From SLT4AI to AI4SLT:

Generalization under scaling, Infrastructure in Lean 4

Fanghui Liu

fanghui.liu@sjtu.edu.cn

Institute of Natural Sciences, School of Mathematical Sciences
Shanghai Jiao Tong University (SJTU)

at MLSTAT 2026

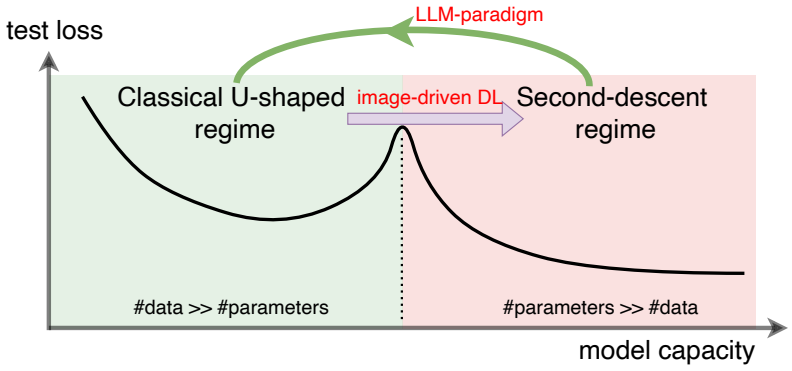
Outline

- SLT4AI: generalization under scaling via norm-based capacities
  - How do generalization curves behave under data, **norm-based capacity**?
  - Which function class can be **efficiently** learned by neural networks?

Outline

- SLT4AI: generalization under scaling via norm-based capacities
  - How do generalization curves behave under data, **norm-based capacity**?
  - Which function class can be **efficiently** learned by neural networks?
- AI4SLT: empirical process in Lean 4 (Infrastructure)
  - Concentration inequality toolbox, Dudley's entropy integral
  - Application to least-squares
  - Human-AI collaborative systems

Learning paradigm in the past twenty years



- U-shaped curve (Vapnik, 1995; Hastie et al., 2009)
- Double descent (Belkin et al., 2019)
- Scaling law (Kaplan et al., 2020)

Model capacity: Norm-based capacity

(Bartlett, 1998)

"The size of the weights is more important than the size of the network!"

Model capacity: Norm-based capacity

(Bartlett, 1998)

"The size of the weights is more important than the size of the network!"

- Min-norm solution (Neyshabur et al., 2015; Savarese et al., 2019; Hastie et al., 2022)
- Stable approximation...
- Applications: pruning, MoE, layerNorm, RMS Norm, EoS, mu-transfer, Muon...

Model capacity: Norm-based capacity

(Bartlett, 1998)

"The size of the weights is more important than the size of the network!"

- Min-norm solution (Neyshabur et al., 2015; Savarese et al., 2019; Hastie et al., 2022)
- Stable approximation...
- Applications: pruning, MoE, layerNorm, RMS Norm, EoS, mu-transfer, Muon...

How do generalization curves behave under norm-based model capacity?

Model capacity: Norm-based capacity

(Bartlett, 1998)

"The size of the weights is more important than the size of the network!"

- Min-norm solution (Neyshabur et al., 2015; Savarese et al., 2019; Hastie et al., 2022)
- Stable approximation...
- Applications: pruning, MoE, layerNorm, RMS Norm, EoS, mu-transfer, Muon...

How do generalization curves behave under norm-based model capacity?

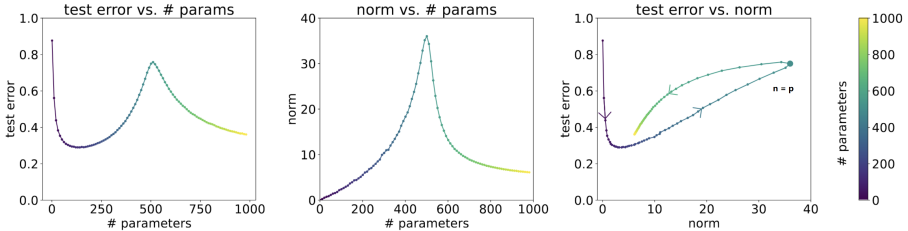
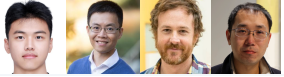
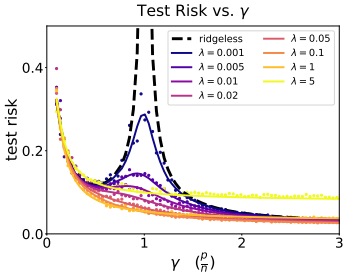


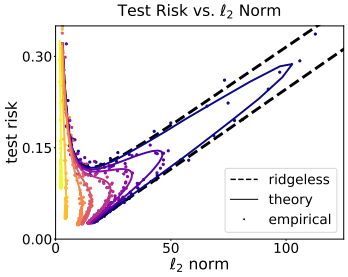
Figure 1: Stanford CS229 lecture notes (Ng and Ma, 2023, Figure 8.12).

The φ curve (Wang, Chen, Rosasco, Liu, NeurIPS'25)

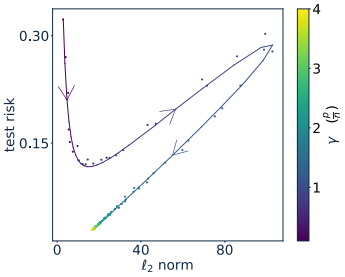




(a) Test Risk vs. γ



(b) Test Risk vs. norm

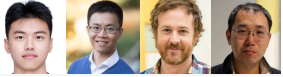


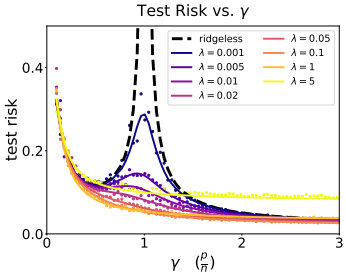
(c) $\lambda = 0.001$

- $\gamma := p/n$, p : model size (width), n : data size

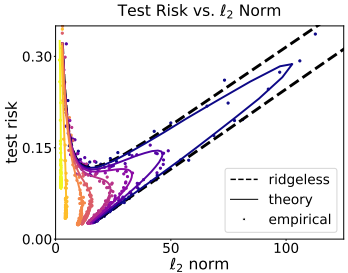
p ($n = 100$)	10	50	100	150	200	250	300	350	400
Test Risk	0.32	0.12	0.29	0.08	0.05	0.04	0.03	0.03	0.03
Norm	2.93	14.66	102.89	35.26	24.76	20.67	18.63	17.47	16.69

The φ curve (Wang, Chen, Rosasco, Liu, NeurIPS'25)

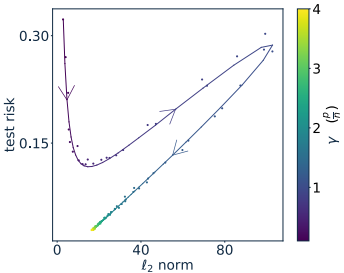




(a) Test Risk vs. γ



(b) Test Risk vs. norm

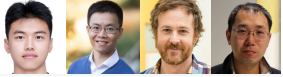


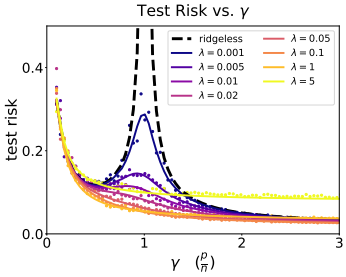
(c) $\lambda = 0.001$

- $\gamma := p/n$, p : model size (width), n : data size

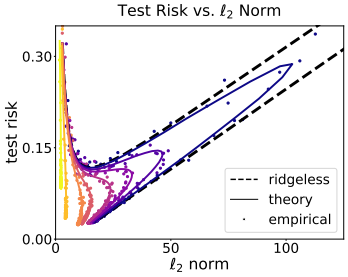
- Phase transition exists but double descent does not exist
- More close to **U-shaped** instead of double descent: A φ paradigm

The φ curve (Wang, Chen, Rosasco, Liu, NeurIPS'25)

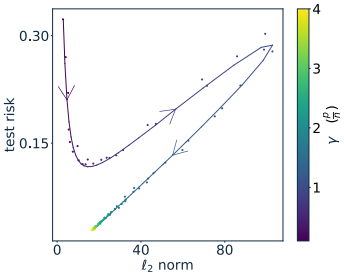




(a) Test Risk vs. γ



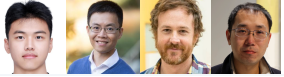
(b) Test Risk vs. norm

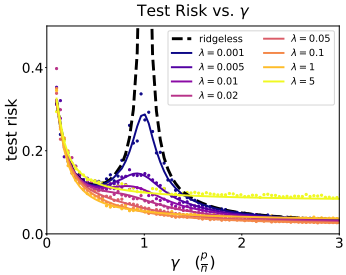


(c) $\lambda = 0.001$

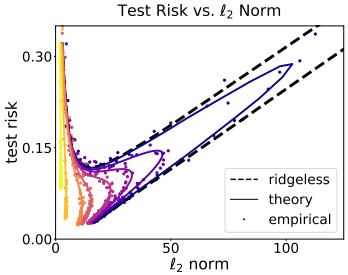
- Reshape scaling-law:
  test loss $=$ A $\times$ Data Size^{-a} $+$ B $\times$ **Model Size**$^{-b}$ $+$ C with $a, b > 0$

The φ curve (Wang, Chen, Rosasco, Liu, NeurIPS'25)

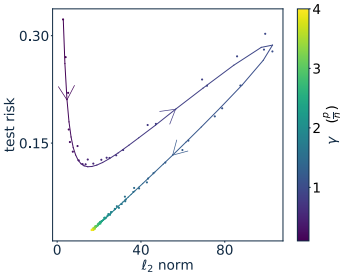




(a) Test Risk vs. γ



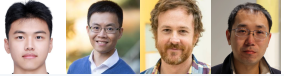
(b) Test Risk vs. norm

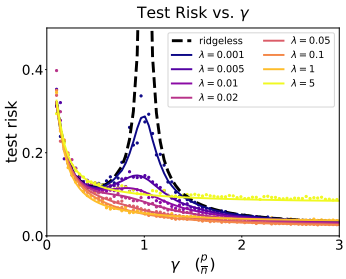


(c) $\lambda = 0.001$

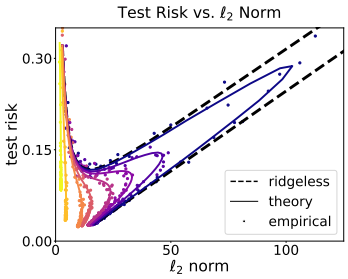
- Reshape scaling-law:
  test loss $=$ A $\times$ Data Size^{-a} $+$ B $\times$ **Model Size**$^{-b}$ $+$ C with $a, b > 0$
  test loss $=$ A $\times$ Data Size^{-a} $\times$ **Norm Capacity**$^{-b}$ with $a > 0$ and $b \in \mathbb{R}$

