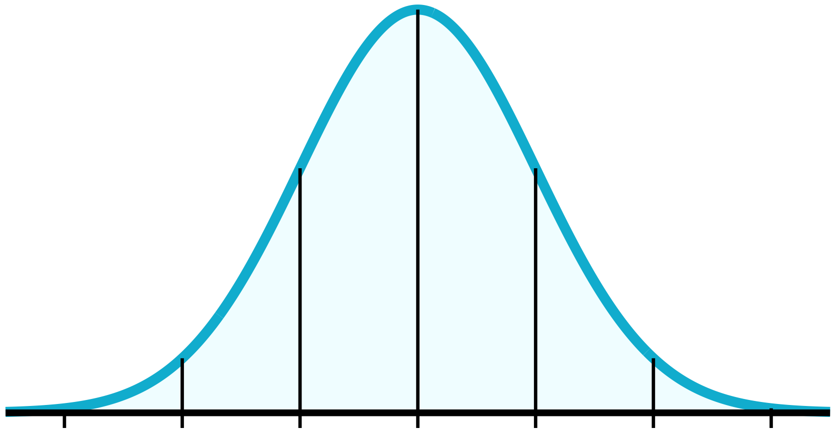
Attention-index models and how to solve them

using tools for quadratic networks

Emanuele Troiani

Multi-index models…



Input data

$d \gg 1$ dimensional Gaussian $\boldsymbol{x}_\mu$

Projections

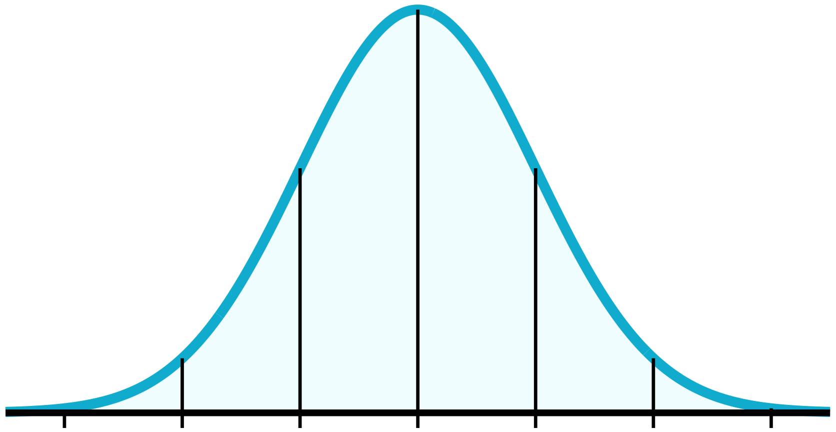
on $p = \mathcal{O}_d(1)$

random directions

Link function

using $g : \mathbb{R}^p \to \mathbb{R}$

Multi-index models…



<u>Input data</u>
$d \gg 1$ dimensional Gaussian $\boldsymbol{x}_\mu$

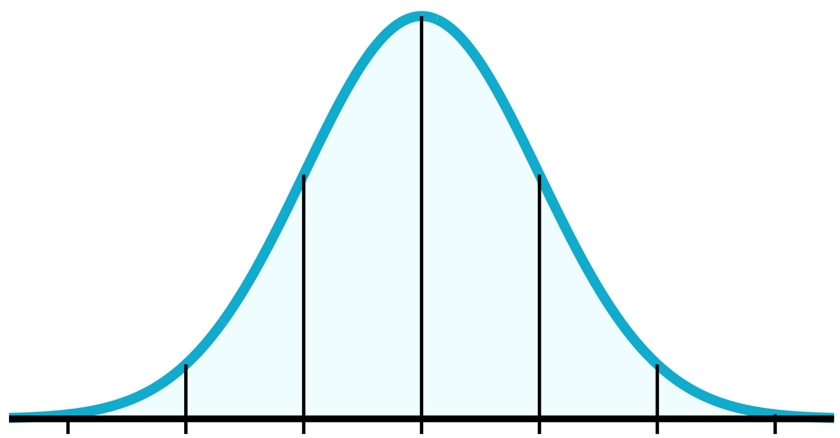
<u>Projections</u>
on $p = \mathcal{O}_d(1)$
random directions

<u>Link function</u>
using $g : \mathbb{R}^p \to \mathbb{R}$

<u>1.</u> Generate dataset $\mathcal{D} = \{\boldsymbol{x}_\mu, y_\mu\}_{\mu=1,\ldots,n}$

$$y_\mu = g(w_1^\star \boldsymbol{x}_\mu^\top, \ldots, w_p^\star \boldsymbol{x}_\mu^\top)$$

Multi-index models…



Input data

$d \gg 1$ dimensional Gaussian $\boldsymbol{x}_\mu$

Projections

on $p = \mathcal{O}_d(1)$

random directions

Link function

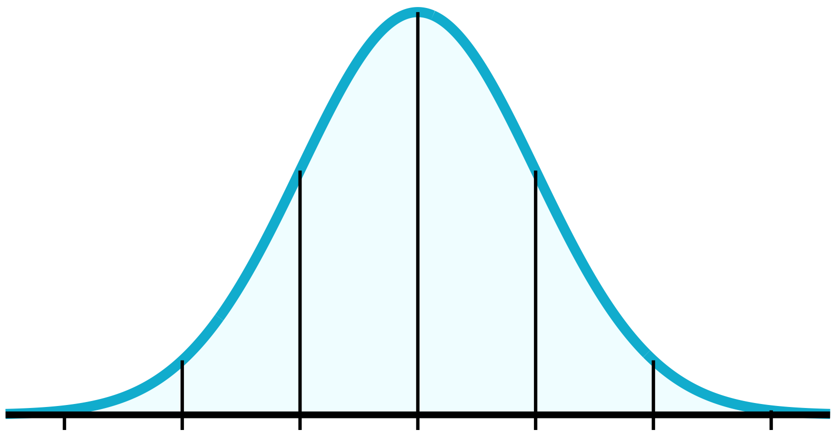
using $g : \mathbb{R}^p \to \mathbb{R}$

1. Generate dataset $\mathcal{D} = \{\boldsymbol{x}_\mu, y_\mu\}_{\mu=1,\ldots,n}$

$$y_\mu = g(w_1^{\star}\boldsymbol{x}_\mu^\top, \ldots, w_p^{\star}\boldsymbol{x}_\mu^\top)$$

2. Learn the function $x \mapsto y$

Multi-index models…



Input data

$d \gg 1$ dimensional Gaussian $\boldsymbol{x}_\mu$

Projections

on $p = \mathcal{O}_d(1)$

random directions

Link function

using $g : \mathbb{R}^p \to \mathbb{R}$

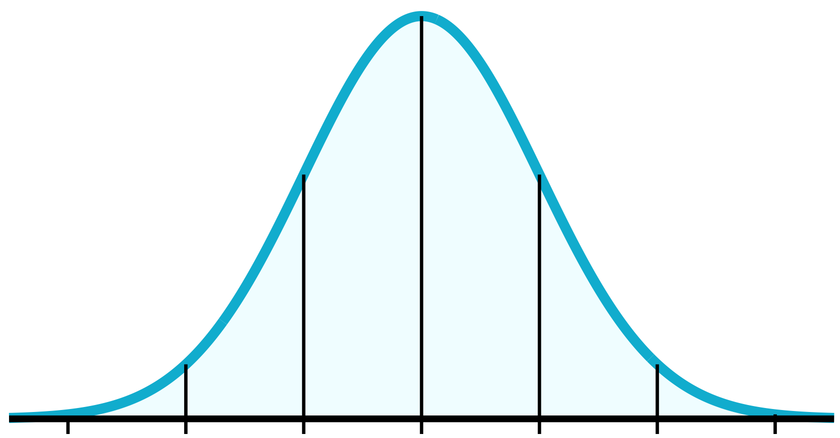
1. Generate dataset $\mathcal{D} = \{\boldsymbol{x}_\mu, y_\mu\}_{\mu=1,\ldots,n}$

$$y_\mu = g(w_1^{\star}\boldsymbol{x}_\mu^\top, \ldots, w_p^{\star}\boldsymbol{x}_\mu^\top)$$

2. Learn the function $x \mapsto y$

Bayes Optimal

Multi-index models…



Input data

$d \gg 1$ dimensional Gaussian $\boldsymbol{x}_\mu$

Projections

on $p = \mathcal{O}_d(1)$

random directions

Link function

using $g : \mathbb{R}^p \to \mathbb{R}$

1. Generate dataset $\mathcal{D} = \{\boldsymbol{x}_\mu, y_\mu\}_{\mu=1,\ldots,n}$

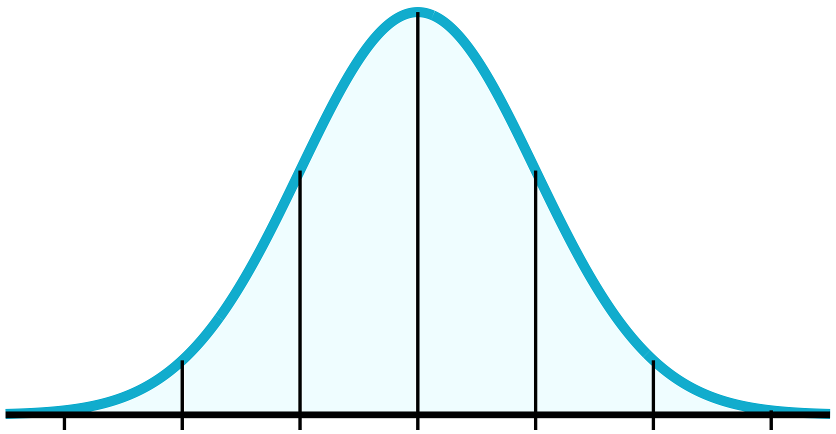
$$y_\mu = g(w_1^{\star}\boldsymbol{x}_\mu^\top, \ldots, w_p^{\star}\boldsymbol{x}_\mu^\top)$$

2. Learn the function $x \mapsto y$

Bayes Optimal

Empirical Risk Minimization

Multi-index models…



Input data

$d \gg 1$ dimensional Gaussian $\boldsymbol{x}_\mu$

Projections

on $p = \mathcal{O}_d(1)$

random directions

Link function

using $g : \mathbb{R}^p \to \mathbb{R}$

1. Generate dataset $\mathcal{D} = \{\boldsymbol{x}_\mu, y_\mu\}_{\mu=1,\ldots,n}$

$$y_\mu = g(w_1^{\star}\boldsymbol{x}_\mu^\top, \ldots, w_p^{\star}\boldsymbol{x}_\mu^\top)$$

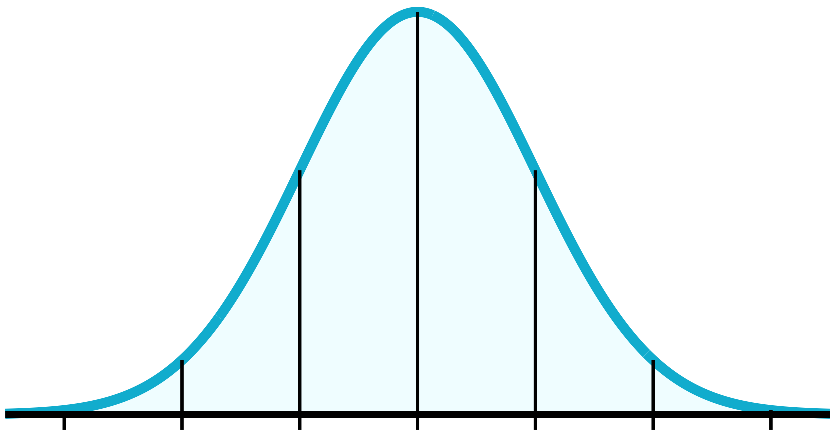
2. Learn the function $x \mapsto y$

Bayes Optimal

Empirical Risk Minimization

Many functions in this class

Multi-index models…



Input data

$d \gg 1$ dimensional Gaussian $\boldsymbol{x}_\mu$

Projections

on $p = \mathcal{O}_d(1)$ random directions

Link function

using $g : \mathbb{R}^p \to \mathbb{R}$

1. Generate dataset $\mathcal{D} = \{\boldsymbol{x}_\mu, y_\mu\}_{\mu=1,\ldots,n}$

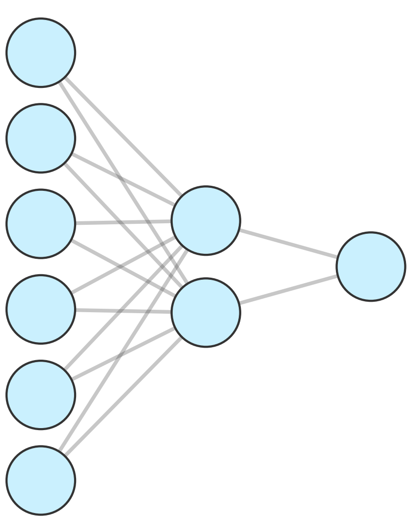
$$y_\mu = g(w_1^{\star}\boldsymbol{x}_\mu^\top, \ldots, w_p^{\star}\boldsymbol{x}_\mu^\top)$$

2. Learn the function $x \mapsto y$

Bayes Optimal

Empirical Risk Minimization

Many functions in this class

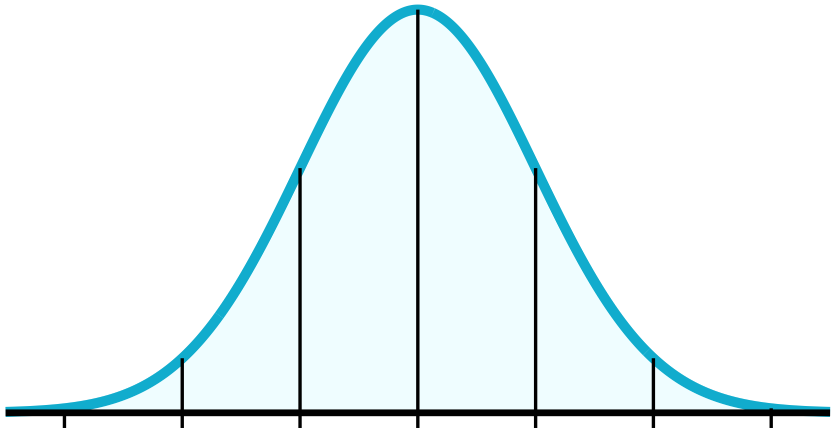


Narrow

two-layer networks…

$$y_\mu = \sum_{i=1}^{p} \sigma(\boldsymbol{w}_i^\top \boldsymbol{x}_\mu)$$

Multi-index models…



Input data

$d \gg 1$ dimensional Gaussian $\boldsymbol{x}_\mu$

Projections

on $p = \mathcal{O}_d(1)$
random directions

Link function

using $g : \mathbb{R}^p \to \mathbb{R}$

1. Generate dataset $\mathcal{D} = \{\boldsymbol{x}_\mu, y_\mu\}_{\mu=1,\ldots,n}$

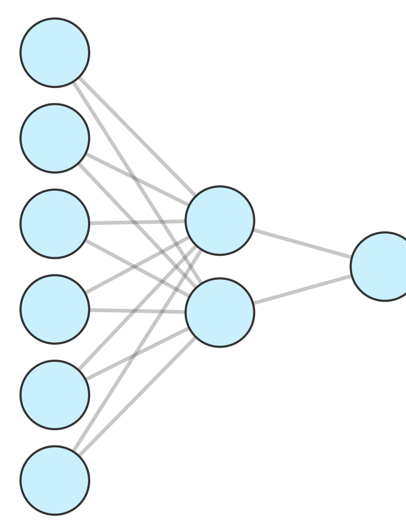
$$y_\mu = g(w_1^{\star}\boldsymbol{x}_\mu^\top, \ldots, w_p^{\star}\boldsymbol{x}_\mu^\top)$$

2. Learn the function $x \mapsto y$

Bayes Optimal

Empirical Risk Minimization

Many functions in this class

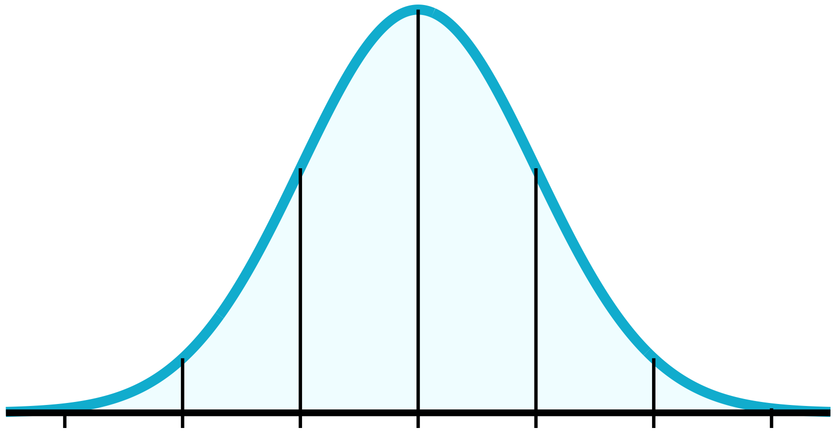


Narrow
two-layer networks…

$$y_\mu = \sum_{i=1}^{p} \sigma\left(\boldsymbol{w}_i^\top \boldsymbol{x}_\mu\right)$$

But also **Finite-rank** attention networks!

Multi-index models… are ideal for attention



Input data

Collections of Gaussian tokens $X_\mu \in \mathbb{R}^{T\times d}$, $T = \mathcal{O}_d(1)$, $d \gg 1$

Projections

on $p = \mathcal{O}_d(1)$ random directions

Link function

using $g : \mathbb{R}^p \to \mathbb{R}^{T\times T}$

1. Generate dataset $\mathcal{D} = \{X_\mu, y_\mu\}_{\mu=1,\ldots,n}$

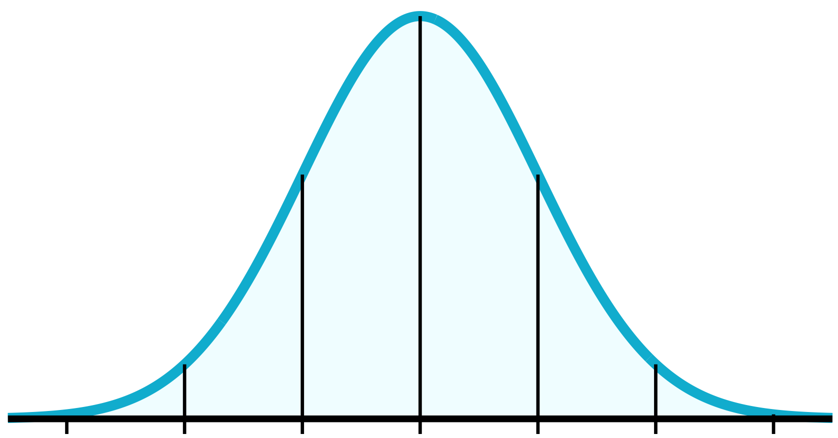
$$y_\mu = g(w_1^\star X_\mu^\top, \ldots, w_p^\star X_\mu^\top)$$

2. Learn the function $x \mapsto y$

Bayes Optimal

Empirical Risk Minimization

Multi-index models… are ideal for attention



Input data

Collections of Gaussian tokens $X_\mu \in \mathbb{R}^{T\times d}$, $T = \mathcal{O}_d(1)$, $d \gg 1$

Projections

on $p = \mathcal{O}_d(1)$ random directions

Link function

using $g : \mathbb{R}^p \to \mathbb{R}^{T\times T}$

1. Generate dataset $\mathcal{D} = \{X_\mu, y_\mu\}_{\mu=1,\ldots,n}$

$$y_\mu = g(w_1^\star X_\mu^\top, \ldots, w_p^\star X_\mu^\top)$$

2. Learn the function $x \mapsto y$

Bayes Optimal

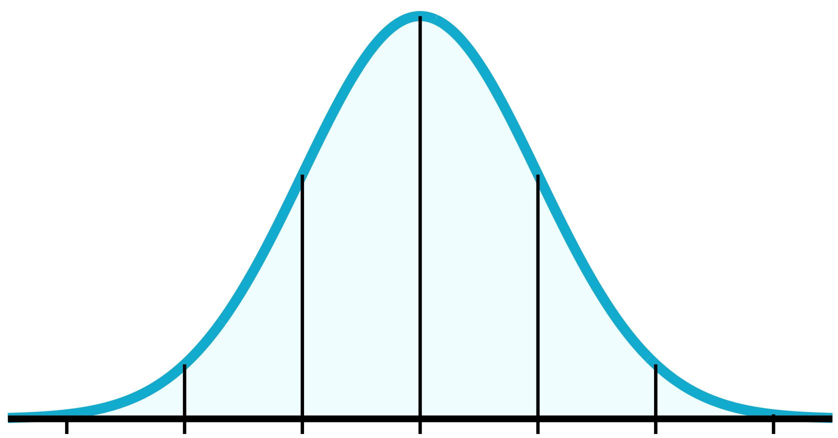
Empirical Risk Minimization

$$y_\mu = \sigma\left(X_\mu \sum_{i=1}^{p} \boldsymbol{w}_i \boldsymbol{w}_i^\top X_\mu^\top \right)$$

Tied keys-queries

[H. Cui, F. Behrens, F. Krzakala, L. Zdeborová, '24]
[L. Arnaboldi, B. Loureiro, L. Stephan, F. Krzakala, L. Zdeborová, '25]
[H. Cui, '25]

Multi-index models… are ideal for attention



Input data
Collections of Gaussian tokens $X_\mu \in \mathbb{R}^{T\times d}$, $T = \mathcal{O}_d(1)$, $d \gg 1$

Projections
on $p = \mathcal{O}_d(1)$ random directions

Link function
using $g : \mathbb{R}^p \to \mathbb{R}^{T\times T}$

1. Generate dataset $\mathscr{D} = \{X_\mu, y_\mu\}_{\mu=1,\ldots,n}$

$$y_\mu = g(w_1^\star X_\mu^\top, \ldots, w_p^\star X_\mu^\top)$$

2. Learn the function $x \mapsto y$

Bayes Optimal

Empirical Risk Minimization

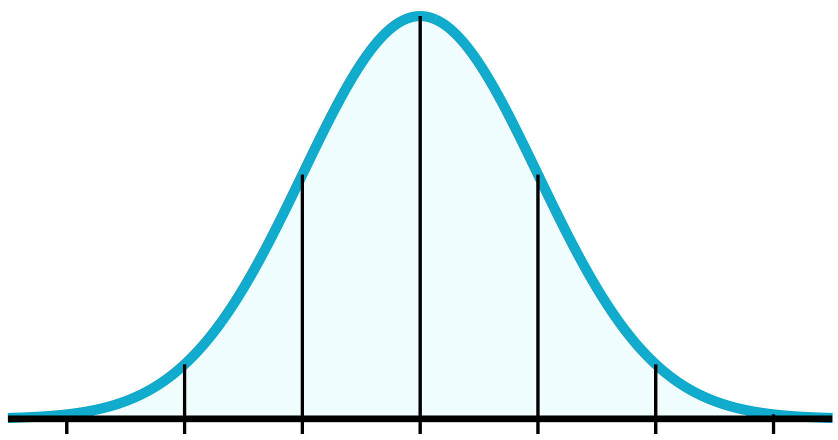
$$y_\mu = \sigma\left(X_\mu \sum_{i=1}^{p} \boldsymbol{w}_i \boldsymbol{w}_i^\top X_\mu^\top \right)$$

Tied keys-queries

[H. Cui, F. Behrens, F. Krzakala, L. Zdeborová, ’24]
[L. Arnaboldi, B. Loureiro, L. Stephan, F. Krzakala, L. Zdeborová, ’25]
[H. Cui, ’25]

Also applies to multi-layer attention!

