Guarantees-Driven Mechanistic Interpretability

Formal Proof Size as a Metric for Mechanistic Detail of Understanding

FAR Seminar, Feb 21

Jason Gross
Collaborators: Rajashree Agrawal, Alex Gibson, Chun Hei Yip, Euan Ong, Somsubhro Bagchi, Soufiane Noubir, Thomas Kwa
Funded by ARC Theory
Loosely advised by Paul Christiano, distillation and write-up additionally co-authored with Lawrence Chan

This talk in three bullet points

Motivation: If we got AGI tomorrow, what would we need to trust any pipeline we build to scalably automate mechanistic explanation discovery?

Solution: Trustworthiness via math (aka formal proof)

Remaining bottleneck: Unstructured noise

Why mech interp?

AI alignment; might help with:

- Catching deception
- Mechanistic anomaly detection (MAD)
- Adversarial training
- Elicit latent knowledge (ELK)
- Provide feedback

Actual causal/historical reason, in my case:

- Neel Nanda’s modular grokking write up is cool!

What is “mechanistic”?

Intuition: “bottom-up”

“Mechanistic interpretability seeks to reverse engineer neural networks, similar to how one might reverse engineer a compiled binary computer program.”

—Chris Olah

“Mechanistic refers to the emphasis on trying to understand the actual mechanisms and algorithms that compose the network”

—Neel Nanda

These are actually about ***faithfulness*** of mechanism — how closely mechanisms corresponds to the mechanisms the model uses

Quotes from https://transformer-circuits.pub/2022/mech-interp-essay/index.html
https://dynalist.io/d/n2ZWtnoYHrU1s4vnFSAQ519J#z=eL6tFQqNwd4LbYlO1DVIen8K

How do we evaluate “mechanistic”?

Existing methods all focus on *faithfulness*

- Casual Scrubbing
- Activation Patching
- Path Patching

We have nothing for *level of mechanistic detail*

Problem 1: Existing metrics are too easy to Goodhart

The brute-force explanation

“I ran the model” i.e., trace the model’s computation on all relevant inputs

100% faithful!

100% bottom-up!

100% useless for many applications!

(also intuitively unsatisfactory)

Very important if we ever want to automate interpretability!

What’s wrong?

1. Infeasible to produce
2. Does not match intuition on “mechanistic”

Common cause:

The explanation is ‘too long’

Problem 2

Existing metrics are limited in what hypotheses they permit

Generally restricted to identifying (sparse) computational subgraphs

Can we get away with minimalism?
Mechanistic detail ∝ 1/(description length of formal proof)

“Mechanistic” = “allows compacting explanation”

Consider theorems that a particular model M achieves a certain level of performance:

$$\mathbb{E}[f(x, M(x))] > b$$

Goal: minimize proof length (in any formal system) for fixed b;
or: maximize b for fixed proof length

Can we get away with minimalism?
Mechanistic detail $\propto$ 1/(description length of formal proof)
What is a proof?

Theorem: $\mathbb{E}[f(x, M(x))] > b$

Goal: minimize proof length (in any formal system) for fixed b;
or: maximize b for fixed proof length

Our proofs consist of two components:

1. Proof that a particular computation C, when run with any model's weights, produces a valid bound on that model's performance
2. A trace of running C proving that C(M) = b

