STAT 535

12/6/22

Lecture 19

Project:
Thu: Test set results
Next Tue: Report due
HW7: optional

- Double Descent
- Wide NN →

Kernel machines $\Rightarrow$ RKHS $= \{ f = \sum_{i=1}^{\infty} \alpha_i k(x^i, \cdot) \} = \mathcal{H}$

Random Fourier features

$\rightarrow$ feature space $\{ \psi_{\omega,\beta} = \frac{1}{\sqrt{2}} \cos(\omega^T x + \beta) \}$

$(e^{-i\omega^T x})$

$k(x,x') = \tilde{\varphi}(x)^T \tilde{\varphi}(x')$

$\tilde{\varphi}^T = [\psi_{\omega^k,\beta^k}, k = 1:D]$, $\omega \sim p_k$

$\beta^k \sim \text{unif}[0, 2\pi]$

$\|f\|_{\mathcal{H}} = \|w\|_2$ (SVM = HWG)

$f(x) = w^T \varphi(x)$ $\Big)$ Riesz

$f(x) \in \mathcal{H}_2$ $\quad$ φ feature map $\equiv \mathcal{H}_2$

$f^{SVM}(x) = \sum \alpha_i y^i k(x^i, x)$

α_i

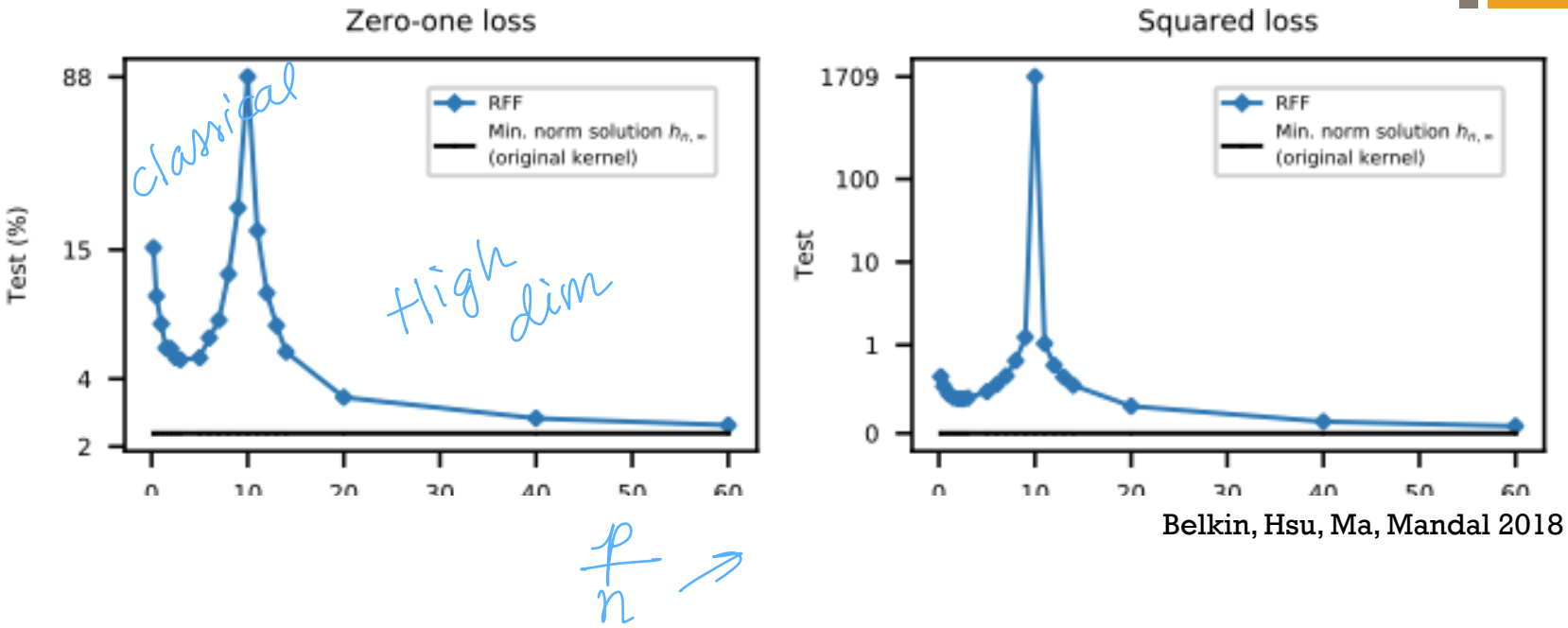
$\Downarrow$

$\|w\|^2 = \alpha^T G \alpha$

$G = [k(x^i, x^j)]_{ij}$

Double Descent

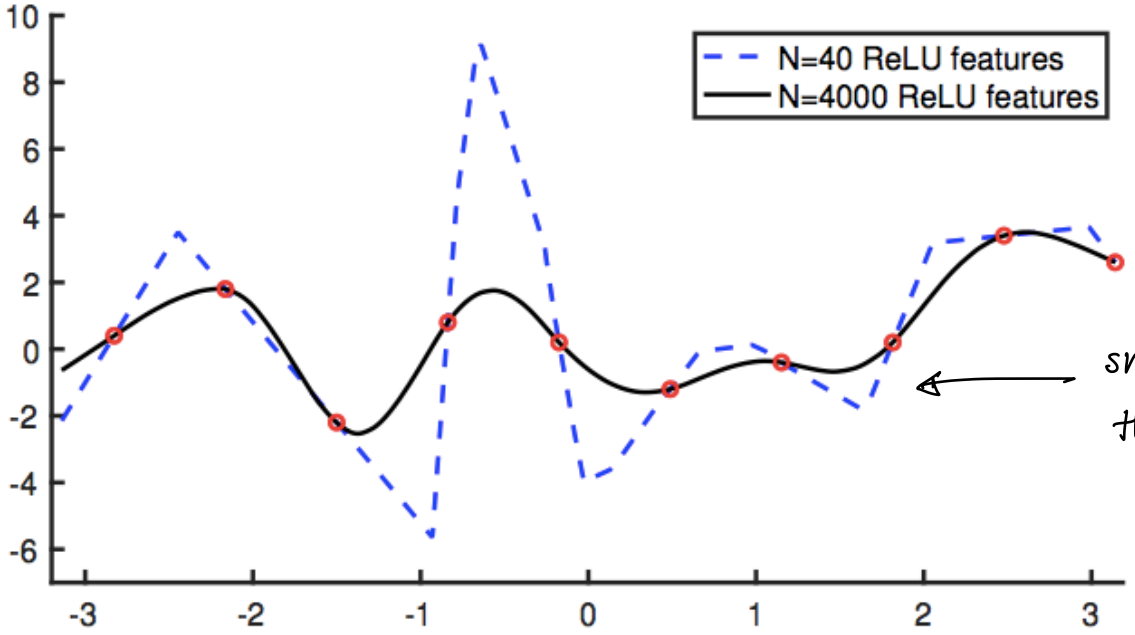
What is observed



Belkin, Hsu, Ma, Mandal 2018

- Double descent curves for the generalization error
  - Random Fourier Features (RFF)
  - ReLU 2 layer networks (with random first layer weights)
  - Random Forests, l2-Adaboost
  - Linear regression
- With and without noise

Main intuition [Belkin et al.]



smoothest $f(x)$ that interpolates

$f(x^i) = y^i \quad i = 1:n$

$\min \|f\|_{\mathcal{H}}$

$\mathcal{F} \subset \mathcal{H}$