The φ curve (Wang, Chen, Rosasco, Liu, NeurIPS'25)

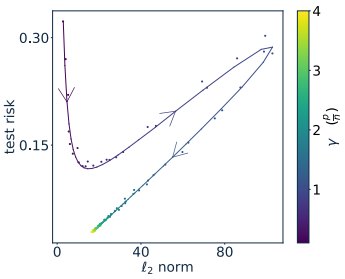




(a) Test Risk vs. γ



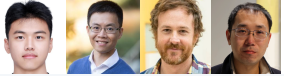
(b) Test Risk vs. norm

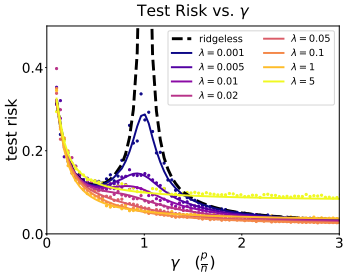


(c) $\lambda = 0.001$

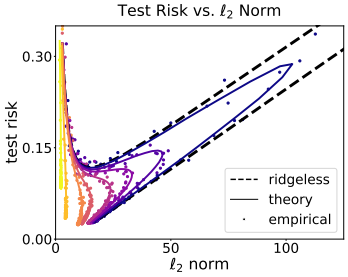
- Reshape scaling-law:
  test loss $=$ A $\times$ Data Size^{-a} $+$ B $\times$ **Model Size**$^{-b}$ $+$ C with $a, b > 0$
  test loss $=$ A $\times$ Data Size^{-a} $\times$ **Norm Capacity**$^{-b}$ with $a > 0$ and $b \in \mathbb{R}$
- Control norm by regularization: L-curve (Hansen, 1992)

The φ curve (Wang, Chen, Rosasco, Liu, NeurIPS'25)

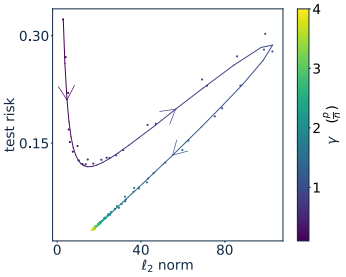




(a) Test Risk vs. γ



(b) Test Risk vs. norm



(c) $\lambda = 0.001$

- Reshape scaling-law:
  test loss $=$ A $\times$ Data Size^{-a} $+$ B $\times$ **Model Size**$^{-b}$ $+$ C with $a, b > 0$
  test loss $=$ A $\times$ Data Size^{-a} $\times$ **Norm Capacity**$^{-b}$ with $a > 0$ and $b \in \mathbb{R}$
- Control norm by regularization: L-curve (Hansen, 1992)
- Characterize generalization: Norm-based capacity ✓ ☺ effective dimension ✗ ☹

Learning with norm-based NNs (Liu, Dadi, Cevher, JMLR'24)

ℓ_1-path norm (Neyshabur et al., 2015; E et al., 2021)

$\|\boldsymbol{\theta}\|_{\mathcal{P}} := \frac{1}{m}\sum_{k=1}^{m} |a_k| \|\boldsymbol{w}_k\|_1$

Learning with norm-based NNs (Liu, Dadi, Cevher, JMLR'24)

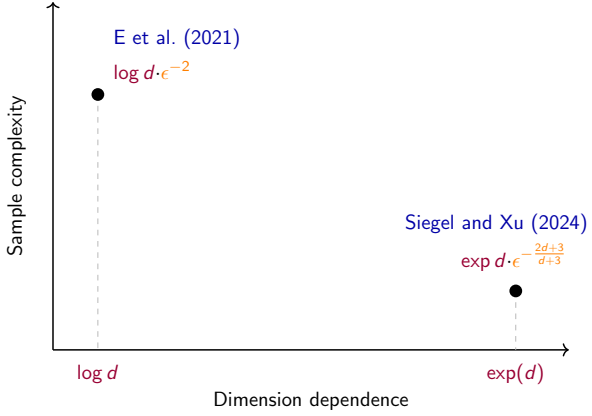
Theorem (Informal, sample complexity of learning $f^\star \in \mathcal{B}$)

- *Kernel methods* require $\Omega(\epsilon^{-d})$ samples.
- *Two-layer (path-norm) neural networks* require $\mathcal{O}_d(\epsilon^{-\frac{2d+2}{d+2}})$ samples. *[smaller than ϵ^{-2}]*

Learning with norm-based NNs (Liu, Dadi, Cevher, JMLR'24)

Theorem (Informal, sample complexity of learning $f^\star \in \mathcal{B}$)

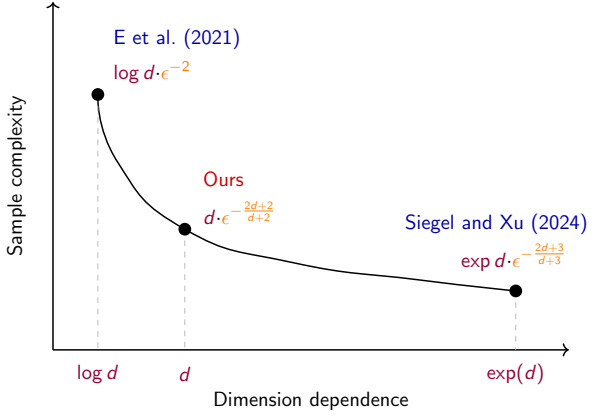
- *Kernel methods* require $\Omega(\epsilon^{-d})$ samples.
- *Two-layer (path-norm) neural networks* require $\mathcal{O}_d(\epsilon^{-\frac{2d+2}{d+2}})$ samples. *[smaller than ϵ^{-2}]*



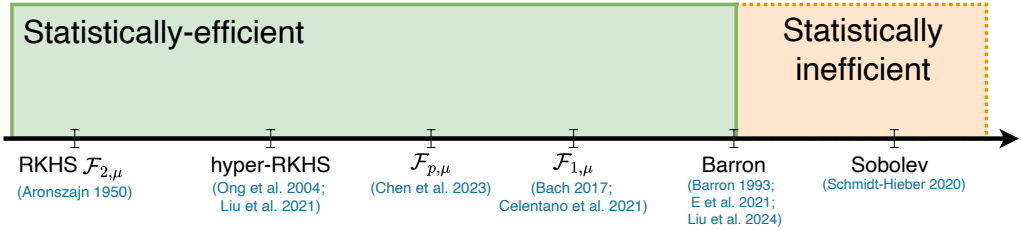
Learning with norm-based NNs (Liu, Dadi, Cevher, JMLR'24)

Theorem (Informal, sample complexity of learning $f^\star \in \mathcal{B}$)

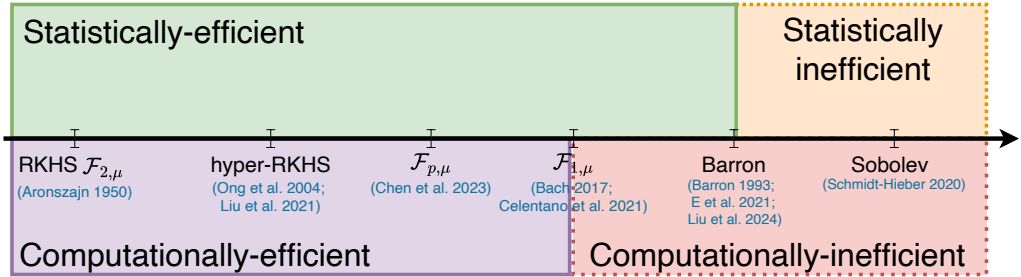
- *Kernel methods* require $\Omega(\epsilon^{-d})$ samples.
- *Two-layer (path-norm) neural networks* require $\mathcal{O}_d(\epsilon^{-\frac{2d+2}{d+2}})$ samples. *[smaller than ϵ^{-2}]*



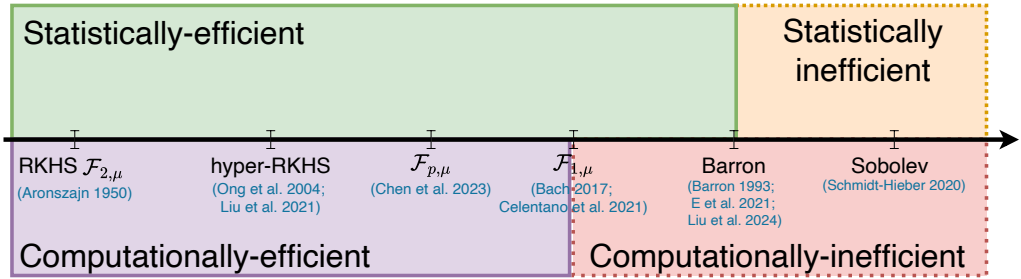
Which function class can be efficiently learned by neural networks



Which function class can be efficiently learned by neural networks



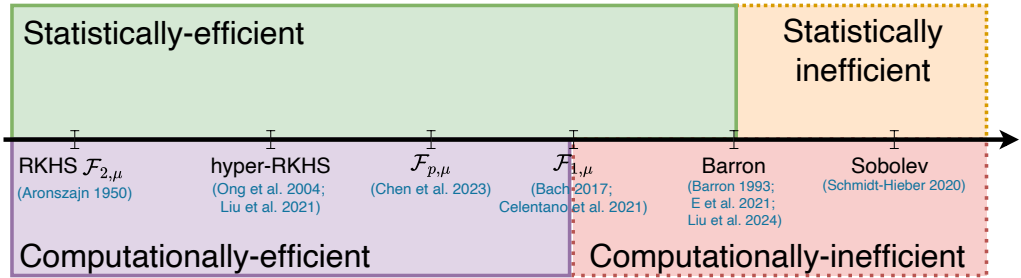
Which function class can be efficiently learned by neural networks



❒ Learning in $\mathcal{F}_1$ space as data-adaptive kernels (Radhakrishnan et al., 2024)

$$\min_{f \in \mathcal{F}_1} \|f - f^*\|^2_{L^2(\mu)} + \lambda \|f\|_{\mathcal{F}_1}$$

Which function class can be efficiently learned by neural networks



- Learning in $\mathcal{F}_1$ space as data-adaptive kernels (Radhakrishnan et al., 2024)

$$\min_{f \in \mathcal{F}_1} \|f - f^*\|_{L^2(\mu)}^2 + \lambda \|f\|_{\mathcal{F}_1}$$

- Single/multi-index models
  - TCS: ReLU neurons (Chen and Narayanan, 2023)
  - Low-dimensional polynomials (Arous et al., 2021; Damian et al., 2022; Abbe et al., 2022; Lee et al., 2024; Dandi et al., 2024; Defilippis et al., 2025)

AI4SLT...

