalues of small matrix & density estimate

Densities under free decompression

Expected density & solution from free decompression

Random Projections (Kesten-McKay Law)

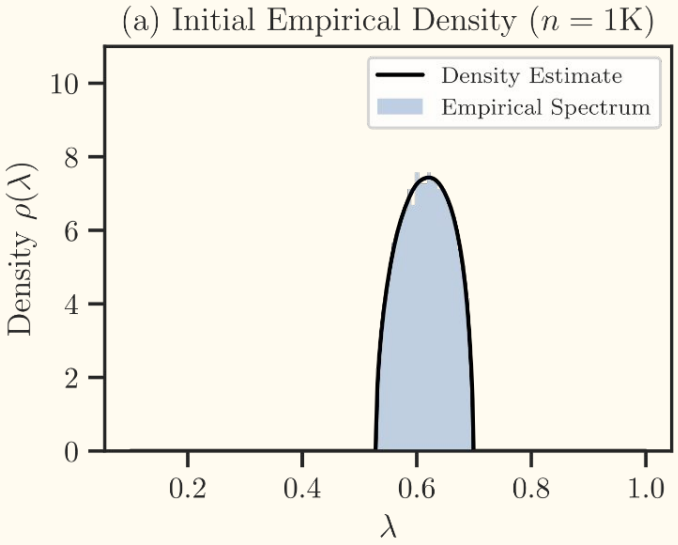


Histogram of eigenvalues of small matrix & density estimate

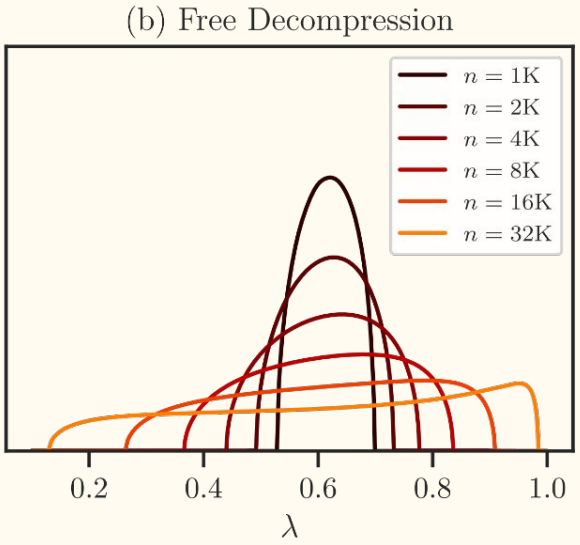
Densities under free decompression

Expected density & solution from free decompression

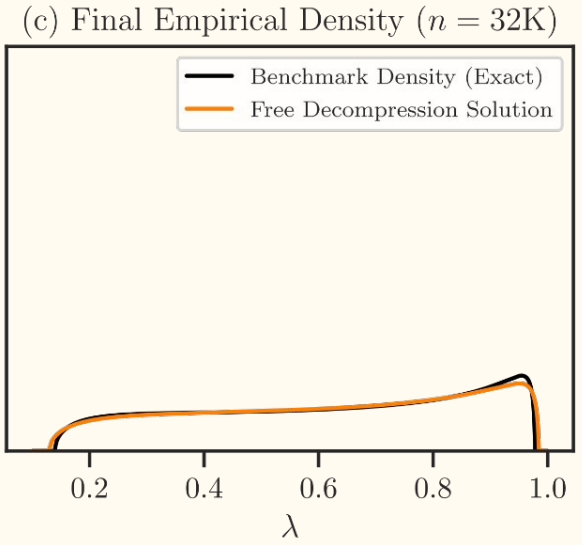
Generalized Eigenvalue Problems (Wachter Law)