Multi-index models… are ideal for attention



Input data

Collections of Gaussian tokens $X_\mu \in \mathbb{R}^{T \times d}$, $T = \mathcal{O}_d(1)$, $d \gg 1$

Projections

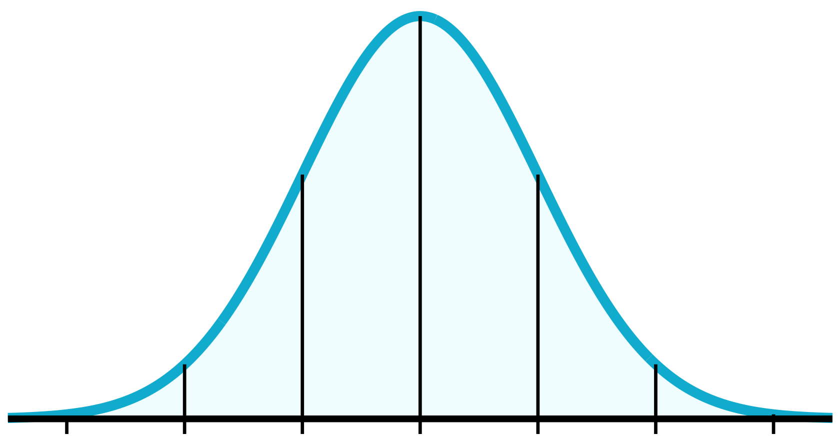
$P = p_1 + \ldots + p_L = \mathcal{O}_d(1)$ random directions

Link function

using $g : \mathbb{R}^P \to \mathbb{R}^{T \times T}$

$$y_\mu = \sigma\left(X_\mu^L \sum_{i=1}^{p_L} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X_\mu^{L\top} \right) \qquad X_\mu^{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\mu^\ell \sum_{i=1}^{p_\ell} \boldsymbol{w}_i^\ell \boldsymbol{w}_i^{\ell\top} X_\mu^{\ell\top} \right) \right] X_\mu^\ell$$

Multi-index models… are ideal for attention



Input data

Collections of Gaussian tokens $X_\mu \in \mathbb{R}^{T\times d}$, $T = \mathcal{O}_d(1)$, $d \gg 1$

Projections

$P = p_1 + \ldots + p_L = \mathcal{O}_d(1)$ random directions

Link function

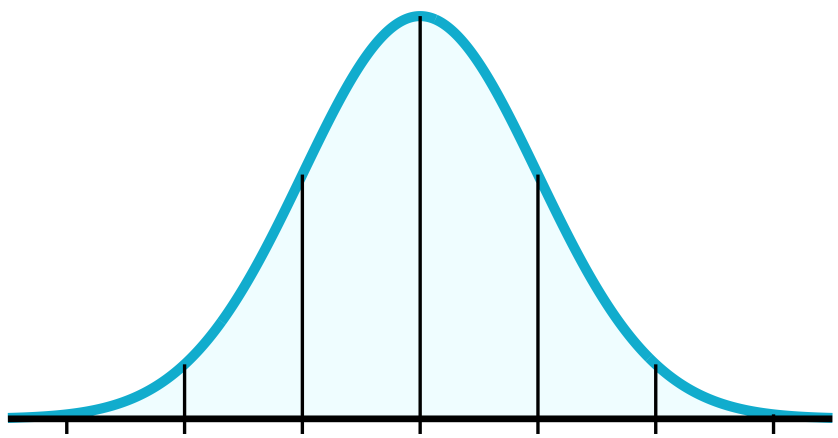
using $g : \mathbb{R}^P \to \mathbb{R}^{T\times T}$

$$y_\mu = \sigma\left(X_\mu^L \sum_{i=1}^{p_L} \mathbf{w}_i^L {\mathbf{w}_i^L}^\top {X_\mu^L}^\top\right) \qquad X_\mu^{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\mu^\ell \sum_{i=1}^{p_\ell} \mathbf{w}_i^\ell {\mathbf{w}_i^\ell}^\top {X_\mu^\ell}^\top\right)\right] X_\mu^\ell$$

$$y_\mu = g\left(X_\mu \sum_{i=1}^{p_1} \mathbf{w}_i^1 {\mathbf{w}_i^1}^\top X_\mu^\top, \ldots, X_\mu \sum_{i=1}^{p_L} \mathbf{w}_i^L {\mathbf{w}_i^L}^\top X_\mu^\top\right)$$

Exact g found by recursion

Multi-index models… are ideal for attention



Input data

Collections of Gaussian tokens $X_\mu \in \mathbb{R}^{T\times d}$, $T = \mathcal{O}_d(1)$, $d \gg 1$

Projections

$P = p_1 + \ldots + p_L = \mathcal{O}_d(1)$ random directions

Link function

using $g : \mathbb{R}^P \to \mathbb{R}^{T\times T}$

$$y_\mu = \sigma\left(X_\mu^L \sum_{i=1}^{p_L} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X_\mu^{L\top} \right) \qquad X_\mu^{\ell+1} = \Big[c\mathbf{1}_T + \ldots$$

$$y_\mu = g\left(X_\mu \sum_{i=1}^{p_1} \boldsymbol{w}_i^1 \boldsymbol{w}_i^{1\top} X_\mu^\top, \ldots, X_\mu \sum_{i=1}^{p_L} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X_\mu^\top \right)$$

Exact g found by recursion

Also multiple heads,
untied keys-queries, ecc…

[E. Troiani, H. Cui, Y. Dandi, F. Krzakala, L. Zdeborová, '25]

yet these are **finite rank…**
attention networks!

Multi-index models… are limited

Can we remove the $P \ll d$ assumption?

Multi-index models… are limited

Can we remove the $P \ll d$ assumption?

IDEA: the attention is an index

Define a new class of models

Multi-index models… are limited

Can we remove the $P \ll d$ assumption?

IDEA: the attention is an index

Define a new class of models

$$y_\mu = g(\boldsymbol{w}^\top \boldsymbol{x}_\mu)$$

"Linear" index model

$\boldsymbol{x}_\mu \in \mathbb{R}^d$

$d \to \infty$

Multi-index models… are limited

Can we remove the $P \ll d$ assumption?

IDEA: the attention is an index

Define a new class of models

$$y_\mu = g(\boldsymbol{w}^\top \boldsymbol{x}_\mu)$$

"Linear" index model

$\boldsymbol{x}_\mu \in \mathbb{R}^d$
$d \to \infty$

$$y_\mu = g(X_\mu^\top A X_\mu)$$

"Attention" index model

$X_\mu \in \mathbb{R}^{T\times d}$

$\mathrm{rank}(A) = \rho d$
$d \to \infty$

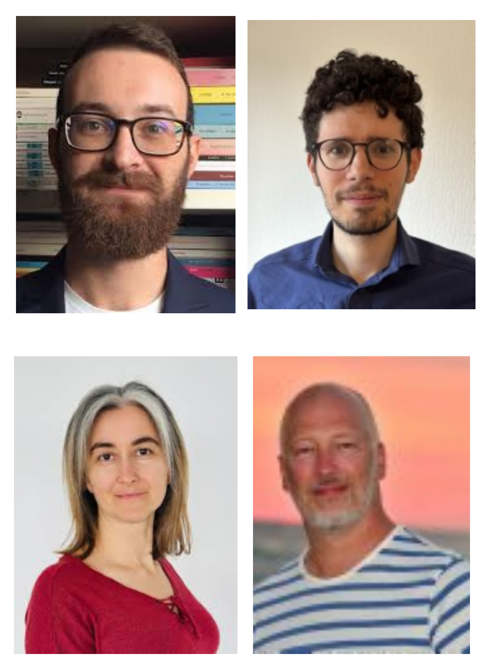
Structure of the talk

I

Bayes-Optimal learning of quadratic networks

II

Asymptotics of ERM in overparametrised quadratic networks



III

Bayes-Optimal of attention networks

Simplest possible model: quadratic networks

Data model:

Proportional-width
two-layer networks with centered quadratic activations

$$y_\mu = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}_\mu}{\sqrt{d}}\right)$$

$$\sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \frac{(\boldsymbol{w}_i^\top \boldsymbol{x})^2}{d} - \frac{\|w_i\|^2}{d}$$

Simplest possible model: quadratic networks

Data model:

Proportional-width
two-layer networks with centered quadratic activations

$$y_\mu = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}_\mu}{\sqrt{d}}\right) = \mathrm{Tr}\left[\frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sum_{i=1}^{p} \frac{\boldsymbol{w}_i \boldsymbol{w}_i^\top}{\sqrt{pd}}\right]$$

$$\sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \frac{(\boldsymbol{w}_i^\top \boldsymbol{x})^2}{d} - \frac{\|w_i\|^2}{d}$$

Simplest possible model: quadratic networks

Data model:

Proportional-width
two-layer networks with centered quadratic activations

$$y_\mu = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}_\mu}{\sqrt{d}}\right) = \mathrm{Tr}\left[\frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sum_{i=1}^{p} \frac{\boldsymbol{w}_i \boldsymbol{w}_i^\top}{\sqrt{pd}}\right] = \mathrm{Tr}[G_\mu A]$$

$$\sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \frac{(\boldsymbol{w}_i^\top \boldsymbol{x})^2}{d} - \frac{\|w_i\|^2}{d}$$

Simplest possible model: quadratic networks

Data model:

Proportional-width
two-layer networks with centered quadratic activations

$$y_\mu = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}_\mu}{\sqrt{d}}\right) = \mathrm{Tr}\left[\frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sum_{i=1}^{p} \frac{\boldsymbol{w}_i \boldsymbol{w}_i^\top}{\sqrt{pd}}\right] = \mathrm{Tr}[G_\mu A]$$

$$\sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \frac{(\boldsymbol{w}_i^\top \boldsymbol{x})^2}{d} - \frac{\|w_i\|^2}{d}$$

Goal: Bayes optimal study of simplest index model

$$y_\mu = \boldsymbol{x}_\mu^\top A^\star \boldsymbol{x}_\mu \qquad A^\star \in \mathrm{Sym}(d), \quad \mathrm{rank}(A^\star) = \rho d, \;\; \boldsymbol{x} \in \mathbb{R}^d, \; d \to \infty$$

Simplest possible model: quadratic networks

Data model:

Proportional-width
two-layer networks with centered quadratic activations

$$y_\mu = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}_\mu}{\sqrt{d}}\right) = \mathrm{Tr}\left[\frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sum_{i=1}^{p} \frac{\boldsymbol{w}_i \boldsymbol{w}_i^\top}{\sqrt{pd}}\right] = \mathrm{Tr}[G_\mu A]$$

$$\sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \frac{(\boldsymbol{w}_i^\top \boldsymbol{x})^2}{d} - \frac{\|w_i\|^2}{d}$$

Goal: Bayes optimal study of simplest index model

$$y_\mu = \boldsymbol{x}_\mu^\top A^\star \boldsymbol{x}_\mu \qquad A^\star \in \mathrm{Sym}(d), \quad \mathrm{rank}(A^\star) = \rho d, \ \ \boldsymbol{x} \in \mathbb{R}^d, \, d \to \infty$$

Results:

Fundamental limits :

You have $n = \alpha d^2$ samples $\{\boldsymbol{x}_\mu, y_\mu\}$

What is the minimal $\mathscr{E} = \|\hat{A} - A^\star\|_F$ over choices of $\hat{A}$?

Simplest possible model: quadratic networks

Data model:

Proportional-width
two-layer networks with
centered quadratic
activations

$$y_\mu = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}_\mu}{\sqrt{d}}\right) = \mathrm{Tr}\left[\frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sum_{i=1}^{p} \frac{\boldsymbol{w}_i \boldsymbol{w}_i^\top}{\sqrt{pd}}\right] = \mathrm{Tr}[G_\mu A]$$

$$\sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \frac{(\boldsymbol{w}_i^\top \boldsymbol{x})^2}{d} - \frac{\|w_i\|^2}{d}$$

Goal: Bayes optimal study of simplest index model

$$y_\mu = \boldsymbol{x}_\mu^\top A^\star \boldsymbol{x}_\mu \qquad A^\star \in \mathrm{Sym}(d), \quad \mathrm{rank}(A^\star) = \rho d, \; \boldsymbol{x} \in \mathbb{R}^d, d \to \infty$$

Results:

Fundamental limits :

You have $n = \alpha d^2$ samples $\{\boldsymbol{x}_\mu, y_\mu\}$

What is the minimal $\mathscr{E} = \|\hat{A} - A^\star\|_F$ over choices of $\hat{A}$?

Optimal algorithms :

Write an algorithm that achieves $\mathscr{E}$

Sketch of the analysis though AMP

1. <u>Gaussian equivalence on input</u>

$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA]$$

$$G_\mu = \frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sim \mathrm{GOE}(d)$$

Matrix compressed sensing
[Y. Xu, A. Maillard, L. Zdeborová, F. Krzakala]

Sketch of the analysis though AMP

1. Gaussian equivalence on input

$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA]$$

$$G_\mu = \frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sim \mathrm{GOE}(d)$$

Matrix compressed sensing
[Y. Xu, A. Maillard, L. Zdeborová, F. Krzakala]

2. Write an AMP iteration

$$A^{t+1} = \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

$$h_\mu^{t+1} = y_\mu - \mathrm{Tr}[G_\mu A^{t+1}] + \nabla \cdot \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

Sketch of the analysis though AMP

$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA]$$

1. **Gaussian equivalence on input**

$$G_\mu = \frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sim \mathrm{GOE}(d)$$

Matrix compressed sensing
[Y. Xu, A. Maillard, L. Zdeborová, F. Krzakala]

2. **Write an AMP iteration**

$$A^{t+1} = \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

Denoising subproblem

$$h_\mu^{t+1} = y_\mu - \mathrm{Tr}[G_\mu A^{t+1}] + \nabla \cdot \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

Sketch of the analysis though AMP

1. Gaussian equivalence on input

$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA]$$

$$G_\mu = \frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sim \mathrm{GOE}(d)$$

Matrix compressed sensing
[Y. Xu, A. Maillard, L. Zdeborová, F. Krzakala]

2. Write an AMP iteration

$$A^{t+1} = \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

Denoising subproblem

$$h_\mu^{t+1} = y_\mu - \mathrm{Tr}[G_\mu A^{t+1}] + \nabla \cdot \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

3. Study BO denoising

Observe $\mathscr{Y} = \mathscr{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathscr{S}}$ rotationally invariant

Sketch of the analysis though AMP

1. **Gaussian equivalence on input**

$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA]$$

$$G_\mu = \frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sim \mathrm{GOE}(d)$$

Matrix compressed sensing
[Y. Xu, A. Maillard, L. Zdeborová, F. Krzakala]

2. **Write an AMP iteration**

$$A^{t+1} = \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

Denoising subproblem

$$h_\mu^{t+1} = y_\mu - \mathrm{Tr}[G_\mu A^{t+1}] + \nabla \cdot \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

3. **Study BO denoising**

Observe $\mathcal{Y} = \mathcal{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathcal{S}}$ rotationally invariant

Estimate $\mathcal{S}$!

Sketch of the analysis though AMP

1. Gaussian equivalence on input

$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA]$$

$$G_\mu = \frac{\boldsymbol{x}_\mu \boldsymbol{x}_\mu^\top - \mathbf{1}}{\sqrt{d}} \sim \mathrm{GOE}(d)$$

Matrix compressed sensing
[Y. Xu, A. Maillard, L. Zdeborová, F. Krzakala]

2. Write an AMP iteration

$$A^{t+1} = \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

Denoising subproblem

$$h_\mu^{t+1} = y_\mu - \mathrm{Tr}[G_\mu A^{t+1}] + \nabla \cdot \eta\left(A^t + \sum_{\mu=1}^{N} G_\mu h_\mu^t\right)$$

3. Study BO denoising

Observe $\mathscr{Y} = \mathscr{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathscr{S}}$ rotationally invariant

Estimate $\mathscr{S}$!

Not quite a trivial problem…

[A. Maillard, F. Krzakala, M. Mézard, L. Zdeborová '21]
[E. Troiani, V. Erba, F. Krzakala, A. Maillard, L. Zdeborová '22]
[F. Pourkamali, N. Macris '23]
[G. Semerjian '24]

BO denoising and HCIZ integrals

Observe $\mathscr{Y} = \mathscr{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathscr{S}}$ rotationally invariant

Minimise MSE on $\mathscr{S}$!

BO denoising and HCIZ integrals

Observe $\mathcal{Y} = \mathcal{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathcal{S}}$ rotationally invariant

Minimise MSE on $\mathcal{S}$!

<u>Rotationally Invariant Estimator (RIE)</u>

[O. Ledoit, S. Péché '11]
[J. Bun, J.P. Bouchaud, M. Potters '16]

BO denoising and HCIZ integrals

Observe $\mathscr{Y} = \mathscr{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathscr{S}}$ rotationally invariant
Minimise MSE on $\mathscr{S}$!

Rotationally Invariant Estimator (RIE)

[O. Ledoit, S. Péché '11]
[J. Bun, J.P. Bouchaud, M. Potters '16]

$$\mathscr{Y} = U\Lambda_{\mathscr{Y}}U^\top$$

$$\hat{\mathscr{S}} = UF(\Lambda_{\mathscr{Y}})U^\top$$

$$F[\Lambda_{\mathscr{Y}}]_i = f_{\mathscr{Y}}[(\Lambda_{\mathscr{Y}})_i]$$

BO denoising and HCIZ integrals

Observe $\mathcal{Y} = \mathcal{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathcal{S}}$ rotationally invariant
Minimise MSE on $\mathcal{S}$!

Rotationally Invariant Estimator (RIE)

[O. Ledoit, S. Péché '11]
[J. Bun, J.P. Bouchaud, M. Potters '16]

$$\mathcal{Y} = U \Lambda_{\mathcal{Y}} U^\top$$

$$\hat{\mathcal{S}} = U F(\Lambda_{\mathcal{Y}}) U^\top$$

$$F[\Lambda_{\mathcal{Y}}]_i = f_{\mathcal{Y}}[(\Lambda_{\mathcal{Y}})_i]$$

Integrate posterior over rotations

$$P(\mathcal{S} \,|\, \mathcal{Y}) \propto e^{-\frac{\|\mathcal{Y} - \mathcal{S}\|^2}{2\delta^2}} P(\mathcal{S}) = e^{-\frac{\|\mathcal{Y}\|^2 + \|\mathcal{S}\|^2}{2\delta^2}} P(\mathcal{S})\ e^{-\frac{\mathrm{Tr}[\mathcal{Y}\mathcal{S}]}{\delta^2}}$$

BO denoising and HCIZ integrals

Observe $\mathcal{Y} = \mathcal{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathcal{S}}$ rotationally invariant
Minimise MSE on $\mathcal{S}$!

Rotationally Invariant Estimator (RIE)

[O. Ledoit, S. Péché '11]
[J. Bun, J.P. Bouchaud, M. Potters '16]

$$\mathcal{Y} = U\Lambda_{\mathcal{Y}}U^\top \longrightarrow \hat{\mathcal{S}} = UF(\Lambda_{\mathcal{Y}})U^\top$$

$$F[\Lambda_{\mathcal{Y}}]_i = f_{\mathcal{Y}}[(\Lambda_{\mathcal{Y}})_i]$$

Integrate posterior over rotations

$$P(\mathcal{S}\,|\,\mathcal{Y}) \propto e^{-\frac{\|\mathcal{Y}-\mathcal{S}\|^2}{2\delta^2}} P(\mathcal{S}) = e^{-\frac{\|\mathcal{Y}\|^2+\|\mathcal{S}\|^2}{2\delta^2}} P(\mathcal{S})\, e^{-\frac{\mathrm{Tr}[\mathcal{Y}\mathcal{S}]}{\delta^2}}$$

HCIZ!

$$\int e^{-\frac{\mathrm{Tr}[U\Lambda_{\mathcal{Y}}U^\top\Lambda_{\mathcal{S}}]}{\delta^2}}\,\mathrm{d}U$$

Recall A. Maillard's talk…

BO denoising and HCIZ integrals

Observe $\mathcal{Y} = \mathcal{S} + \delta Z$ with $Z \sim \mathrm{GOE}(d)$ and $P_{\mathcal{S}}$ rotationally invariant
Minimise MSE on $\mathcal{S}$!

Rotationally Invariant Estimator (RIE)

[O. Ledoit, S. Péché '11]
[J. Bun, J.P. Bouchaud, M. Potters '16]

$$\mathcal{Y} = U\Lambda_{\mathcal{Y}}U^\top \longrightarrow \hat{\mathcal{S}} = UF(\Lambda_{\mathcal{Y}})U^\top$$

$$F[\Lambda_{\mathcal{Y}}]_i = f_{\mathcal{Y}}[(\Lambda_{\mathcal{Y}})_i]$$

Integrate posterior over rotations

$$P(\mathcal{S}\,|\,\mathcal{Y}) \propto e^{-\frac{\|\mathcal{Y}-\mathcal{S}\|^2}{2\delta^2}}P(\mathcal{S}) = e^{-\frac{\|\mathcal{Y}\|^2+\|\mathcal{S}\|^2}{2\delta^2}}P(\mathcal{S})\,e^{-\frac{\mathrm{Tr}[\mathcal{Y}\mathcal{S}]}{\delta^2}}$$

HCIZ!

Asymptotic Minimum MSE

$$\|\hat{\mathcal{S}} - \mathcal{S}\|_F \xrightarrow{d\to\infty} \delta^2 - \frac{4\pi^2\delta^4}{3}\int \mu_{\mathcal{Y}}(\lambda)^3\,\mathrm{d}\lambda$$

$$\int e^{-\frac{\mathrm{Tr}[U\Lambda_{\mathcal{Y}}U^\top\Lambda_{\mathcal{S}}]}{\delta^2}}\,\mathrm{d}U$$

Recall A. Maillard's talk...