- The target function h^* is (mostly) smooth
  - i.e. $||h^*||_{RKHS}$ is small
- $p > N$, no noise, hence h_p interpolates data
- Train to minimize $||h_p||$ subject to 0 training error
- Then $||h_p||$ will decrease with p!

Assume smoothest interpolator $f \in \mathcal{H}$ can be found

Random Fourier Features (RFF)

Random Fourier features. We first consider a popular class of non-linear parametric models called *Random Fourier Features* (*RFF*) [30], which can be viewed as a class of two-layer neural networks with fixed weights in the first layer. The RFF model family $\mathcal{H}_N$ with N (complex-valued) parameters consists of functions $h : \mathbb{R}^d \to \mathbb{C}$ of the form

$$h(x) = \sum_{k=1}^{N} a_k \phi(x; v_k) \quad \text{where} \quad \phi(x; v) := e^{\sqrt{-1}\langle v, x \rangle},$$

and the vectors $v_1, \ldots, v_N$ are sampled independently from the standard normal distribution in $\mathbb{R}^d$. (We consider $\mathcal{H}_N$ as a class of real-valued functions with $2N$ real-valued parameters by taking real and imaginary parts separately.) Note that $\mathcal{H}_N$ is a randomized function class, but as $N \to \infty$, the function class becomes a closer and closer approximation to the Reproducing Kernel Hilbert Space (RKHS) corresponding to the Gaussian kernel, denoted by $\mathcal{H}_\infty$.