Histogram of eigenvalues of small matrix & density estimate

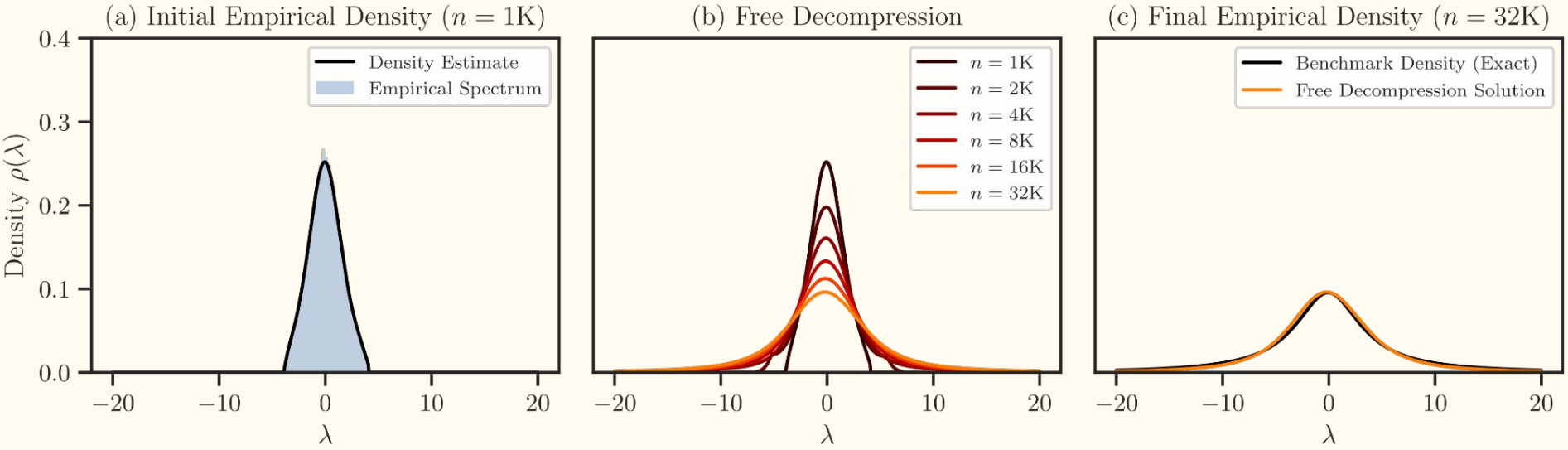


Densities under free decompression



Expected density & solution from free decompression

General Family of Meixner Law



Histogram of eigenvalues of small matrix & density estimate

Densities under free decompression

Expected density & solution from free decompression

Experiments with Real Data

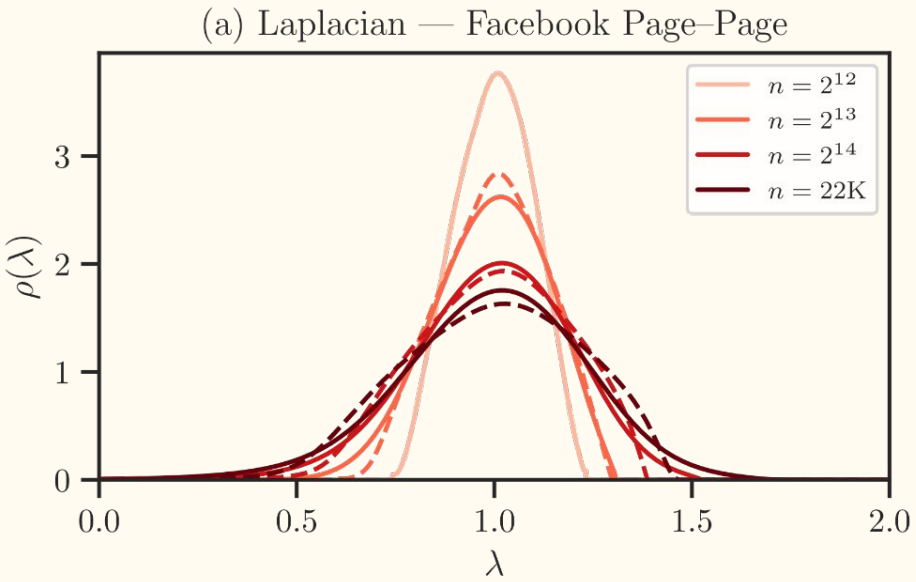
Large covariance and kernel matrices involving real data typically exhibit disconnected spectral densities with support over multiple orders of magnitude.