Learning curves for quadratic networks

Minimal MMSE

$$\sum_{\mu=1}^{n} (\hat{y}[x_\mu^{\text{new}}] - y_\mu^{\text{new}})^2 \xrightarrow{d\to\infty} \frac{2\alpha\rho}{\hat{q}} \qquad 1 - 2\alpha = \frac{4\pi^2}{3\hat{q}} \int \mu_{\mathscr{Y}}(\lambda)^3 \, \mathrm{d}\lambda$$

$$\mathscr{Y} = A + \hat{q}^{-1/2} G$$

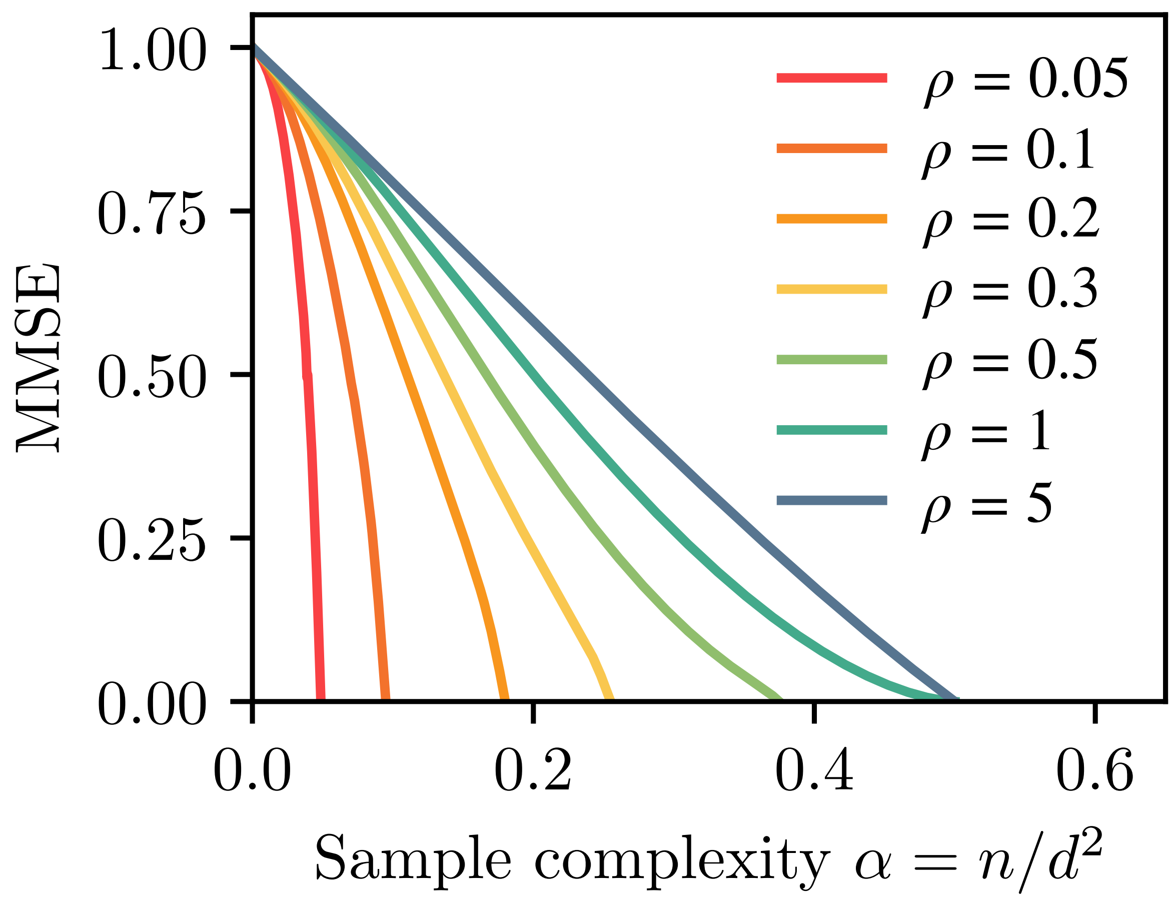
$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA] \qquad G \sim \mathrm{GOE}(d) \quad A \sim \mathscr{W}_{p,d} \qquad p = \rho d \quad n = \alpha d^2$$

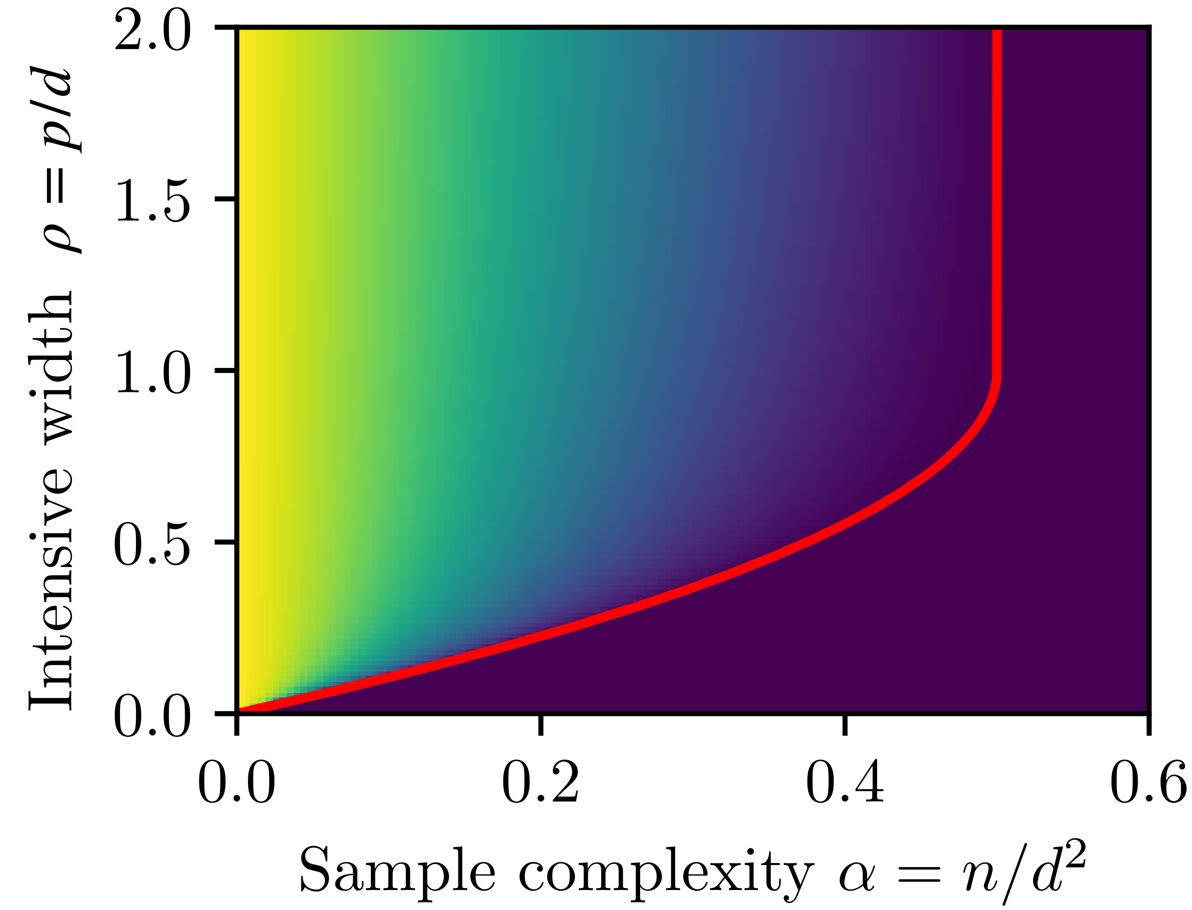
Learning curves for quadratic networks

Minimal MMSE

$$\sum_{\mu=1}^{n} (\hat{y}[x_\mu^{\text{new}}] - y_\mu^{\text{new}})^2 \xrightarrow{d\to\infty} \frac{2\alpha\rho}{\hat{q}} \qquad 1 - 2\alpha = \frac{4\pi^2}{3\hat{q}} \int \mu_{\mathcal{Y}}(\lambda)^3 \, \mathrm{d}\lambda$$

$$\mathcal{Y} = A + \hat{q}^{-1/2} G$$





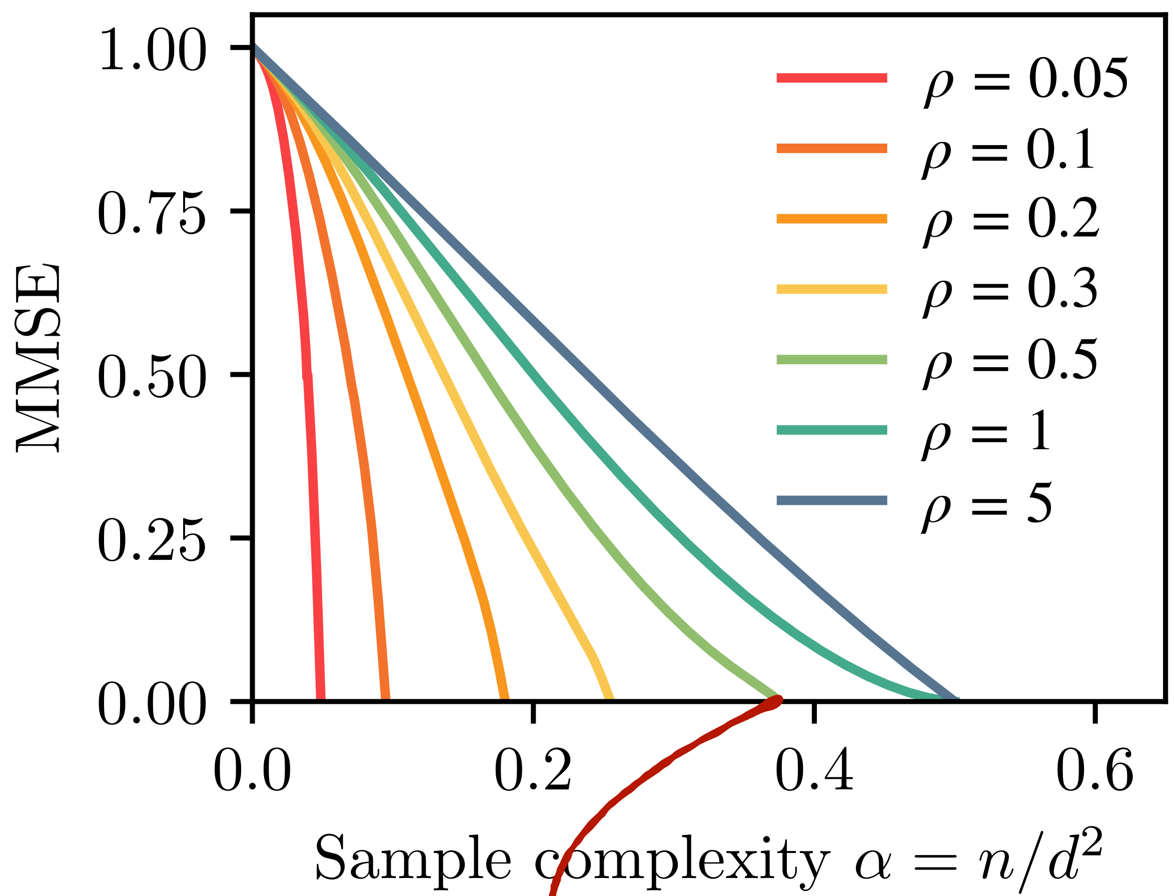
$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA] \qquad \begin{array}{ll} G \sim \mathrm{GOE}(d) & p = \rho d \\ A \sim \mathcal{W}_{p,d} & n = \alpha d^2 \end{array}$$

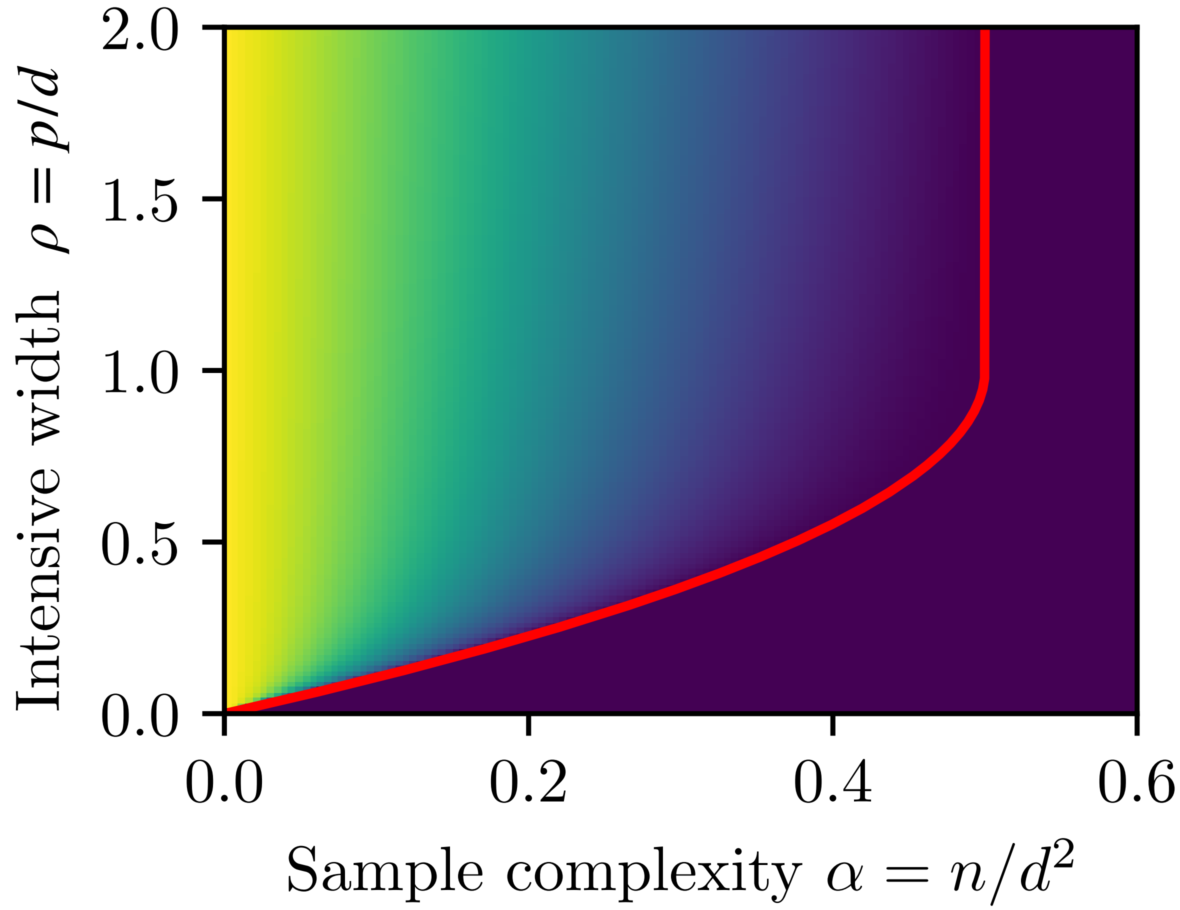
Learning curves for quadratic networks

Minimal MMSE

$$\sum_{\mu=1}^{n} (\hat{y}[x_\mu^{\text{new}}] - y_\mu^{\text{new}})^2 \xrightarrow{d\to\infty} \frac{2\alpha\rho}{\hat{q}} \qquad 1 - 2\alpha = \frac{4\pi^2}{3\hat{q}} \int \mu_{\mathcal{Y}}(\lambda)^3 \, \mathrm{d}\lambda$$

$\mathcal{Y} = A + \hat{q}^{-1/2} G$





$\alpha_c = \rho - \rho^2/2$

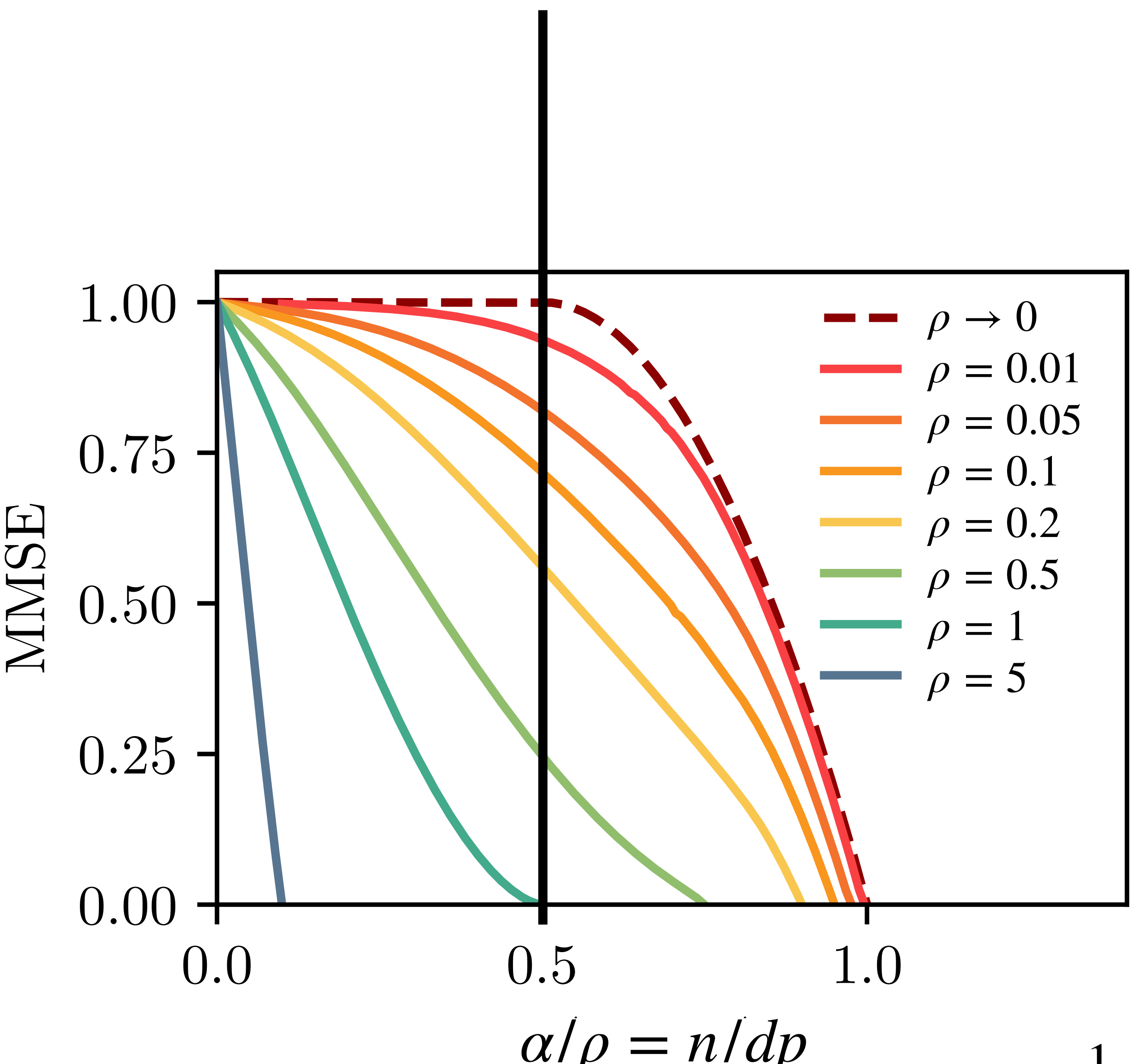
$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA] \qquad \begin{array}{l} G \sim \mathrm{GOE}(d) \\ A \sim \mathscr{W}_{p,d} \end{array} \qquad \begin{array}{l} p = \rho d \\ n = \alpha d^2 \end{array}$$

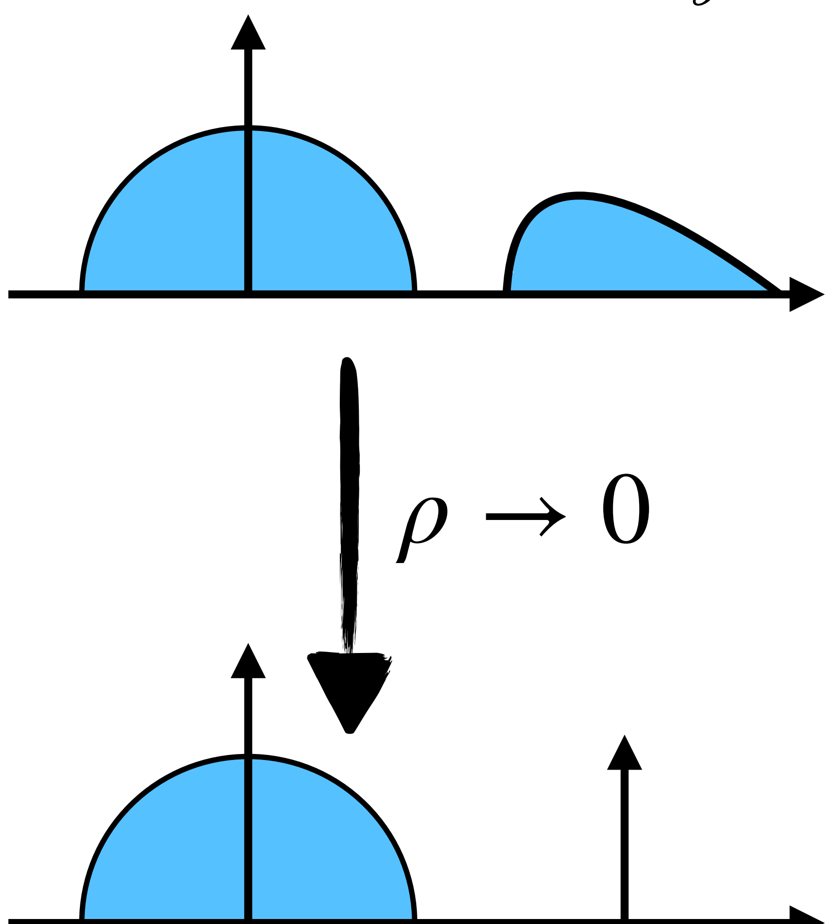
Narrow limit: a BBP transition for weak recovery

Narrow width limit $\rho \to 0$

$$1 - 2\alpha = \frac{4\pi^2}{3\hat{q}} \int \mu_{\mathcal{Y}}(\lambda)^3 \, \mathrm{d}\lambda$$

$$\mathcal{Y} = A + \hat{q}^{-1/2} G$$





$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA] \qquad G \sim \mathrm{GOE}(d) \qquad p = \rho d$$