(Industrial) Revolution in Mathematics via LLMs

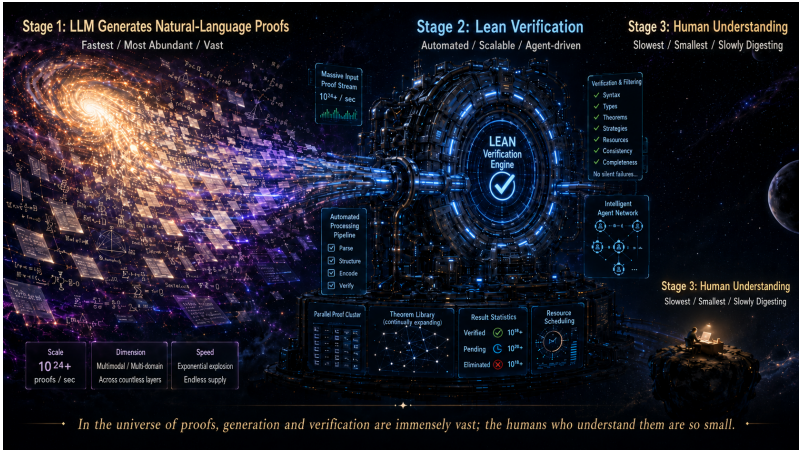
Proof generation (LLMs)
Natural-language ideas, lemmas, candidate arguments

→

Proof validation/-formalization (Lean)
Machine-checked formal proof in LEAN 4

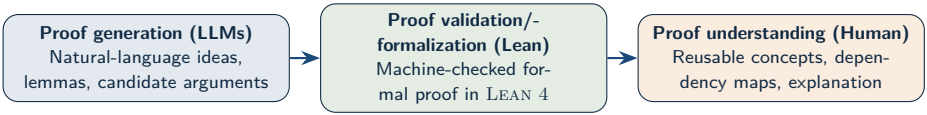
→

Proof understanding (Human)
Reusable concepts, dependency maps, explanation



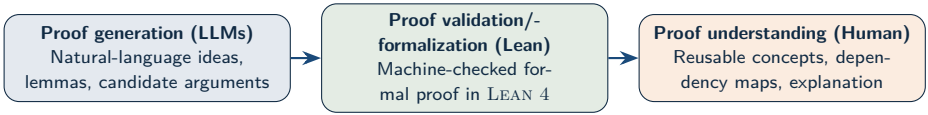
(Generated by GPT5.5)

(Industrial) Revolution in Mathematics via LLMs



- **Validation is bottleneck**
  - plausibility is not correctness;
  - small gaps can invalidate a proof;
  - informal checking does not scale to libraries.
- **Understanding is the final goal**
  - Math aims to advance human understanding
  - not just a machine for generating formal proofs

(Industrial) Revolution in Mathematics via LLMs



- **Validation is bottleneck**
  - plausibility is not correctness;
  - small gaps can invalidate a proof;
  - informal checking does not scale to libraries.
- **Understanding is the final goal**
  - Math aims to advance human understanding
  - not just a machine for generating formal proofs

Question shift

Can AI suggest a proof? $\Longrightarrow$ Can **WE** validate, reuse, and help understanding?

Brief introduction of Lean 4

Lean as a programming language

- We write a mathematical statement and construct a proof as **coding**
- Lean's trusted kernel checks that the proof is valid.

❒ **State it, but not prove it**

Adding zero changes nothing

```
theorem add_zero
    (n : Nat) :
    n + 0 = n := by
  sorry
```

`sorry` as a **placeholder**:

"I have not provided the proof yet."

❒ **State it and prove it**

Verified by Lean

```
theorem add_zero
    (n : Nat) :
    n + 0 = n := by
  simp
```

The tactic `simp` as a **strategy**:
constructs a proof via Lean's simplification rules.

Brief introduction of Lean 4

Not a silver bullet but helps reduce and transfer risk

- **consistency** vs. **complete** [Gödel's incompleteness theorems]
- statement inconsistency

❒ **State it, but not prove it**

Adding zero changes nothing

```
theorem add_zero
    (n : Nat) :
    n + 0 = n := by
  sorry
```

`sorry` as a **placeholder**:

"I have not provided the proof yet."

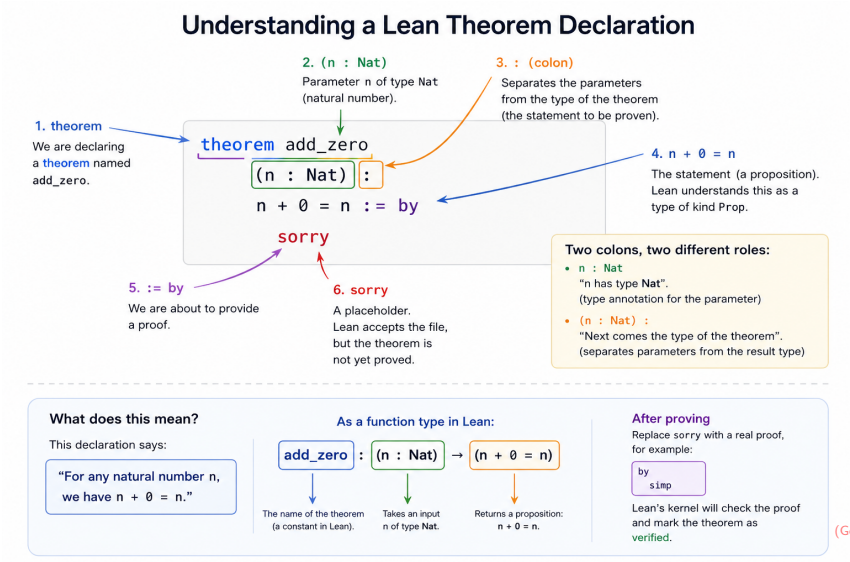
❒ **State it and prove it**

Verified by Lean

```
theorem add_zero
    (n : Nat) :
    n + 0 = n := by
  simp
```

The tactic `simp` as a **strategy**:
constructs a proof via Lean's simplification rules.

https://lean-lang.org/

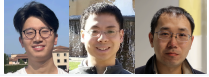


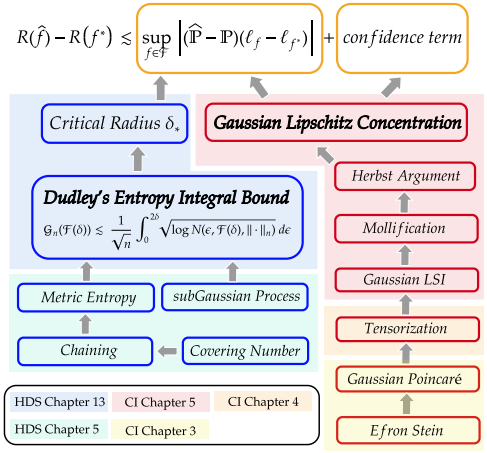
(Generated by GPT5.5)

Natural-language SLT vs. Lean 4 obligations

Mathematical phrase	What Lean asks for
"Take a supremum over a class"	Topology, measurability, boundedness or extended-real formulation
"Apply concentration"	Exact Lipschitz structure, distributional assumptions, integrability
"Use covering numbers"	Metric-space structure, finite nets, cardinal coercions, monotonicity
"Integrate entropy"	$\mathbb{R}$ vs. $\mathbb{R}_{\geq 0} \cup \{\infty\}$ integral interface
"By density"	Approximation sequence, convergence mode, preservation of constants

AI4SLT from empirical process (Zhang, Lee, Liu, ICML'26)





Wainwright (2019) (High-dimensional Statistics, HDS)

Boucheron et al. (2013) (Concentration Inequality, CI)

Concentration

High-probability control of the empirical process fluctuation.

$$\mathbb{P}\{|f(X) - \mathbb{E}f(X)| \geq t\} \leq 2\exp\left(-\frac{t^2}{2L^2}\right).$$

Capacity control

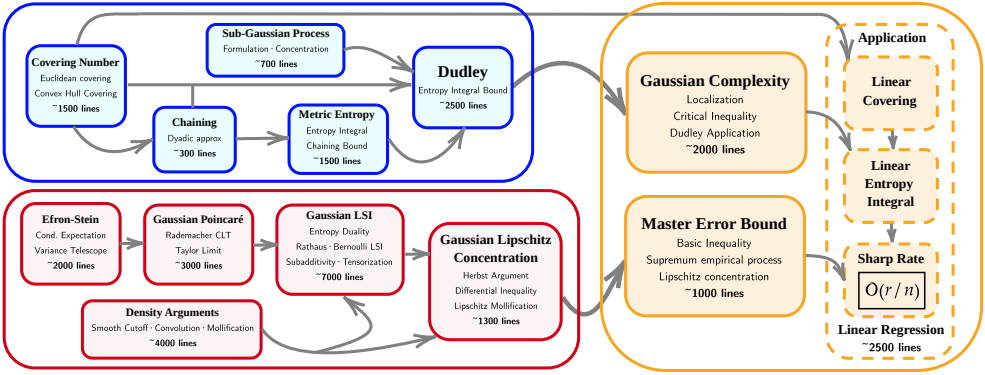
Localize around $f^\star$:

$$\mathcal{F}(\delta) = \{f \in \mathcal{F} : d(f, f^\star) \leq \delta\},$$

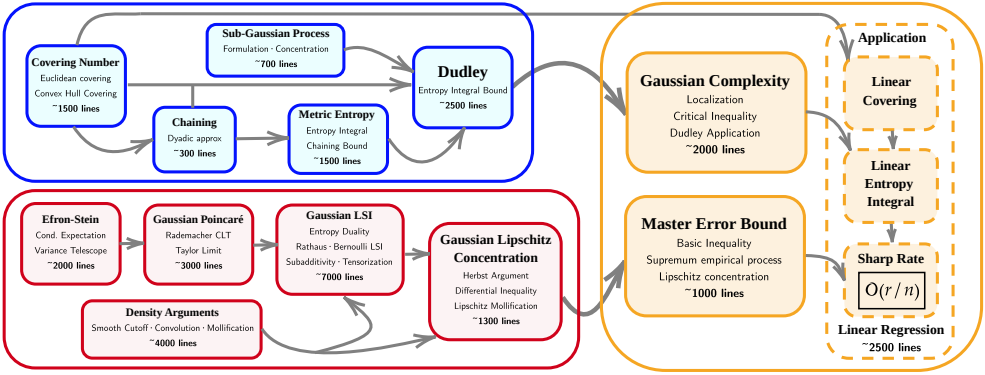
then bound complexity by metric entropy:

$$G_n(\mathcal{F}(\delta)) \lesssim \frac{1}{\sqrt{n}} \int_0^{2\delta} \sqrt{\log \mathcal{N}(\epsilon, \mathcal{F}(\delta), d)}\, \mathrm{d}\epsilon.$$

Dependency graph and deliverables



Dependency graph and deliverables



Main implementation

- Gaussian concentration analysis toolbox.
- Dudley's entropy integral.
- Human-AI workflow.