Density estimation remains a significant challenge here

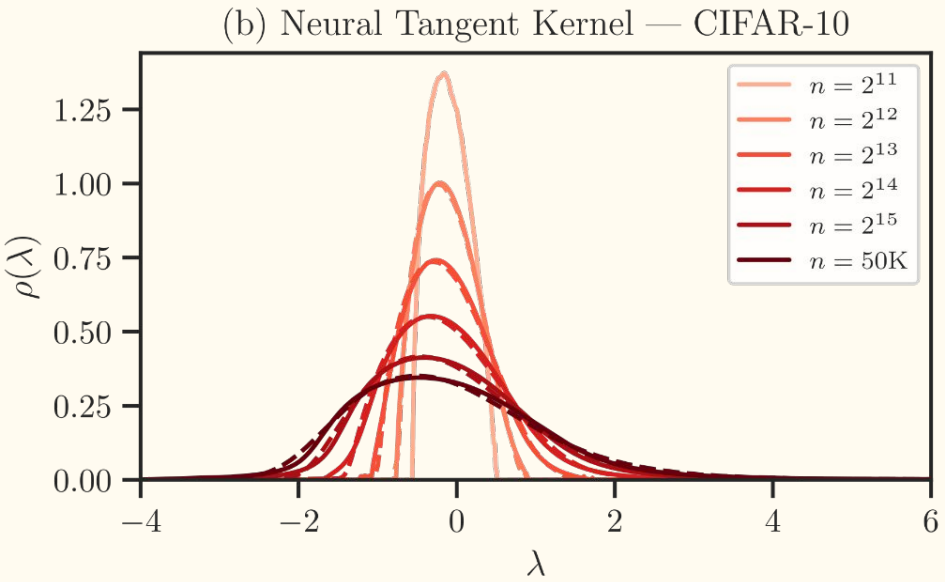
We consider two examples of real data matrices to demonstrate efficacy of our current procedure:

1. **Facebook SNAP Graph** Dataset (22,470 x 22,470 adjacency matrix) perturbed by an Erdős-Rényi graph to reduce leaf nodes.
2. Log-**neural tangent kernel** Gram matrix from ResNet50 trained on CIFAR-10 with low-rank components removed (50,000 x 50,000 dense matrix).

Experiments with Real Data



Symmetrically normalized Laplacian matrix of the SNAP Facebook dataset



log-NTK matrix computed from the CIFAR-10 dataset using a ResNet-50 model

Empirical spectral density (solid) vs. free decompression estimate from $\eta = 2^{11}$ (dashed)

Experiments with Real Data

Table: Comparison of runtime of direct computation of spectral density versus the free decompression of the NTK dataset, and accuracy in terms of statistical distance and moments.

Size	Process Time (sec)		Divergences		Rel. Error	
n_s	Direct	FD (ours)	TV	JS	μ_1	μ_2
2^{11}	10.2	10.2 + 0.00	0.0%	0.0%	0.0%	0.0%
2^{12}	50.9	10.2 + 54.2	1.2%	3.7%	0.4%	0.3%
2^{13}	358.9	10.2 + 56.6	1.9%	5.2%	0.9%	0.2%
2^{14}	2820.2	10.2 + 54.9	2.4%	5.8%	0.9%	0.1%
2^{15}	20451.2	10.2 + 61.9	2.6%	5.8%	1.2%	0.5%
50K	67331.1	10.2 + 16.2	2.9%	5.5%	2.4%	0.4%

FREEALG

freealg is our Python package that implements free decompression for estimating eigenspectra.

`pip install freealg`

(work in progress!)

Listing 1: A minimal usage example of the `freealg` package.

```
# Install freealg with "pip install freealg"
import freealg as fa

# Create an object for the Marchenko--Pastur distribution with the parameter λ = 1/50
mp = fa.distributions.MarchenkoPastur(1/50)

# Generate a matrix of size n_s = 1000 corresponding to this distribution
A = mp.matrix(size=1000)

# Create a free-form object for the matrix within the support I = [λ_-, λ_+]
ff = fa.FreeForm(A, support=(mp.lam_m, mp.lam_p))

# Fit the distribution using Jacobi polynomials of degree K = 20, with α = β = 1/2
# Also fit the glue function via Pade of degree [(p+q)/q] with p = 0, q = 1.
psi = ff.fit(method='jacobi', K=20, alpha=0.5, beta=0.5, reg=0.0, damp='jackson',
             pade_p=0, pade_q=1, optimizer='ls', plot=True)

# Decompress the spectral density corresponding to a larger matrix of size n = 2^5 × n_s,
rho_large = ff.decompress(size=32_000, plot=True)
```

arXiv

Siavash Ameli, Chris van der Heide, Liam Hodgkinson, Michael W. Mahoney. (2025) *Spectral Estimation with Free Decompression*. arxiv: 2506.11994