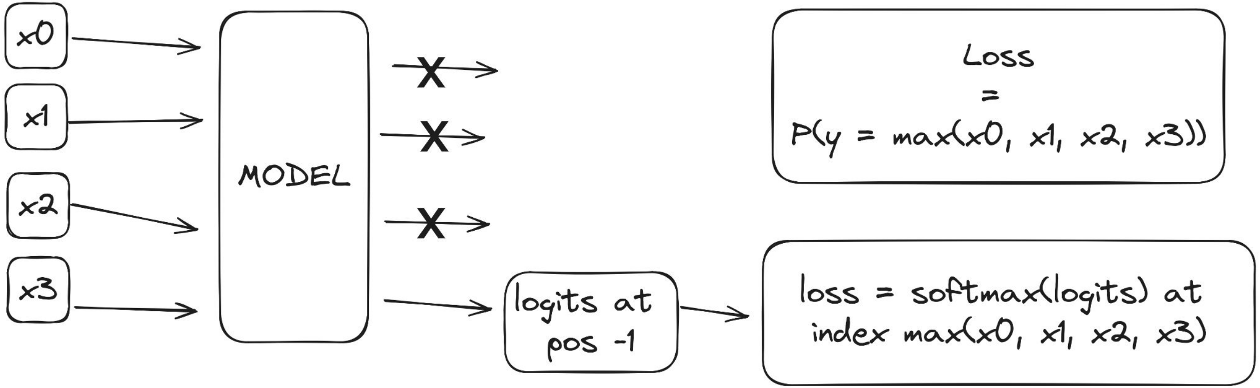
Outline of the Technical Part of the Talk

Goal: walkthrough of a toy model to assess this definition of mechanistic detail

- Toy algorithmic task
- Small transformer architecture
- Basic model interpretation
- Proof Size vs. Tightness of Bound (table of proofs with four complexities)
- Sketch of the proof at each complexity
- Noise problem
- Conclusions, Limitations, & Future Work

Anchor: Tying mechanistic detail and size of proof

Model Setup: Task



Max-of-K (K=4)

one-hot encoded numbers

Accuracy: argmax(model(xs)[-1]) == max(xs)

Loss: softmax(model(xs)[-1])[max(xs)]

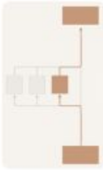
model([40,**62**,3,0]) == [[_, _, _, [-10, -16, -18, …, 16.8, **32.6**, 0.6]]] (position 62 = 64 - 2)

Model Setup

1L, attn-only, no layernorm
1 attn head
d_vocab = 64
d_head = d_model = 32
n_ctx = 4

$$T_i = (t_i \cdot W_E + (W_{\text{pos}})_i) W_U + \sum_{h\in H}\sum_j A^h_{i,j}(t_j \cdot W_E + (W_{\text{pos}})_j) W_V W_O W_U$$

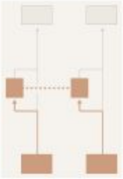
"Direct path" term contributes to bigram statistics.



The **attention head** terms describe the effects of attention heads in linking input tokens to logits. A^h describes which tokens are attended to while $W_U W_{OV}^h W_E$ describes how each token changes the logits if attended to.

where $$A^h_{q,k} = \text{softmax}^*\left((t_q \cdot W_E + (W_{\text{pos}})_q) \cdot W_Q W_K^T (W_E^T \cdot t_k^T + (W_{\text{pos}})_k^T)\right)$$

Softmax with autoregressive masking



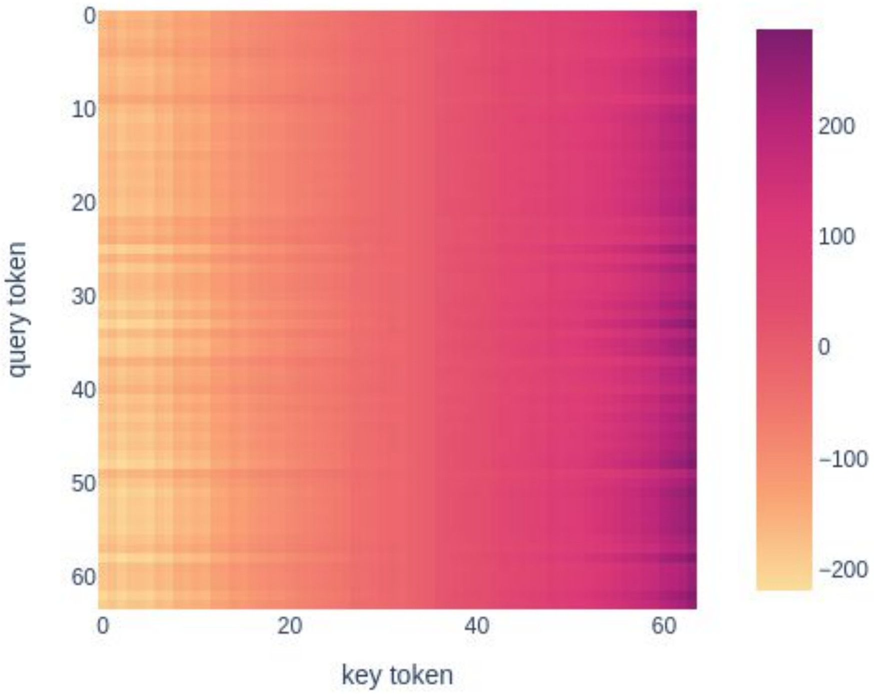
Attention pattern logits are produced by multiplying pairs of tokens through different sides of W_{QK}^h.