Scale

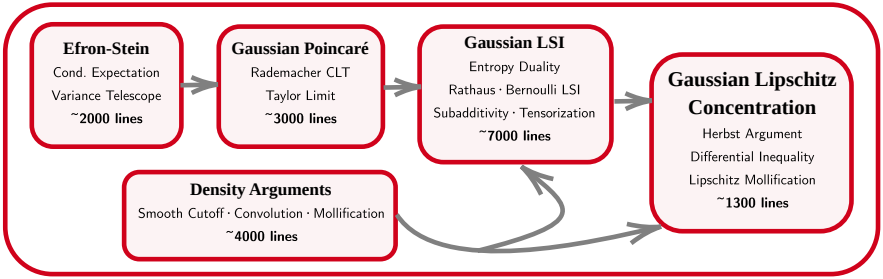
Lean 4 code	$\sim 30{,}000$ lines
Theorems/lemmas	$1000+$

Gaussian concentration: target theorem

Gaussian Lipschitz concentration

Let $X \sim \mathcal{N}(0, I_n)$ and $f : \mathbb{R}^n \to \mathbb{R}$ be L-Lipschitz. Then, for all $t \geq 0$,

$$\mathbb{P}\{f(X) - \mathbb{E}f(X) \geq t\} \leq \exp\left(-\frac{t^2}{2L^2}\right).$$



Example interface: Efron–Stein inequality

Mathematical statement

Let $X = (X_1, \ldots, X_n)$ be a vector of n independent random variables and $Z = f(X)$ square-integrable. With $\mathbb{E}^{(i)}$ denoting conditional expectation over all coordinates except i,

$$\operatorname{Var}(Z) \le \sum_{i=1}^{n} \mathbb{E}\left[\left(Z - \mathbb{E}^{(i)}[Z]\right)^2\right].$$

❒ Leave-one-coordinate conditional expectation

```
noncomputable def condExpExceptCoord
  (i : Fin n) (f : (Fin n -> Omega) -> R) :
  (Fin n -> Omega) -> R :=
  fun x => integral y, f (Function.update x i y) d(mus i)

theorem efronStein
  (f : (Fin n -> Omega) -> R) (hf : MemLp f 2 mus) :
  variance f mus <=
    sum i : Fin n, integral x, (f x - condExpExceptCoord i f x)^2 dmus := by
```

Why is Lean formalization complex?

We want to prove

$$\mathrm{Law}(X^{(i)}) = \mu_1 \otimes \cdots \otimes \mu_n$$

Key Point

- Natural language: *Resampling one independent coordinate preserves the joint distribution*
- Lean: *Prove that a pushforward of a product measure equals the original product measure*

Why is Lean formalization complex?

We want to prove

$$\mathrm{Law}(X^{(i)}) = \mu_1 \otimes \cdots \otimes \mu_n$$

Key Point

- Natural language: *Resampling one independent coordinate preserves the joint distribution*
- Lean: *Prove that a pushforward of a product measure equals the original product measure*

product measures measurable maps pushforwards Fubini's theorem measure rectangles.

Why is Lean formalization complex?

We want to prove

$$\mathrm{Law}(X^{(i)}) = \mu_1 \otimes \cdots \otimes \mu_n$$

Key Point

- Natural language: *Resampling one independent coordinate preserves the joint distribution*
- Lean: *Prove that a pushforward of a product measure equals the original product measure*

product measures measurable maps pushforwards Fubini's theorem measure rectangles.

Density argument (missing in the textbook (Boucheron et al., 2013))

Gaussian LSI on $C_c^\infty \Longrightarrow$ Gaussian LSI on $W^{1,2}(\gamma)$

Gaussian **Lipschitz** concentration on $C_c^\infty \Longrightarrow$ ...

Human–AI collaborative formalization

Structured `TASK.md` protocol

Each task specification contains:

1. also need NL target statement;
2. tactic-level proof plan (for agents);
3. relevant file paths and expected crossref;
4. hard boundaries for what the agent should not rewrite.

Collaborative level

Autonomous	$\sim 15\%$
Sketch-guided	$\sim 30\%$
Specified under NL proofs	$\sim 40\%$
Human-critical	$\sim 15\%$

Human–AI collaborative formalization

Structured `TASK.md` protocol

Each task specification contains:

1. also need NL target statement;
2. tactic-level proof plan (for agents);
3. relevant file paths and expected crossref;
4. hard boundaries for what the agent should not rewrite.

Collaborative level

Autonomous	$\sim 15\%$
Sketch-guided	$\sim 30\%$
Specified under NL proofs	$\sim 40\%$
Human-critical	$\sim 15\%$

Takeaway

The dominant bottleneck is not tactic writing; it is converting compressed mathematical prose into precise, decomposed proof architecture.

Practical lessons for formalization

- **Decompose aggressively.** Target small lemmas that fit within one agent context window.

Practical lessons for formalization

- ❒ **Decompose aggressively.** Target small lemmas that fit within one agent context window.
- ❒ **Specify more than the theorem.** Include local APIs, intended proof route.

Practical lessons for formalization

- **Decompose aggressively.** Target small lemmas that fit within one agent context window.
- **Specify more than the theorem.** Include local APIs, intended proof route.
- **Treat persistent failure as mathematical signal.** Recheck the statement and look for missing assumptions or counterexamples.

Practical lessons for formalization

- **Decompose aggressively.** Target small lemmas that fit within one agent context window.
- **Specify more than the theorem.** Include local APIs, intended proof route.
- **Treat persistent failure as mathematical signal.** Recheck the statement and look for missing assumptions or counterexamples.
- **Make hidden regularity reusable.** Integrability, continuity, nonemptiness, and boundedness hypotheses become reusable theorem contracts.

Practical lessons for formalization

- **Decompose aggressively.** Target small lemmas that fit within one agent context window.
- **Specify more than the theorem.** Include local APIs, intended proof route.
- **Treat persistent failure as mathematical signal.** Recheck the statement and look for missing assumptions or counterexamples.
- **Make hidden regularity reusable.** Integrability, continuity, nonemptiness, and boundedness hypotheses become reusable theorem contracts.
- **To agent: don't frequently change the proof!**

Practical lessons for formalization

- ❒ **Decompose aggressively.** Target small lemmas that fit within one agent context window.
- ❒ **Specify more than the theorem.** Include local APIs, intended proof route.
- ❒ **Treat persistent failure as mathematical signal.** Recheck the statement and look for missing assumptions or counterexamples.
- ❒ **Make hidden regularity reusable.** Integrability, continuity, nonemptiness, and boundedness hypotheses become reusable theorem contracts.
- ❒ **To agent: don't frequently change the proof!**

- Know more details (e.g., implicit assumptions)
- Know which key infrastructure is reused (structure view, multi-scale understanding)
- proof connections (e.g., Gaussian Poincaré by Efron-Stein, Hermite decomposition, OU semigroup)

Go beyond...arXiv: 2606.05400

If Mathlib is “complete”, can **we** autoformalize research-level problems even with proof gap?

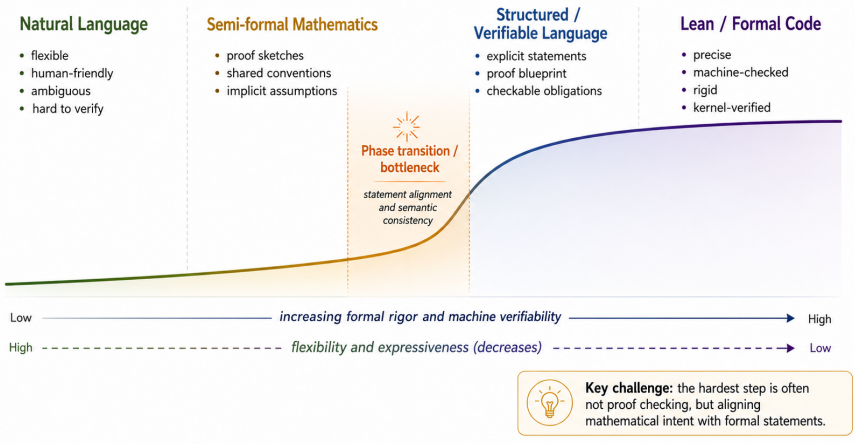
Go beyond...arXiv: 2606.05400

If Mathlib is “complete”, can **we** autoformalize research-level problems even with proof gap?

LeanMarathon: Toward Reliable AI Co-Mathematicians through Long-Horizon Lean Autoformalization

Yuanhe Zhang* Yuekai Sun† Taiji Suzuki‡ Jason D. Lee§ Fanghui Liu¶

Erdős problems (#1051, #1196, #164, #1217) (Barreto et al., 2026; Alexeev et al., 2026)

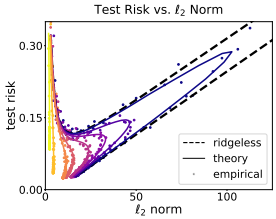


(Generated by GPT5.5)

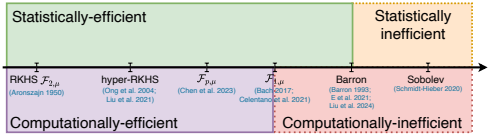
Yuanhe Zhang, Ilja Kuzborskij, Jason D. Lee, Chenlei Leng, Fanghui Liu. DAG-Math: Graph-of-Thought-Guided Mathematical Reasoning in LLMs. ICLR'26.

Takeaway: From SLT4AI to AI4SLT

❒ The φ curve: be aware of model capacity!

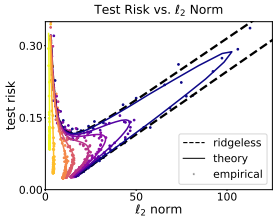


❒ Trade-off between sample complexity and dimension dependence

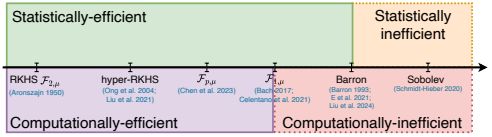


Takeaway: From SLT4AI to AI4SLT

❒ The φ curve: be aware of model capacity!



❒ Trade-off between sample complexity and dimension dependence



AI4SLT: End-to-end reusable infrastructure

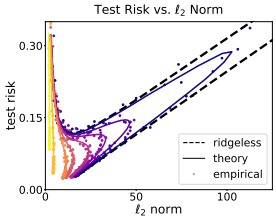
- Verified reusable SLT toolbox;
- Template for human–AI large-scale formalization.

Why it matters

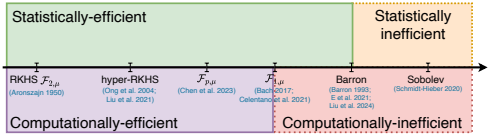
- Machine-checked used for the review system and insights for AI4AI/ML;
- **Multi-scale understanding**.

Takeaway: From SLT4AI to AI4SLT

❒ The φ curve: be aware of model capacity!



❒ Trade-off between sample complexity and dimension dependence



AI4SLT: End-to-end reusable infrastructure

- Verified reusable SLT toolbox;
- Template for human–AI large-scale formalization.