Overview

Motivations:

- WeightWatcher, Weight Diagnostics for Analyzing ML Models (with Charles H. Martin)
- Randomized Numerical Linear Algebra for Modern ML (with Michal Derezinski)

Some Theory:

- RMT for NNs: Linear to Nonlinear; Shallow to Deep; etc. (with Zhenyu Liao)

Applications:

- Models of Heavy-Tailed Mechanistic Universality (with Zhichao Wang and Liam Hodgkinson)
- Spectral Estimation with Free Decompression (with Siavash Ameli, Chris van der Heide, and Liam Hodgkinson)
- Determinant Estimation under Memory Constraints and Neural Scaling Laws (with S. Ameli, C. van der Heide, L. Hodgkinson, and F. Roosta)

Determinant Estimation under Memory Constraints and Neural Scaling Laws

Siavash Ameli [1,2] Chris van der Heide [3] Liam Hodgkinson [4] Fred Roosta [5]
Michael W. Mahoney [1,2,6]

[1] *Department of Statistics, UC Berkeley*
[2] *International Computer Science Institute*
[3] *Dept. of Electrical and Electronic Eng., University of Melbourne*
[4] *School of Mathematics and Statistics, University of Melbourne*
[5] *CIRES and School of Mathematics and Physics, University of Queensland*
[6] *Lawrence Berkeley National Laboratory*

ICML 2025

Overview

Log-determinant is widely encountered in linear algebra and statistics:

- Gaussian process (kernel methods)
- Determinantal point process
- Volume form (Bayesian computation)

Challenges

- It is often **the most difficult term** to compute in these applications.
- **Memory-wall** (time complexity isn't the only bottleneck)

Outline

I. Large Matrices

- Neural Tangent Kernels
- Arithmetic Precision

II. MEMDET

- Compute exact log-det
- Out-of-core

III. FLODANCE

- Approximate log-det
- Utilize scale law

IIII. Results

- NTK matrices
- Matérn kernel

I. Large Matrices

Example of Extremely Challenging Matrices

Neural Tangent Kernel (NTK)

- Neural network $f_\theta : \mathcal{X} \to \mathbb{R}^d$
- θ: parameters
- $\mathbf{J}_\theta(f_\theta(x))$: Jacobian of f_θ
- NTK is Gramian of $\mathbf{J}_\theta$:

$$\kappa_\theta(x, x') \coloneqq \mathbf{J}_\theta(f_\theta(x))\mathbf{J}_\theta(f_\theta(x'))^\intercal$$

Compute time of NTK (using NVIDIA H100 GPU)

		Compute Time (hrs)		
Dataset	Model	float16	float32	float64
MNIST	MobileNet	6	25	50
CIFAR-10	ResNet9	6	24	70
	ResNet18	14	63	65
	ResNet50	37	177	297
	ResNet101	107	442	1178

Challenges

Challenge I. Forming NTK

- Takes days/months to compute on H100 GPU
- Need large storage (from **Terabytes** to **Exabytes**)
- **Precision loss** when forming Gram matrix
- **double precision** to retain **positive-definiteness**

Challenge II. Computing LogDet

- **Cubic** complexity $\mathcal{O}(m^3)$
- NTK is nearly **singular**
- CIFAR-10: 10% of eigenvalues near zero
- Cannot load on **memory**

Neural Tangent Kernel Sizes

			Matrix Size		
Dataset	Training Set	Classes	float16	float32	float64
CIFAR-10	50,000	10	0.5 TB	1.0 TB	2.0 TB
MNIST	60,000	10	0.72 TB	1.5 TB	2.9 TB
SVHN	73,257	10	1.1 TB	2.2 TB	4.2 TB
ImageNet-1k	1,281,167	1000	3,282,778 TB	6,565,556 TB	13,131,111 TB*

* 13.1 exabytes is an order of magnitude larger than CERN's current data storage capacity.