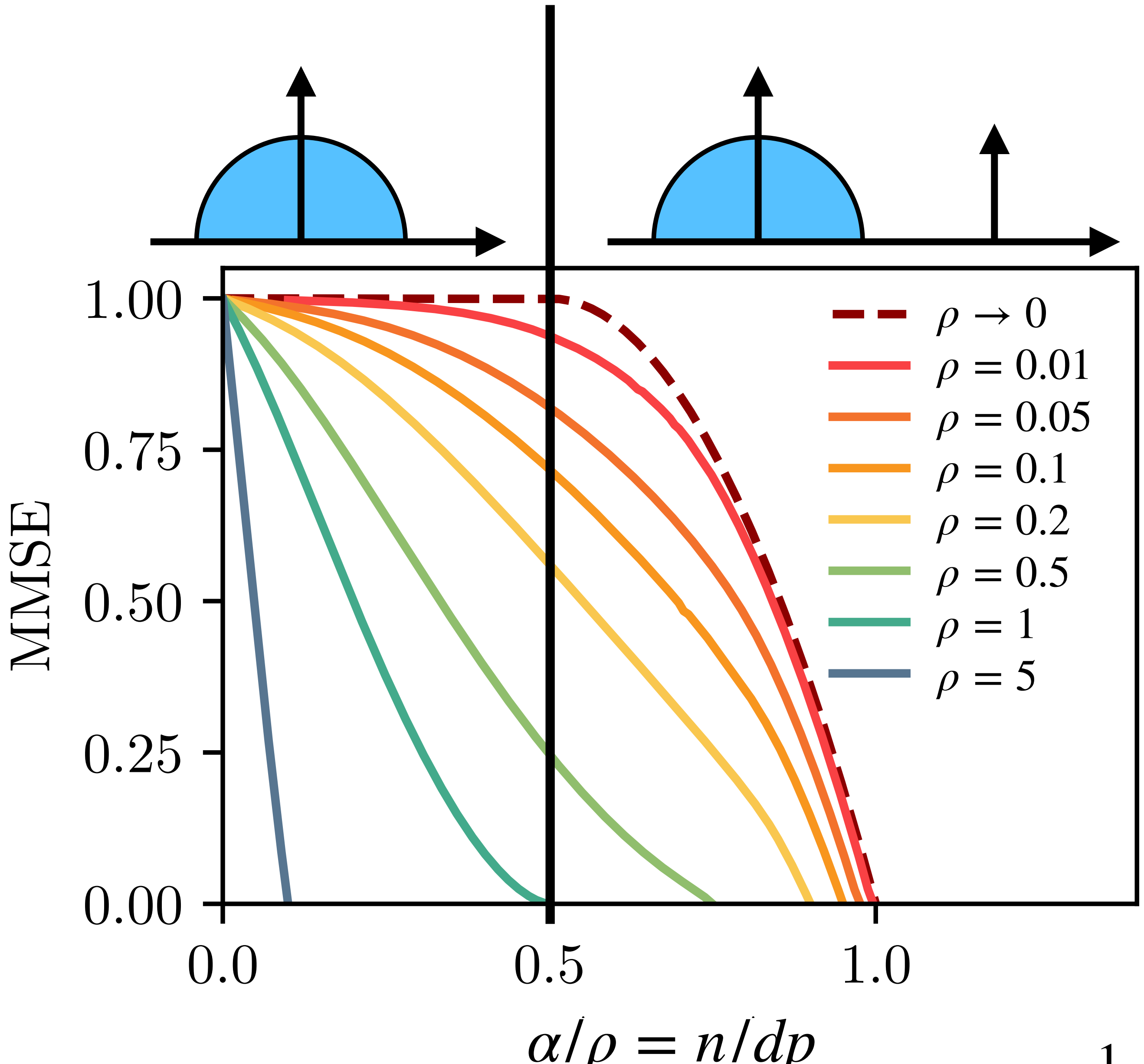
$$A \sim \mathcal{W}_{p,d} \qquad n = \alpha d^2$$

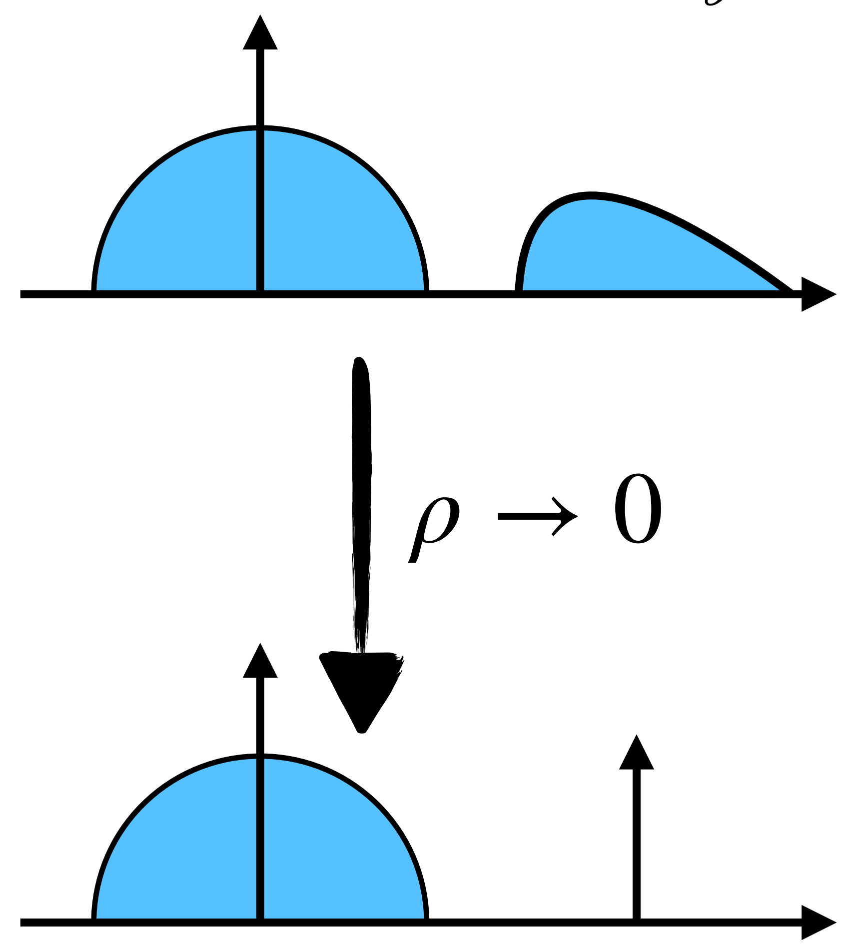
Narrow limit: a BBP transition for weak recovery

Narrow width limit $\rho \to 0$

$$1 - 2\alpha = \frac{4\pi^2}{3\hat{q}} \int \mu_{\mathscr{Y}}(\lambda)^3 \, \mathrm{d}\lambda$$

$$\mathscr{Y} = A + \hat{q}^{-1/2} G$$





$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA]$$

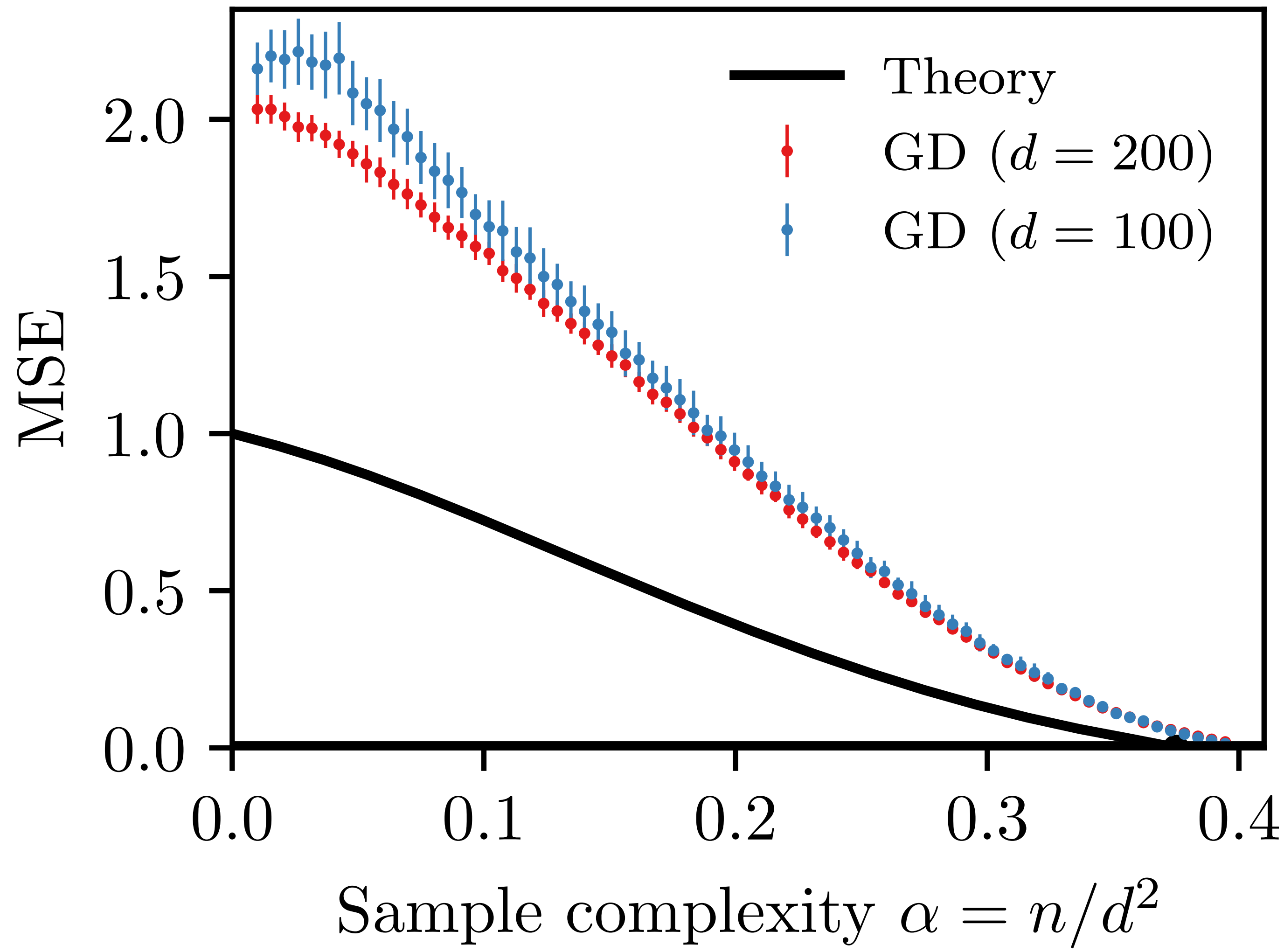
$G \sim \mathrm{GOE}(d)$ $\quad p = \rho d$

$A \sim \mathscr{W}_{p,d}$ $\quad n = \alpha d^2$

(Some form of) GD can have BO performance

Everyone: I don't care about AMP, what about GD?

Minimise $$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2$$



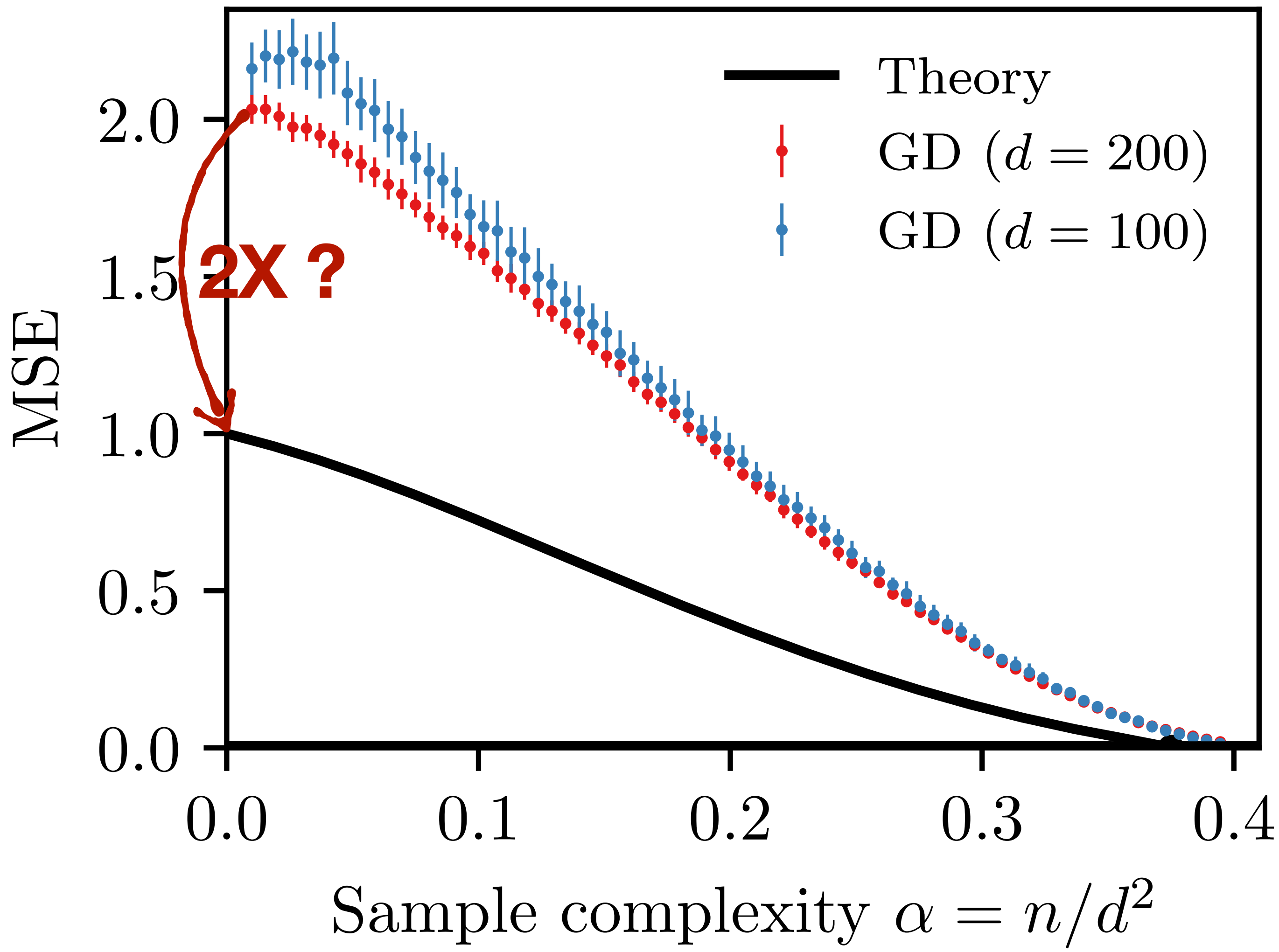
$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA^\star] \qquad \hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

$$G \sim \mathrm{GOE}(d) \qquad p = \rho d$$
$$A^\star \sim \mathscr{W}_{p,d} \qquad n = \alpha d^2$$

(Some form of) GD can have BO performance

Everyone: I don't care about AMP, what about GD?

Minimise $$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2$$



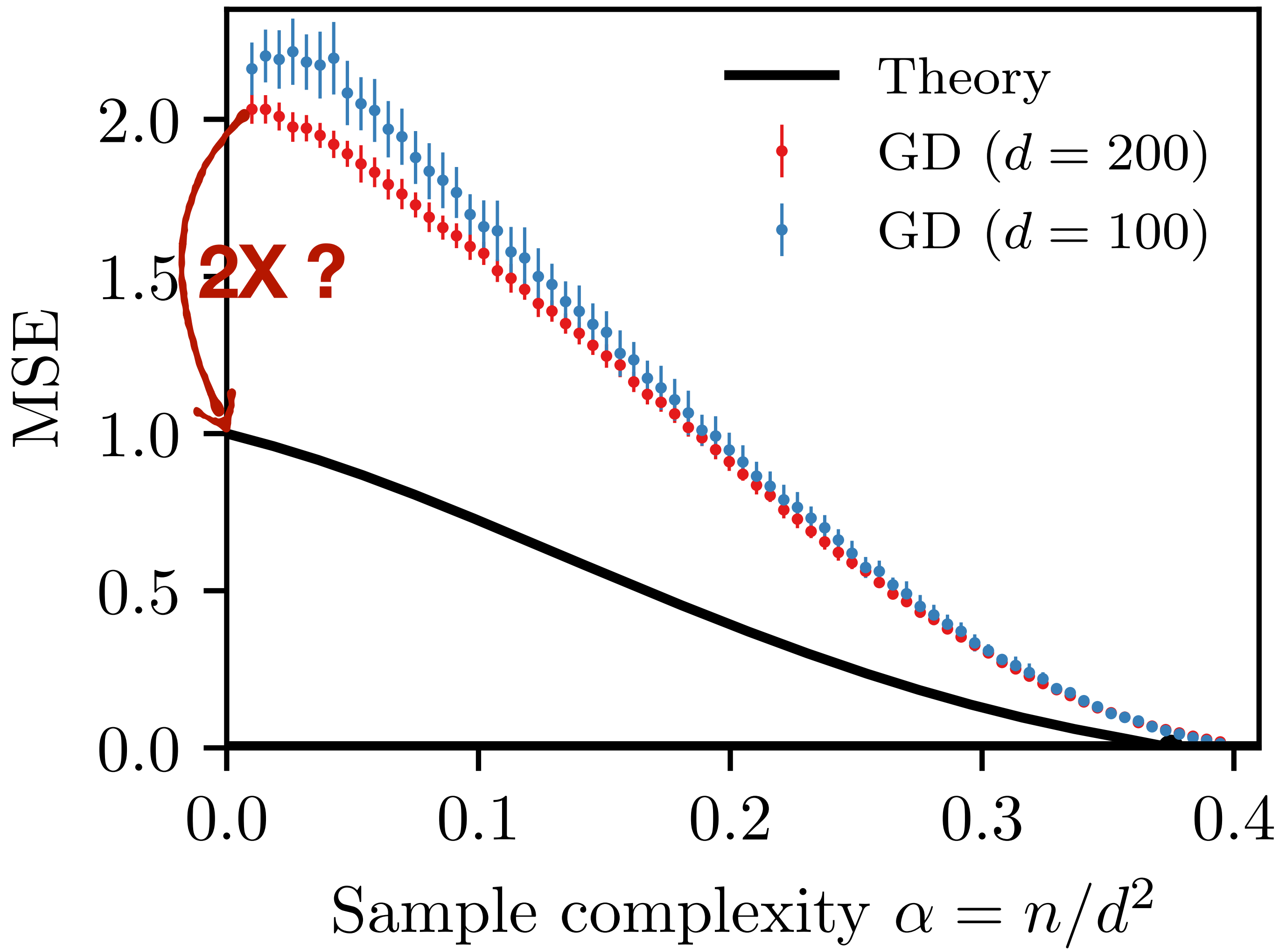
$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA^\star] \qquad \hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

$$G \sim \mathrm{GOE}(d) \qquad p = \rho d$$
$$A^\star \sim \mathscr{W}_{p,d} \qquad n = \alpha d^2$$

(Some form of) GD can have BO performance

Everyone: I don't care about AMP, what about GD?

Minimise $\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2$



Averaged GD:

1. Run multiple times initializing in prior;
2. Average the labels at convergence

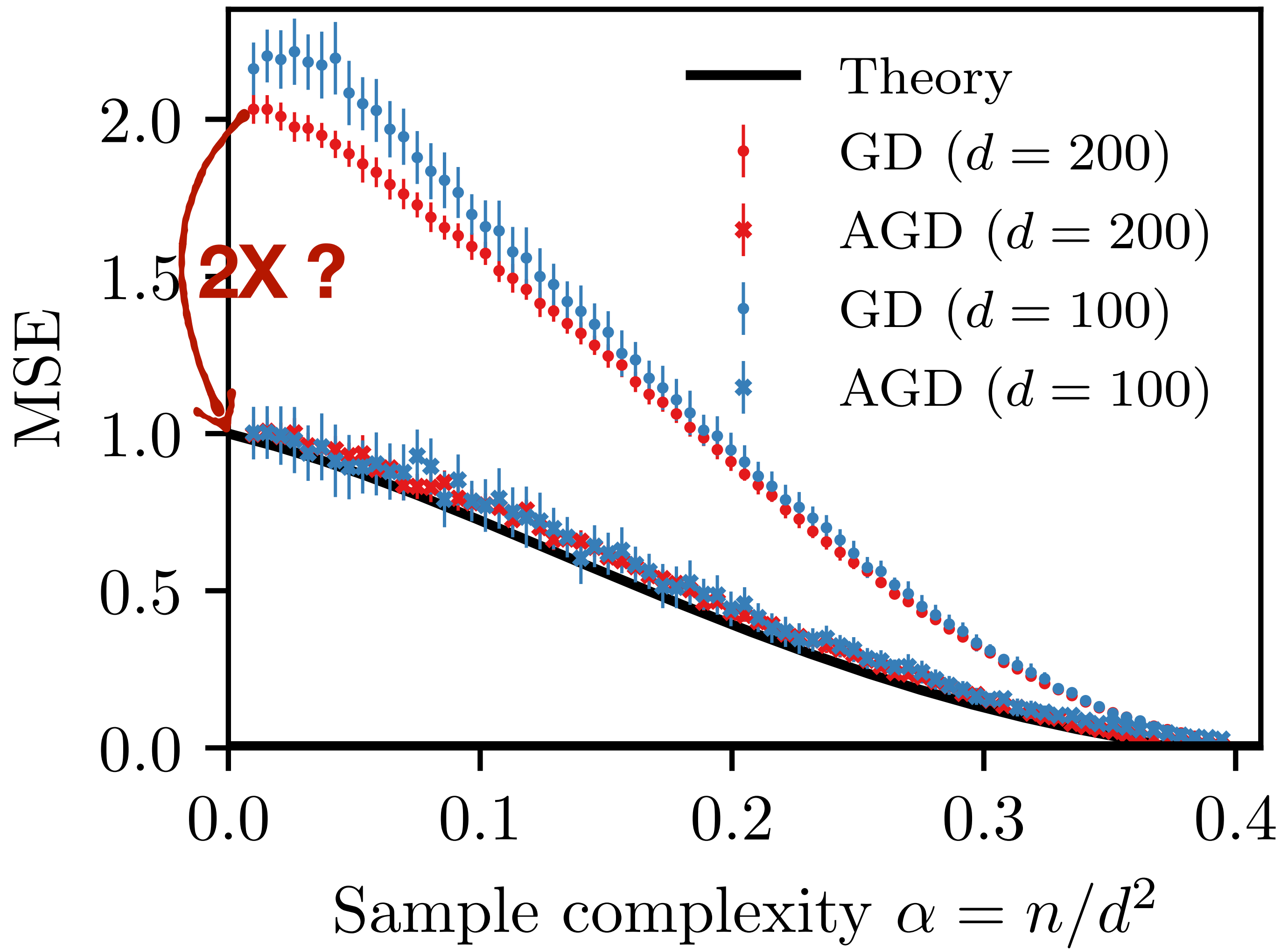
$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA^\star] \qquad \hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

$$G \sim \mathrm{GOE}(d) \qquad p = \rho d$$
$$A^\star \sim \mathscr{W}_{p,d} \qquad n = \alpha d^2$$

(Some form of) GD can have BO performance

Everyone: I don't care about AMP, what about GD?

Minimise $$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2$$



Averaged GD:

1. Run multiple times initializing in prior;
2. Average the labels at convergence

BO performance!

