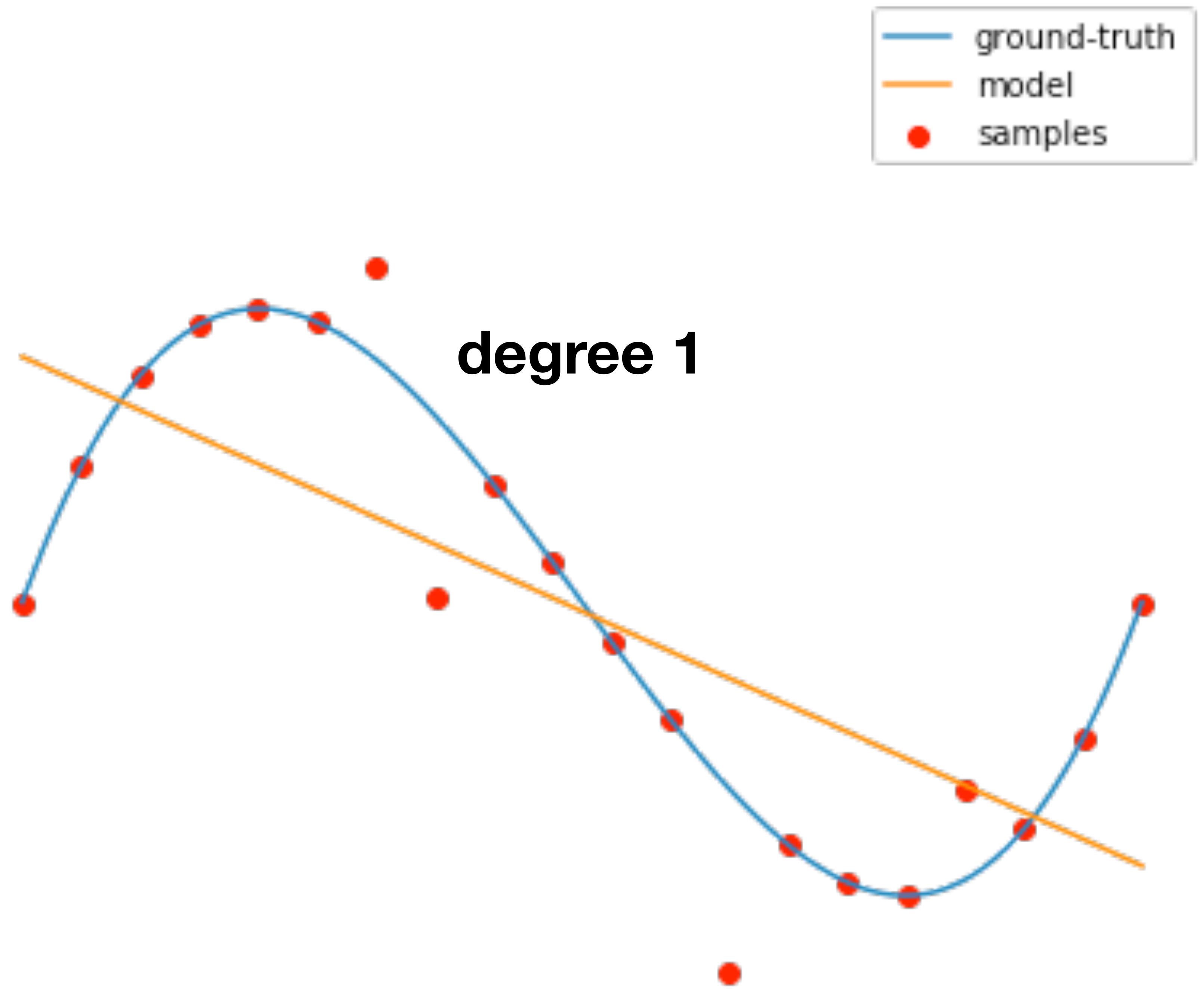
Double Descent / Implicit Regularization + Neural Tangent Kernels

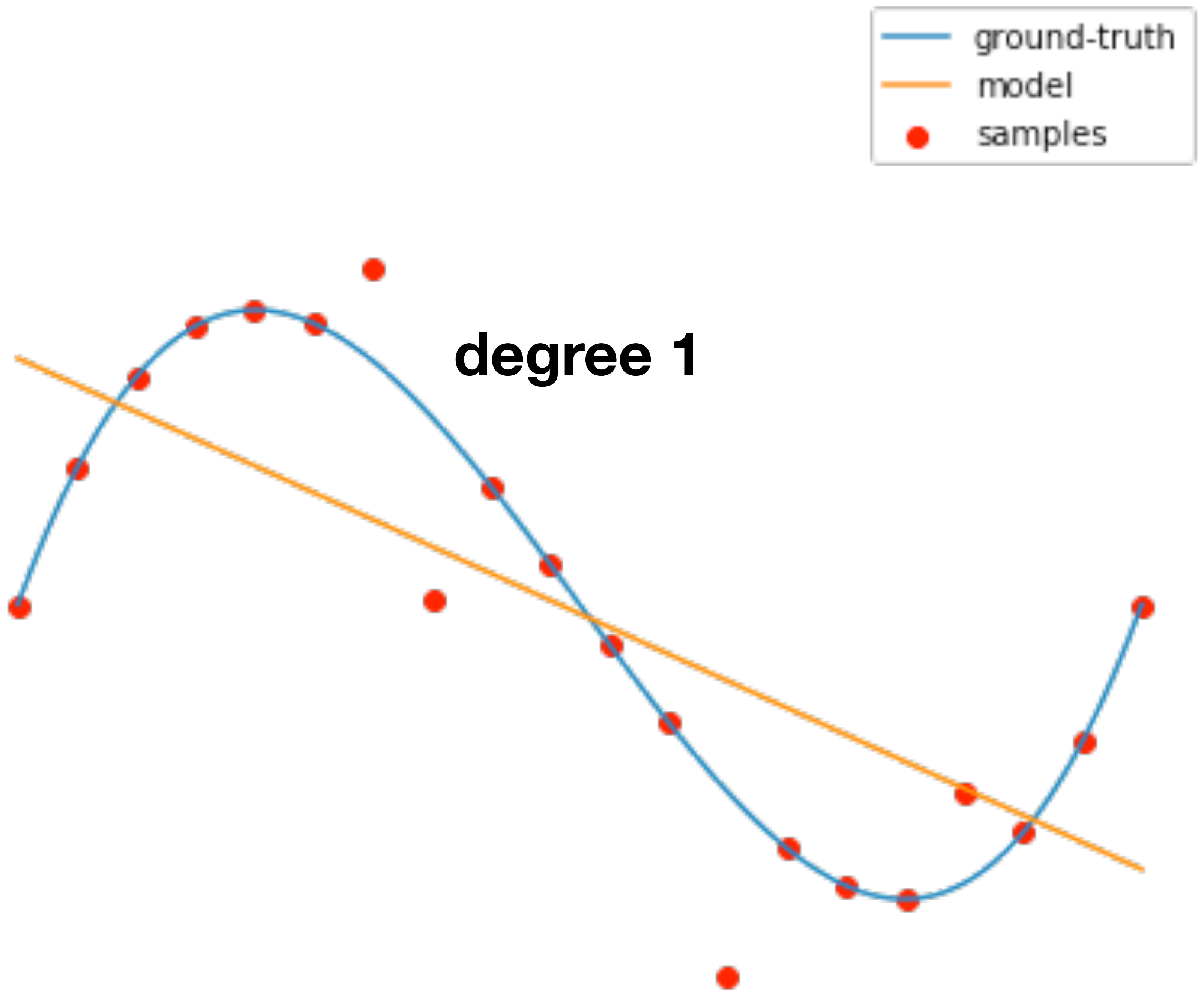
CPSC 532D: Modern Statistical Learning Theory