Why it matters

- Machine-checked used for the review system and insights for AI4AI/ML;
- **Multi-scale understanding**.

Q & A

fanghui.liu@sjtu.edu.cn
www.lfhsgre.org

References

Emmanuel Abbe, Enric Boix Adsera, and Theodor Misiakiewicz. The merged-staircase property: a necessary and nearly sufficient condition for sgd learning of sparse functions on two-layer neural networks. In *Conference on Learning Theory*, pages 4782–4887, 2022.

Boris Alexeev, Kevin Barreto, Yanyang Li, Jared Duker Lichtman, Liam Price, Jibran Iqbal Shah, Quanyu Tang, and Terence Tao. Primitive sets and von Mangoldt chains: Erdős Problem #1196 and beyond, 2026. URL https://arxiv.org/abs/2605.00301.

Gerard Ben Arous, Reza Gheissari, and Aukosh Jagannath. Online stochastic gradient descent on non-convex losses from high-dimensional inference. *Journal of Machine Learning Research*, 22(106):1–51, 2021.

References ii

Francis Bach. High-dimensional analysis of double descent for linear regression with random projections. *SIAM Journal on Mathematics of Data Science*, 6(1):26–50, 2024.

Kevin Barreto, Jiwon Kang, Sang-hyun Kim, Vjekoslav Kovač, and Shengtong Zhang. Irrationality of rapidly converging series: a problem of Erdős and Graham, 2026. URL https://arxiv.org/abs/2601.21442.

Peter Bartlett. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Transactions on Information Theory*, 44(2):525–536, 1998.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *the National Academy of Sciences*, 116(32):15849–15854, 2019.

References iii

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 02 2013. URL https://doi.org/10.1093/acprof:oso/9780199535255.001.0001.

Sitan Chen and Shyam Narayanan. A faster and simpler algorithm for learning shallow networks. *arXiv preprint arXiv:2307.12496*, 2023.

Chen Cheng and Andrea Montanari. Dimension free ridge regression. *The Annals of Statistics*, 52(6):2879–2912, 2024.

Alexandru Damian, Jason Lee, and Mahdi Soltanolkotabi. Neural networks can learn representations with gradient descent. In *Conference on Learning Theory*, pages 5413–5452. PMLR, 2022.

References iv

Yatin Dandi, Emanuele Troiani, Luca Arnaboldi, Luca Pesce, Lenka Zdeborova, and Florent Krzakala. The benefits of reusing batches for gradient descent in two-layer networks: Breaking the curse of information and leap exponents. In *Forty-first International Conference on Machine Learning*, 2024.

Leonardo Defilippis, Bruno Loureiro, and Theodor Misiakiewicz. Dimension-free deterministic equivalents for random feature regression. In *Advances in Neural Information Processing Systems*, 2024.

Leonardo Defilippis, Yatin Dandi, Pierre Mergny, Florent Krzakala, and Bruno Loureiro. Optimal spectral transitions in high-dimensional multi-index models. In *Advances in Neural Information Processing Systems*, volume 38, pages 174966–175002, 2025.

Weinan E, Chao Ma, and Lei Wu. The barron space and the flow-induced function spaces for neural network models. *Constructive Approximation*, pages 1–38, 2021.

References v

Per Christian Hansen. Analysis of discrete ill-posed problems by means of the l-curve. *SIAM Review*, 34(4):561–580, 1992.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *Annals of Statistics*, 50(2):949–986, 2022.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*, 2020.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

References vi

Jason D Lee, Kazusato Oko, Taiji Suzuki, and Denny Wu. Neural network learns low-dimensional polynomials with sgd near the information-theoretic limit. *arXiv preprint arXiv:2406.01581*, 2024.

Fanghui Liu, Xiaolin Huang, Yudong Chen, and Johan AK Suykens. Random features for kernel approximation: A survey on algorithms, theory, and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7128–7148, 2021.

Theodor Misiakiewicz and Basil Saeed. A non-asymptotic theory of kernel ridge regression: deterministic equivalents, test error, and gcv estimator. *arXiv preprint arXiv:2403.08938*, 2024.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. Norm-based capacity control in neural networks. In *Conference on Learning Theory*, pages 1376–1401. PMLR, 2015.

Andrew Ng and Tengyu Ma. CS229 lecture notes. 2023. URL https://cs229.stanford.edu/main_notes.pdf.

References vii

Adityanarayanan Radhakrishnan, Daniel Beaglehole, Parthe Pandit, and Mikhail Belkin. Mechanism for feature learning in neural networks and backpropagation-free machine learning models. *Science*, 383(6690):1461–1467, 2024.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pages 1177–1184, 2007.

Pedro Savarese, Itay Evron, Daniel Soudry, and Nathan Srebro. How do infinite width bounded norm networks look in function space? In *Conference on Learning Theory*, pages 2667–2690. PMLR, 2019.

Aad W Van Der Vaart, Adrianus Willem van der Vaart, Aad van der Vaart, and Jon Wellner. *Weak convergence and empirical processes: with applications to statistics*. Springer Science & Business Media, 1996.

Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

References viii

Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Denny Wu and Ji Xu. On the optimal weighted ℓ_2 regularization in overparameterized linear regression. In *Advances in Neural Information Processing Systems*, pages 10112–10123, 2020.

Background: Random features ridge regression

$$f_p(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{i=1}^{p} a_i \phi(\boldsymbol{x}, \boldsymbol{w}_i), \quad \boldsymbol{\theta} := \{(a_i, \boldsymbol{w}_i)\}_{i=1}^{p}$$

- $\phi : \mathcal{X} \times \mathcal{W} \to \mathbb{R}$, e.g., ReLU: $\phi(\boldsymbol{x}, \boldsymbol{w}) = \max(\langle \boldsymbol{x}, \boldsymbol{w} \rangle, 0)$

Background: Random features ridge regression

$$f_p(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{i=1}^{p} a_i \phi(\boldsymbol{x}, \boldsymbol{w}_i), \quad \boldsymbol{\theta} := \{(a_i, \boldsymbol{w}_i)\}_{i=1}^{p}$$

- $\phi : \mathcal{X} \times \mathcal{W} \to \mathbb{R}$, e.g., ReLU: $\phi(\boldsymbol{x}, \boldsymbol{w}) = \max(\langle \boldsymbol{x}, \boldsymbol{w} \rangle, 0)$
- Random features models (RFMs) (Rahimi and Recht, 2007; Liu et al., 2021):
  - $\{\boldsymbol{w}_i\}_{i=1}^{p} \overset{iid}{\sim} \mu$ for a given $\mu \in \mathcal{P}(\mathcal{W})$
  - only train the second layer

Background: Random features ridge regression

$$f_p(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{i=1}^{p} a_i \phi(\boldsymbol{x}, \boldsymbol{w}_i), \quad \boldsymbol{\theta} := \{(a_i, \boldsymbol{w}_i)\}_{i=1}^{p}$$

- $\phi : \mathcal{X} \times \mathcal{W} \to \mathbb{R}$, e.g., ReLU: $\phi(\boldsymbol{x}, \boldsymbol{w}) = \max(\langle \boldsymbol{x}, \boldsymbol{w} \rangle, 0)$
- Random features models (RFMs) (Rahimi and Recht, 2007; Liu et al., 2021):
  - $\{\boldsymbol{w}_i\}_{i=1}^{p} \overset{iid}{\sim} \mu$ for a given $\mu \in \mathcal{P}(\mathcal{W})$
  - only train the second layer

$$\hat{\boldsymbol{a}} := \operatorname{argmin}_{\boldsymbol{a} \in \mathbb{R}^p} \left\{ \sum_{i=1}^{n} \left(y_i - f(\boldsymbol{x}_i; \boldsymbol{a})\right)^2 + \lambda \|\boldsymbol{a}\|_2^2 \right\} = \left(\boldsymbol{Z}^\top \boldsymbol{Z} + \lambda \boldsymbol{I}_p\right)^{-1} \boldsymbol{Z}^\top \boldsymbol{y}\,.$$

- $\boldsymbol{Z} \in \mathbb{R}^{n \times p}$ with $[\boldsymbol{Z}]_{ij} = \frac{1}{\sqrt{m}} \phi(\boldsymbol{x}_i, \boldsymbol{w}_j)$.

Background: Random features ridge regression

$$f_p(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{i=1}^{p} a_i \phi(\boldsymbol{x}, \boldsymbol{w}_i), \quad \boldsymbol{\theta} := \{(a_i, \boldsymbol{w}_i)\}_{i=1}^{p}$$

- $\phi : \mathcal{X} \times \mathcal{W} \to \mathbb{R}$, e.g., ReLU: $\phi(\boldsymbol{x}, \boldsymbol{w}) = \max(\langle \boldsymbol{x}, \boldsymbol{w} \rangle, 0)$
- Random features models (RFMs) (Rahimi and Recht, 2007; Liu et al., 2021):
  - $\{\boldsymbol{w}_i\}_{i=1}^{p} \overset{iid}{\sim} \mu$ for a given $\mu \in \mathcal{P}(\mathcal{W})$
  - only train the second layer

$$\hat{\boldsymbol{a}} := \operatorname{argmin}_{\boldsymbol{a} \in \mathbb{R}^p} \left\{ \sum_{i=1}^{n} \left(y_i - f(\boldsymbol{x}_i; \boldsymbol{a})\right)^2 + \lambda \|\boldsymbol{a}\|_2^2 \right\} = \left(\boldsymbol{Z}^\top \boldsymbol{Z} + \lambda \boldsymbol{I}_p\right)^{-1} \boldsymbol{Z}^\top \boldsymbol{y}\,.$$

- $\boldsymbol{Z} \in \mathbb{R}^{n \times p}$ with $[\boldsymbol{Z}]_{ij} = \frac{1}{\sqrt{m}} \phi(\boldsymbol{x}_i, \boldsymbol{w}_j)$.
  - Norm over the first-layer (untrained) $\|\boldsymbol{W}\|_{\mathrm{F}}$
  - Norm over the second-layer $\|\hat{\boldsymbol{a}}\|_2^2$

Background: Test risk of random features model

- A **compact** integral operator $\mathbb{T} : L^2(\rho_X) \to L^2(\mu_{\mathcal{W}})$ for any $f \in L^2(\rho_X)$ (Defilippis et al., 2024)

$$(\mathbb{T}f)(\boldsymbol{w}) := \int_{\mathbb{R}^d} \phi(\boldsymbol{x}, \boldsymbol{w}) f(\boldsymbol{x}) \mathrm{d}\rho(\boldsymbol{x}) \,, \quad \mathbb{T} = \sum_{k=1}^{\infty} \xi_k \psi_k \varphi_k^* \,.$$