(hard) open problem

$$y = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\boldsymbol{w}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[GA^\star] \qquad \hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

$$G \sim \mathrm{GOE}(d) \qquad p = \rho d$$
$$A^\star \sim \mathscr{W}_{p,d} \qquad n = \alpha d^2$$

L2 regularized quadratic networks

Let's add L2 regularization and learn with overparametrized $p \gg p^\star$ network

Minimise $$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2 + \lambda \|W\|_F^2$$

$$y = \mathrm{Tr}[GA^\star] + \sqrt{\Delta}\zeta$$

$$\hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

$$G \sim \mathrm{GOE}(d)$$
$$A^\star \sim \mathscr{W}_{p^\star,d}$$

$$p \geq d$$
$$p^\star = \rho^\star d$$
$$n = \alpha d^2$$

L2 regularized quadratic networks

Let's add L2 regularization and learn with overparametrized $p \gg p^\star$ network

Minimise $$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2 + \lambda \|W\|_F^2$$

(i) Training: Optimisation with ***GD is easy***, all minima are global minima

[Venturi, Bandeira, Bruna '19]

$$y = \mathrm{Tr}[GA^\star] + \sqrt{\Delta}\zeta \qquad \hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

$$G \sim \mathrm{GOE}(d) \qquad A^\star \sim \mathscr{W}_{p^\star,d}$$

$$p \geq d \qquad p^\star = \rho^\star d \qquad n = \alpha d^2$$

L2 regularized quadratic networks

Let's add L2 regularization and learn with overparametrized $p \gg p^\star$ network

Minimise $$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2 + \lambda \|W\|_F^2$$

(i) Training: Optimisation with ***GD is easy***, all minima are global minima

[Venturi, Bandeira, Bruna '19]

(ii) Equivalent to a (convex) matrix compressed sensing with a ***nuclear norm regularisation***

[Fazel, Candes, Recht, Parrilo '08; Donoho, Gavish, Montanari '13; Gunasekar, Woodworth, Bhojanapalli, Neyshabur, Srebro '17]

$$\sum_{\mu=1}^{n} \|(A^\star - A)G_\mu + \sqrt{\Delta}\zeta\|_F^2 + \lambda \|A\|_*$$

$$y = \mathrm{Tr}[GA^\star] + \sqrt{\Delta}\zeta \qquad \hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}] \qquad \begin{matrix} G \sim \mathrm{GOE}(d) \\ A^\star \sim \mathscr{W}_{p^\star,d} \end{matrix} \qquad \begin{matrix} p \geq d \\ p^\star = \rho^\star d \\ n = \alpha d^2 \end{matrix}$$

L2 regularized quadratic networks

Let's add L2 regularization and learn with overparametrized $p \gg p^\star$ network

Minimise $$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2 + \lambda \|W\|_F^2$$

(i) Training: Optimisation with ***GD is easy***, all minima are global minima

[Venturi, Bandeira, Bruna '19]

(ii) Equivalent to a (convex) matrix compressed sensing with a ***nuclear norm regularisation***

[Fazel, Candes, Recht, Parrilo '08; Donoho, Gavish, Montanari '13; Gunasekar, Woodworth, Bhojanapalli, Neyshabur, Srebro '17]

$$\sum_{\mu=1}^{n} \|(A^\star - A)G_\mu + \sqrt{\Delta}\zeta\|_F^2 + \lambda \|A\|_*$$

(iii) NTK/Lazy is instead equivalent to matrix compressed sensing with a Frobenius ***norm regularisation***

$$\sum_{\mu=1}^{n} \|(A^\star - A)G_\mu + \sqrt{\Delta}\zeta\|_F^2 + \lambda \|A\|_F^2$$

$$y = \mathrm{Tr}[GA^\star] + \sqrt{\Delta}\zeta \qquad \hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}] \qquad G \sim \mathrm{GOE}(d), \quad A^\star \sim \mathcal{W}_{p^\star,d}$$

$$p \geq d, \quad p^\star = \rho^\star d, \quad n = \alpha d^2$$

Asymptotics of overparametrized quadratic nets

Generalization error

$$\sum_{\mu=1}^{n} (\hat{y}[x_\mu^{\text{new}}] - y_\mu^{\text{new}})^2 \xrightarrow{d\to\infty} 2\alpha\delta^2 - \frac{\Delta}{2}$$

Training loss

$$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2 \xrightarrow{d\to\infty} \frac{\delta^2}{4\epsilon^2} - \frac{\lambda}{2}\partial_2 J(\delta, \lambda\epsilon)$$

$$y = \mathrm{Tr}[GA^\star] + \sqrt{\Delta}\zeta \qquad \hat{y} = \frac{1}{\sqrt{p}}\sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

$$G \sim \mathrm{GOE}(d) \qquad A^\star \sim \mathscr{W}_{p^\star,d}$$

$$p \geq d \qquad p^\star = \rho^\star d \qquad n = \alpha d^2$$

Asymptotics of overparametrized quadratic nets

Generalization error

$$\sum_{\mu=1}^{n} (\hat{y}[x_\mu^{\text{new}}] - y_\mu^{\text{new}})^2 \xrightarrow{d\to\infty} 2\alpha\delta^2 - \frac{\Delta}{2}$$

Training loss

$$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2 \xrightarrow{d\to\infty} \frac{\delta^2}{4\epsilon^2} - \frac{\lambda}{2}\partial_2 J(\delta, \lambda\epsilon)$$

$$4\alpha\delta - \frac{\delta}{\epsilon} = \partial_1 J(\delta, \lambda\epsilon)$$

$$Q^\star + \frac{\Delta}{2} + 2\alpha\delta^2 - \frac{\delta^2}{\epsilon} = (1 - \epsilon\lambda\partial_2) J(\delta, \lambda\epsilon)$$

$$J(a,b) = \int_b^{+\infty} dx\, \mu_{\mathcal{Y}}(x)\,(x-b)^2$$

$$\mathcal{Y} = A^\star + a^{-1/2} G$$

$$y = \mathrm{Tr}[GA^\star] + \sqrt{\Delta}\zeta$$

$$\hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

$G \sim \mathrm{GOE}(d)$

$A^\star \sim \mathscr{W}_{p^\star, d}$

$p \geq d$

$p^\star = \rho^\star d$

$n = \alpha d^2$

Asymptotics of overparametrized quadratic nets

Generalization error

$$\sum_{\mu=1}^{n} (\hat{y}[x_\mu^{\text{new}}] - y_\mu^{\text{new}})^2 \xrightarrow{d\to\infty} 2\alpha\delta^2 - \frac{\Delta}{2}$$

Training loss

$$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2 \xrightarrow{d\to\infty} \frac{\delta^2}{4\epsilon^2} - \frac{\lambda}{2}\partial_2 J(\delta, \lambda\epsilon)$$

$$4\alpha\delta - \frac{\delta}{\epsilon} = \partial_1 J(\delta, \lambda\epsilon)$$

$$J(a,b) = \int_b^{+\infty} dx\, \mu_{\mathcal{Y}}(x)\,(x-b)^2$$

$$Q^\star + \frac{\Delta}{2} + 2\alpha\delta^2 - \frac{\delta^2}{\epsilon} = (1 - \epsilon\lambda\partial_2) J(\delta, \lambda\epsilon)$$

$$\mathcal{Y} = A^\star + a^{-1/2} G$$

Spectrum of $\hat{A}$ at convergence $\quad \mu_{\hat{W}}(x) = C\delta(x) + I(x>0)\left[2x\mu_{\mathcal{Y}}(x^2 + \lambda\epsilon)\right]$

$$y = \text{Tr}[GA^\star] + \sqrt{\Delta}\zeta \qquad \hat{y} = \frac{1}{\sqrt{p}}\sum_{i=1}^{p}\sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \text{Tr}[G\hat{A}]$$

$$G \sim \text{GOE}(d) \qquad A^\star \sim \mathcal{W}_{p^\star,d}$$

$$p \geq d \qquad p^\star = \rho^\star d \qquad n = \alpha d^2$$

Asymptotics of overparametrized quadratic nets

Generalization error

$$\sum_{\mu=1}^{n} (\hat{y}[x_\mu^{\text{new}}] - y_\mu^{\text{new}})^2 \xrightarrow{d\to\infty} 2\alpha\delta^2 - \frac{\Delta}{2}$$

Training loss

$$\sum_{\mu=1}^{n} (y_\mu - \hat{y}_\mu[W])^2 \xrightarrow{d\to\infty} \frac{\delta^2}{4\epsilon^2} - \frac{\lambda}{2}\partial_2 J(\delta, \lambda\epsilon)$$

$$4\alpha\delta - \frac{\delta}{\epsilon} = \partial_1 J(\delta, \lambda\epsilon)$$

$$Q^\star + \frac{\Delta}{2} + 2\alpha\delta^2 - \frac{\delta^2}{\epsilon} = (1 - \epsilon\lambda\partial_2) J(\delta, \lambda\epsilon)$$

$$J(a,b) = \int_b^{+\infty} dx\, \mu_{\mathscr{Y}}(x)\,(x-b)^2$$

$$\mathscr{Y} = A^\star + a^{-1/2} G$$

Spectrum of $\hat{A}$ at convergence $\quad \mu_{\hat{W}}(x) = C\delta(x) + I(x>0)\left[2x\mu_{\mathscr{Y}}(x^2 + \lambda\epsilon)\right]$

<u>Analysis similar to BO:</u>

Different spectral denoiser: shifted ReLU on eigenvalues

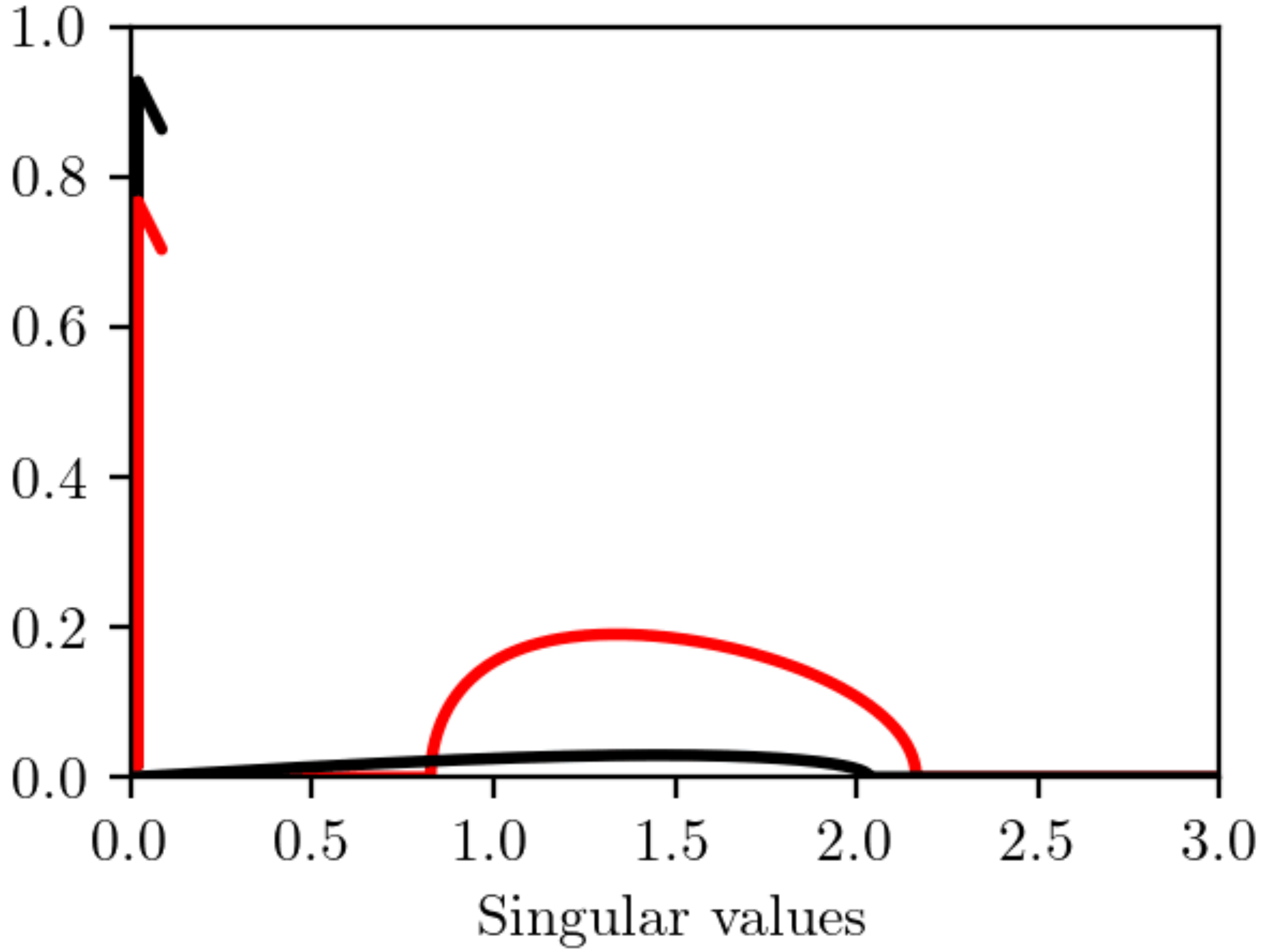
$$y = \mathrm{Tr}[GA^\star] + \sqrt{\Delta}\zeta \qquad \hat{y} = \frac{1}{\sqrt{p}} \sum_{i=1}^{p} \sigma\left(\frac{\hat{\boldsymbol{w}}_i^\top \boldsymbol{x}}{\sqrt{d}}\right) = \mathrm{Tr}[G\hat{A}]$$

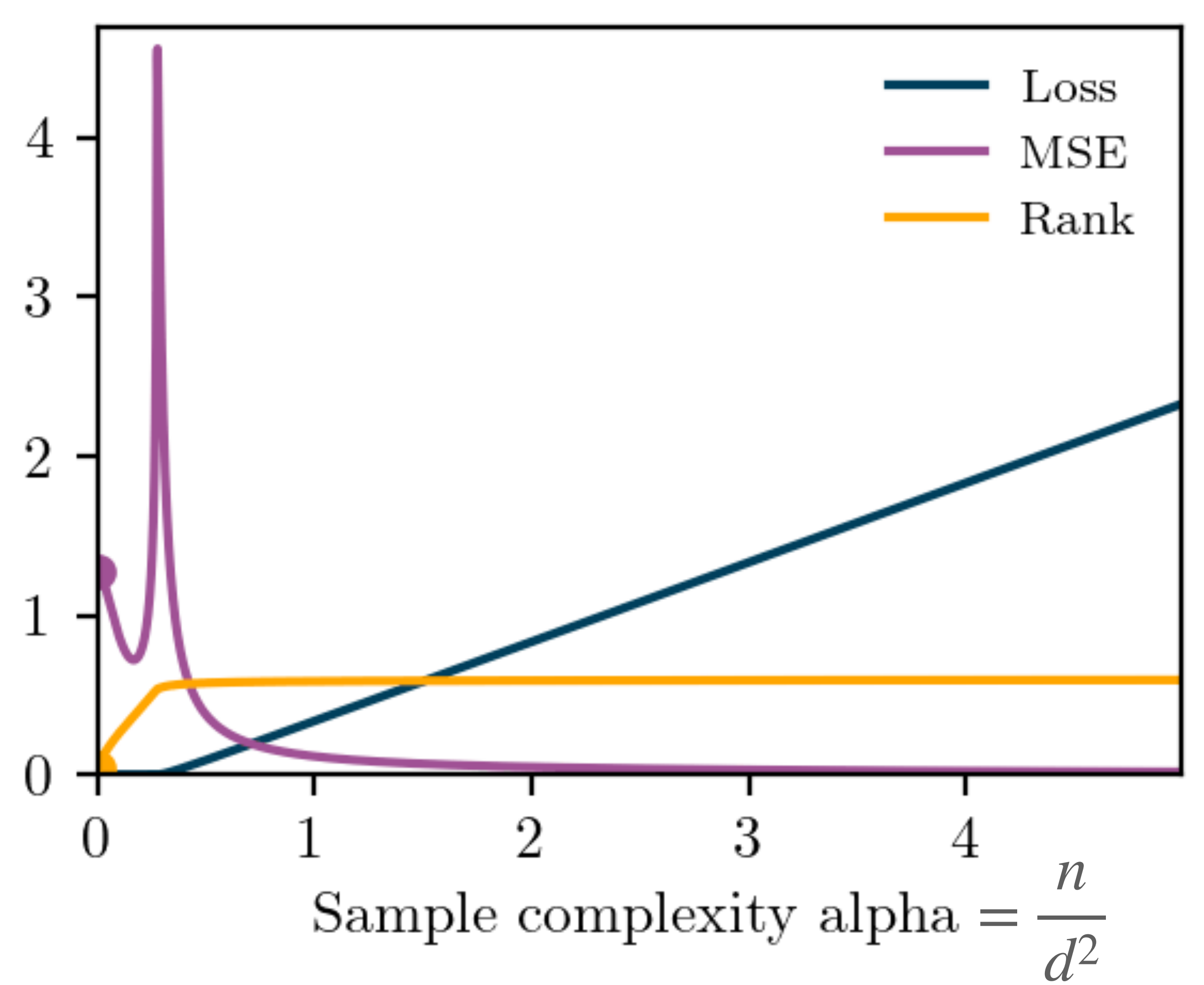
$$G \sim \mathrm{GOE}(d) \qquad A^\star \sim \mathscr{W}_{p^\star, d}$$

$$p \geq d \qquad p^\star = \rho^\star d \qquad n = \alpha d^2$$

Vignette I : spectrum of matrix $\hat{W}$, lower noise

Noisy data, low regularisation





Double descent

Memorization
(But min-ℓ_1 norm interpolator)

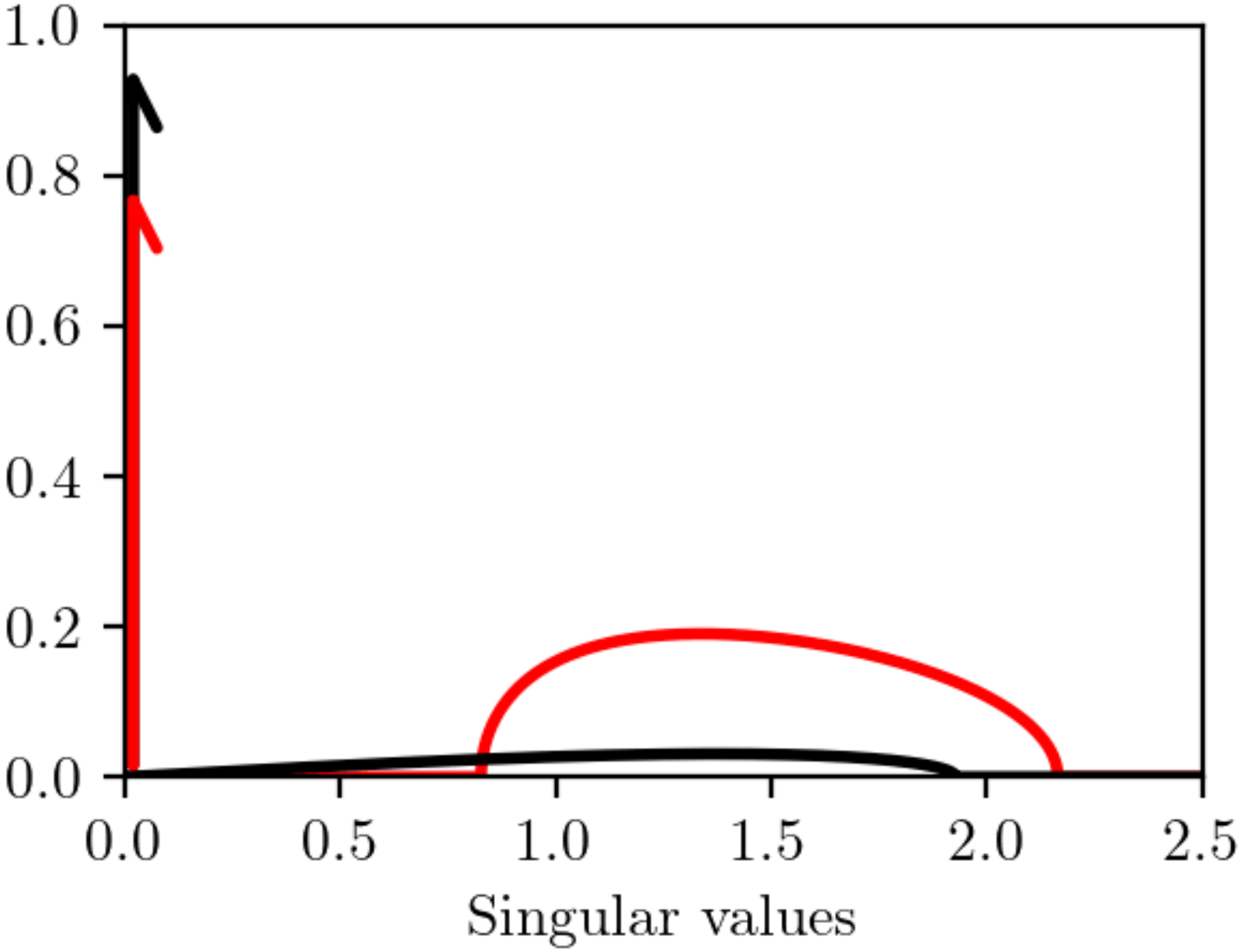
Interpolation peak

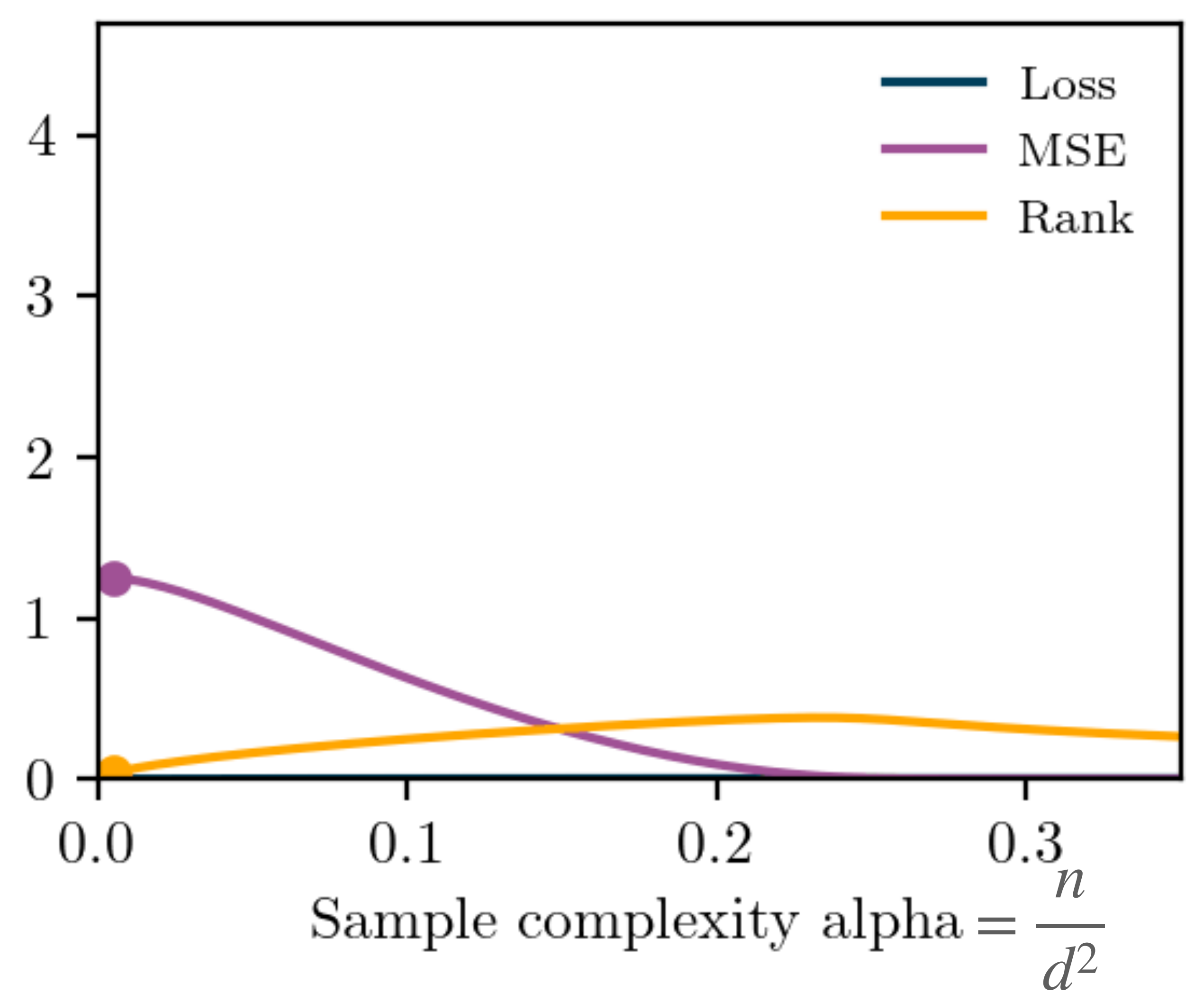
Generalization

$p \geq d$
$p^\star = \rho^\star d$
$n = \alpha d^2$

Vignette I : spectrum of matrix $\hat{W}$, lower noise

Less noisy data, low regularisation





Memorization
(But min-ℓ_1 norm interpolator)