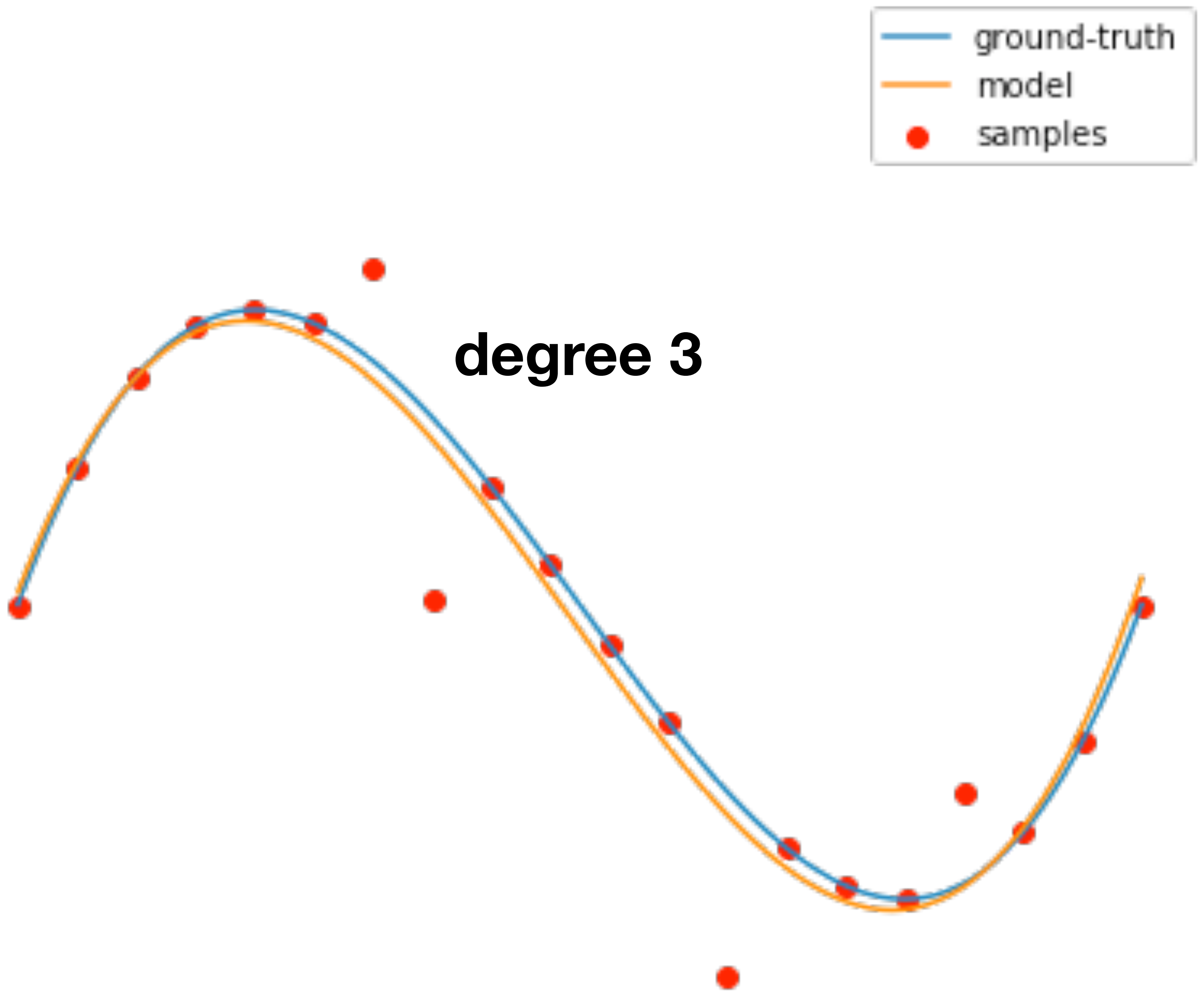
28 November 2022

cs.ubc.ca/~dsuth/532D/22w1/



Nakkiran et al. blog post's companion notebook

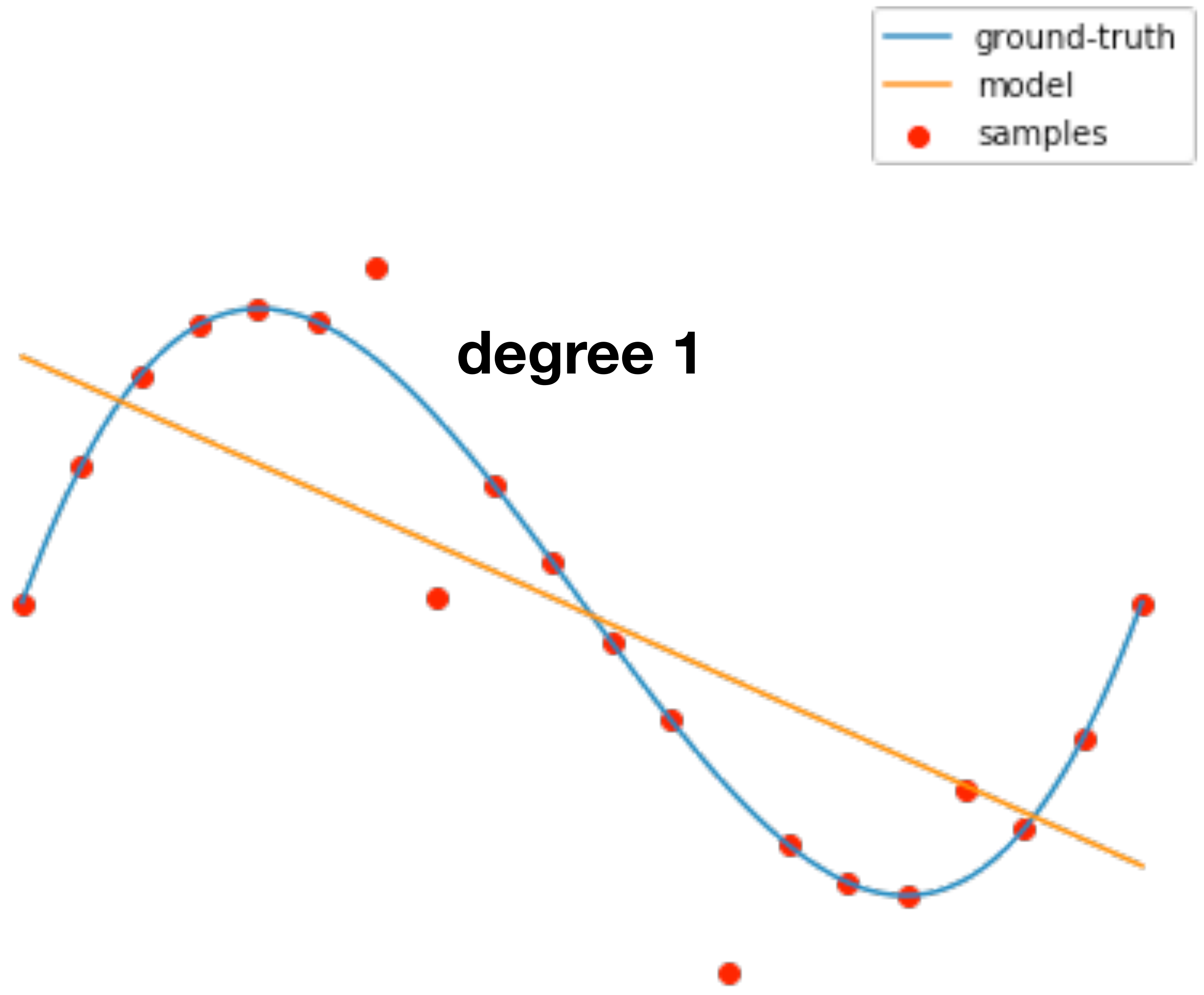


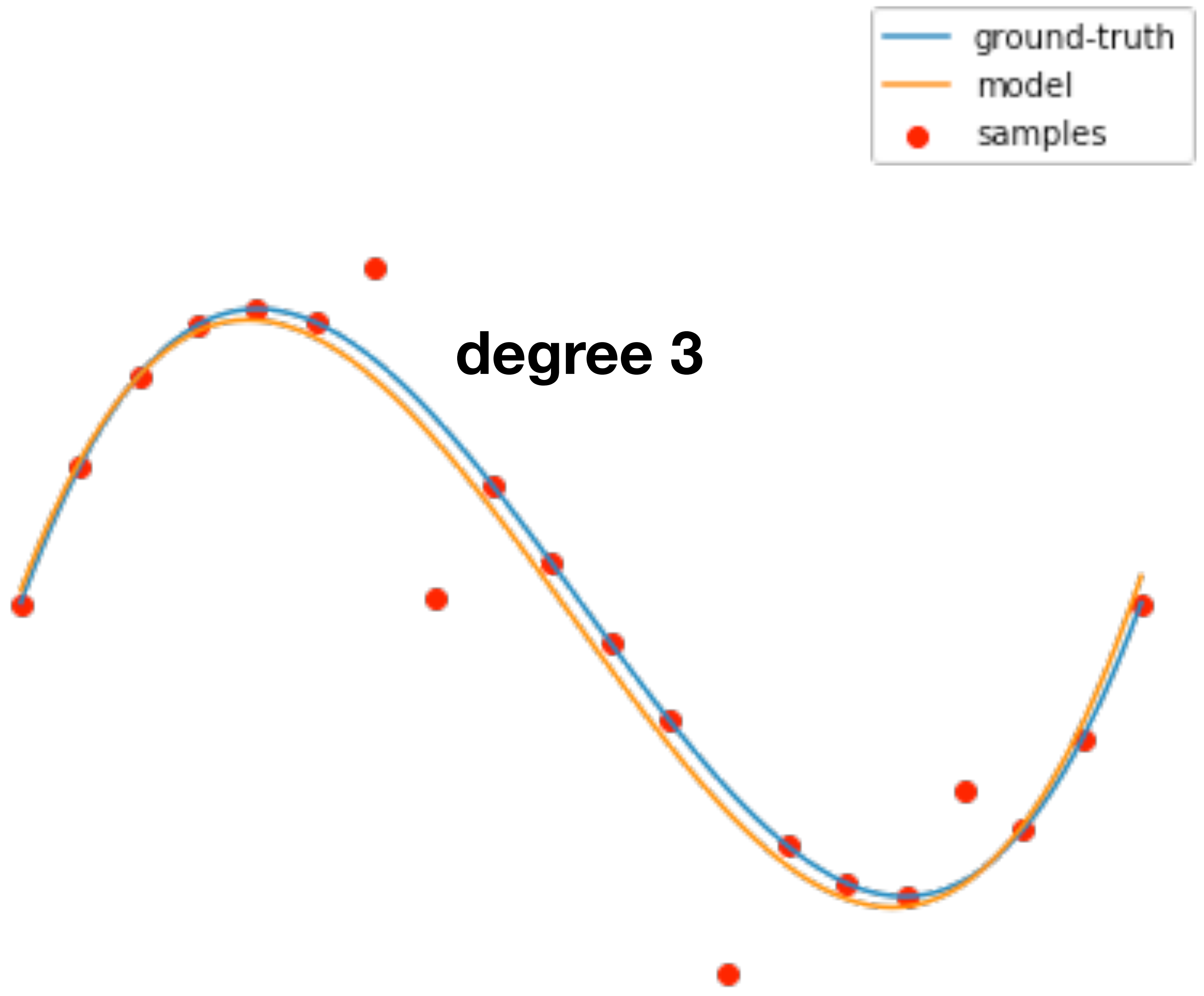


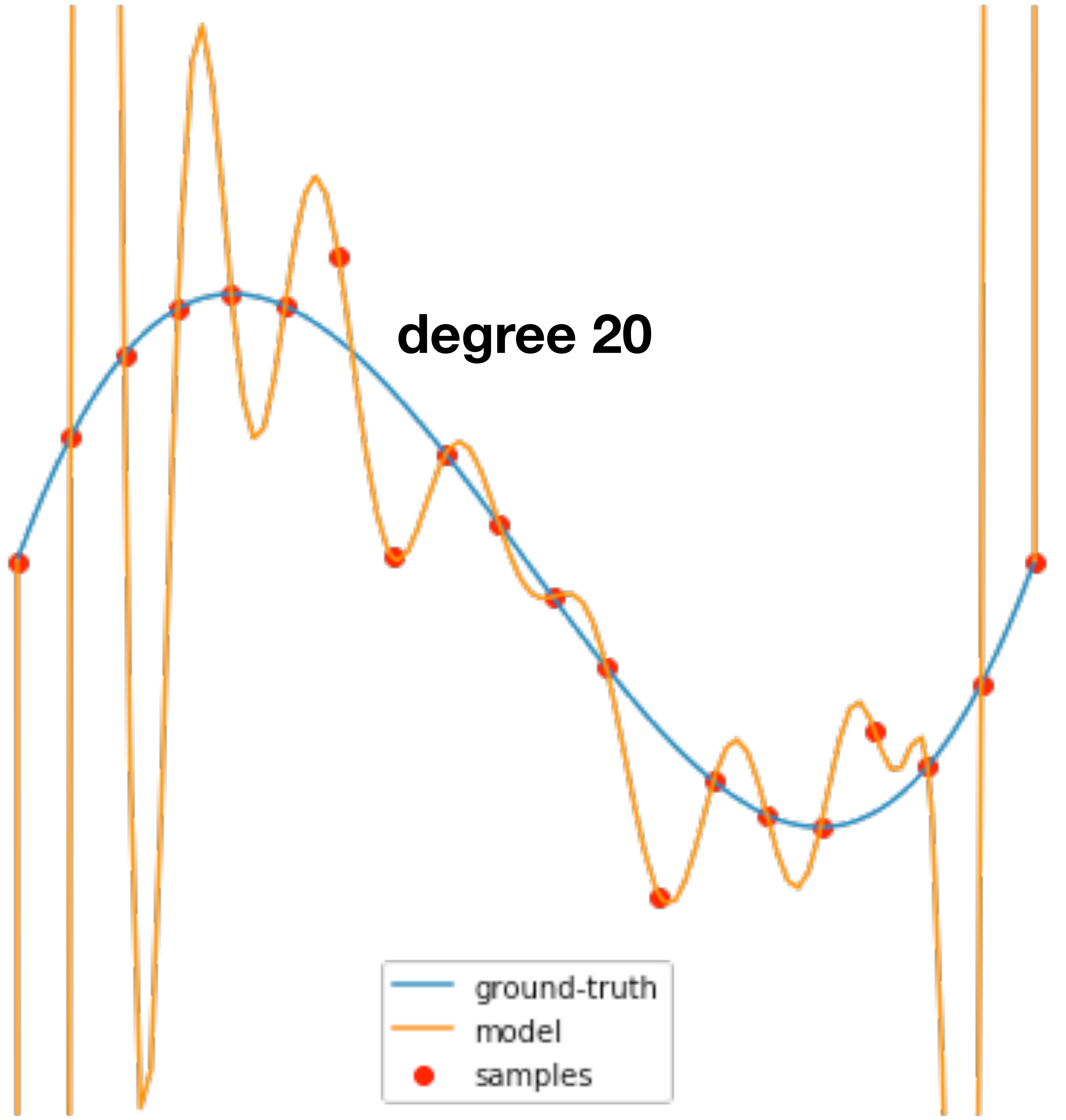
Nakkiran et al. blog post's companion notebook

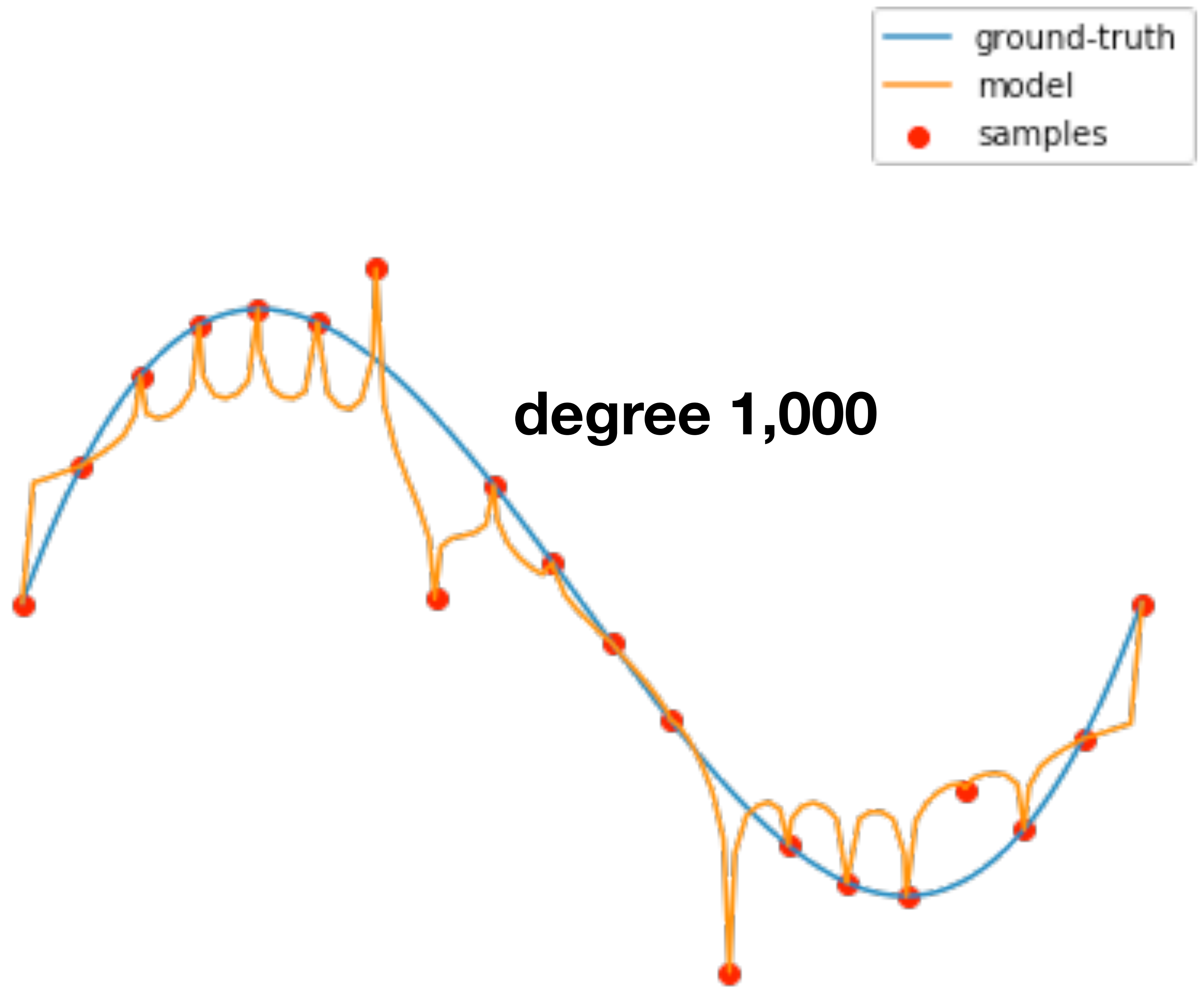
Nakkiran et al. blog post's companion notebook