II. MEMDET

Memory-Constrained LogDet Computation

MEMDET

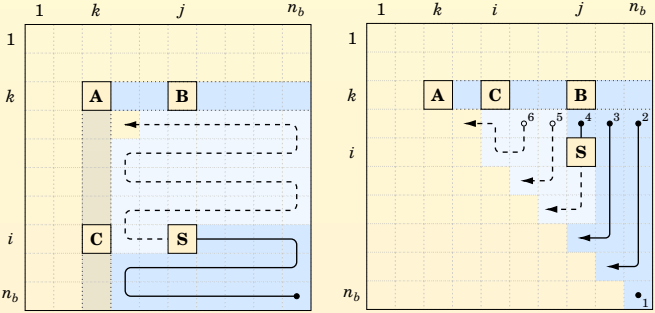
- Out-of-core algorithm
- Can process matrix of any scale
- Eliminates memory wall

Block decompositions:

- LU decomposition: generic matrices

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{11} & \mathbf{0} \\ \mathbf{L}_{21} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{11} & \mathbf{U}_{12} \\ \mathbf{0} & \mathbf{S} \end{bmatrix}$$

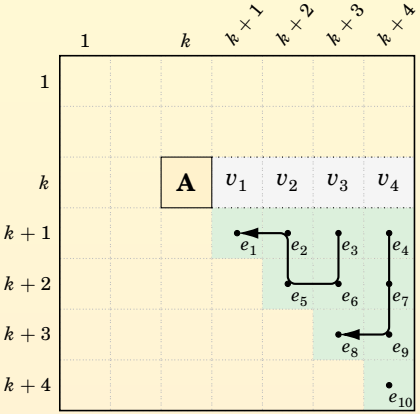
- Repeat decomposition on block **S**.
- LDL: for symmetric matrices
- Cholesky: for symmetric PD matrices

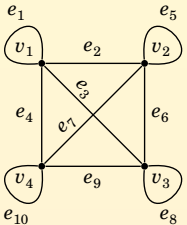


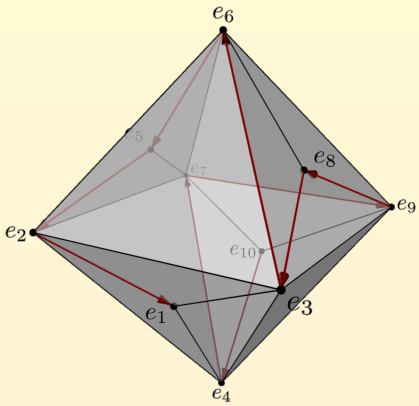
MEMDET Algorithm

- Only **four blocks A**, **B**, **C**, **S** on **memory**
- Blue blocks are written to disk (**scratch space**)
- Efficient **order of processing** of blocks
- *Figure*: LU (left) and LDL/Cholesky decompositions (right).