$y =$

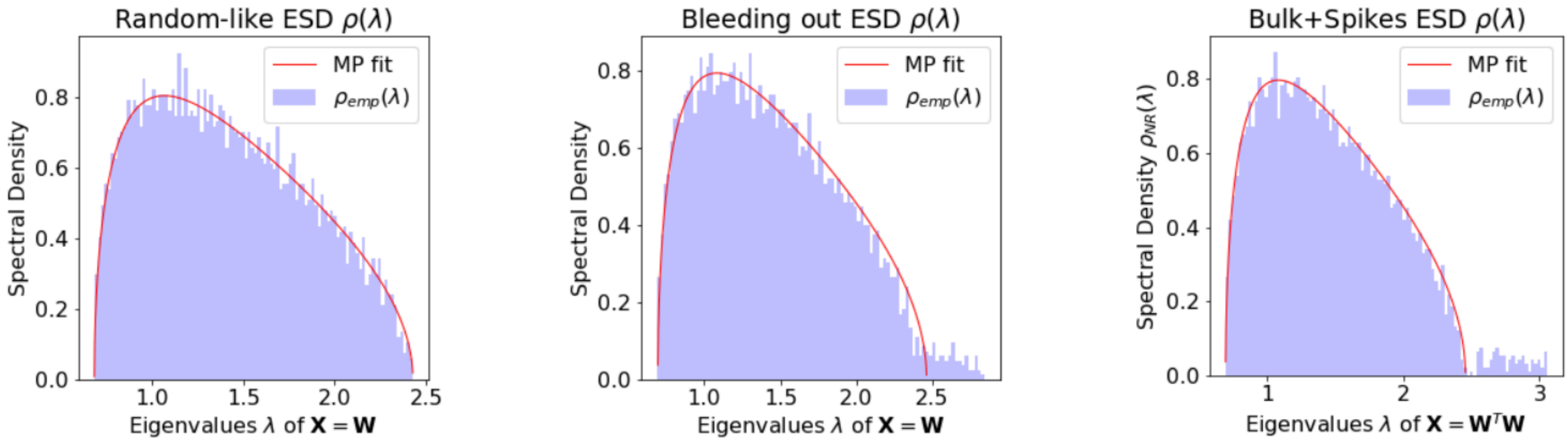
$\sum_{i=1}^{p} \sigma\left(\frac{\hat{w}_i^\top x}{\sqrt{d}}\right) = \mathrm{Tr}[GA]$

Generalization

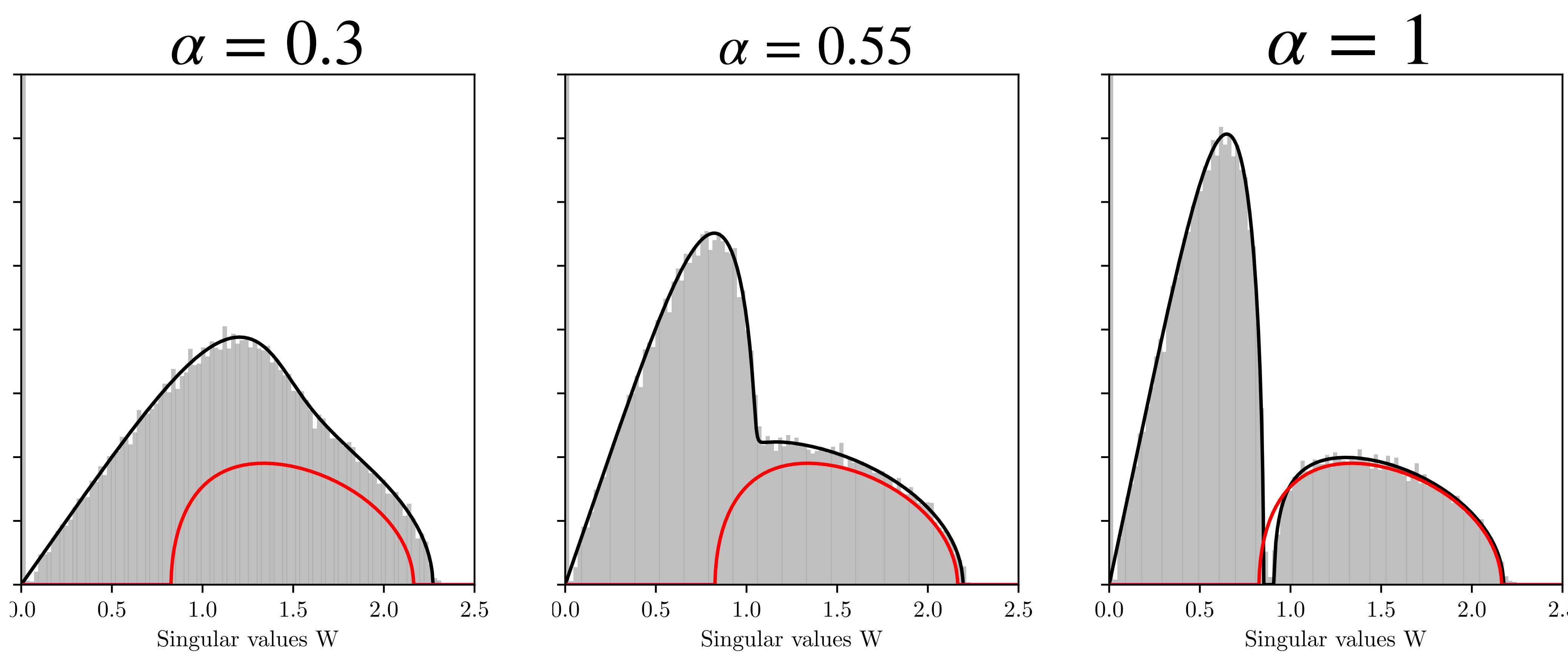
$p \geq d$
$p^\star = \rho^\star d$
$n = \alpha d^2$

Vignette I : spectrum reproduces deep learning ones!

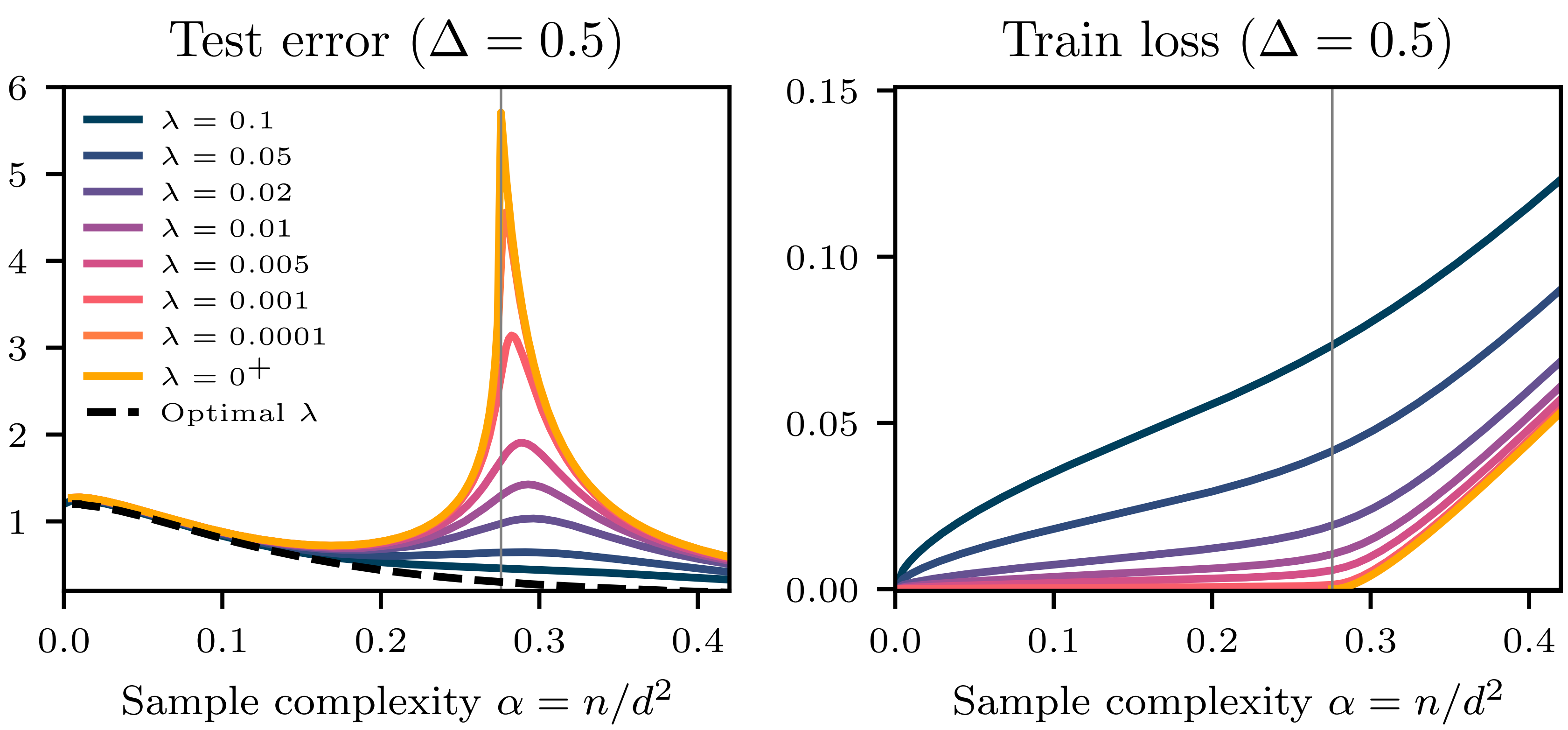
M. Michael W. Mahoney ‘2020



(a) Random-like. (b) Bleeding-out. (c) Bulk+Spikes.



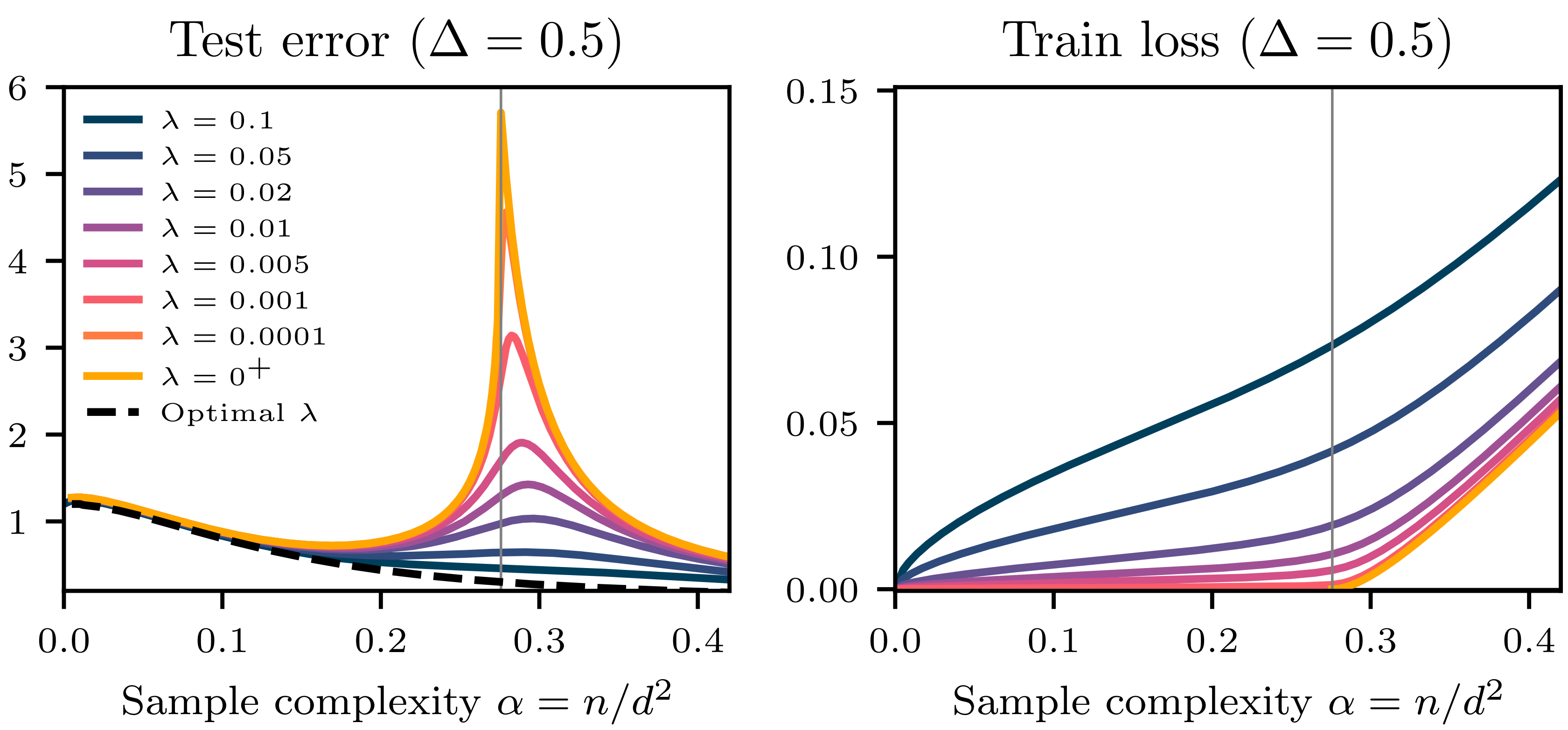
Vignette II : Interpolation threshold & double descent



Interpolation threshold

Largest value of $\alpha = n/d^2$ such that a perfect fit with "zero training loss" (aka, an interpolator) solution exists

Vignette II : Interpolation threshold & double descent



Interpolation threshold

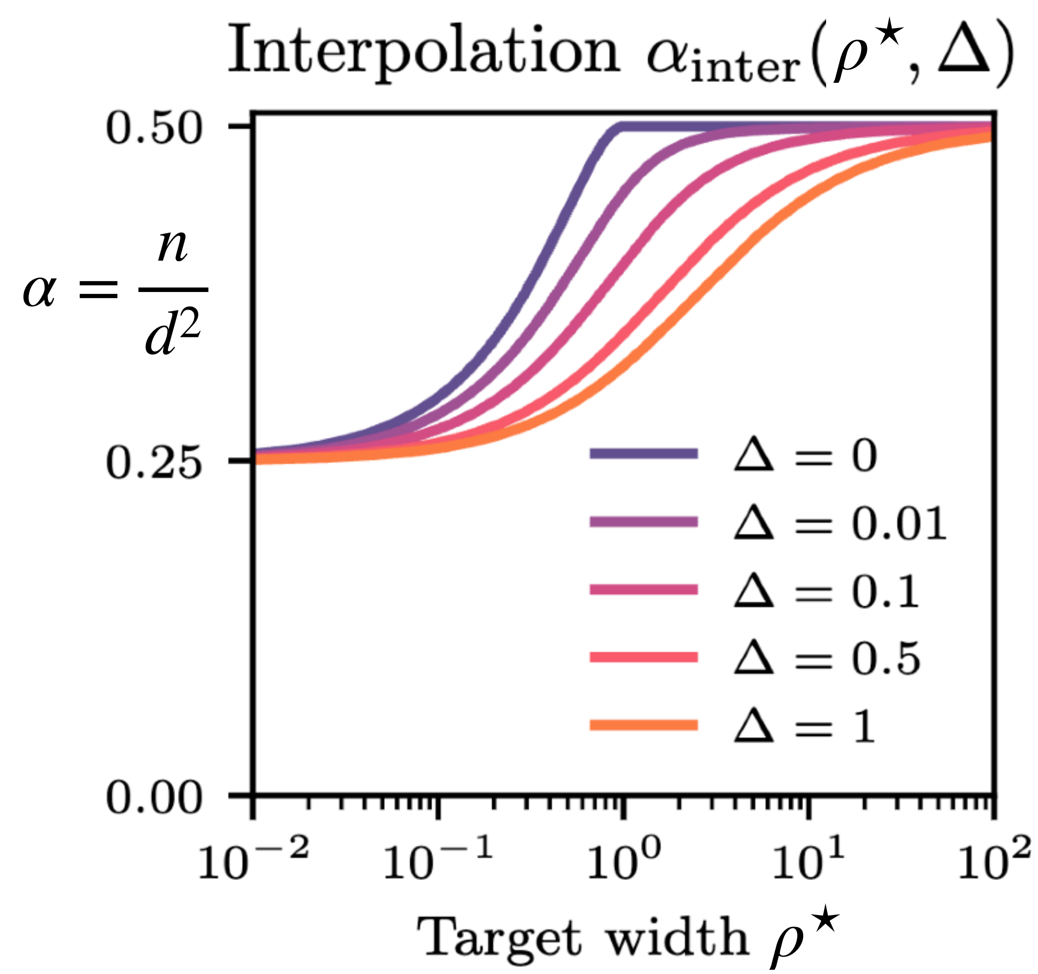
Largest value of $\alpha = n/d^2$ such that a perfect fit with "zero training loss" (aka, an interpolator) solution exists

Location of the interpolation threshold

Here with the objective a Marcenko-Pastur with parameter $\rho^\star$

$$\lim_{\Delta \to 0^+} \alpha_{\text{inter}}(\rho^\star, \Delta) = \frac{1}{4} \begin{cases} 1 + 2\rho^\star - \rho^{\star 2}, & \text{if } 0 < \rho^\star < 1, \\ 2, & \text{if } \rho^\star > 1. \end{cases}$$

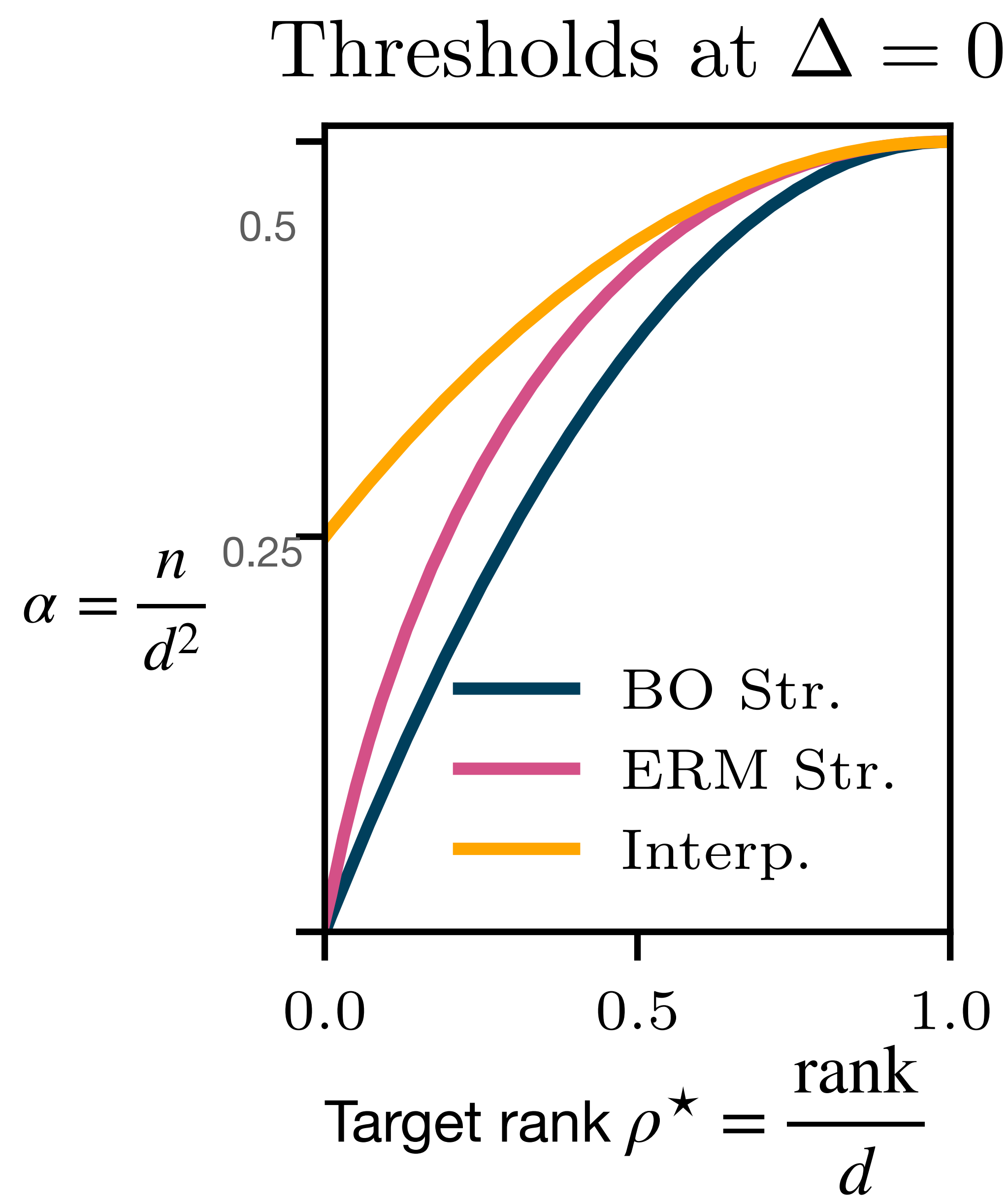
$$\lim_{\Delta \to \infty} \alpha_{\text{inter}}(\rho^\star, \Delta) = \frac{1}{4}$$ as in [Maillard and Bandeira '24]



Vignette III : Perfect recovery ℓ_1 versus ℓ_2

Strong recovery:

Number of measurement to reach perfect recovery with noiseless measurements



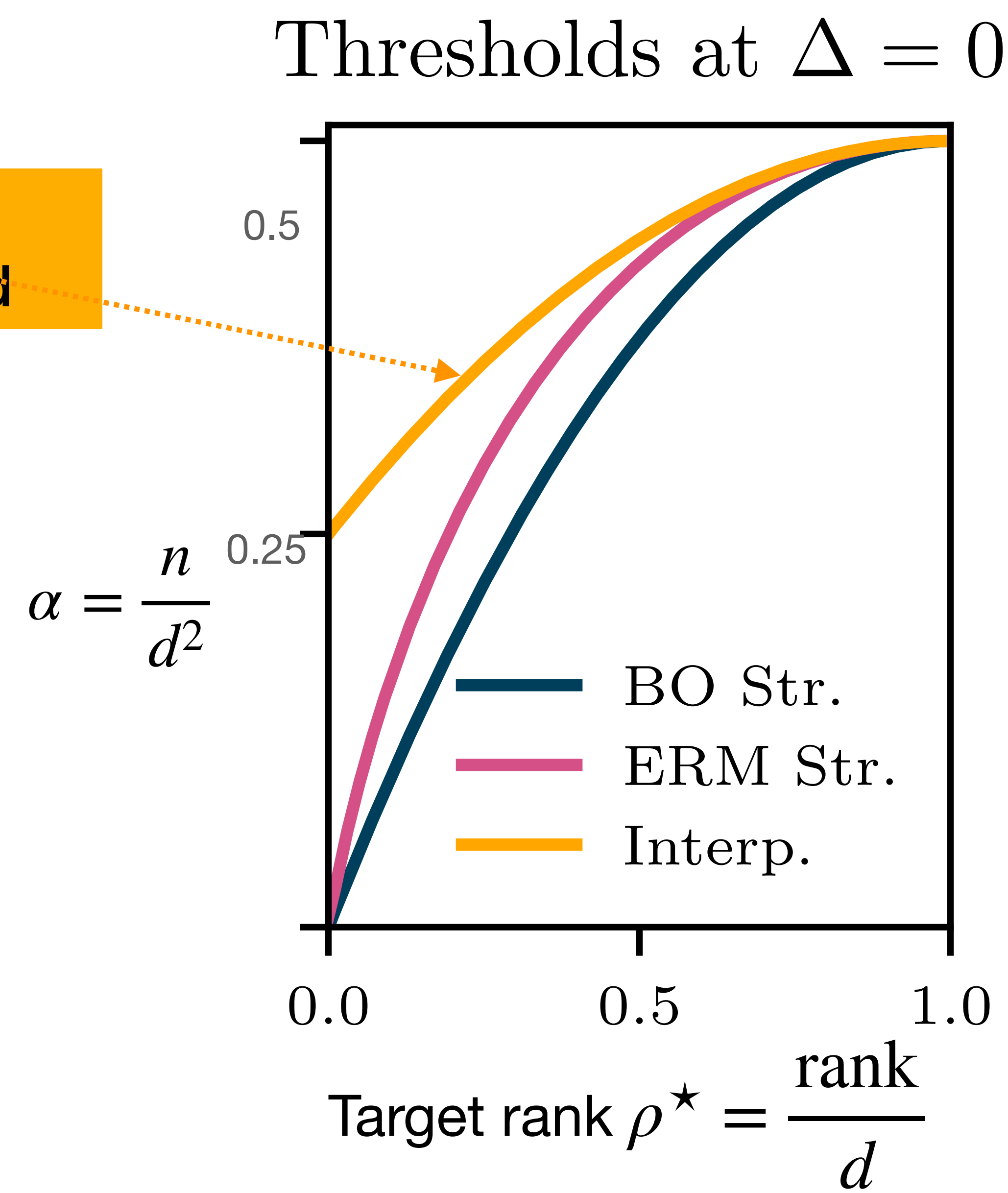
See also [DL Donoho, M Gavish, A Montanari '13, Oymak Hassibi '13, D Amelunxen, M Lotz, MB McCoy, JA Tropp '14]

Vignette III : Perfect recovery ℓ_1 versus ℓ_2

Strong recovery:

Number of measurement to reach perfect recovery with noiseless measurements

NTK, KernelLazy…
Interpolation Threshold

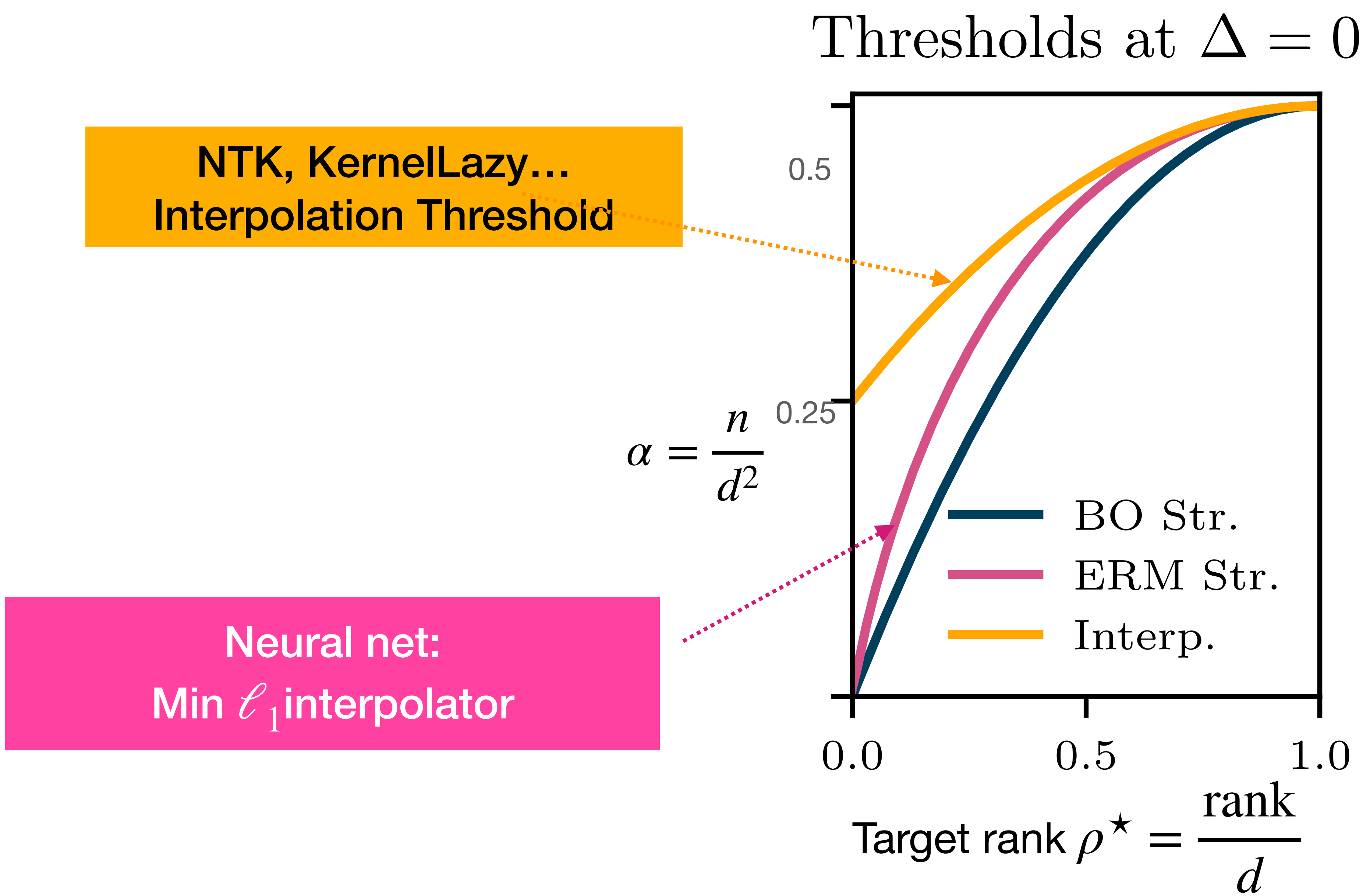


See also [DL Donoho, M Gavish, A Montanari ’13, Oymak Hassibi ’13, D Amelunxen, M Lotz, MB McCoy, JA Tropp ’14]

Vignette III : Perfect recovery ℓ_1 versus ℓ_2

Strong recovery:

Number of measurement to reach perfect recovery with noiseless measurements



See also [DL Donoho, M Gavish, A Montanari ’13, Oymak Hassibi ’13, D Amelunxen, M Lotz, MB McCoy, JA Tropp ’14]

Vignette III : Perfect recovery ℓ_1 versus ℓ_2

Strong recovery:

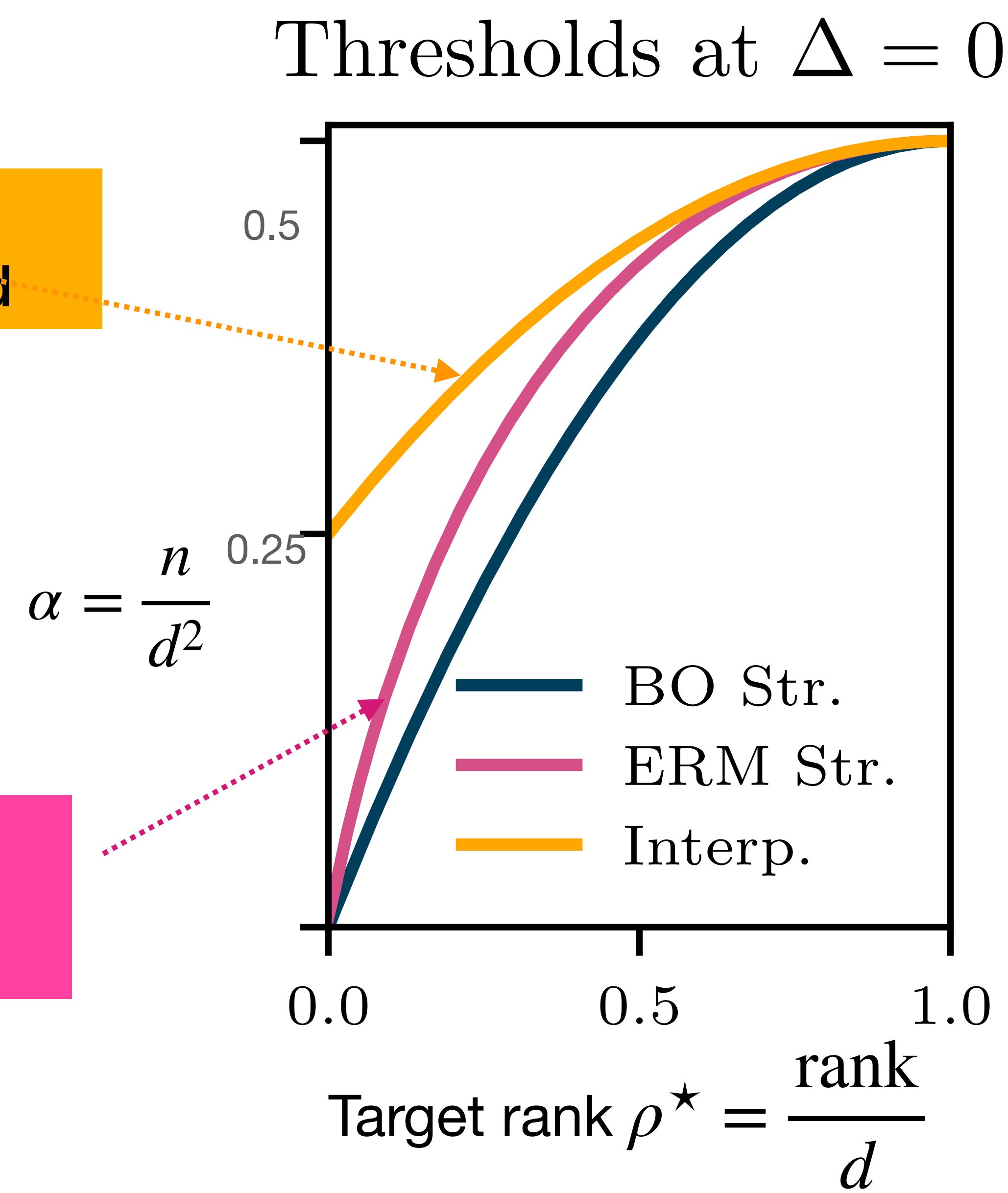
Number of measurement to reach perfect recovery with noiseless measurements

NTK, KernelLazy…
Interpolation Threshold

NTK, Kernel
(aka ℓ_2)

$$\alpha_C \to \frac{1}{4}$$

Neural net:
Min ℓ_1 interpolator



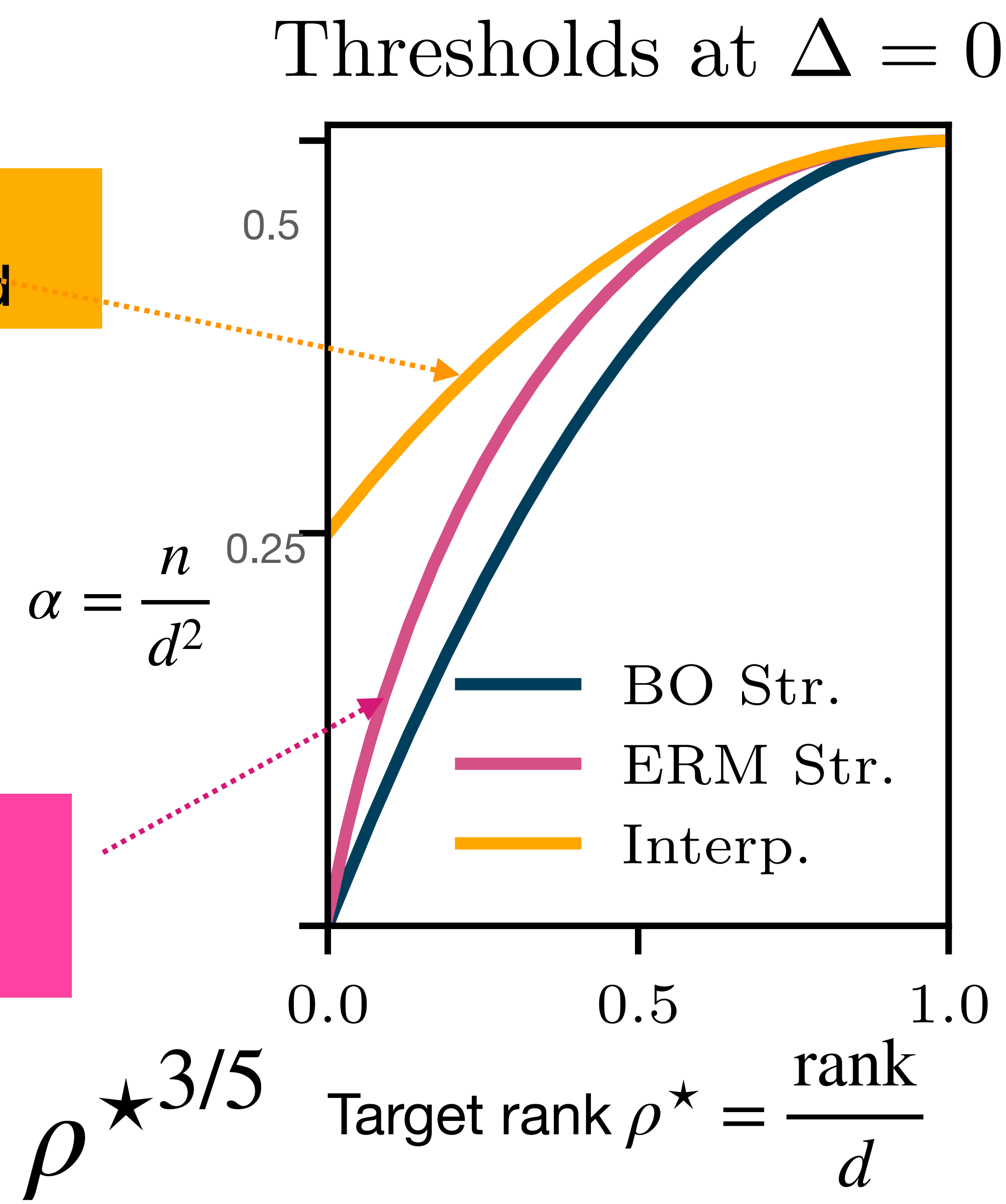
See also [DL Donoho, M Gavish, A Montanari '13, Oymak Hassibi '13, D Amelunxen, M Lotz, MB McCoy, JA Tropp '14]

Vignette III : Perfect recovery ℓ_1 versus ℓ_2

Strong recovery:

Number of measurement to reach perfect recovery with noiseless measurements

NTK, KernelLazy…
Interpolation Threshold



NTK, Kernel (aka ℓ_2)

$$\alpha_C \to \frac{1}{4}$$

Neural net:
Min ℓ_1 interpolator

Neural Networks (aka ℓ_1)

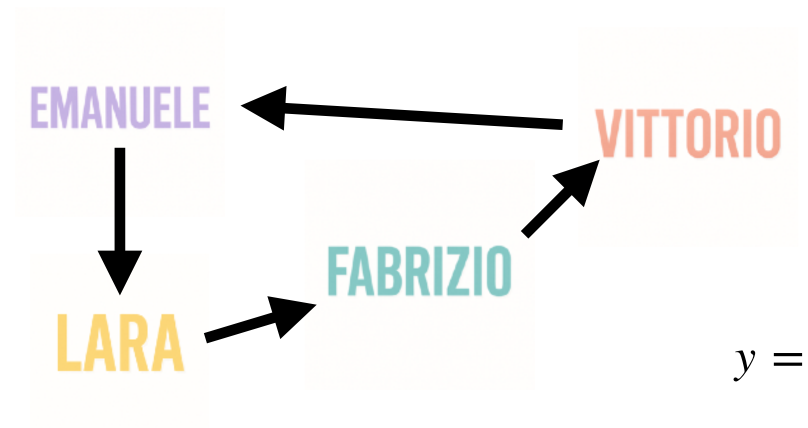
$$\alpha_C \to 3\rho^\star \quad \rho_{\text{eff}} \to \rho^{\star 3/5}$$

See also [DL Donoho, M Gavish, A Montanari '13, Oymak Hassibi '13, D Amelunxen, M Lotz, MB McCoy, JA Tropp '14]

Methodologically BO attention is analogous to quadratic network

<u>**Back to attention**</u> $X \in \mathbb{R}^{T \times d}, X_{a\mu} \sim \mathcal{N}(0, \mathbf{1}_d)$

$$y_\mu = \mathrm{hardmax}(X_\mu A X_\mu^\top)$$



$$y = \begin{array}{c c} & \begin{array}{cccc} E & L & V & F \end{array} \\ \begin{array}{c} E \\ L \\ V \\ F \end{array} & \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

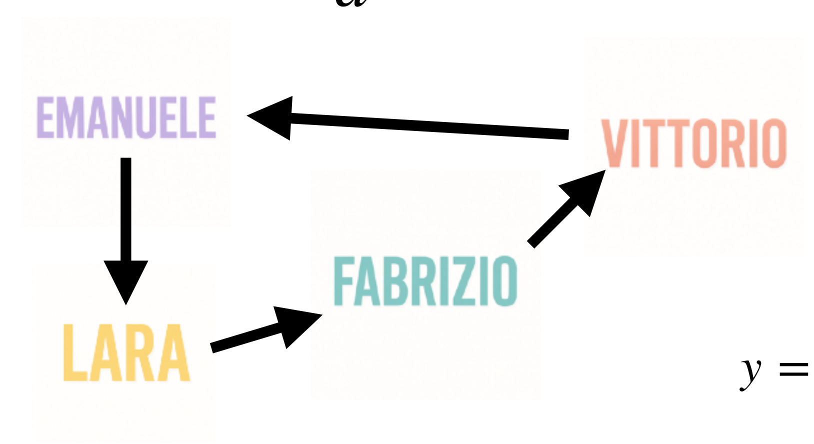
$X \in \mathbb{R}^{T \times d}$ $\quad p = \rho d$

$\mathrm{rank}(A) = p$ $\quad d \to \infty$

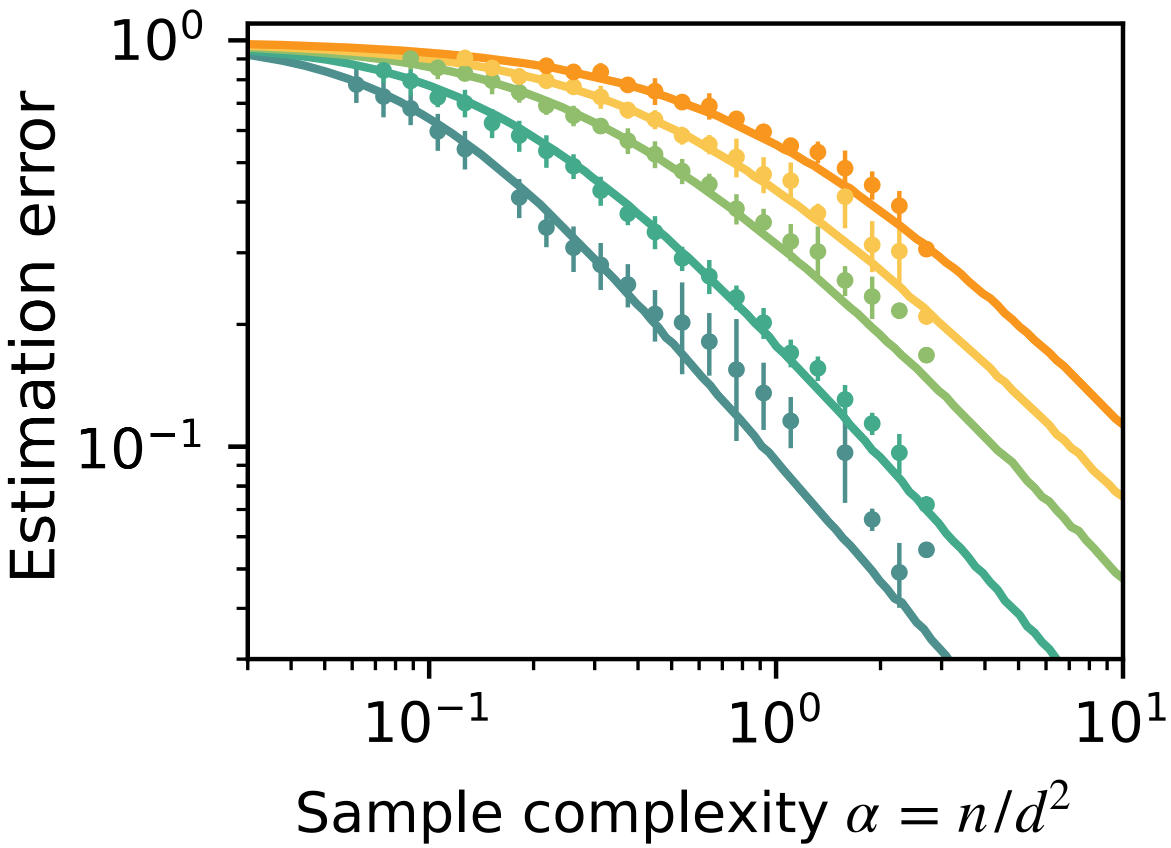
Methodologically BO attention is analogous to quadratic network