Nakkiran et al. blog post's companion notebook



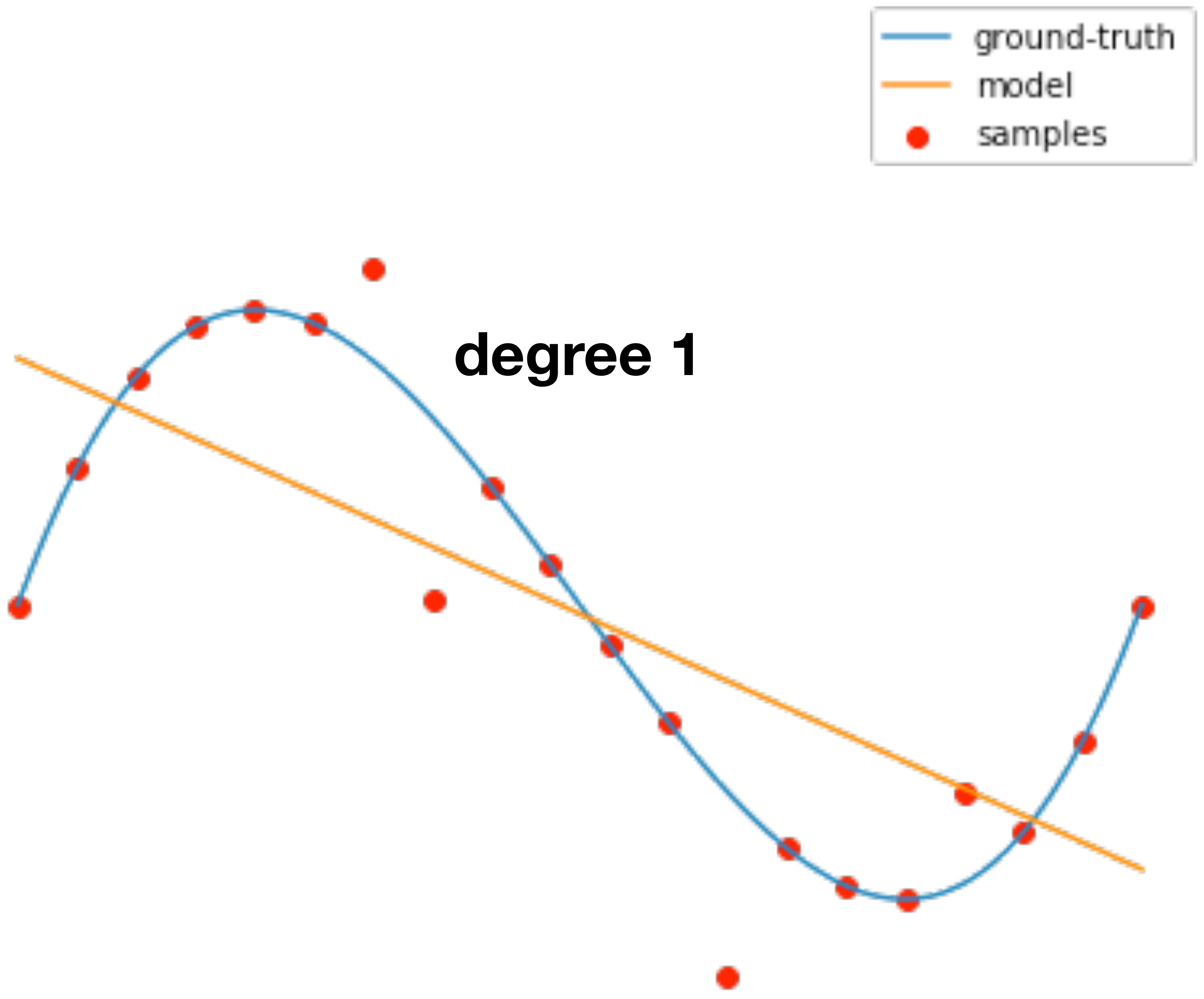


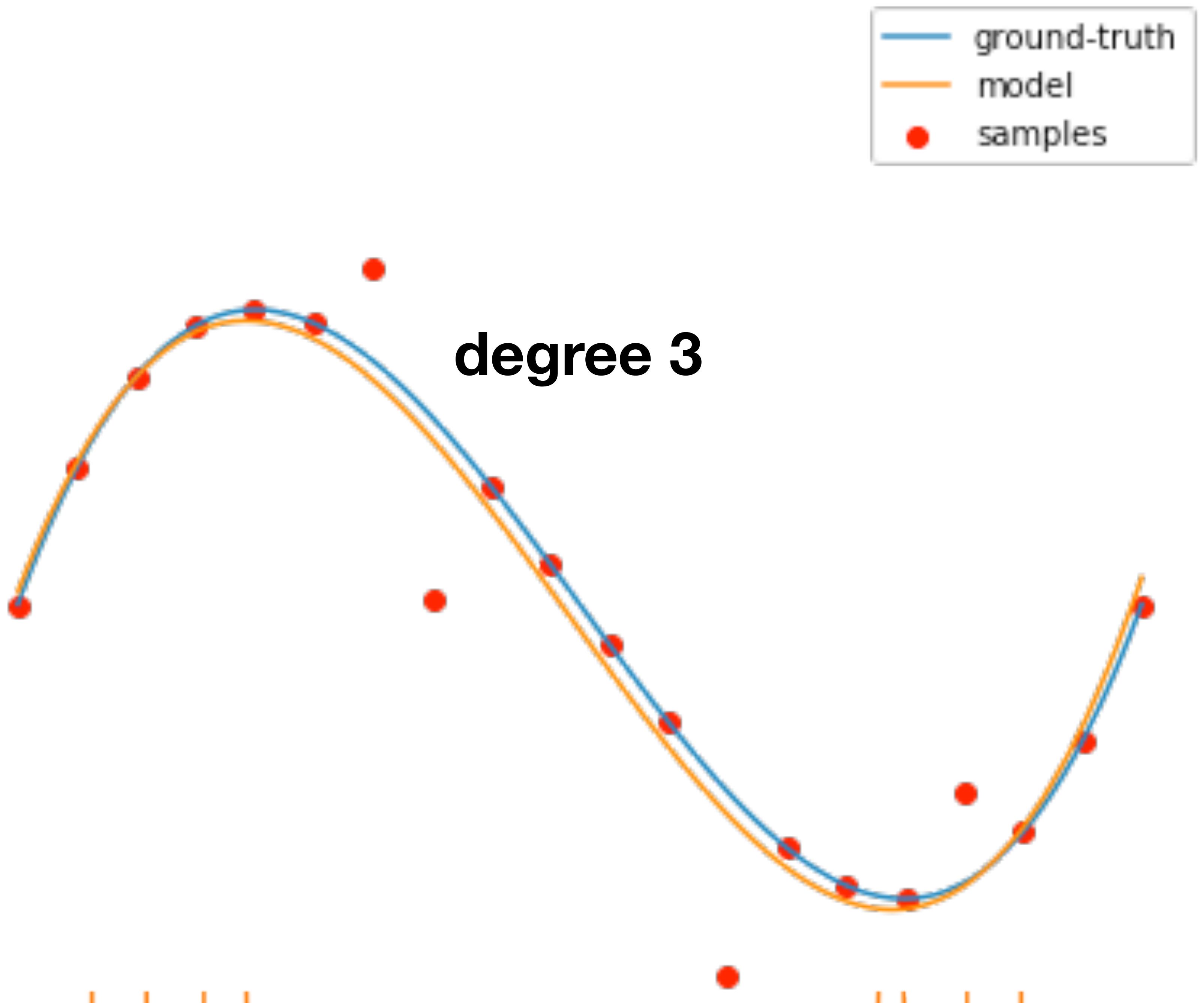


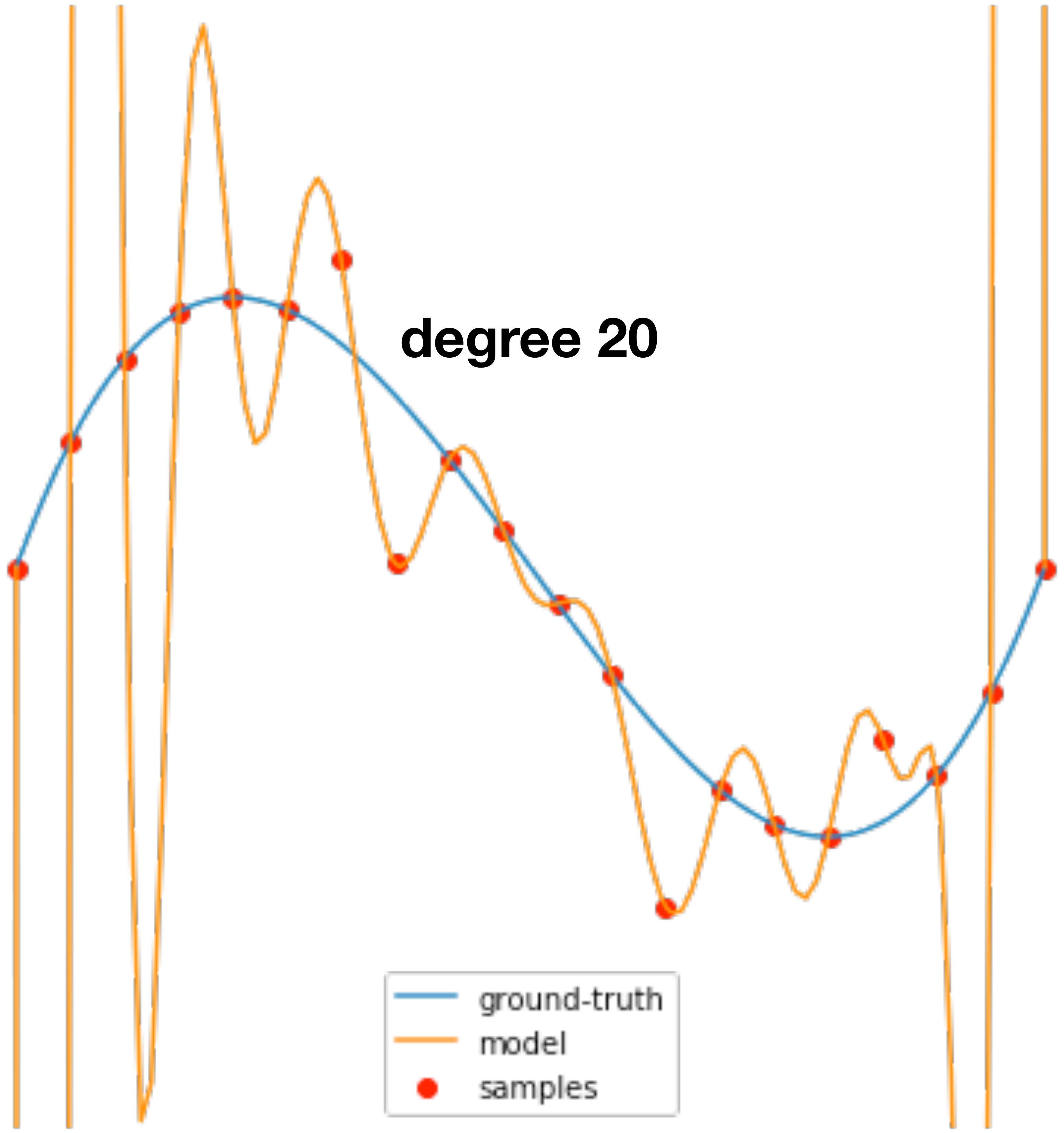


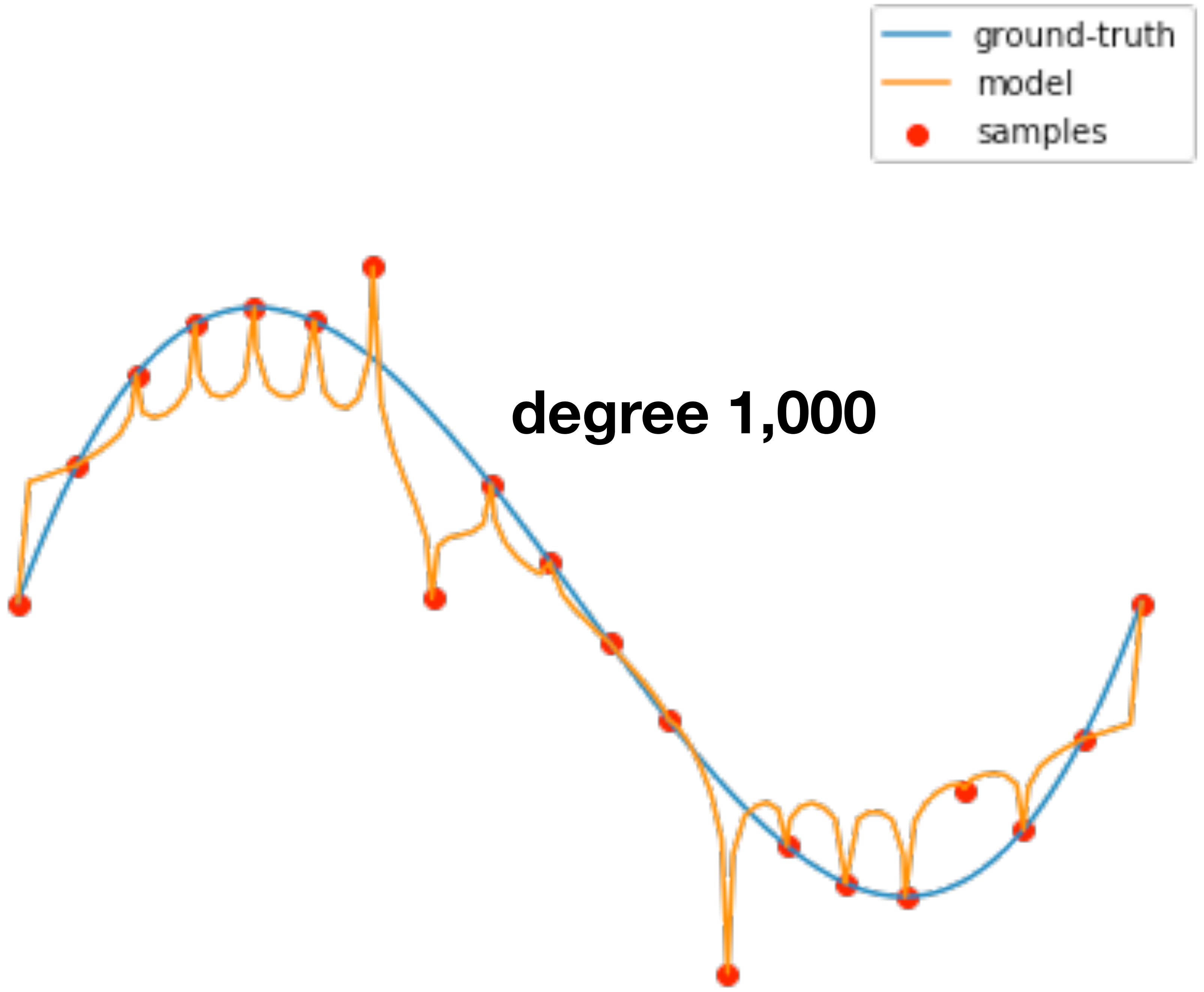
Important: this is the *minimum norm* solution, with the particular Legendre basis!