Optimal Block Ordering

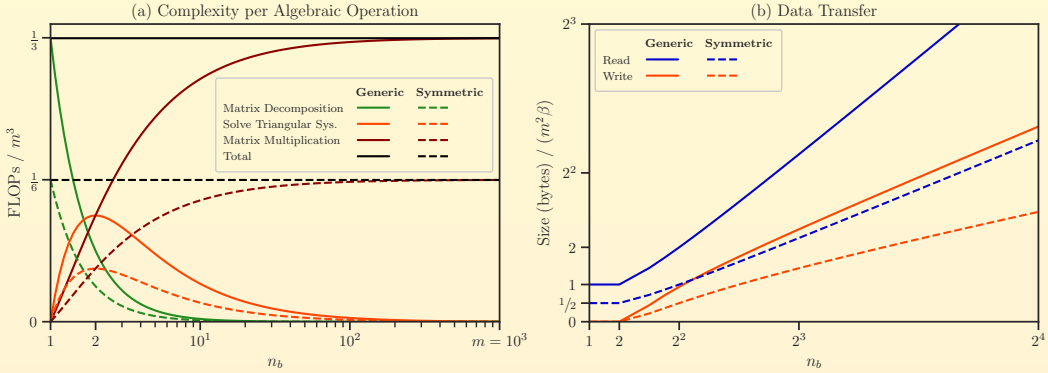






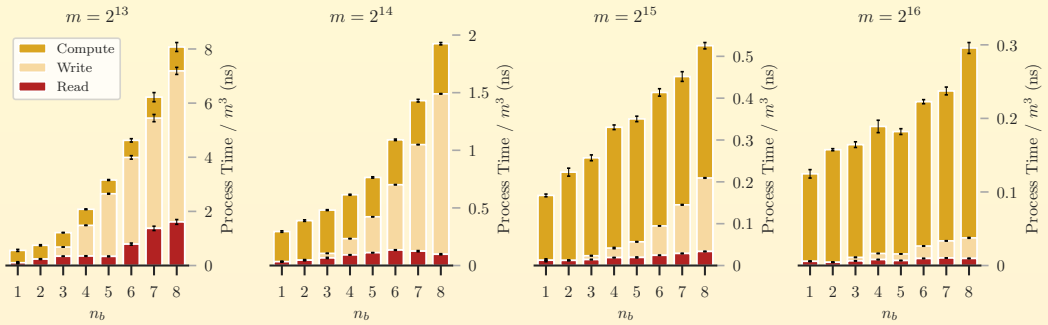
- *Left:* Processing order of blocks for a symmetric matrix at the k-th hierarchical step.
- Two memory blocks are selected from the set $V = \{v_1, v_2, v_3, v_4\}$.
- *Middle:* Complete graph $G(V, E)$.
- *Right:* Line graph $L(G)$, with one possible Hamiltonian path highlighted in red.

Complexity and Data Transfer



- *Left:* Complexity of MEMDET by the increasing number of blocks n_b.
- The **total complexity** (black) remains **constant**.
- Workload transitions from **decompositions** (green) to **solving linear system** (orange) & **matvec** (red).
- *Right:* Data transfer between disk/memory increases with n_b.

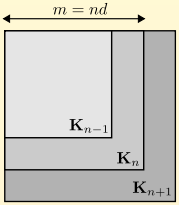
Performance



- Breakdown of MEMDET runtime into computation (ochre) and data transfer times (rea/write).
- At large matrix sizes, **data transfer time** becomes **negligible** compared to compute time.
- **Compute time** is **consistent** across varying number of blocks.

III. FLODANCE

Scale Law



$$\frac{\det(\mathbf{K}_n)}{\det(\mathbf{K}_{n-1})} \sim n^{\nu}$$

- n: num dataset
- d: num classes
- $m = nd$: matrix size

Lemma

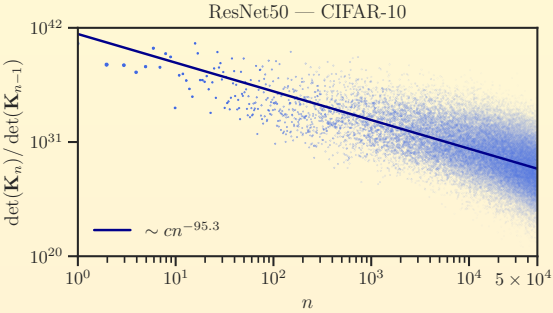
Let $f : \mathcal{X} \to \mathbb{R}^d$ be a zero-mean vector-valued m-dimensional Gaussian process with covariance kernel κ. For each $n \geq 2$, let

$$E(n) := \mathbb{E}[d^{-\frac{1}{2}} \|f(x_n)\|^2 \mid f(x_i) = 0$$

denote the mean-squared error of fitting the f to the zero function using $x_1, \ldots, x_{n-1}$. Then

$$\frac{\mathsf{pdet}(\mathbf{K}_n)}{\mathsf{pdet}(\mathbf{K}_{n-1})} \leq E(n)^d, \quad \forall n > 1,$$

with equality if $d = 1$.



- NTK of ResNet50 on CIFAR-10
- Number of classes: $d = 10$
- Dataset images: $n = 50\text{K}$
- Matrix size: $m = 500\text{K}$.