Image from https://transformer-circuits.pub/2021/framework/index.html#onel-path-expansion

Basic Mech Interp: Attend More to Bigger Tokens & Copy

Attention Score

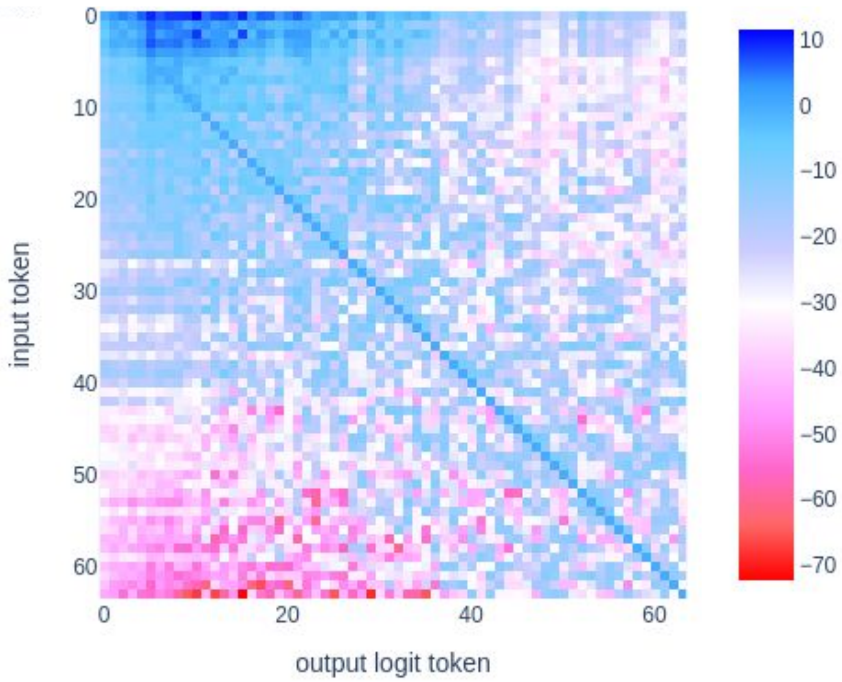
EQKE := $(W_E + W_{pos}[-1])W_Q W_K^T(W_E + \mathbb{E}_p W_{pos}[p])^T$



Attention Computation (centered)

EVOU := $(W_E + \mathbb{E}_p W_{pos}[p])W_V W_O W_U$

EVOU - EVOU.diag()[:, None]



Results: Proof Size vs. Tightness of Bound

Description of Proof	Complexity Cost Budget	Bound
Brute force	Exponential: $\text{d\_vocab}^{\text{n\_ctx}}\text{n\_ctx}$ d_vocab d_model	99.73%
Convexity of Softmax	Cubic: $\text{d\_vocab}^3\text{n\_ctx}^2$	98.4%
Sub-cubic	$\text{d\_vocab}^2\text{n\_ctx}^2$ + **$\text{d\_vocab}^2\text{d\_model}$**	54.5% – 56.9%
low-rank avg+diff on EU, QK	$\text{d\_vocab}^2\text{n\_ctx}^2$ + **d_vocab d_model^2** + (OV only) $\text{d\_vocab}^2\text{d\_model}$	48.9% – 54.8% (27.8% – 33.2% if via SVD)
Convex Hull on OV (WIP)	$\text{d\_vocab}^2\text{n\_ctx}^2$ + **d_vocab d_model^2**	WIP

What do & don’t we understand?