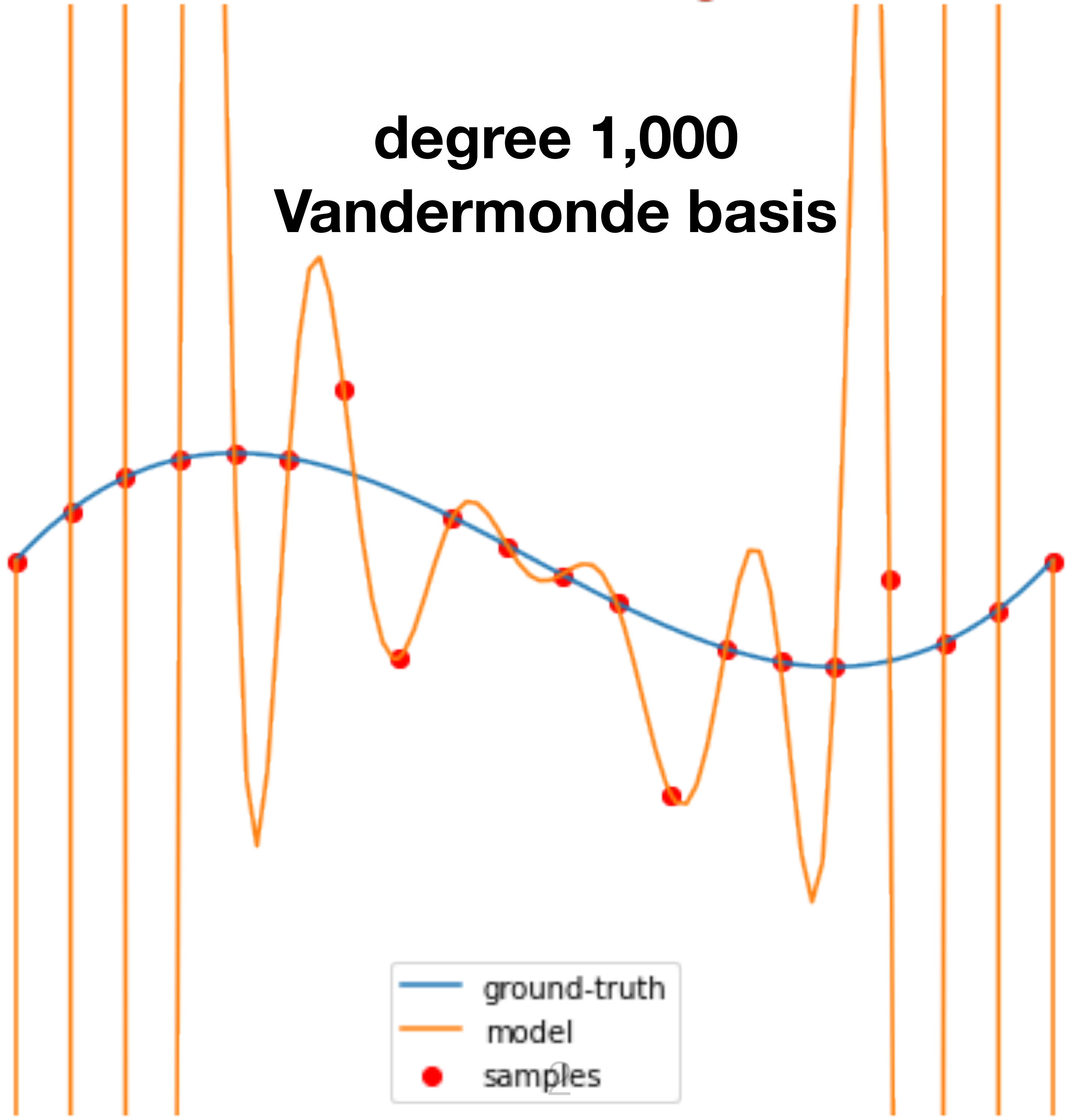
Nakkiran et al. blog post's companion notebook











$$\operatorname*{argmin}_{L_S(w)=0} \|w\| = X^\dagger y$$

Important: this is the *minimum norm* solution, with the particular Legendre basis!

Nakkiran et al. blog post's companion notebook

$$f(w) = \frac{n}{2} L_S(w) = \frac{1}{2}\|\underset{n\times d}{X} w - \underset{n\times 1}{y}\|^2 \qquad \nabla f(w) = X^T(Xw - y)$$

$$w^{(1)} = 0$$

$$w^{(t+1)} = w^{(t)} - \eta \nabla f(w^{(t)}) = (I - \eta X^T X) w^{(t)} + \eta X^T y$$

$$= \eta \sum_{k=0}^{t} (I - \eta X^T X)^k X^T y$$

$$= \eta \sum_{k=0}^{t} (I - \eta V \Sigma^2 V^T)^k V \Sigma U^T y$$

$$= \eta \sum_{k=0}^{t} V (I - \eta \Sigma^2)^k V^T V \Sigma U^T y$$

$$= \eta V \underbrace{\left[\sum_{k=0}^{t} (I - \eta \Sigma^2)^k \right]}_{} V^T V \Sigma U^T y$$

$$\xrightarrow{k \to \infty} \eta V (I - \underbrace{(I - \eta \Sigma^2)}_{\eta \Sigma^2})^{-1} V^T V \Sigma U^T y$$

$$= \eta V \frac{1}{\eta} \Sigma^{-2} V^T V \Sigma U^T y$$

$$= V \Sigma^{-1} U^T y$$

$$= X^+ y$$

$$\sum_{k=0}^{\infty} a^k = \frac{1}{1-a} \quad \text{if } |a| < 1$$

$V: n \times d$

$$X = U \Sigma V^T \qquad (U: n \times r,\ \Sigma \text{ diagonal matrix } r \times r) \qquad r = \mathrm{rank}(X) \le n,\ \le d$$

$$U^T U = I_r \qquad U U^T \text{ if } n = r,\ U U^T = I_n$$

$$V^T V = I_r$$

$$\eta X^T X = \eta V \Sigma \underbrace{U^T U}_{} \Sigma U^T = \eta V \Sigma^2 V^T$$

$$V = V V^T V$$

$$\sum_{k=0}^{\infty} A^k = (I - A)^{-1} \quad \text{if } \|A\|_{op} < 1 \quad (A \text{ symmetric})$$

$$\lim_{N\to\infty} (I - A) \sum_{k=0}^{N} A^k = \lim_{N\to\infty} \sum_{k=0}^{N} A^k - \sum_{k=1}^{N+1} A^k$$

$$= \lim_{N\to\infty} I - \underbrace{A^{N+1}}_{\to 0} = I$$

$$\lambda_{max}(I - \eta \Sigma^2) = 1 - \eta \lambda_{min}(\Sigma^2) < 1$$

$$\lambda_{min}(I - \eta \Sigma^2) = 1 - \eta \lambda_{max}(\Sigma^2) \quad \text{if } -1 < \lambda_i(A) < 1$$

$$= 1 - \eta \|\Sigma\|_{op}^2 > -1 \quad \text{if } \eta < \frac{2}{\sigma_{max}(X)^2}$$

$$(I - \eta \Sigma^2)_{ii} = 1 - \eta \Sigma_{ii}^2$$

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$

$UU^T U \Sigma V^T = U \Sigma V^T = X$

assume $Xw = y$

$\underbrace{UU^T X}_{X} w = UU^T y$

$X(X^\dagger y + q) = y$

$Xw = y$

$\therefore y = UU^T y$

$U \Sigma V^T (V \Sigma^{-1} U^T y + q) = y$

if $\mathrm{rank}(X) = n$

$UU^T y + U \Sigma V^T q = y$

$Xq = 0 = U \Sigma V^T q = 0 \Rightarrow V^T q = 0$

$\| V \Sigma^{-1} U^T y + q \|^2 = \underbrace{y^T U \Sigma^{-2} U^T y} + y^T U \Sigma^{-1} \underbrace{V^T q}_{0} + \underbrace{\|q\|^2}$

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
- "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + n\lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + n\lambda I)^{-1} y$

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
  - "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0^{(1)} \neq 0$ in same analysis, get $w_\infty^{(\infty)} = (I - VV^\top) w_0^{(1)} + X^\dagger y$ (proof)

$\overset{?}{=} \operatorname{argmin}_{Xw=y} \|x - w_0\|^2$

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$

  $\approx \frac{2n}{(\sqrt{n}+\sqrt{d})^2} = \frac{2}{1+\sqrt{\frac{d}{n}}+\frac{d}{n}}$ if $d = \omega(n) \to 0$

  - "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0 \neq 0$ in same analysis, get $w_\infty = (I - VV^\top) w_0 + X^\dagger y$ (proof)

- So, the 1,000-degree polynomial picture is what (small-LR) GD would give

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < \cancel{2} n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
  - "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0 \neq 0$ in same analysis, get $w_\infty = (I - V V^\top) w_0 + X^\dagger y$ (proof)

- So, the 1,000-degree polynomial picture is what (small-LR) GD would give
- Does this same idea hold for other losses / models? **Not necessarily.**

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
  - "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0 \neq 0$ in same analysis, get $w_\infty = (I - VV^\top) w_0 + X^\dagger y$ (proof)

- So, the 1,000-degree polynomial picture is what (small-LR) GD would give
- Does this same idea hold for other losses / models? **Not necessarily.**
  - Logistic regression:

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
  - "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0 \neq 0$ in same analysis, get $w_\infty = (I - VV^\top) w_0 + X^\dagger y$ (proof)

- So, the 1,000-degree polynomial picture is what (small-LR) GD would give
- Does this same idea hold for other losses / models? **Not necessarily.**
  - Logistic regression:
    - Separable: norm diverges in direction of max-margin separator (Soudry et al.)

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
  - "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0 \neq 0$ in same analysis, get $w_\infty = (I - V V^\top) w_0 + X^\dagger y$ (proof)

- So, the 1,000-degree polynomial picture is what (small-LR) GD would give
- Does this same idea hold for other losses / models? **Not necessarily.**
  - Logistic regression:
    - Separable: norm diverges in direction of max-margin separator (Soudry et al.)
    - Non-separable: biased towards max-margin, but complicated (Ji/Telgarsky)

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
  - "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0 \neq 0$ in same analysis, get $w_\infty = (I - V V^\top) w_0 + X^\dagger y$ (proof)

- So, the 1,000-degree polynomial picture is what (small-LR) GD would give
- Does this same idea hold for other losses / models? **Not necessarily.**
  - Logistic regression:
    - Separable: norm diverges in direction of max-margin separator (Soudry et al.)
    - Non-separable: biased towards max-margin, but complicated (Ji/Telgarsky)
    - Also see Telgarsky notes section 10

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
  - "Ridgeless" regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0 \neq 0$ in same analysis, get $w_\infty = (I - VV^\top) w_0 + X^\dagger y$ (proof)

- So, the 1,000-degree polynomial picture is what (small-LR) GD would give
- Does this same idea hold for other losses / models? **Not necessarily.**
  - Logistic regression:
    - Separable: norm diverges in direction of max-margin separator (Soudry et al.)
    - Non-separable: biased towards max-margin, but complicated (Ji/Telgarsky)
    - Also see Telgarsky notes section 10

$X = UV^T$

  - Matrix factorization models: conjectured min nuclear norm, slightly controversial

Implicit regularization of gradient descent

- We just showed that gradient descent for OLS with X of rank n, starting from zero with $\eta < 2n / \sigma_{\max}(X)^2$, converges to the minimum-norm interpolator $X^\dagger y$
  - “Ridgeless” regression: $\lim_{\lambda \to 0} (X^\top X + \lambda I)^{-1} X^\top y = X^\dagger y = \lim_{\lambda \to 0} X^\top (X X^\top + \lambda I)^{-1} y$
  - If we track $w_0 \neq 0$ in same analysis, get $w_\infty = (I - VV^\top) w_0 + X^\dagger y$ (proof)

- So, the 1,000-degree polynomial picture is what (small-LR) GD would give
- Does this same idea hold for other losses / models? **Not necessarily.**
  - Logistic regression:
    - Separable: norm diverges in direction of max-margin separator (Soudry et al.)
    - Non-separable: biased towards max-margin, but complicated (Ji/Telgarsky)
    - Also see Telgarsky notes section 10
  - Matrix factorization models: conjectured min nuclear norm, slightly controversial
  - Deep learning: ???