LogDet as Stochastic Process

Proposition

Let $L_n := \frac{1}{n}\text{logdet}(\mathbf{K}_n)$. *Then*

$$\hat{L}_n \approx L_1 + \left(1 - \frac{1}{n}\right) c_0 - \nu \frac{\log(n!)}{n}$$

- *Law of large numbers (LLN):*

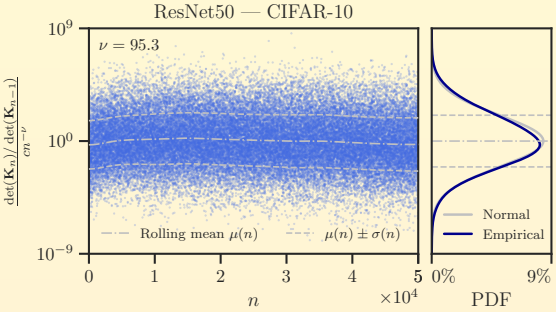
$$L_n = \hat{L}_n + o_p(1).$$

- *Central limit theorem (CLT):*

$$\frac{n}{\sqrt{n-1}}(L_n - \hat{L}_n) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2).$$

Algorithm:

- **Fit** $\hat{L}_n$ on submatrices $n = 1, \ldots, n_s \ll n$
- (Linear regression on parameters c_0, ν)
- **Extrapolate** to larger $n \gg n_s$

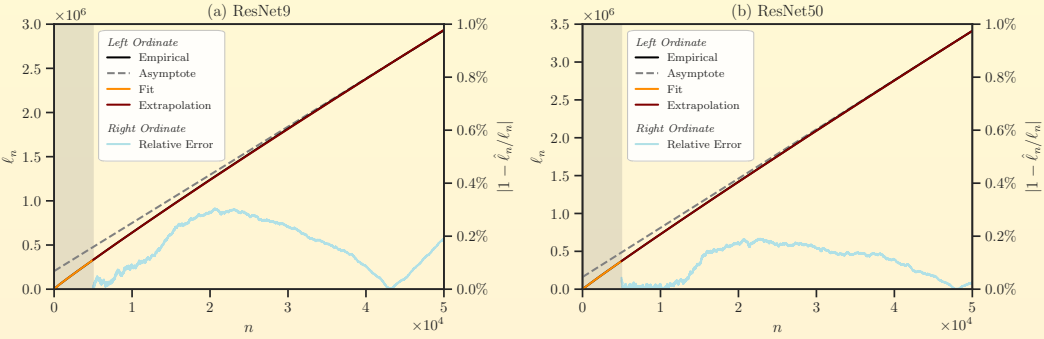


Assumptions:

- Stochastic process: $\dfrac{\det(\mathbf{K}_n) \,/\, \det(\mathbf{K}_{n-1})}{cn^{\nu}}$
- **Stationary** logarithmic process
- **Ergodic process**

IIII. Results

Estimating Log-Det — NTK Matrix

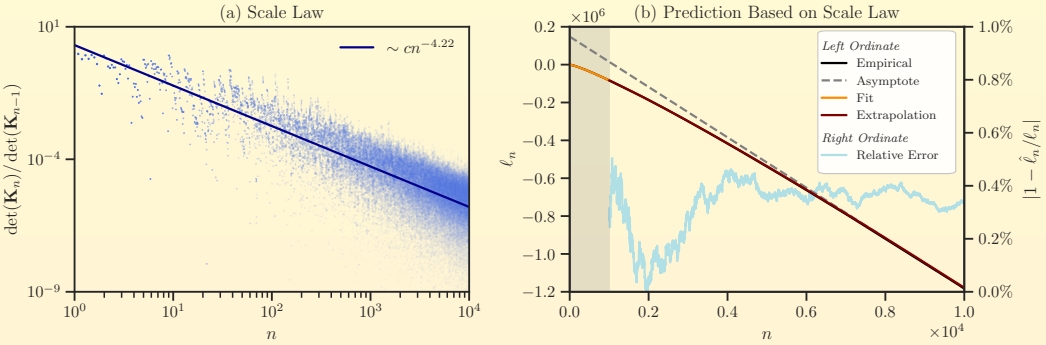


- Full CIFAR-10 data with all $n = 50$K images
- Matrix size $m = 500{,}000$ dense matrix, **double precision**, **2TB** size.
- **Fit:** on 10% of total matrix size (shaded gray region, yellow curve)
- **Extrapolation:** in much larger interval (red curve)
- Error compared to MEMDET: (blue curve right axis in each panel), **0.2%** (left), **0.02%** (right).