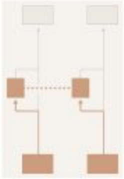
$$T_i = (t_i \cdot W_E + (W_{\text{pos}})_i)W_U + \sum_{h \in H} \sum_{k} A^h_{i,k}(t_k \cdot W_E + (W_{\text{pos}})_k)W_V^h W_O^h W_U$$

“Direct path” term contributes to bigram statistics.

The **attention head** terms describe the effects of attention heads in linking input tokens to logits. A^h describes which tokens are attended to while $W_U W_{OV}^h W_E$ describes how each token changes the logits if attended to.

where $$A^h_{q,k} = \text{softmax}^*\left((t_q \cdot W_E + (W_{\text{pos}})_q) \cdot W_Q^h {W_K^h}^T (W_E^T \cdot t_k^T + (W_{\text{pos}})_k^T)\right)$$

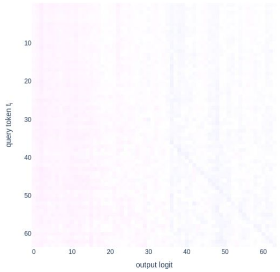
Softmax with autoregressive masking



Attention pattern logits are produced by multiplying pairs of tokens through different sides of W_{QK}^h.

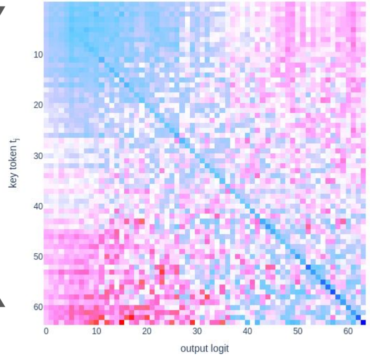
What do & don’t we understand?

$T_{-1} =$



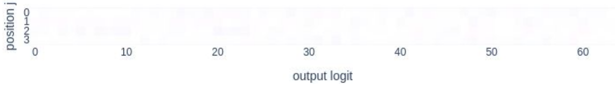
direct contribution

$+\sum_k A_{-1,k} \cdot \Big($



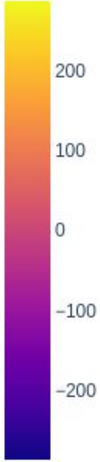
key token contribution

$+$

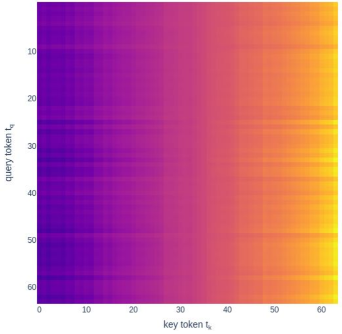


key position contribution

$\Big)$

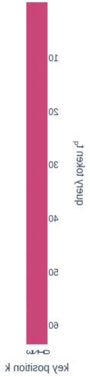


$A_{-1,k}$ = softmax* $\Big($



key token contribution

$+$

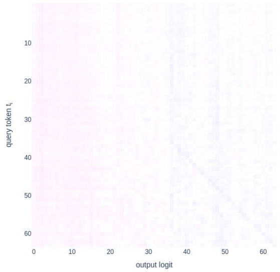


key position contribution

$\Big)$

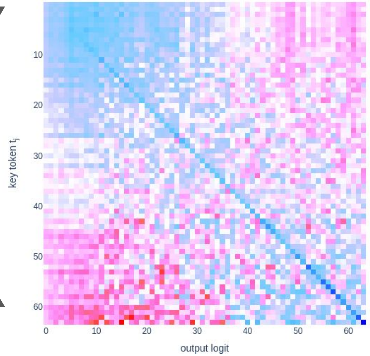
Brute force accuracy: 99.73% (16,777,216 sequences)