Double descent

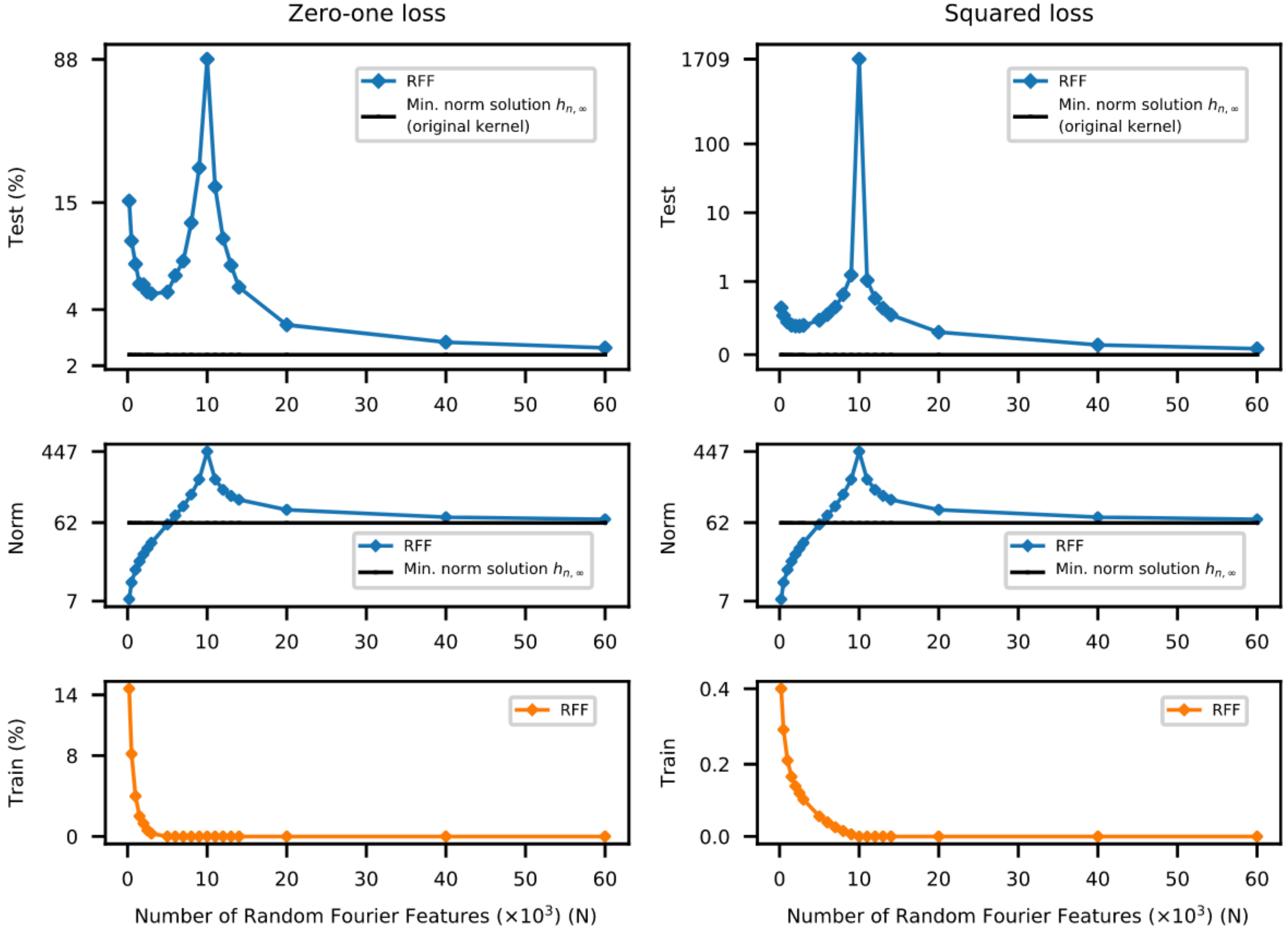


Fig. 2. Double-descent risk curve for the RFF model on MNIST. Shown are test risks (log scale), coefficient ℓ_2 norms (log scale), and training risks of the RFF model predictors $h_{n,N}$ learned on a subset of MNIST ($n = 10^4$, 10 classes). The interpolation threshold is achieved at $N = 10^4$.

Belkin/Hsu/Ma/Mandal, PNAS 2019

Double descent

Classical regime (left of peak): unique ERM

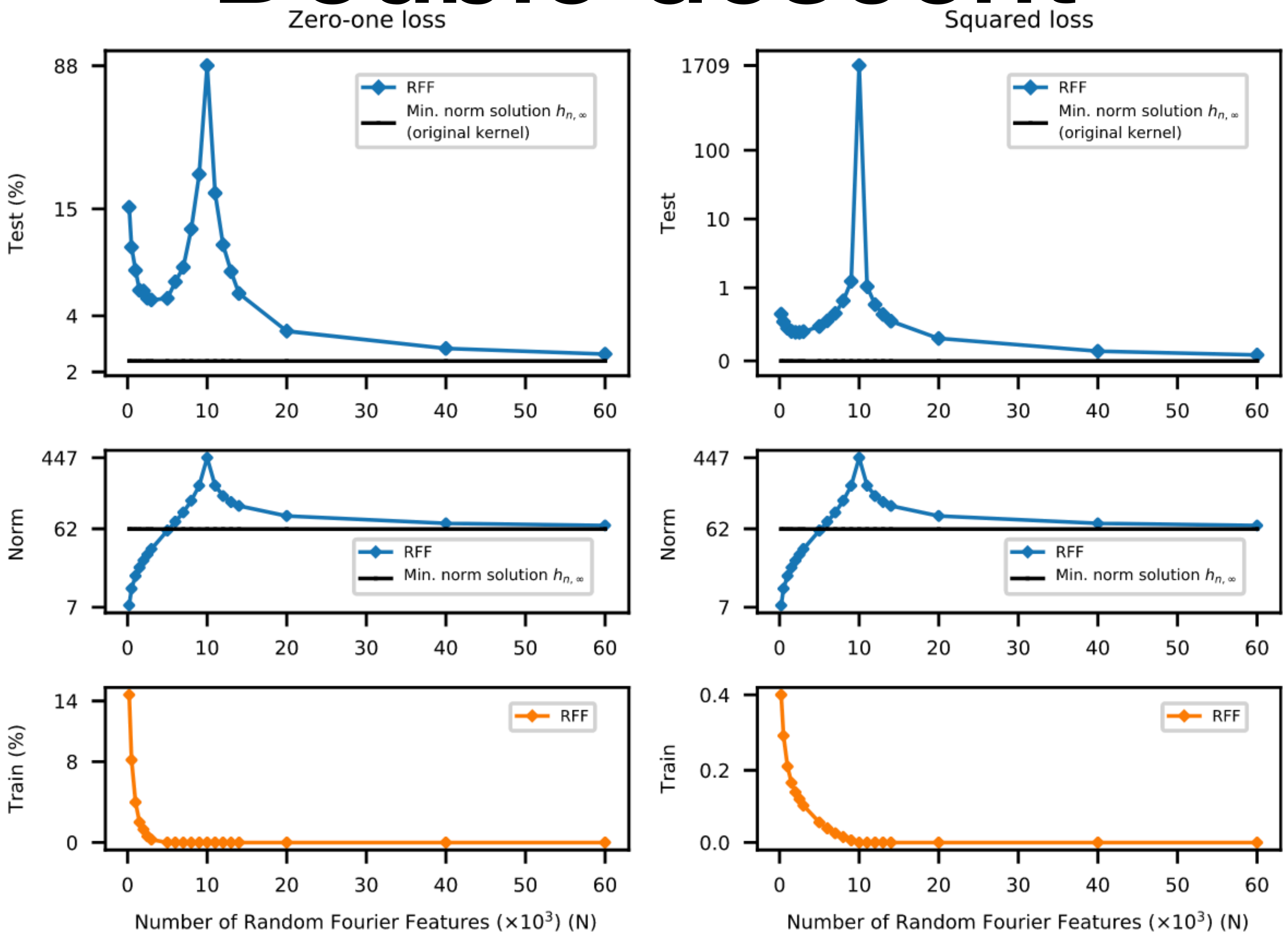
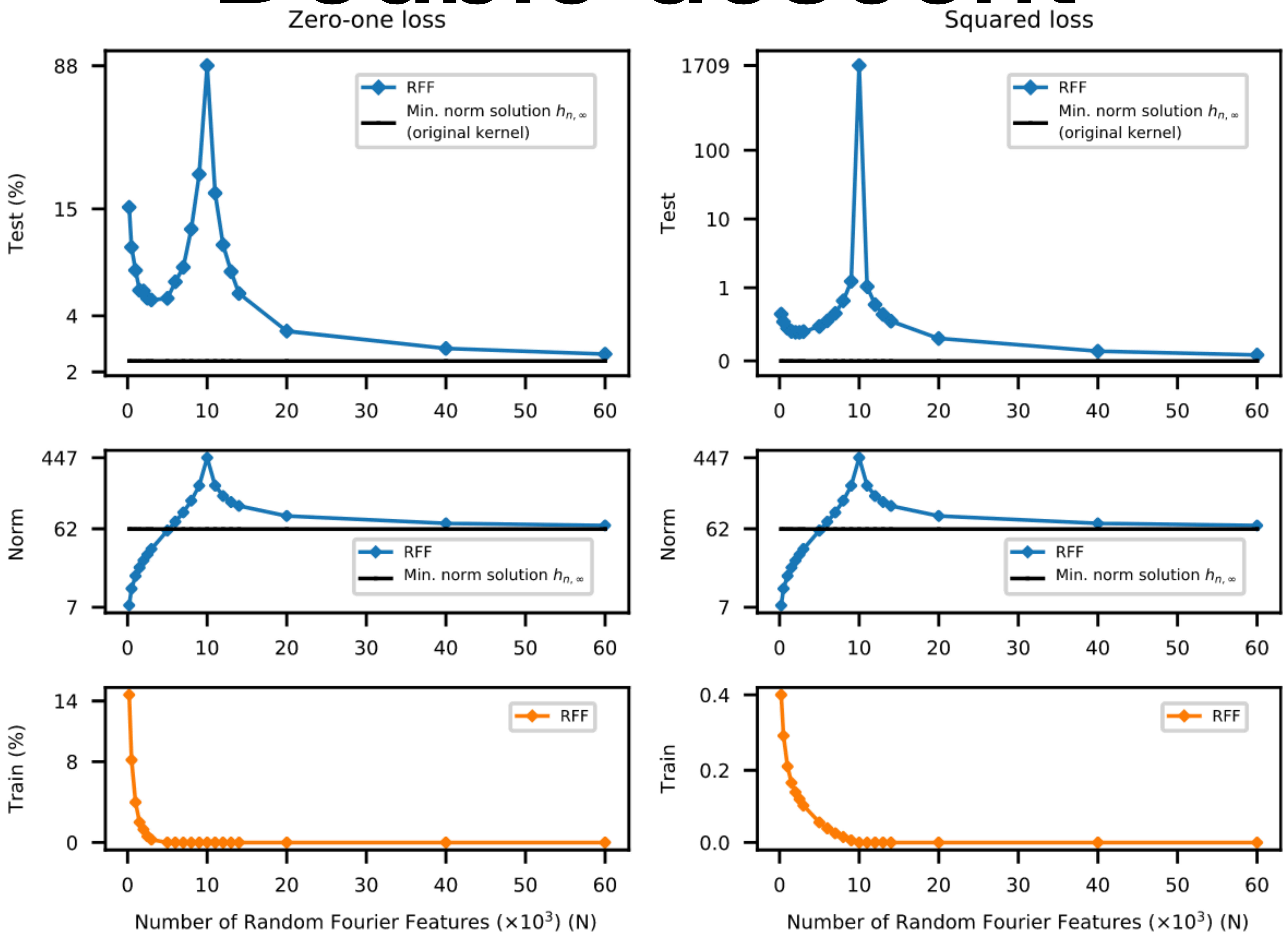


Fig. 2. Double-descent risk curve for the RFF model on MNIST. Shown are test risks (log scale), coefficient ℓ_2 norms (log scale), and training risks of the RFF model predictors $h_{n,N}$ learned on a subset of MNIST ($n = 10^4$, 10 classes). The interpolation threshold is achieved at $N = 10^4$.