Background: Test risk of random features model

- A **compact** integral operator $\mathbb{T} : L^2(\rho_X) \to L^2(\mu_{\mathcal{W}})$ for any $f \in L^2(\rho_X)$ (Defilippis et al., 2024)

$$(\mathbb{T}f)(\boldsymbol{w}) := \int_{\mathbb{R}^d} \phi(\boldsymbol{x}, \boldsymbol{w}) f(\boldsymbol{x}) \mathrm{d}\rho(\boldsymbol{x}) \,, \quad \mathbb{T} = \sum_{k=1}^{\infty} \xi_k \psi_k \varphi_k^* \,.$$

- Covariate feature matrix $\boldsymbol{G} := [\boldsymbol{g}_1, \ldots, \boldsymbol{g}_n]^\top \in \mathbb{R}^{n \times \infty}$ with $\boldsymbol{g}_i := (\psi_k(\boldsymbol{x}_i))_{k \geq 1}$
- Weight feature matrix $\boldsymbol{H} := [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_p]^\top \in \mathbb{R}^{p \times \infty}$ with $\boldsymbol{h}_j := (\xi_k \varphi_k(\boldsymbol{w}_j))_{k \geq 1}$

Background: Test risk of random features model

- A **compact** integral operator $\mathbb{T} : L^2(\rho_X) \to L^2(\mu_{\mathcal{W}})$ for any $f \in L^2(\rho_X)$ (Defilippis et al., 2024)

$$(\mathbb{T}f)(\boldsymbol{w}) := \int_{\mathbb{R}^d} \phi(\boldsymbol{x}, \boldsymbol{w}) f(\boldsymbol{x}) \mathrm{d}\rho(\boldsymbol{x}) \,, \quad \mathbb{T} = \sum_{k=1}^{\infty} \xi_k \psi_k \varphi_k^* \,.$$

- Covariate feature matrix $\boldsymbol{G} := [\boldsymbol{g}_1, \ldots, \boldsymbol{g}_n]^\top \in \mathbb{R}^{n \times \infty}$ with $\boldsymbol{g}_i := (\psi_k(\boldsymbol{x}_i))_{k \geq 1}$
- Weight feature matrix $\boldsymbol{H} := [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_p]^\top \in \mathbb{R}^{p \times \infty}$ with $\boldsymbol{h}_j := (\xi_k \varphi_k(\boldsymbol{w}_j))_{k \geq 1}$
- target function: $f_*(\boldsymbol{x}) = \sum_{k \geq 1} \boldsymbol{\theta}_{*,k} \psi_k(\boldsymbol{x})$

Background: Test risk of random features model

- A **compact** integral operator $\mathbb{T} : L^2(\rho_X) \to L^2(\mu_{\mathcal{W}})$ for any $f \in L^2(\rho_X)$ (Defilippis et al., 2024)

$$(\mathbb{T}f)(\boldsymbol{w}) := \int_{\mathbb{R}^d} \phi(\boldsymbol{x}, \boldsymbol{w}) f(\boldsymbol{x}) \mathrm{d}\rho(\boldsymbol{x}) \,, \quad \mathbb{T} = \sum_{k=1}^{\infty} \xi_k \psi_k \varphi_k^* \,.$$

- Covariate feature matrix $\boldsymbol{G} := [\boldsymbol{g}_1, \ldots, \boldsymbol{g}_n]^\top \in \mathbb{R}^{n \times \infty}$ with $\boldsymbol{g}_i := (\psi_k(\boldsymbol{x}_i))_{k \geq 1}$
- Weight feature matrix $\boldsymbol{H} := [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_p]^\top \in \mathbb{R}^{p \times \infty}$ with $\boldsymbol{h}_j := (\xi_k \varphi_k(\boldsymbol{w}_j))_{k \geq 1}$
- target function: $f_*(\boldsymbol{x}) = \sum_{k \geq 1} \boldsymbol{\theta}_{*,k} \psi_k(\boldsymbol{x})$

$$\mathcal{R}^{\mathtt{RFM}} := \mathbb{E}_\varepsilon \left\| \boldsymbol{\theta}_* - \frac{1}{\sqrt{p}} \boldsymbol{H}^\top \hat{\boldsymbol{a}} \right\|_2^2$$

Control norm by tuning λ: L-curve (Hansen, 1992)

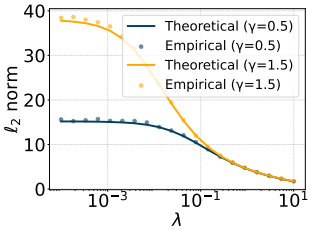
Explicit (model size) *vs.* Implicit (norm)

One-to-one mapping between norm and λ

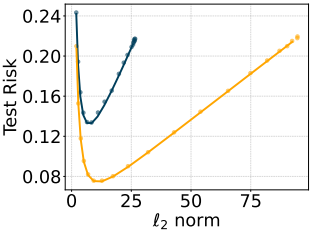
Control norm by tuning λ: L-curve (Hansen, 1992)

Explicit (model size) *vs.* Implicit (norm)

One-to-one mapping between norm and λ



(a) Norm vs. λ (varying λ)



(b) Risk vs. Norm (varying λ)

Control norm by tuning λ: L-curve (Hansen, 1992)

Explicit (model size) *vs.* Implicit (norm)

One-to-one mapping between norm and λ

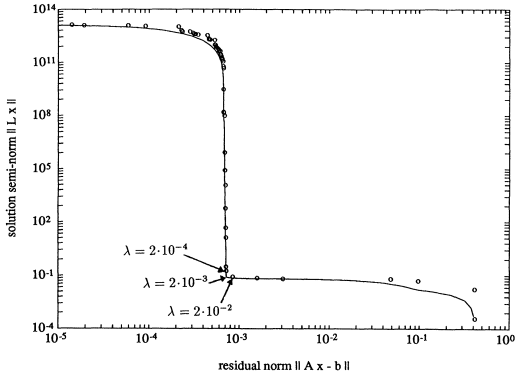


Figure 2: Source from Hansen (1992).

An example of linear regression: Textbook level and beyond

- n i.i.d. samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ with $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$
- $y = \langle \boldsymbol{\beta}_*, \boldsymbol{x} \rangle + \varepsilon$, $\mathbb{E}(\varepsilon) = 0$ and $\mathbb{V}(\varepsilon) = \sigma^2$, covariance matrix $\boldsymbol{\Sigma} = \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top]$
- ridge regression: $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$

An example of linear regression: Textbook level and beyond

- n i.i.d. samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ with $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$
- $y = \langle \boldsymbol{\beta}_*, \boldsymbol{x} \rangle + \varepsilon$, $\mathbb{E}(\varepsilon) = 0$ and $\mathbb{V}(\varepsilon) = \sigma^2$, covariance matrix $\boldsymbol{\Sigma} = \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top]$
- ridge regression: $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$

Target: precise analysis

The expected test risk $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_*\|_{\boldsymbol{\Sigma}}^2$ vs. the norm $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}}\|_2^2$

An example of linear regression: Textbook level and beyond

- n i.i.d. samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ with $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$
- $y = \langle \boldsymbol{\beta}_*, \boldsymbol{x} \rangle + \varepsilon$, $\mathbb{E}(\varepsilon) = 0$ and $\mathbb{V}(\varepsilon) = \sigma^2$, covariance matrix $\boldsymbol{\Sigma} = \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top]$
- ridge regression: $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$

Target: precise analysis

The expected test risk $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_*\|_{\boldsymbol{\Sigma}}^2$ vs. the norm $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}}\|_2^2$

❒ Deterministic equivalence (Cheng and Montanari, 2024; Misiakiewicz and Saeed, 2024; Bach, 2024)

The empirical spectral measure converges to a deterministic limit.

An example of linear regression: Textbook level and beyond

- n i.i.d. samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ with $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$
- $y = \langle \boldsymbol{\beta}_*, \boldsymbol{x} \rangle + \varepsilon$, $\mathbb{E}(\varepsilon) = 0$ and $\mathbb{V}(\varepsilon) = \sigma^2$, covariance matrix $\boldsymbol{\Sigma} = \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top]$
- ridge regression: $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$

Target: precise analysis

The expected test risk $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_*\|_{\boldsymbol{\Sigma}}^2$ vs. the norm $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}}\|_2^2$

❒ Deterministic equivalence (Cheng and Montanari, 2024; Misiakiewicz and Saeed, 2024; Bach, 2024)

$$\mathrm{Tr}\left(\boldsymbol{X}^\top \boldsymbol{X} (\boldsymbol{X}^\top \boldsymbol{X} + \lambda)^{-1}\right) \sim \mathrm{Tr}\left(\boldsymbol{\Sigma}(\boldsymbol{\Sigma} + \lambda_*)^{-1}\right), w.h.p.$$

- $\sim$ can be **asymptotic** or **non-asymptotic** at the rate of $\mathcal{O}(1/\sqrt{n})$.
- λ_* is the non-negative solution to the self-consistent equation $n - \frac{\lambda}{\lambda_*} = \mathrm{Tr}(\boldsymbol{\Sigma}(\boldsymbol{\Sigma} + \lambda_*)^{-1})$.

An example of linear regression: Textbook level and beyond

- n i.i.d. samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ with $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$
- $y = \langle \boldsymbol{\beta}_*, \boldsymbol{x} \rangle + \varepsilon$, $\mathbb{E}(\varepsilon) = 0$ and $\mathbb{V}(\varepsilon) = \sigma^2$, covariance matrix $\boldsymbol{\Sigma} = \mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top]$
- ridge regression: $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$

Target: precise analysis

The expected test risk $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_*\|_{\boldsymbol{\Sigma}}^2$ vs. the norm $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}}\|_2^2$

- ❒ Deterministic equivalence (Cheng and Montanari, 2024; Misiakiewicz and Saeed, 2024; Bach, 2024)
- ❒ Bias-variance decomposition on the test risk

- $\mathcal{B}_{\mathcal{R},\lambda}^{\mathrm{LS}} = \lambda^2 \langle \boldsymbol{\beta}_*, (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{\Sigma} (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{\beta}_* \rangle$
- $\mathcal{V}_{\mathcal{R},\lambda}^{\mathrm{LS}} = \sigma^2 \mathrm{Tr}(\boldsymbol{\Sigma} \boldsymbol{X}^\top \boldsymbol{X} (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-2})$

Our results

Theorem (asymptotic/non-asymptotic results)

We have a bias-variance decomposition $\mathbb{E}_\varepsilon \|\hat{\beta}\|_2^2 = \mathcal{B}_{\mathcal{N},\lambda} + \mathcal{V}_{\mathcal{N},\lambda}$.