$T_{-1} =$



direct contribution

$+ \sum_k A_{-1,k} \cdot \Big($

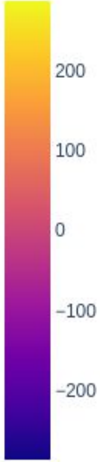


key token contribution

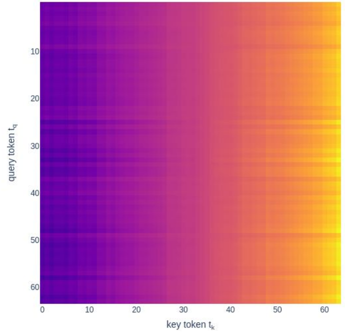
$+$

key position contribution

$\Big)$

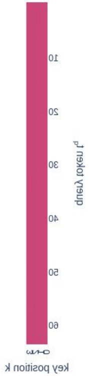


$A_{-1,k}$ = softmax* $\Big($



key token contribution

$+$



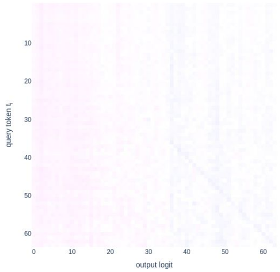
key position contribution

$\Big)$

$\mathcal{O}(\text{d\_vocab}^3\text{n\_ctx}^2)$

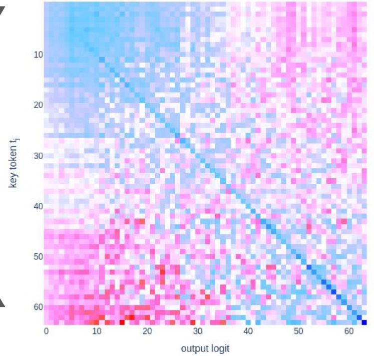
Accuracy with cubic proof: 98.4% (1,048,576 sequences)

$T_{-1} =$



direct contribution

$+\sum_k A_{-1,k} \cdot \Bigg($



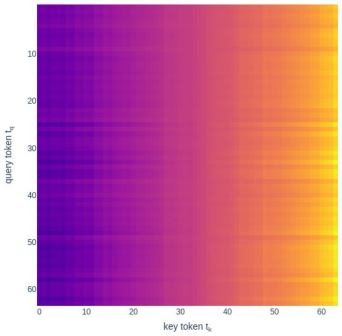
key token contribution

$+$

key position contribution

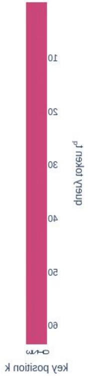
$\Bigg)$

$A_{-1,k} = \text{softmax*}\Bigg($



key token contribution

$+$



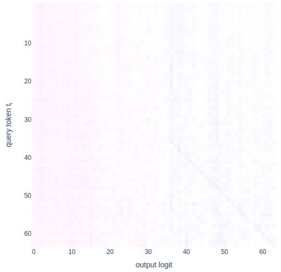
key position contribution

$\Bigg)$

$\mathcal{O}$(d_vocab^{2}n_ctx^2 + **d_vocab^{2}d_model**)

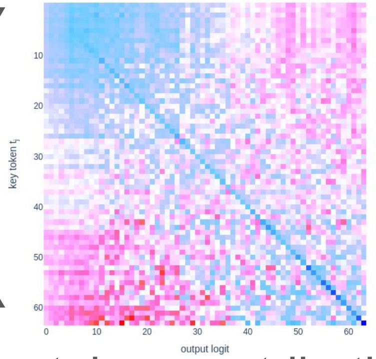
Accuracy with sub-cubic proof: ≈55% (≈ 65,536 sequences)