Belkin/Hsu/Ma/Mandal, PNAS 2019

Double descent

Classical regime (left of peak): unique ERM



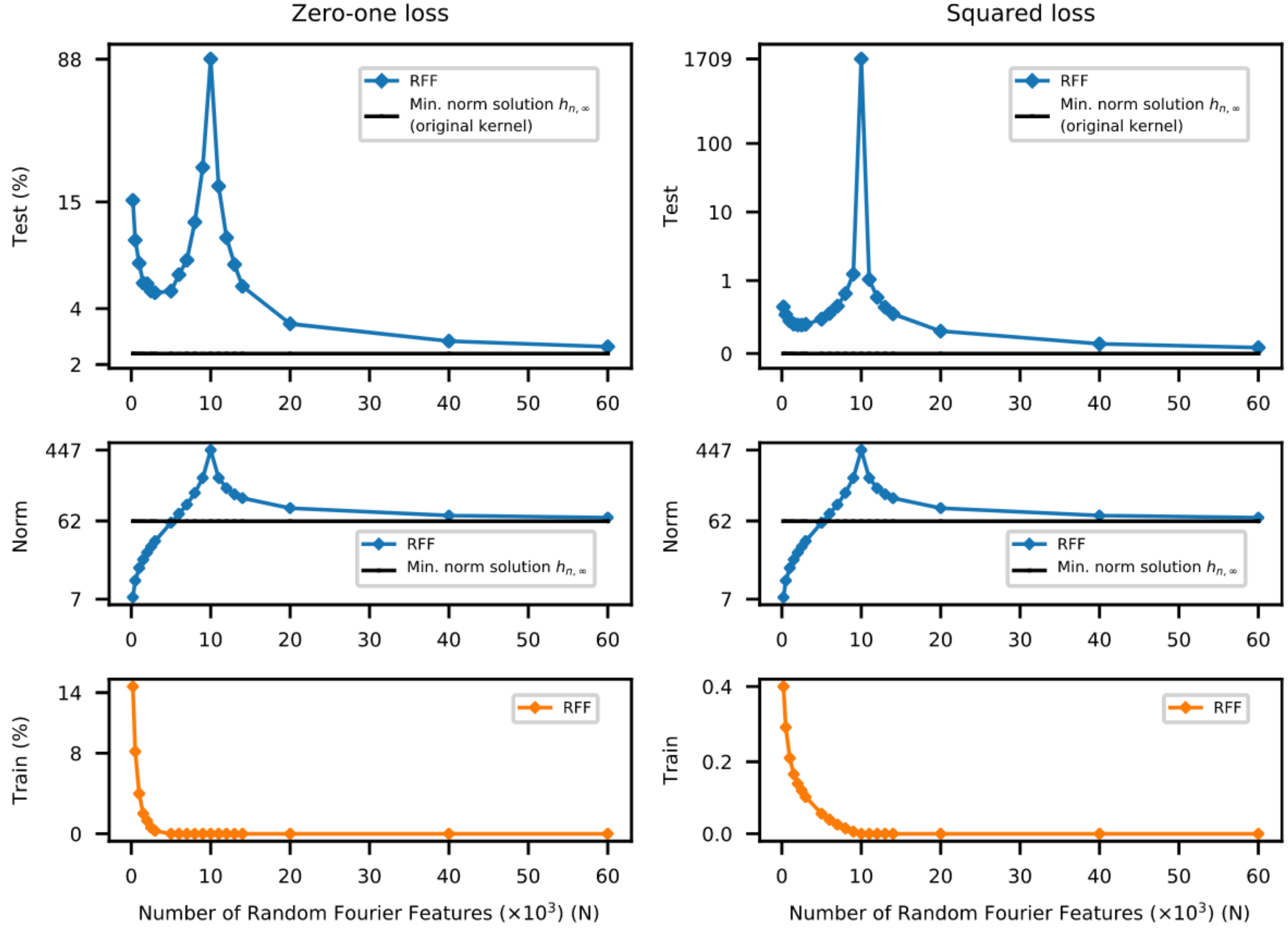
Interpolating regime (right of peak): many possible interpolators

Fig. 2. Double-descent risk curve for the RFF model on MNIST. Shown are test risks (log scale), coefficient ℓ_2 norms (log scale), and training risks of the RFF model predictors $h_{n,N}$ learned on a subset of MNIST ($n = 10^4$, 10 classes). The interpolation threshold is achieved at $N = 10^4$.

Belkin/Hsu/Ma/Mandal, PNAS 2019

Double descent

Classical regime (left of peak): unique ERM



Interpolating regime (right of peak): many possible interpolators

which one we get depends on alg.'s implicit bias

Fig. 2. Double-descent risk curve for the RFF model on MNIST. Shown are test risks (log scale), coefficient ℓ_2 norms (log scale), and training risks of the RFF model predictors $h_{n,N}$ learned on a subset of MNIST ($n = 10^4$, 10 classes). The interpolation threshold is achieved at $N = 10^4$.

Belkin/Hsu/Ma/Mandal, PNAS 2019

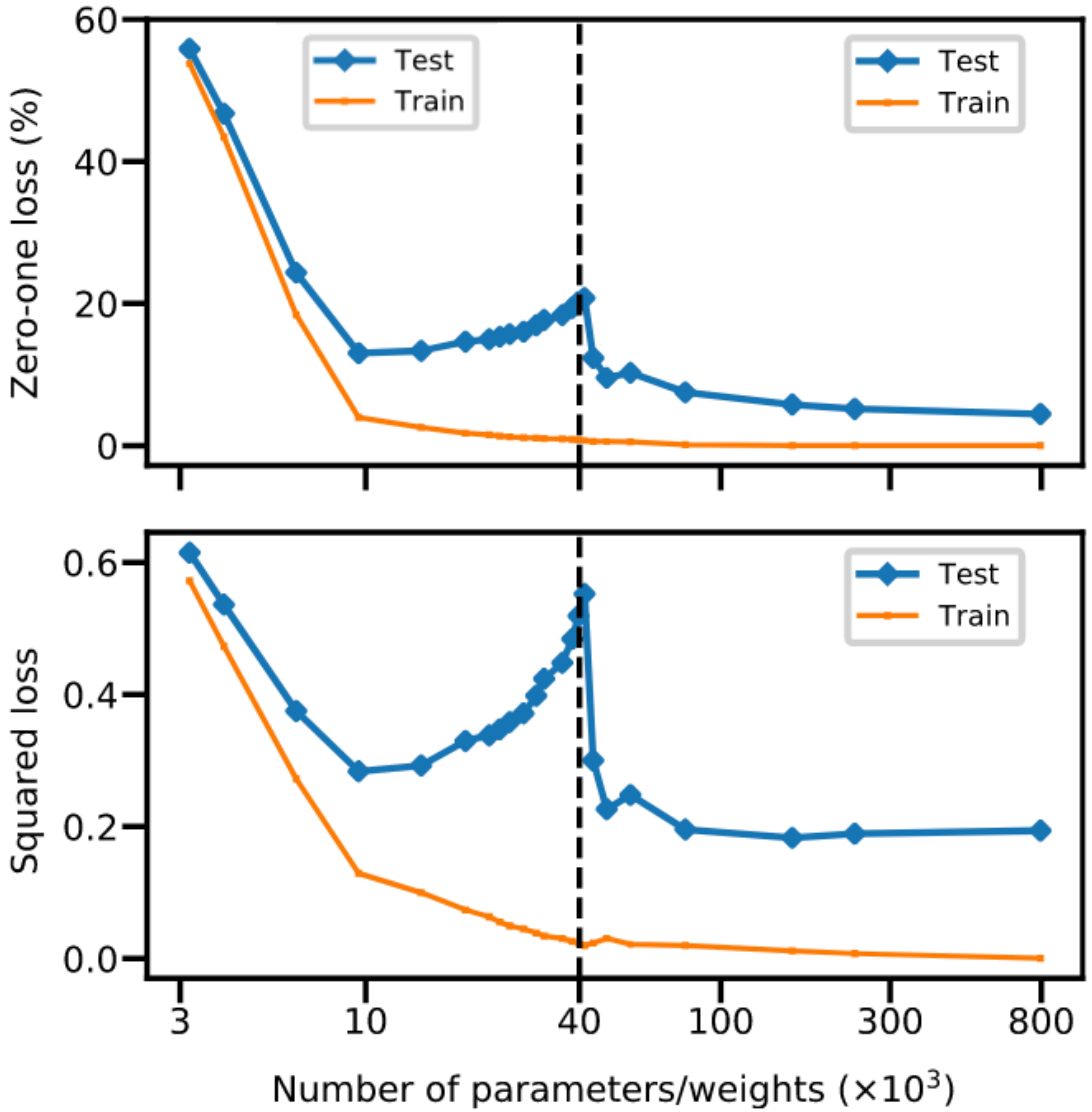


Fig. 3. Double-descent risk curve for a fully connected neural network on MNIST. Shown are training and test risks of a network with a single layer of H hidden units, learned on a subset of MNIST ($n = 4 \cdot 10^3$, $d = 784$, $K = 10$ classes). The number of parameters is $(d+1) \cdot H + (H+1) \cdot K$. The interpolation threshold (black dashed line) is observed at $n \cdot K$.

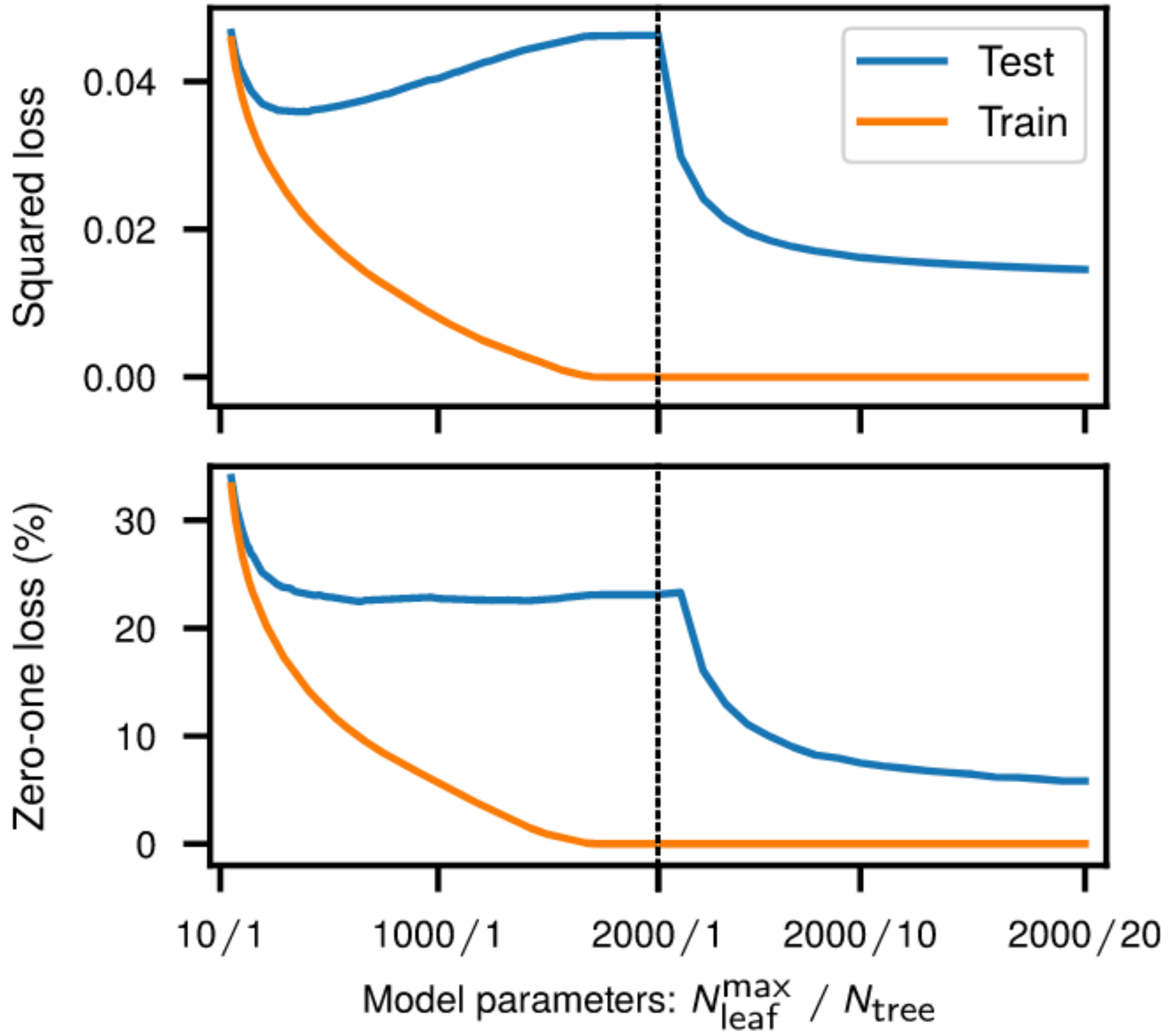
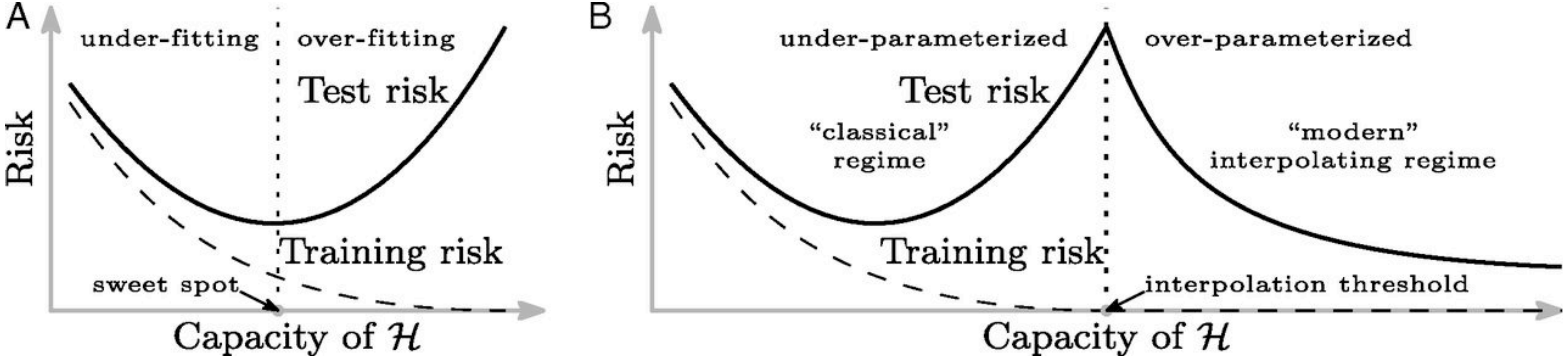
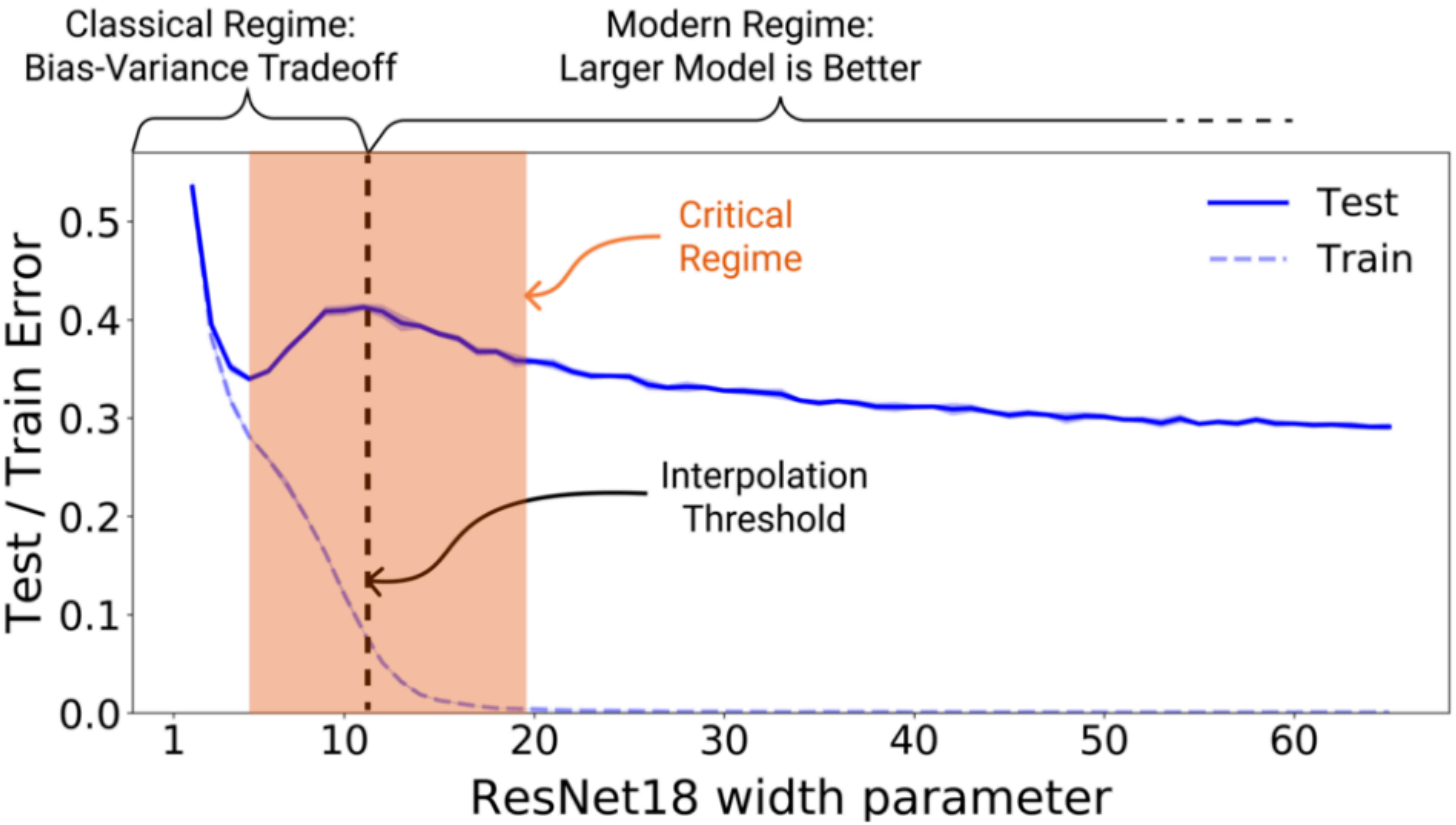