Comparison of Methods

Method			Rel.	Est.	Wall
Name	Settings	TFLOPs	Error	Cost	Time
SLQ	$l = 100, s = 104$	5203	55%	\$83	1.8 days
MEMDET	LDL, $n_b = 32$	41,667	**0%**	\$601	13.8 days
FLODANCE	$n_s = 500, \quad q = 0$	**0.04**	4%	\$0.04	1 min
FLODANCE	$n_s = 5000, q = 4$	41.7	**0.02%**	\$4	1.5 hr

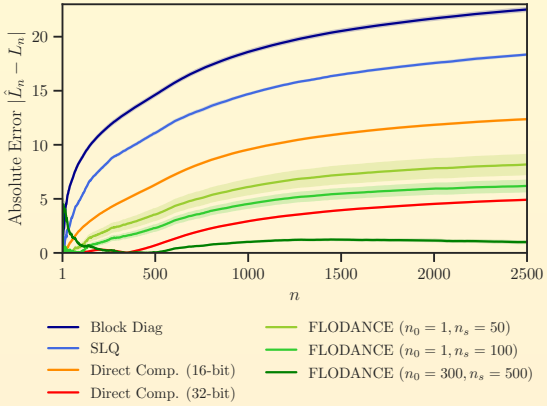
- **Largest NTK formation** and **exact logdet computation** to our knowledge
- ResNet50, full CIFAR-10 with all $n = 50$K images
- Matrix size $m = 500{,}000$ dense matrix, **double precision**, **2TB** size.
- MEMDET computes the **exact** log-determinant, serves as **benchmark**.
- Costs and wall time are based on an NVIDIA H100 GPU (\$2/hour).
- Wall time include NTK formation.



- Gaussian process with a 10-dimensional output using Matérn kernel
- Data points $n = 10\text{K}$
- Covariance matrix of size $m = 100{,}000$
- **Fit:** on 10% of total matrix size (shaded gray region, yellow curve)
- **Extrapolation:** in much larger interval (red curve)
- Error compared to MEMDET: (blue curve right axis) **0.4%**

Comparison of Methods

Method	Approach
MEMDET	Direct factorization
FLODANCE	Submatrix extrapolation
SLQ	Stochastic trace estimation
Pseudo NTK	Cross-class block reduction
Block Diagonal	Class-wise block approx.



Experiment:

- ResNet9 with CIFAR-10
- Smaller matrices to compare with other methods
- Uncertainty quantification: submatrix samples
- Shaded region: standard deviation
- Benchamrk: MEMDET in double precision

Results:

- FLODANCE out performs other methods
- FLODANCE comparable to 32-bit exact method

Reference

Ameli, S., van der Heide, C., Hodgkinson, L., Roosta, F., Mahoney, M.W., (2025). Determinant Estimation under Memory Constraints and Neural Scaling Laws, *The 42nd International Conference on Machine Learning*.

Related Work

Ameli, S., van der Heide, C., Hodgkinson, L., Mahoney, M.W., (2025). Spectral Estimation with Free Decompression. *arXiv: 2506.11994*

Software

Package	Documentation	Install	Implements
detkit	`ameli.github.io/detkit`	`pip install detkit`	MEMDET FLODANCE
imate	`ameli.github.io/imate`	`pip install imate`	SLQ
freealg	`ameli.github.io/freealg`	`pip install freealg`	(Related work)

Overview

Motivations:

- WeightWatcher, Weight Diagnostics for Analyzing ML Models
  (with Charles H. Martin)
- Randomized Numerical Linear Algebra for Modern ML
  (with Michal Derezinski)

Some Theory:

- RMT for NNs: Linear to Nonlinear; Shallow to Deep; etc.
  (with Zhenyu Liao)

Applications:

- Models of Heavy-Tailed Mechanistic Universality
  (with Zhichao Wang and Liam Hodgkinson)
- Spectral Estimation with Free Decompression
  (with Siavash Ameli, Chris van der Heide, and Liam Hodgkinson)
- Determinant Estimation under Memory Constraints and Neural Scaling Laws
  (with S. Ameli, C. van der Heide, L. Hodgkinson, and F. Roosta)