$T_{-1} =$



direct contribution

$+\sum_k A_{-1,k} \cdot \Big($



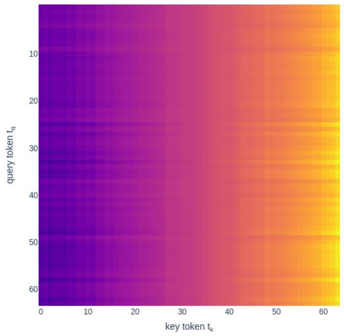
key token contribution

$+$

key position contribution

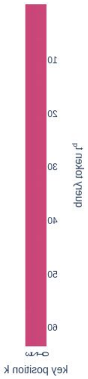
$\Big)$

$A_{-1,k}$ = softmax* $\Big($



key token contribution

$+$



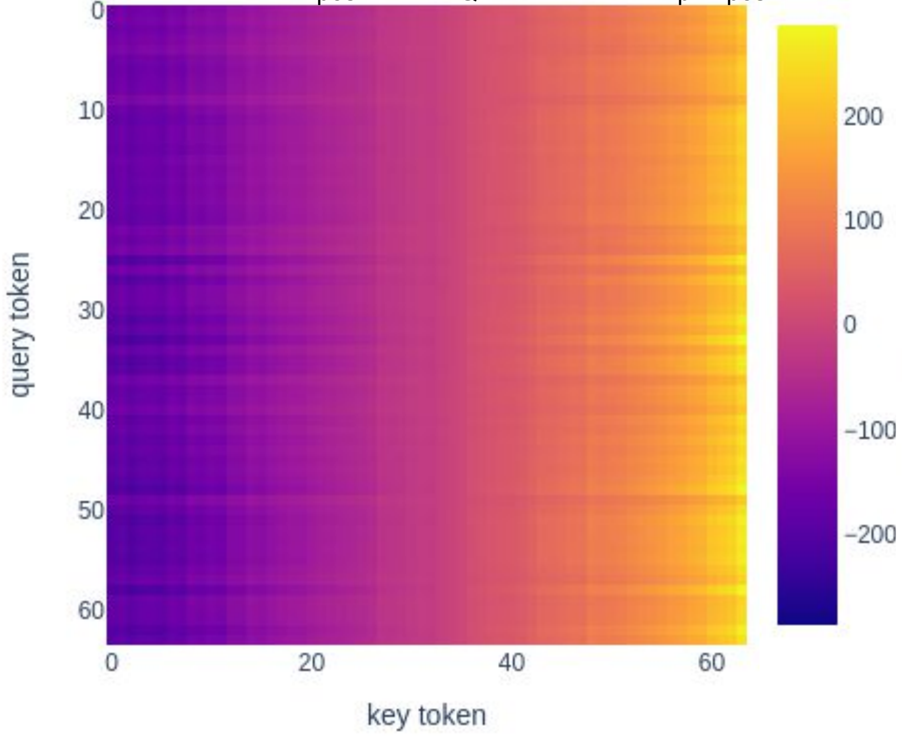
key position contribution

$\Big)$

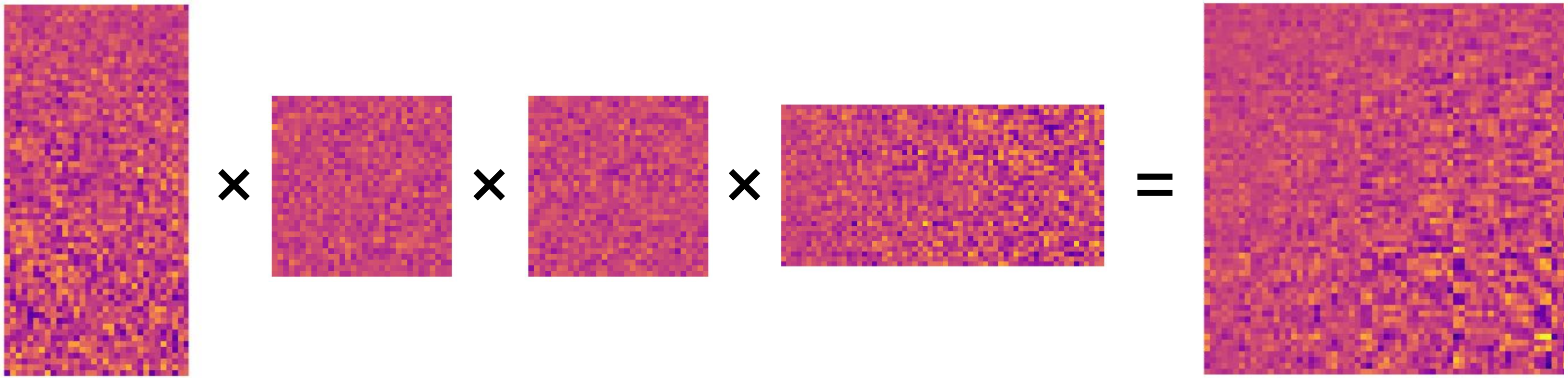
Mechanistic detail in proofs: d_vocab^{2}d_model ⇒ d_vocab d_model^2

Attention Score

EQKE := $(W_E + W_{pos}[-1])W_QW_K^T(W_E + \mathbb{E}_pW_{pos}[p])^T$



Noise

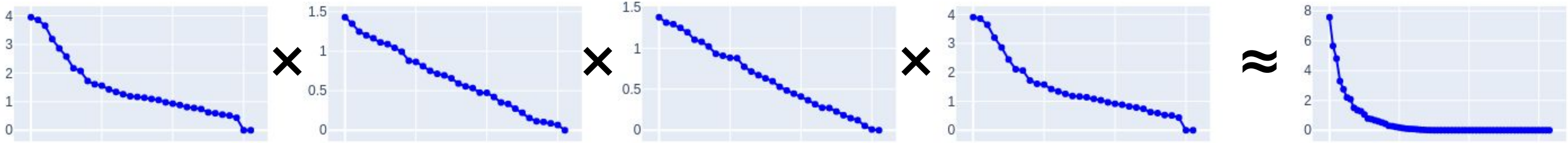


max row diff ≈ 1.85

How???

Noise: SVD?

$\sqrt{2} \cdot \sigma_1 \approx 4\sqrt{2}$ × $\sqrt{2} \cdot \sigma_1 \approx 1.4\sqrt{2}$ × $\sqrt{2} \cdot \sigma_1 \approx 1.4\sqrt{2}$ × $\sqrt{2} \cdot \sigma_1 \approx 4\sqrt{2}$ ≈ $30\sqrt{2} \approx 43 \geq 10.7$



max row diff		≈ 1.85
$2 \cdot$ max abs value		≈ 2
$\sqrt{2} \cdot \sigma_1$	$\approx 7.6\sqrt{2}$	≈ 10.7

Even lower complexity bound:
Frobenius norm: $10 \times 4 \times 4 \times 10\ \sqrt{2} \approx 1932\ \sqrt{2} \approx 2732$(!)

(best bound with another ~SVD-complexity method: ≈ 5.67)

Conclusions

- Proofs *are* possible!!!!!
  - But really hard
- Small noise is a problem (no mechanistic understanding)
  - Most existing work glosses over this
  - Do we even want an explanation of it?
- Proofs *can* be used as a minimalist “grounding” of mech interp
  - Confused about X in mech interp ⇒ convert to proof frame
- Link between mechanistic understanding and proof length
  - Shorter proofs require more mechanistic understanding
  - After improving bound tightness (fixed complexity), we can extract mechanistic detail
  - Failure to compact proof ⇒ lack mechanistic understanding
- Objective, numerical standard for mechanistic detail
  - Can be tailored to subcomponents

Limitations & Future Work

In progress:

- Max of 10
- Modular addition (including MLPs)
- Sorted list

How to solve noise?

- Lagrange multiplier on various parts of the proof
- Heuristic arguments

Limitations / Future Work:

- 2L
- Layer norm on > 1L
- SAEs
- Automation

Thank You!