Fig. 4. Double-descent risk curve for random forests on MNIST. The double-descent risk curve is observed for random forests with increasing model complexity trained on a subset of MNIST ($n = 10^4$, 10 classes). Its complexity is controlled by the number of trees N_{tree} and the maximum number of leaves allowed for each tree $N_{\text{leaf}}^{\max}$.

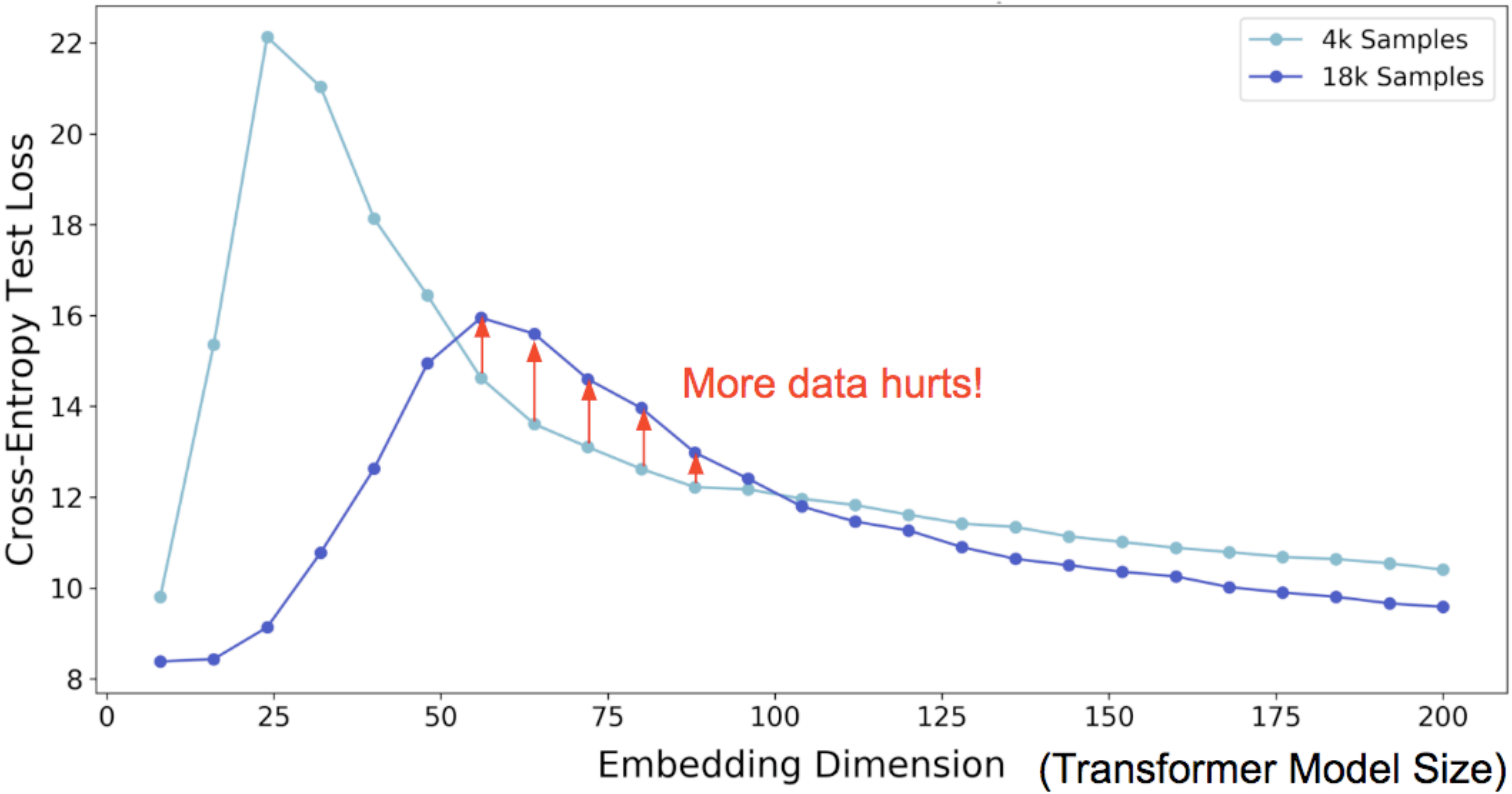
Belkin/Hsu/Ma/Mandal, PNAS 2019



Belkin/Hsu/Ma/Mandal, PNAS 2019

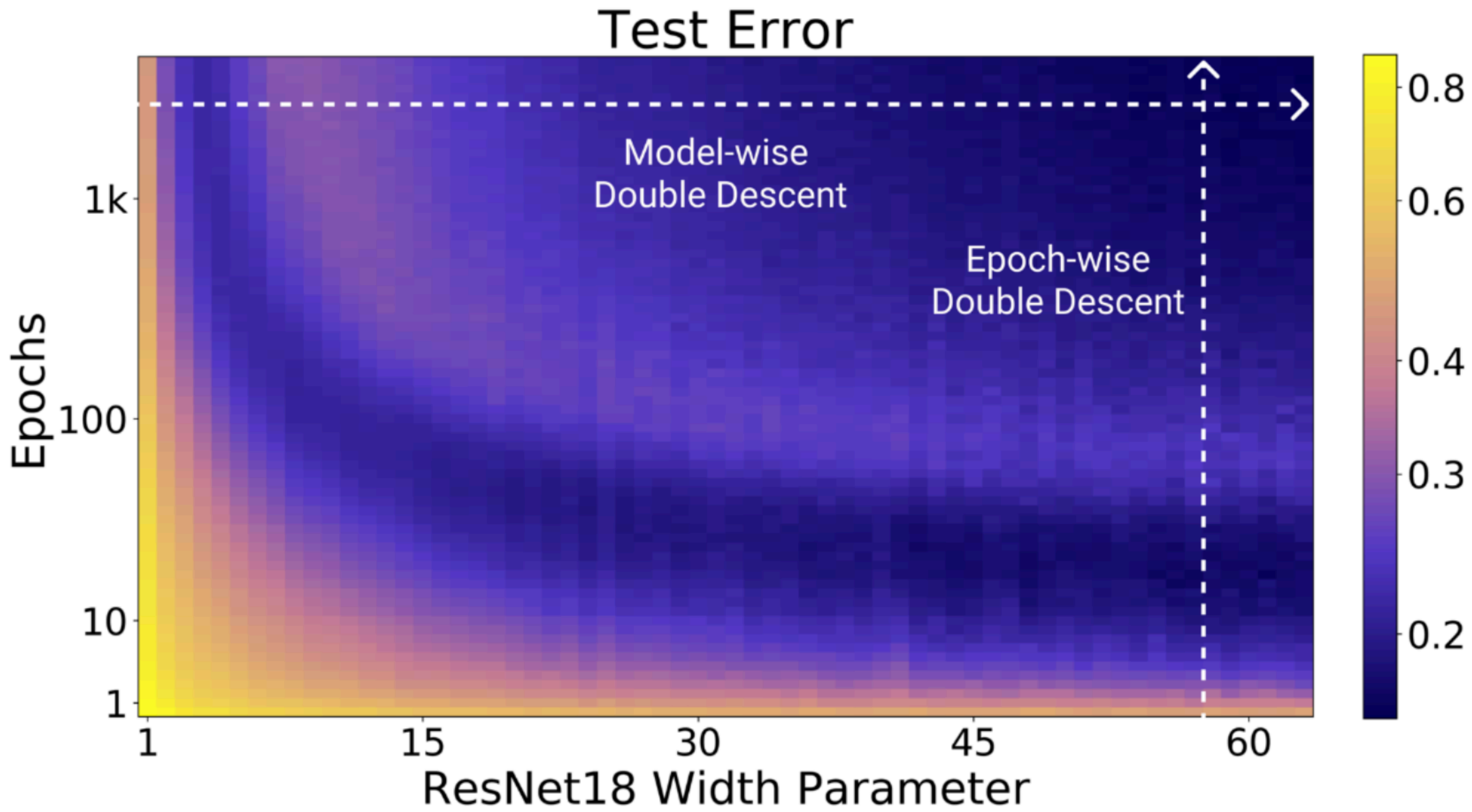


Nakkiran et al. ICLR-20



(Transformer Model Size)

Nakkiran et al. ICLR-20



Nakkiran et al. ICLR-20

Definition 1 (Effective Model Complexity) *The* Effective Model Complexity *(EMC) of a training procedure $\mathcal{T}$, with respect to distribution $\mathcal{D}$ and parameter $\epsilon > 0$, is defined as:*

$$\mathrm{EMC}_{\mathcal{D},\epsilon}(\mathcal{T}) := \max\left\{n \mid \mathbb{E}_{S\sim\mathcal{D}^n}[\mathrm{Error}_S(\mathcal{T}(S))] \leq \epsilon\right\}$$

where $\mathrm{Error}_S(M)$ is the mean error of model M on train samples S.