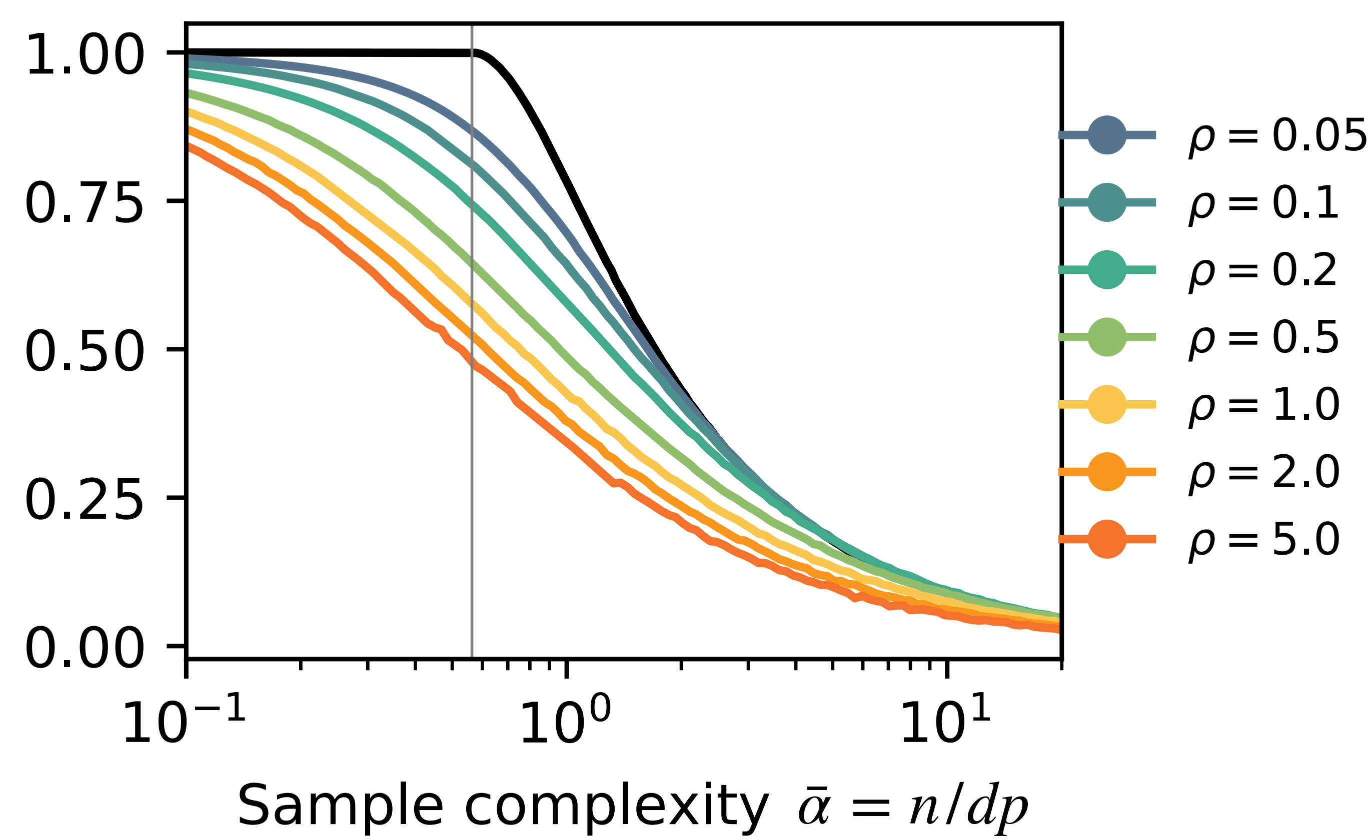
Back to attention $X \in \mathbb{R}^{T\times d}, X_{a\mu} \sim \mathcal{N}(0,\mathbf{1}_d)$

$$y_\mu = \mathrm{hardmax}(X_\mu A X_\mu^\top)$$



$$y = \begin{array}{c c} & \begin{array}{cccc} E & L & V & F \end{array} \\ \begin{array}{c} E \\ L \\ V \\ F \end{array} & \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$





$X \in \mathbb{R}^{T\times d}$ $\quad p = \rho d$

$\mathrm{rank}(A) = p$ $\quad d \to \infty$

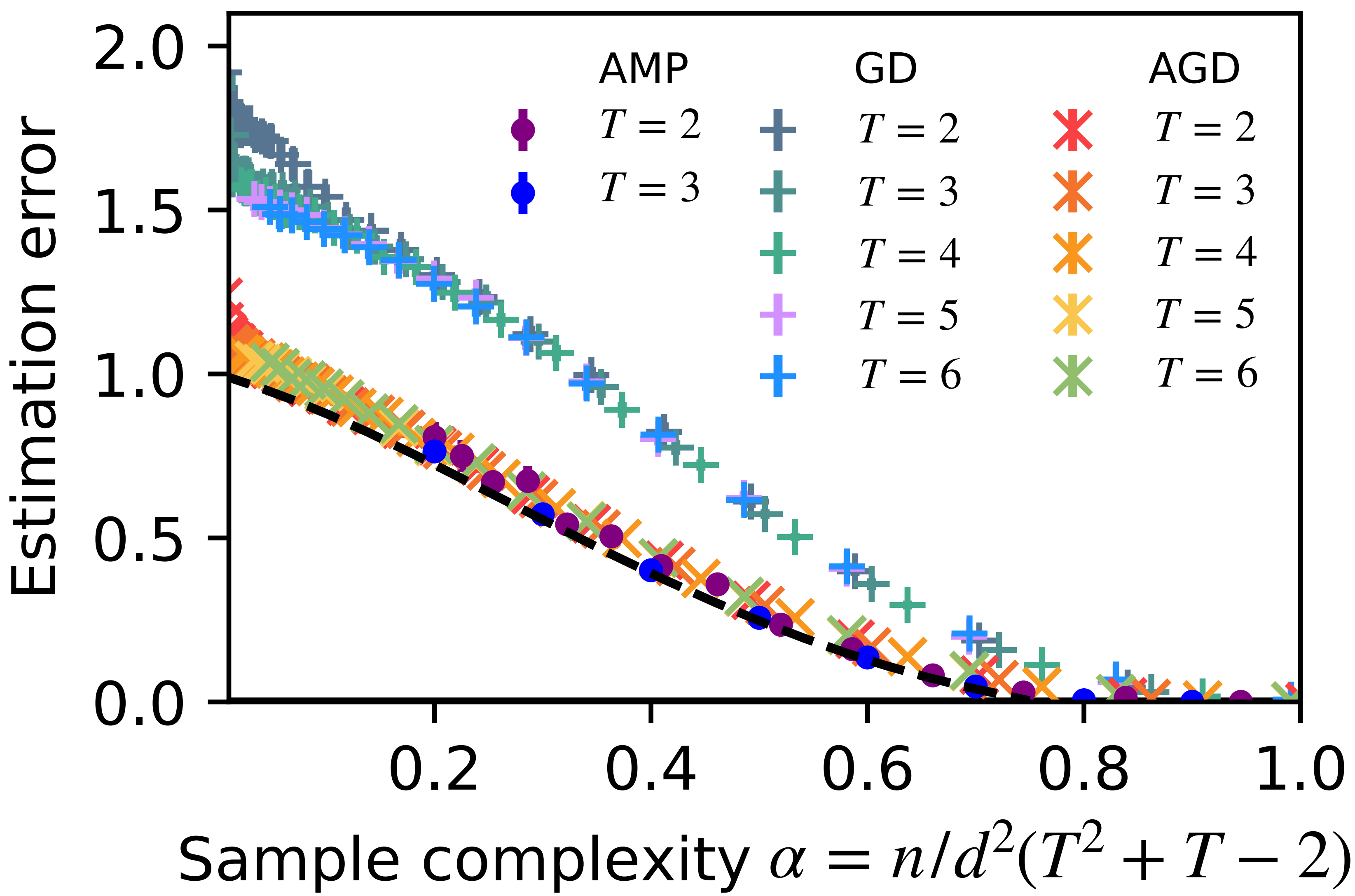
Methodologically BO attention is analogous to quadratic network

Back to attention

$$y_\mu = \mathrm{softmax}(X_\mu A X_\mu^\top)$$

$$\text{Estimation error} \xrightarrow{d\to\infty} \frac{2\alpha\rho}{\hat{q}}$$

$$1 - \alpha(T^2 + T - 2) = \frac{4\pi^2}{3\hat{q}} \int \mu_{\mathcal{Y}}(\lambda)^3 \, \mathrm{d}\lambda$$



$X \in \mathbb{R}^{T\times d}$ $\quad p = \rho d$

$\mathrm{rank}(A) = p$ $\quad d \to \infty$

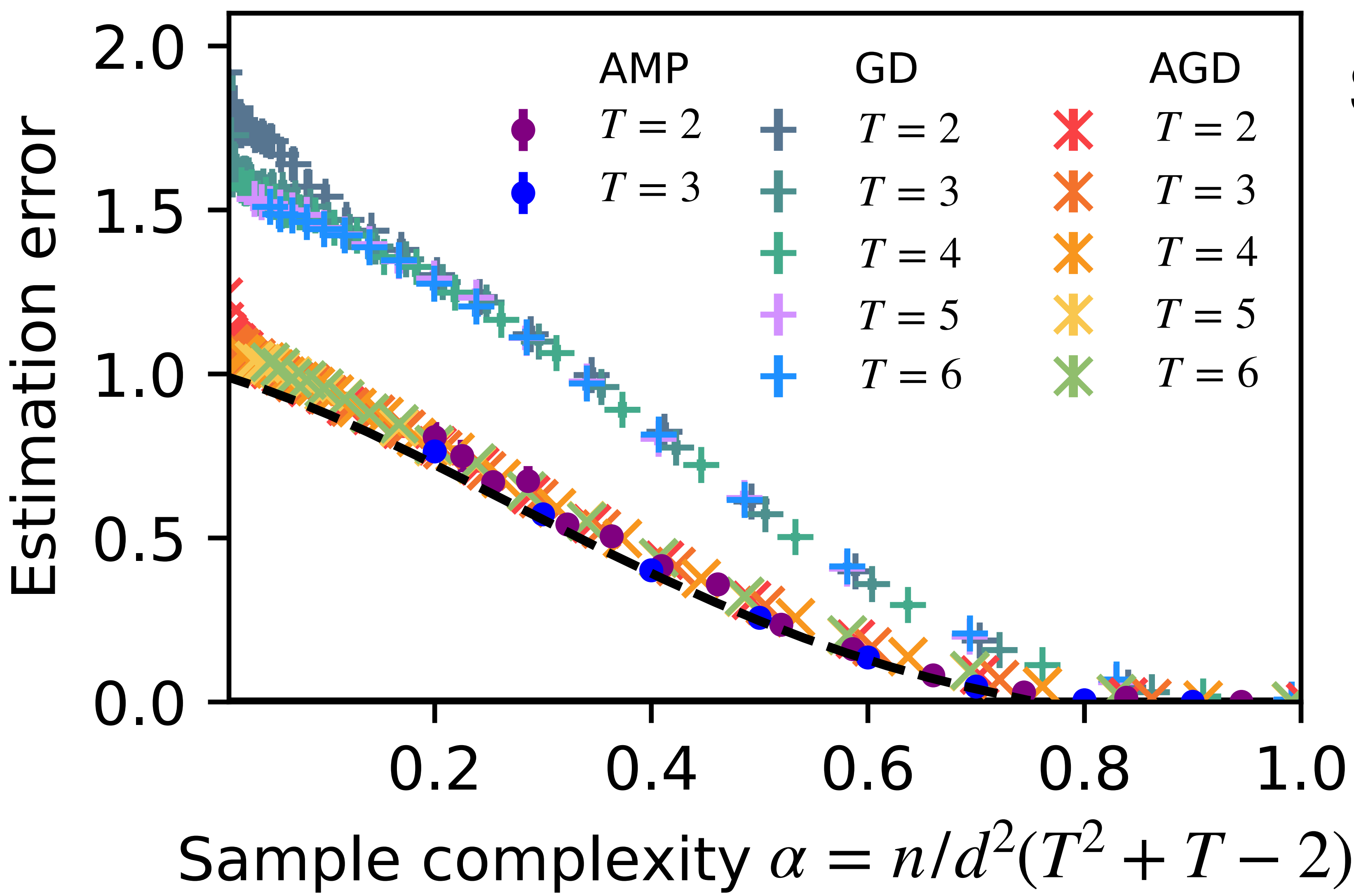
Methodologically BO attention is analogous to quadratic network

Back to attention

$$y_\mu = \mathrm{softmax}(X_\mu A X_\mu^\top)$$

$$\text{Estimation error} \xrightarrow{d\to\infty} \frac{2\alpha\rho}{\hat{q}}$$

$$1 - \alpha(T^2 + T - 2) = \frac{4\pi^2}{3\hat{q}} \int \mu_{\mathcal{Y}}(\lambda)^3 \, \mathrm{d}\lambda$$



Simple scaling for **tokens**

$$n \to n/(T^2 + T - 2)$$

$X \in \mathbb{R}^{T\times d}$ $\quad p = \rho d$

$\mathrm{rank}(A) = p$ $\quad d \to \infty$

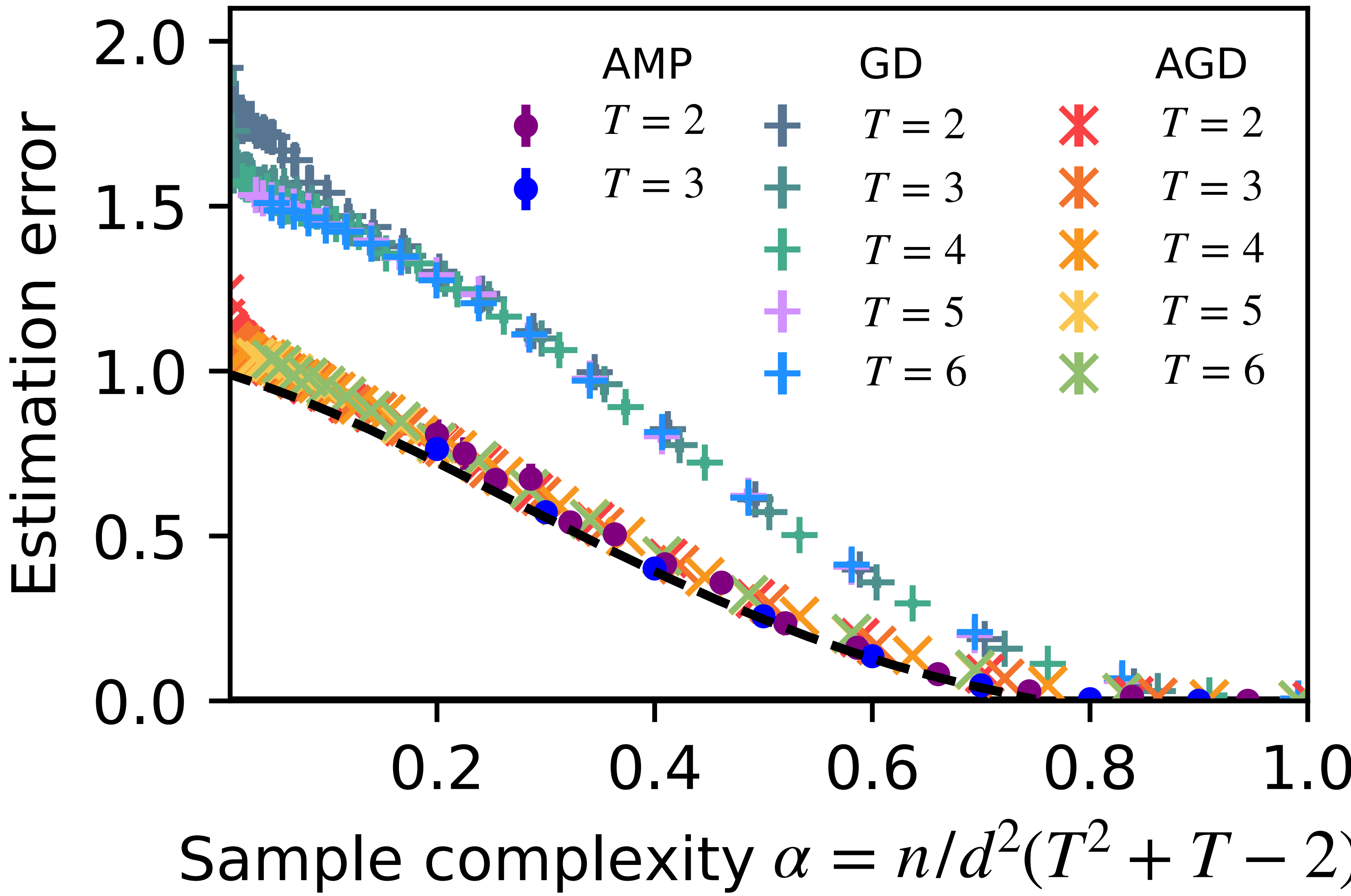
Methodologically BO attention is analogous to quadratic network

Back to attention

$$y_\mu = \mathrm{softmax}(X_\mu A X_\mu^\top)$$

$$\text{Estimation error} \xrightarrow{d\to\infty} \frac{2\alpha\rho}{\hat{q}}$$

$$1 - \alpha(T^2 + T - 2) = \frac{4\pi^2}{3\hat{q}} \int \mu_{\mathcal{Y}}(\lambda)^3 \, \mathrm{d}\lambda$$



Simple scaling for **tokens**

$n \to n/(T^2 + T - 2)$

No effect of $\beta < \infty$

WORK IN PROGRESS

It's a BO artifact

$X \in \mathbb{R}^{T\times d}$ $\quad p = \rho d$

$\mathrm{rank}(A) = p$ $\quad d \to \infty$

What about multiple layers ?

Single head, no MLP layer

$$y = \sigma\left(X_L \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X_L^\top \right) \qquad X_{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\ell \sum_{i=1}^{p} \boldsymbol{w}_i^\ell \boldsymbol{w}_i^{\ell\top} X_\ell^\top \right) \right] X_\ell$$

$X \in \mathbb{R}^{T\times d}$ $\quad p = \rho d$

$\text{rank}(A) = p$ $\quad d \to \infty$

What about multiple layers ?

Single head, no MLP layer

$$y = \sigma\left(X_L \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X_L^\top \right) \qquad X_{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\ell \sum_{i=1}^{p} \boldsymbol{w}_i^\ell \boldsymbol{w}_i^{\ell\top} X_\ell^\top \right) \right] X_\ell$$

$$y = g\left(X \sum_{i=1}^{p} \boldsymbol{w}_i^1 \boldsymbol{w}_i^{1\top} X^\top, \ldots, X \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X^\top \right)$$

$X \in \mathbb{R}^{T\times d}$ $\qquad p = \rho d$

$\mathrm{rank}(A) = p$ $\qquad d \to \infty$

What about multiple layers ?

Single head, no MLP layer

$$y = \sigma\left(X_L \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X_L^\top \right) \qquad X_{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\ell \sum_{i=1}^{p} \boldsymbol{w}_i^\ell \boldsymbol{w}_i^{\ell\top} X_\ell^\top \right)\right] X_\ell$$

$$y = g\left(X \sum_{i=1}^{p} \boldsymbol{w}_i^1 \boldsymbol{w}_i^{1\top} X^\top, \ldots, X \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X^\top \right) = g\left(XA^1X^\top, \ldots, XA^LX^\top \right)$$

$X \in \mathbb{R}^{T\times d}$ $\qquad p = \rho d$

$\mathrm{rank}(A) = p$ $\qquad d \to \infty$

What about multiple layers ?

Single head, no MLP layer

$$y = \sigma\left(X_L \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X_L^\top \right) \qquad X_{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\ell \sum_{i=1}^{p} \boldsymbol{w}_i^\ell \boldsymbol{w}_i^{\ell\top} X_\ell^\top \right) \right] X_\ell$$

$$y = g\left(X \sum_{i=1}^{p} \boldsymbol{w}_i^1 \boldsymbol{w}_i^{1\top} X^\top, \ldots, X \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X^\top \right) = g\left(XA^1X^\top, \ldots, XA^LX^\top \right)$$

Shown by induction: assume $X_\ell = g_\ell\left(XA^1X^\top, \ldots, XA^\ell X^\top \right) X$

$$X_{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\ell A^\ell X_\ell^\top \right) \right] X_\ell = \left[c\mathbf{1}_T + \sigma\left(g_\ell XA^\ell X^\top g_\ell^\top \right) \right] g_\ell X$$

$X \in \mathbb{R}^{T\times d}$ $\qquad p = \rho d$

$\mathrm{rank}(A) = p$ $\qquad d \to \infty$

What about multiple layers ?

Single head, no MLP layer

$$y = \sigma\left(X_L \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X_L^\top \right) \qquad X_{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\ell \sum_{i=1}^{p} \boldsymbol{w}_i^\ell \boldsymbol{w}_i^{\ell\top} X_\ell^\top \right) \right] X_\ell$$

$$y = g\left(X \sum_{i=1}^{p} \boldsymbol{w}_i^1 \boldsymbol{w}_i^{1\top} X^\top, \dots, X \sum_{i=1}^{p} \boldsymbol{w}_i^L \boldsymbol{w}_i^{L\top} X^\top \right) = g\left(XA^1X^\top, \dots, XA^LX^\top\right)$$

Shown by induction: assume $X_\ell = g_\ell\left(XA^1X^\top, \dots, XA^\ell X^\top\right) X$

$$X_{\ell+1} = \left[c\mathbf{1}_T + \sigma\left(X_\ell A^\ell X_\ell^\top\right)\right] X_\ell = \left[c\mathbf{1}_T + \sigma\left(g_\ell XA^\ell X^\top g_\ell^\top\right)\right] g_\ell X$$

Hinges on new problem:

Multiplexed denoising

$$\mathcal{Y}_\ell = \mathcal{S}_\ell + \sum_{k=1}^{L} \sqrt{\delta}_{\ell k} G_k$$

$X \in \mathbb{R}^{T\times d}$ $\quad p = \rho d$

$\mathrm{rank}(A) = p$ $\quad d \to \infty$

Thanks for your attention!

Bayes-optimal learning of an extensive-width neural network from quadratically many samples

A. Maillard, E. Troiani, S. Martin, F. Krzakala, L. Zdeborová

arXiv:2408.03733

The Nuclear Route: Sharp Asymptotics of ERM in Overparameterized Quadratic Networks

V. Erba, E. Troiani, L. Zdeborová, F. Krzakala

arXiv:2505.17958

Bayes optimal learning of attention-indexed models

F. Boncoraglio, E. Troiani, V. Erba, L. Zdeborová

arXiv:2506.01582

Thanks for your attention!

Bayes-optimal learning of an extensive-width neural network from quadratically many samples

A. Maillard, E. Troiani, S. Martin, F. Krzakala, L. Zdeborová

The Nuclear Route: Sharp Asymptotics of ERM in Overparameterized Quadratic Networks

V. Erba, E. Troiani, L. Zdeborová, F. Krzakala

Bayes optimal learning of attention-indexed models

F. Boncoraglio, E. Troiani, V. Erba, L. Zdeborová