Our results

Theorem (asymptotic/non-asymptotic results)

We have a bias-variance decomposition $\mathbb{E}_\varepsilon\|\hat{\boldsymbol{\beta}}\|_2^2 = \mathcal{B}_{\mathcal{N},\lambda} + \mathcal{V}_{\mathcal{N},\lambda}$.
For well-behaved data and $\boldsymbol{\Sigma}$, *we have* $\mathcal{B}_{\mathcal{N},\lambda} \sim \mathsf{B}_{\mathsf{N},\lambda}$ *and* $\mathcal{V}_{\mathcal{N},\lambda} \sim \mathsf{V}_{\mathsf{N},\lambda}, w.h.p.$

$$\mathsf{B}_{\mathsf{N},\lambda} := \langle \boldsymbol{\beta}_*, \boldsymbol{\Sigma}^2(\boldsymbol{\Sigma}+\lambda_*)^{-2}\boldsymbol{\beta}_*\rangle + \frac{\operatorname{Tr}(\boldsymbol{\Sigma}(\boldsymbol{\Sigma}+\lambda_*)^{-2})}{n} \underbrace{\frac{\lambda_*^2\langle \boldsymbol{\beta}_*, \boldsymbol{\Sigma}(\boldsymbol{\Sigma}+\lambda_*)^{-2}\boldsymbol{\beta}_*\rangle}{1-\frac{1}{n}\operatorname{Tr}(\boldsymbol{\Sigma}^2(\boldsymbol{\Sigma}+\lambda_*)^{-2})}}_{\mathcal{B}_{\mathcal{R},\lambda}},$$

$$\mathsf{V}_{\mathsf{N},\lambda} := \frac{\sigma^2 \operatorname{Tr}(\boldsymbol{\Sigma}(\boldsymbol{\Sigma}+\lambda_*)^{-2})}{n - \operatorname{Tr}(\boldsymbol{\Sigma}^2(\boldsymbol{\Sigma}+\lambda_*)^{-2})}.$$

Our results

Theorem (asymptotic/non-asymptotic results)

We have a bias-variance decomposition $\mathbb{E}_\varepsilon \|\hat{\boldsymbol{\beta}}\|_2^2 = \mathcal{B}_{\mathcal{N},\lambda} + \mathcal{V}_{\mathcal{N},\lambda}$.
For well-behaved data and $\boldsymbol{\Sigma}$, *we have* $\mathcal{B}_{\mathcal{N},\lambda} \sim \mathsf{B}_{\mathsf{N},\lambda}$ *and* $\mathcal{V}_{\mathcal{N},\lambda} \sim \mathsf{V}_{\mathsf{N},\lambda}, w.h.p.$

$$\mathsf{B}_{\mathsf{N},\lambda} := \langle \boldsymbol{\beta}_*, \boldsymbol{\Sigma}^2(\boldsymbol{\Sigma}+\lambda_*)^{-2}\boldsymbol{\beta}_*\rangle + \frac{\operatorname{Tr}(\boldsymbol{\Sigma}(\boldsymbol{\Sigma}+\lambda_*)^{-2})}{n} \underbrace{\frac{\lambda_*^2\langle \boldsymbol{\beta}_*, \boldsymbol{\Sigma}(\boldsymbol{\Sigma}+\lambda_*)^{-2}\boldsymbol{\beta}_*\rangle}{1-\frac{1}{n}\operatorname{Tr}(\boldsymbol{\Sigma}^2(\boldsymbol{\Sigma}+\lambda_*)^{-2})}}_{\mathcal{B}_{\mathcal{R},\lambda}},$$

$$\mathsf{V}_{\mathsf{N},\lambda} := \frac{\sigma^2 \operatorname{Tr}(\boldsymbol{\Sigma}(\boldsymbol{\Sigma}+\lambda_*)^{-2})}{n - \operatorname{Tr}(\boldsymbol{\Sigma}^2(\boldsymbol{\Sigma}+\lambda_*)^{-2})}.$$

Remark: Which model capacity suffices to characterize the test risk?

- Norm-based capacity: ✓ ☺
- effective dimension-style $\operatorname{Tr}(\boldsymbol{\Sigma}(\boldsymbol{\Sigma}+\lambda I)^{-1})$: ✗ ☹

Example: Relationship under isotropic features ($\Sigma = I_d$)

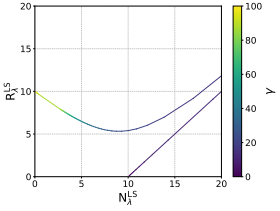
- Test risk R_λ and norm N_λ formulates a cubic curve (complex but precise).

Example: Relationship under isotropic features ($\Sigma = I_d$)

- Test risk R_λ and norm N_λ formulates a cubic curve (complex but precise).

- min-norm interpolator ($\lambda = 0$):

$$\mathsf{R}_0 = \begin{cases} \mathsf{N}_0 - \|\boldsymbol{\beta}_*\|_2^2; \text{in under-parameterized regimes} \\ \sqrt{\left[\mathsf{N}_0 - (\|\boldsymbol{\beta}_*\|_2^2 - \sigma^2)\right]^2 + 4\|\boldsymbol{\beta}_*\|_2^2\sigma^2} - \sigma^2 \,. \end{cases}$$

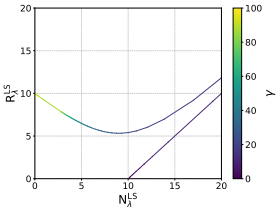


Example: Relationship under isotropic features ($\Sigma = I_d$)

- Test risk R_λ and norm N_λ formulates a cubic curve (complex but precise).

- min-norm interpolator ($\lambda = 0$):

$$\mathsf{R}_0 = \begin{cases} \mathsf{N}_0 - \|\boldsymbol{\beta}_*\|_2^2; \text{in under-parameterized regimes} \\ \sqrt{\left[\mathsf{N}_0 - (\|\boldsymbol{\beta}_*\|_2^2 - \sigma^2)\right]^2 + 4\|\boldsymbol{\beta}_*\|_2^2\sigma^2} - \sigma^2 . \end{cases}$$



Why? ◦ Variance is the same

◦ $\lambda_* = 0$ (under-parameterized)

◦ $\lambda_* = \frac{d-n}{n}$ (over-parameterized)

Example: Relationship under isotropic features ($\Sigma = I_d$)

❒ Test risk R_λ and norm N_λ formulates a cubic curve (complex but precise).

- min-norm interpolator ($\lambda = 0$):

$$\mathsf{R}_0 = \begin{cases} \mathsf{N}_0 - \|\boldsymbol{\beta}_*\|_2^2; \text{in under-parameterized regimes} \\ \sqrt{\left[\mathsf{N}_0 - (\|\boldsymbol{\beta}_*\|_2^2 - \sigma^2)\right]^2 + 4\|\boldsymbol{\beta}_*\|_2^2\sigma^2} - \sigma^2 \,. \end{cases}$$

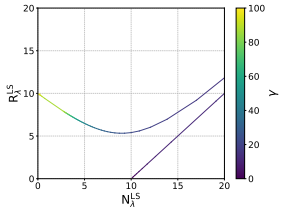
Why? ∘ Variance is the same

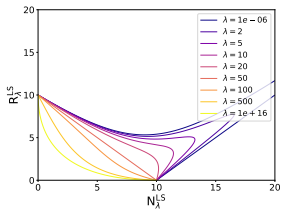
∘ $\lambda_* = 0$ (under-parameterized)

∘ $\lambda_* = \frac{d-n}{n}$ (over-parameterized)

- optimal regularization $\lambda = \frac{d\sigma^2}{\|\boldsymbol{\beta}^*\|_2^2}$ (Wu and Xu, 2020):

$\mathsf{R}_\lambda = \|\boldsymbol{\beta}_*\|_2^2 - \mathsf{N}_\lambda$





Example: Relationship under isotropic features ($\Sigma = I_d$)

❒ Test risk R_λ and norm N_λ formulates a cubic curve (complex but precise).

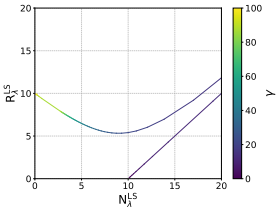
- min-norm interpolator ($\lambda = 0$):

$$\mathsf{R}_0 = \begin{cases} \mathsf{N}_0 - \|\boldsymbol{\beta}_*\|_2^2; \text{in under-parameterized regimes} \\ \sqrt{\left[\mathsf{N}_0 - (\|\boldsymbol{\beta}_*\|_2^2 - \sigma^2)\right]^2 + 4\|\boldsymbol{\beta}_*\|_2^2\sigma^2} - \sigma^2 \,. \end{cases}$$

Why? ∘ Variance is the same

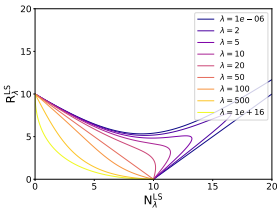
∘ $\lambda_* = 0$ (under-parameterized)

∘ $\lambda_* = \frac{d-n}{n}$ (over-parameterized)



- optimal regularization $\lambda = \frac{d\sigma^2}{\|\boldsymbol{\beta}^*\|_2^2}$ (Wu and Xu, 2020):

$\mathsf{R}_\lambda = \|\boldsymbol{\beta}_*\|_2^2 - \mathsf{N}_\lambda$



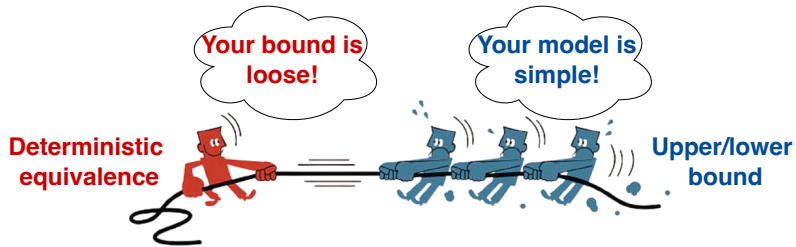
- $\lambda \to \infty$: $\mathsf{R}_\lambda = \left(\|\boldsymbol{\beta}_*\|_2 - \sqrt{\mathsf{N}_\lambda}\right)^2$

Precise analysis via deterministic equivalence

- Precisely describe the learning curve.
  - phase transitions, (non-)monotonicity, etc.
- Enables *accurate comparison* between estimators/algorithms.
  - Foundations of scaling law: data or parameter under the same budget, etc.

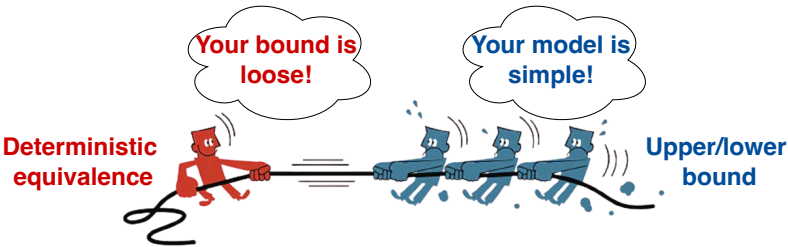
Precise analysis via deterministic equivalence

- Precisely describe the learning curve.
  - phase transitions, (non-)monotonicity, etc.
- Enables *accurate comparison* between estimators/algorithms.
  - Foundations of scaling law: data or parameter under the same budget, etc.