Our main hypothesis can be informally stated as follows:

Hypothesis 1 (Generalized Double Descent hypothesis, informal) *For any natural data distribution $\mathcal{D}$, neural-network-based training procedure $\mathcal{T}$, and small $\epsilon > 0$, if we consider the task of predicting labels based on n samples from $\mathcal{D}$ then:*

Under-paremeterized regime. *If $\mathrm{EMC}_{\mathcal{D},\epsilon}(\mathcal{T})$ is sufficiently smaller than n, any perturbation of $\mathcal{T}$ that increases its effective complexity will decrease the test error.*

Over-parameterized regime. *If $\mathrm{EMC}_{\mathcal{D},\epsilon}(\mathcal{T})$ is sufficiently larger than n, any perturbation of $\mathcal{T}$ that increases its effective complexity will decrease the test error.*

Critically parameterized regime. *If $\mathrm{EMC}_{\mathcal{D},\epsilon}(\mathcal{T}) \approx n$, then a perturbation of $\mathcal{T}$ that increases its effective complexity might decrease* **or increase** *the test error.*

Nakkiran et al. ICLR-20

(pause)

Neural Tangent Kernels (NTKs)

Neural Tangent Kernels (NTKs)

- Gradient descent for square loss finds min-norm interpolator (“ridgeless” regression)

Neural Tangent Kernels (NTKs)

- Gradient descent for square loss finds min-norm interpolator ("ridgeless" regression)

- As we'll see, training an "ultrawide" deep network for square loss ends up being equivalent to "ridgeless" regression with a neural tangent kernel

Neural Tangent Kernels (NTKs)

- Gradient descent for square loss finds min-norm interpolator
  (“ridgeless” regression)

- As we’ll see, training an “ultrawide” deep network for square loss
  ends up being equivalent to “ridgeless” regression
  with a neural tangent kernel

- So, *in the infinite-width limit*, we know things correspond to finding the solution that has small RKHS norm for the neural tangent kernel

Neural Tangent Kernels (NTKs)

- Gradient descent for square loss finds min-norm interpolator
("ridgeless" regression)

- As we'll see, training an "ultrawide" deep network for square loss
ends up being equivalent to "ridgeless" regression
with a neural tangent kernel

- So, *in the infinite-width limit*, we know things correspond to finding the solution that has small RKHS norm for the neural tangent kernel

Neural Tangent Kernels (NTKs)

- Gradient descent for square loss finds min-norm interpolator ("ridgeless" regression)

- As we'll see, training an "ultrawide" deep network for square loss ends up being equivalent to "ridgeless" regression with a neural tangent kernel

- So, *in the infinite-width limit*, we know things correspond to finding the solution that has small RKHS norm for the neural tangent kernel

- Another POV:

$$L_{\mathscr{D}}(\mathscr{A}(S)) - L^* = \underbrace{L_{\mathscr{D}}(\mathscr{A}(S)) - L_{\mathscr{D}}(\mathrm{ERM}_{\mathscr{H}}(S))}_{\text{optimization error}} + \underbrace{L_{\mathscr{D}}(\mathrm{ERM}_{\mathscr{H}}(S))) - \inf_{h \in \mathscr{H}} L_{\mathscr{D}}(h)}_{\text{estimation error}} + \underbrace{\inf_{h \in \mathscr{H}} L_{\mathscr{D}}(h) - L^*}_{\text{approximation error}}$$

Nonconvex optimization

- Neural nets are not convex $\ell(w; (x, y)) = (f_w(x) - y)^2$

Nonconvex optimization

- Neural nets are not convex
- Even **deep linear networks** are not convex

$$f_W(x) = \underbrace{\overset{1\times d_L}{W_L} \cdots \overset{d_3\times d_2}{W_2}\ \overset{d_2\times d_1}{W_1}}_{\in \mathbb{R}^{1\times d_1}} \overset{d_1\times 1}{x}$$

Nonconvex optimization

- Neural nets are not convex
- Even **deep linear networks** are not convex

- But we do know that SGD converges to a *critical point* under fairly mild conditions

Nonconvex optimization

- Neural nets are not convex
- Even **deep linear networks** are not convex

- But we do know that SGD converges to a *critical point* under fairly mild conditions
  - e.g.: if $f \geq f^{\inf}$ is differentiable and β-smooth, and there are A, B, C s.t. for all x, $\mathbb{E}\left[\|\hat{g}(x)\|^2\right] \leq 2A(f(x) - f^{\inf}) + B\|\nabla f(X)\|^2 + C$, then the *best* iterate from $\mathcal{O}(\varepsilon^{-4})$ steps has $\mathbb{E}\left[\|\nabla f(x)\|^2\right] \leq \varepsilon^2$ (Khaled/Richtárik 2020)

Nonconvex optimization

- Neural nets are not convex
- Even **deep linear networks** are not convex

- But we do know that SGD converges to a *critical point* under fairly mild conditions
  - e.g.: if $f \geq f^{\inf}$ is differentiable and β-smooth, and there are A, B, C s.t. for all x, $\mathbb{E}\left[\|\hat{g}(x)\|^2\right] \leq 2A(f(x) - f^{\inf}) + B\|\nabla f(X)\|^2 + C$, then the *best* iterate from $\mathcal{O}(\varepsilon^{-4})$ steps has $\mathbb{E}\left[\|\nabla f(x)\|^2\right] \leq \varepsilon^2$ (Khaled/Richtárik 2020)

- In deep linear nets, local minima are global minima (Kawaguchi 2016, Laurent/von Brecht 2019)

Nonconvex optimization

- Neural nets are not convex
- Even **deep linear networks** are not convex

- But we do know that SGD converges to a *critical point* under fairly mild conditions
  - e.g.: if $f \geq f^{\inf}$ is differentiable and β-smooth, and there are A, B, C s.t. for all x, $\mathbb{E}\left[\|\hat{g}(x)\|^2\right] \leq 2A(f(x) - f^{\inf}) + B\|\nabla f(X)\|^2 + C$, then the *best* iterate from $\mathcal{O}(\varepsilon^{-4})$ steps has $\mathbb{E}\left[\|\nabla f(x)\|^2\right] \leq \varepsilon^2$ (Khaled/Richtárik 2020)

- In deep linear nets, local minima are global minima (Kawaguchi 2016, Laurent/von Brecht 2019)
  - …but there are saddle points, including "bad" ones where $\lambda_{\min}(\nabla^2 f) = 0$

Nonconvex optimization

- Neural nets are not convex
- Even **deep linear networks** are not convex

- But we do know that SGD converges to a *critical point* under fairly mild conditions
  - e.g.: if $f \geq f^{\inf}$ is differentiable and β-smooth, and
    there are A, B, C s.t. for all x, $\mathbb{E}\left[\|\hat{g}(x)\|^2\right] \leq 2A(f(x) - f^{\inf}) + B\|\nabla f(X)\|^2 + C$,
    then the *best* iterate from $\mathcal{O}(\varepsilon^{-4})$ steps has $\mathbb{E}\left[\|\nabla f(x)\|^2\right] \leq \varepsilon^2$ (Khaled/Richtárik 2020)

- In deep linear nets, local minima are global minima (Kawaguchi 2016, Laurent/von Brecht 2019)
  - …but there are saddle points, including "bad" ones where $\lambda_{\min}(\nabla^2 f) = 0$
  - …but gradient descent almost surely escapes saddles, reaches a local min (Lee et al. 2016)

Nonconvex optimization

- Neural nets are not convex
- Even **deep linear networks** are not convex

- But we do know that SGD converges to a *critical point* under fairly mild conditions
  - e.g.: if $f \geq f^{\inf}$ is differentiable and β-smooth, and there are A, B, C s.t. for all x, $\mathbb{E}\left[\|\hat{g}(x)\|^2\right] \leq 2A(f(x) - f^{\inf}) + B\|\nabla f(X)\|^2 + C$, then the *best* iterate from $\mathcal{O}(\varepsilon^{-4})$ steps has $\mathbb{E}\left[\|\nabla f(x)\|^2\right] \leq \varepsilon^2$ (Khaled/Richtárik 2020)

- In deep linear nets, local minima are global minima (Kawaguchi 2016, Laurent/von Brecht 2019)
  - …but there are saddle points, including "bad" ones where $\lambda_{\min}(\nabla^2 f) = 0$
  - …but gradient descent almost surely escapes saddles, reaches a local min (Lee et al. 2016)
  - …but it can take exponential time to escape (Du et al. 2017)

Nonconvex optimization

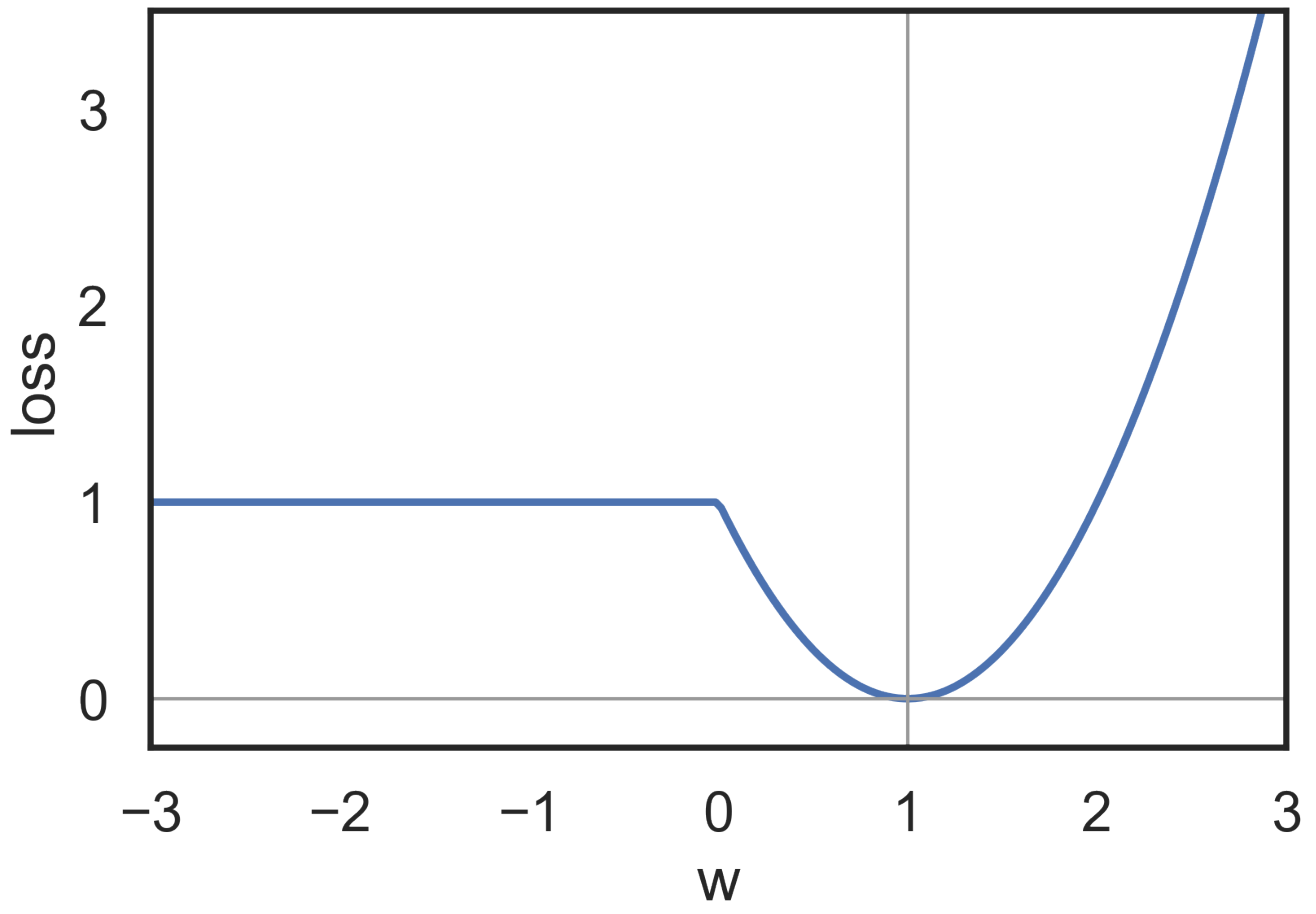
- Neural nets are not convex
- Even **deep linear networks** are not convex

- But we do know that SGD converges to a *critical point* under fairly mild conditions
  - e.g.: if $f \geq f^{\inf}$ is differentiable and β-smooth, and there are A, B, C s.t. for all x, $\mathbb{E}\left[\|\hat{g}(x)\|^2\right] \leq 2A(f(x) - f^{\inf}) + B\|\nabla f(X)\|^2 + C$, then the *best* iterate from $\mathcal{O}(\varepsilon^{-4})$ steps has $\mathbb{E}\left[\|\nabla f(x)\|^2\right] \leq \varepsilon^2$ (Khaled/Richtárik 2020)

- In deep linear nets, local minima are global minima (Kawaguchi 2016, Laurent/von Brecht 2019)
  - …but there are saddle points, including "bad" ones where $\lambda_{\min}(\nabla^2 f) = 0$
  - …but gradient descent almost surely escapes saddles, reaches a local min (Lee et al. 2016)
  - …but it can take exponential time to escape (Du et al. 2017)
  - …but that doesn't happen on deep linear nets [under conditions] (Arora et al. 2019)

Bad local minima in ReLU nets

$h(x) = \mathrm{ReLU}(wx)$ (reals to reals), square loss, $S = \big((1,1)\big)$:



Sub-Optimal Local Minima Exist for Neural Networks with Almost All Non-Linear Activations

Tian Ding* Dawei Li † Ruoyu Sun ‡

Nov 4, 2019