Precise analysis via deterministic equivalence

- Precisely describe the learning curve.
  - phase transitions, (non-)monotonicity, etc.
- Enables *accurate comparison* between estimators/algorithms.
  - Foundations of scaling law: data or parameter under the same budget, etc.



Is ℓ_2 norm-based capacity **best** for characterizing generalization?

Which model capacity is suitable (for neural networks)?

Table 1: Complexity measures compared in the empirical study (Jiang et al., 2020), and their correlation with generalization.

name	definition	rank correlation
Parameter Frobenius norm	$\sum_{i=1}^{L} \|\boldsymbol{W}_i\|_F^2$	0.073

Which model capacity is suitable (for neural networks)?

Table 1: Complexity measures compared in the empirical study (Jiang et al., 2020), and their correlation with generalization.

name	definition	rank correlation
Parameter Frobenius norm	$\sum_{i=1}^{L} \|\boldsymbol{W}_i\|_F^2$	0.073
Frobenius distance to initialization	$\sum_{i=1}^{L} \|\boldsymbol{W}_i - \boldsymbol{W}_i^0\|_{\mathrm{F}}^2$	-0.263
Spectral complexity	$\prod_{i=1}^{L} \|\boldsymbol{W}_i\| \left(\sum_{i=1}^{L} \frac{\|\boldsymbol{W}_i\|_{2,1}^{3/2}}{\|\boldsymbol{W}_i\|^{3/2}} \right)^{2/3}$	$\mathbf{-0.537}$
Fisher-Rao	$\frac{(L+1)^2}{n} \sum_{i=1}^{n} \langle \boldsymbol{W}, \nabla_{\boldsymbol{W}} \ell(h_{\boldsymbol{W}}(\boldsymbol{x}_i), y_i) \rangle$	0.078

Which model capacity is suitable (for neural networks)?

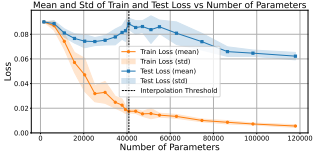
Table 1: Complexity measures compared in the empirical study (Jiang et al., 2020), and their correlation with generalization.

name	definition	rank correlation
Parameter Frobenius norm	$\sum_{i=1}^{L} \|\boldsymbol{W}_i\|_F^2$	0.073
Frobenius distance to initialization	$\sum_{i=1}^{L} \|\boldsymbol{W}_i - \boldsymbol{W}_i^0\|_{\mathrm{F}}^2$	-0.263
Spectral complexity	$\prod_{i=1}^{L} \|\boldsymbol{W}_i\| \left(\sum_{i=1}^{L} \frac{\|\boldsymbol{W}_i\|_{2,1}^{3/2}}{\|\boldsymbol{W}_i\|^{3/2}} \right)^{2/3}$	$\mathbf{-0.537}$
Fisher-Rao	$\frac{(L+1)^2}{n} \sum_{i=1}^{n} \langle \boldsymbol{W}, \nabla_{\boldsymbol{W}} \ell(h_{\boldsymbol{W}}(\boldsymbol{x}_i), y_i) \rangle$	0.078
Path-norm	$\sum_{(i_0,\ldots,i_L)} \prod_{j=1}^{L} \left(\boldsymbol{W}_{i_j, i_{j-1}} \right)^2$	$\mathbf{0.373}$

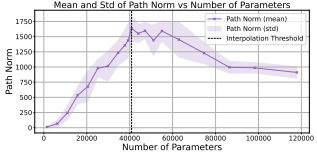
Which model capacity is suitable (for neural networks)?

Table 1: Complexity measures compared in the empirical study (Jiang et al., 2020), and their correlation with generalization.

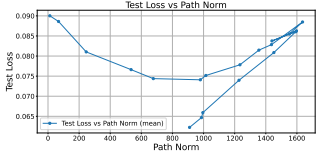
name	definition	rank correlation
Parameter Frobenius norm	$\sum_{i=1}^{L} \|\boldsymbol{W}_i\|_F^2$	0.073
Frobenius distance to initialization	$\sum_{i=1}^{L} \|\boldsymbol{W}_i - \boldsymbol{W}_i^0\|_{\mathrm{F}}^2$	-0.263
Spectral complexity	$\prod_{i=1}^{L} \|\boldsymbol{W}_i\| \left(\sum_{i=1}^{L} \frac{\|\boldsymbol{W}_i\|_{2,1}^{3/2}}{\|\boldsymbol{W}_i\|^{3/2}} \right)^{2/3}$	$\mathbf{-0.537}$
Fisher-Rao	$\frac{(L+1)^2}{n} \sum_{i=1}^{n} \langle \boldsymbol{W}, \nabla_{\boldsymbol{W}} \ell(h_{\boldsymbol{W}}(\boldsymbol{x}_i), y_i) \rangle$	0.078
Path-norm	$\sum_{(i_0,\ldots,i_L)} \prod_{j=1}^{L} \left(\boldsymbol{W}_{i_j, i_{j-1}} \right)^2$	$\mathbf{0.373}$



(a) Test (training) Loss vs. p



(b) Path-norm vs. p



(c) Test Loss vs. Path-norm

Proof sketch: convex hull technique and its constant

- Consider the following function space

$$\mathcal{F} = \{\sigma(\langle \widetilde{\boldsymbol{w}}, \cdot \rangle) : \widetilde{\boldsymbol{w}} \in \mathcal{W}\} \cup \{0\} \cup \{-\sigma(\langle \widetilde{\boldsymbol{w}}, \cdot \rangle) : \widetilde{\boldsymbol{w}} \in \mathbb{S}_1^{d-1} \text{ with the } \ell_1 \text{ ball}\}$$

Proof sketch: convex hull technique and its constant

- Consider the following function space

$$\mathcal{F} = \{\sigma(\langle \widetilde{\boldsymbol{w}}, \cdot \rangle) : \widetilde{\boldsymbol{w}} \in \mathcal{W}\} \cup \{0\} \cup \{-\sigma(\langle \widetilde{\boldsymbol{w}}, \cdot \rangle) : \widetilde{\boldsymbol{w}} \in \mathbb{S}_1^{d-1} \text{ with the } \ell_1 \text{ ball}\}$$

- the convex hull of $\mathcal{F}$ is

$$\overline{\mathrm{conv}}\mathcal{F} = \left\{ \sum_{i=1}^{m} \alpha_i f_i \middle| f_i \in \mathcal{F}, \sum_{i=1}^{m} \alpha_i = 1, \alpha_i \geqslant 0, m \in \mathbb{N} \right\}.$$

Proof sketch: convex hull technique and its constant

- Consider the following function space

$$\mathcal{F} = \{\sigma(\langle \widetilde{\boldsymbol{w}}, \cdot \rangle) : \widetilde{\boldsymbol{w}} \in \mathcal{W}\} \cup \{0\} \cup \{-\sigma(\langle \widetilde{\boldsymbol{w}}, \cdot \rangle) : \widetilde{\boldsymbol{w}} \in \mathbb{S}_1^{d-1} \text{ with the } \ell_1 \text{ ball}\}$$

- the convex hull of $\mathcal{F}$ is

$$\overline{\text{conv}}\mathcal{F} = \left\{ \sum_{i=1}^{m} \alpha_i f_i \middle| f_i \in \mathcal{F}, \sum_{i=1}^{m} \alpha_i = 1, \alpha_i \geqslant 0, m \in \mathbb{N} \right\}.$$

- convex hull technique (Van Der Vaart et al., 1996, Theorem 2.6.9)

$$\log \mathscr{N}_2(\mathcal{G}_1, \epsilon) \leqslant \log \mathscr{N}_2(\overline{\text{conv}}\mathcal{F}, \epsilon, \frac{1}{n}\delta_{\boldsymbol{x}_i}) \leqslant C \left(\frac{1}{\epsilon}\right)^{\frac{2d}{d+2}}.$$

Proof sketch: convex hull technique and its constant

- Consider the following function space

$$\mathcal{F} = \{\sigma(\langle \widetilde{\boldsymbol{w}}, \cdot \rangle) : \widetilde{\boldsymbol{w}} \in \mathcal{W}\} \cup \{0\} \cup \{-\sigma(\langle \widetilde{\boldsymbol{w}}, \cdot \rangle) : \widetilde{\boldsymbol{w}} \in \mathbb{S}_1^{d-1} \text{ with the } \ell_1 \text{ ball}\}$$

- the convex hull of $\mathcal{F}$ is

$$\overline{\text{conv}}\mathcal{F} = \left\{ \sum_{i=1}^{m} \alpha_i f_i \middle| f_i \in \mathcal{F}, \sum_{i=1}^{m} \alpha_i = 1, \alpha_i \geqslant 0, m \in \mathbb{N} \right\}.$$

- convex hull technique (Van Der Vaart et al., 1996, Theorem 2.6.9)

$$\log \mathscr{N}_2(\mathcal{G}_1, \epsilon) \leqslant \log \mathscr{N}_2(\overline{\text{conv}}\mathcal{F}, \epsilon, \frac{1}{n}\delta_{\boldsymbol{x}_i}) \leqslant C \left(\frac{1}{\epsilon}\right)^{\frac{2d}{d+2}}.$$

- estimate the constant C

$$C := \underbrace{D_k}_{=\Theta(d)} [\underbrace{C_k}_{=\Theta(1)} (2^{d+1}+1)^{\frac{1}{d}}]^{\frac{2d}{d+2}} \leq 10^7 d \quad \text{if } d > 5$$

Dudley's entropy integral

Mathematical statement

Let (A, d) be a pseudo-metric space, $T \subseteq A$ totally bounded, $\operatorname{diam}(s) \leq D$, and $\{X_t\}_{t \in T}$ a normalized sub-Gaussian process with parameter σ. Under integrability and path-continuity hypotheses,

$$\mathbb{E}\left[\sup_{t \in T} X_t\right] \leq 12\sqrt{2}\,\sigma \int_0^D \sqrt{\log \mathcal{N}(\epsilon, T, d)}\,\mathrm{d}\epsilon\,.$$

Dudley's entropy integral

Mathematical statement

Let (A, d) be a pseudo-metric space, $T \subseteq A$ totally bounded, $\operatorname{diam}(s) \leq D$, and $\{X_t\}_{t \in T}$ a normalized sub-Gaussian process with parameter σ. Under integrability and path-continuity hypotheses,

$$\mathbb{E}\left[\sup_{t \in T} X_t\right] \leq 12\sqrt{2}\,\sigma \int_0^D \sqrt{\log \mathcal{N}(\epsilon, T, d)}\,\mathrm{d}\epsilon\,.$$

Why it is complex in Lean

The proof combines finite nets, sub-Gaussian maximum inequalities, dyadic chaining, Fatou-type limits, and real vs. nonnegative integration APIs.

One proof-engineering issue

The convenient measure-theoretic integral lives in $\mathbb{R}_{\geq 0} \cup \{\infty\}$, while downstream SLT inequalities are stated over $\mathbb{R}$.