$\mathcal{H}_D \longrightarrow \mathcal{H}_\infty$

RFF

$D \to \infty$

- RFF → $\mathcal{H}_{\text{infinity}}$

Theorem

$x^{1:n}, y^{1:n} \in [0,1]^d$

Theorem 1. *Fix any $h^* \in \mathcal{H}_\infty$. Let $(x_1, y_1), \ldots, (x_n, y_n)$ be independent and identically distributed random variables, where x_i is drawn uniformly at random from a compact cube[2] $\Omega \subset \mathbb{R}^d$, and $y_i = h^*(x_i)$ for all i. There exists absolute constants $A, B > 0$ such that, for any interpolating $h \in \mathcal{H}_\infty$ (i.e., $h(x_i) = y_i$ for all i), so that with high probability*

$$\sup_{x \in \Omega} |h(x) - h^*(x)| < A e^{-B(n/\log n)^{1/d}} \left(\|h^*\|_{\mathcal{H}_\infty} + \|h\|_{\mathcal{H}_\infty}\right).$$

no noise

any interpolator

err

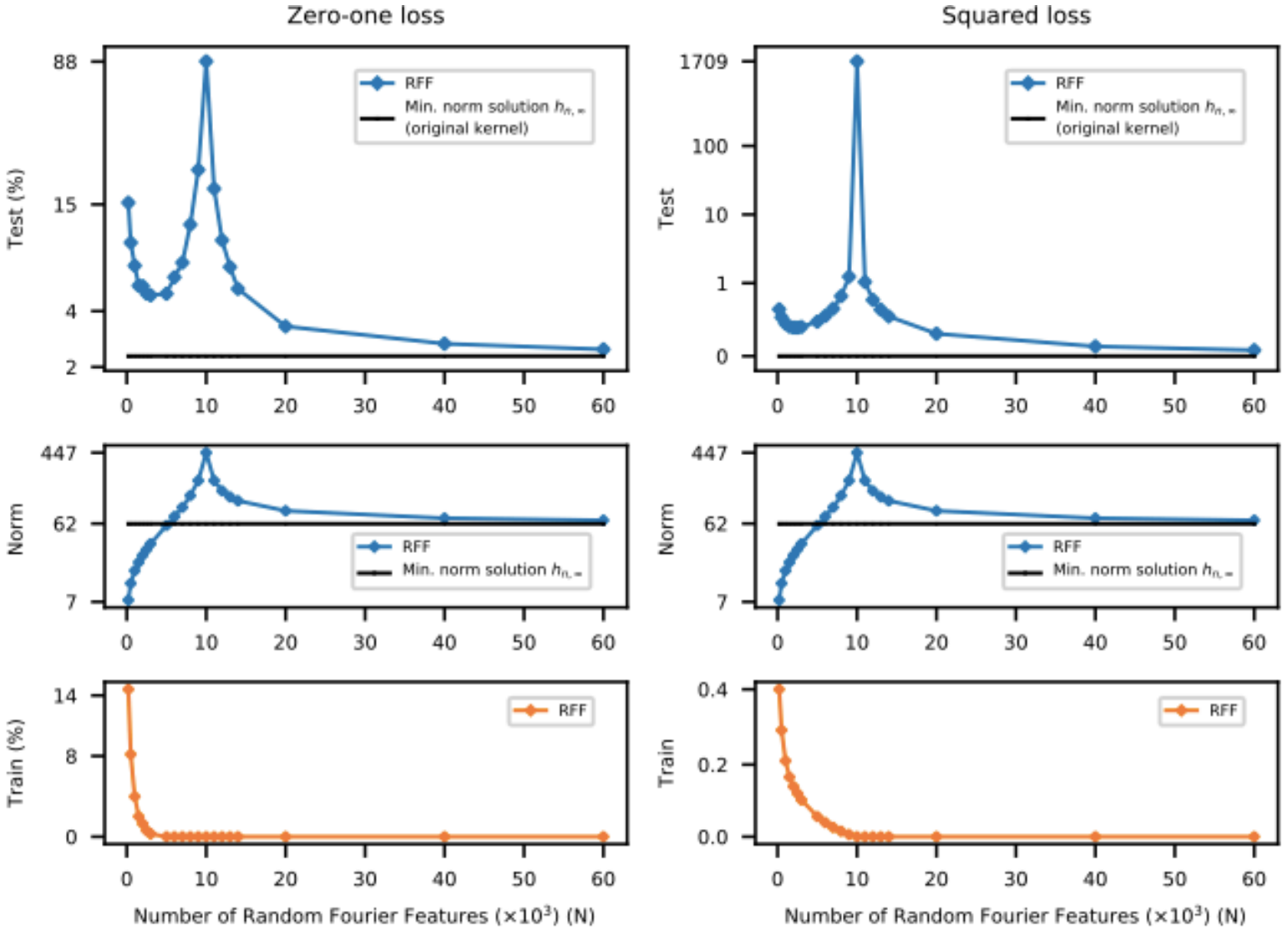
$\to 0$ for $n \to \infty$

smooth target

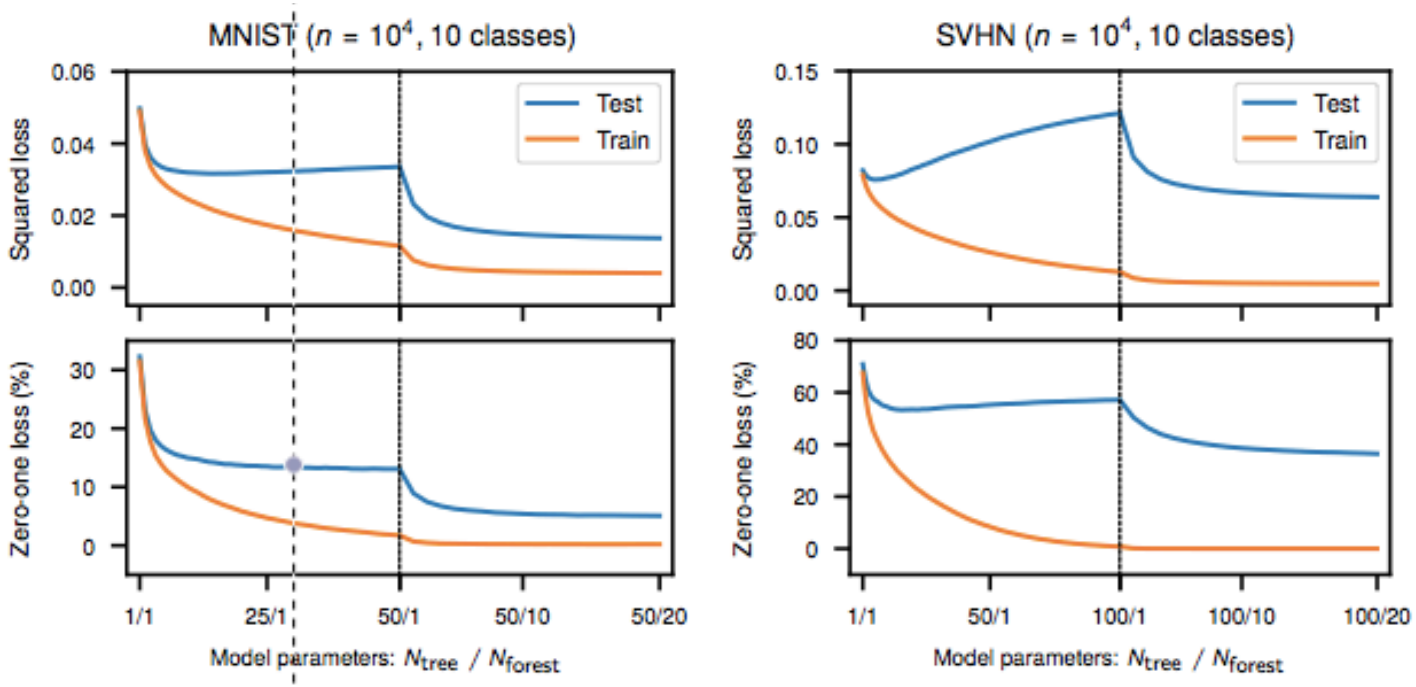
smooth predictor

err small

RFF


Zero-one loss
Squared loss
Test (%)
Test
Norm
Train (%)
Train
RFF
Min. norm solution $h_{n,*}$
(original kernel)
Number of Random Fourier Features (×10³) (N)

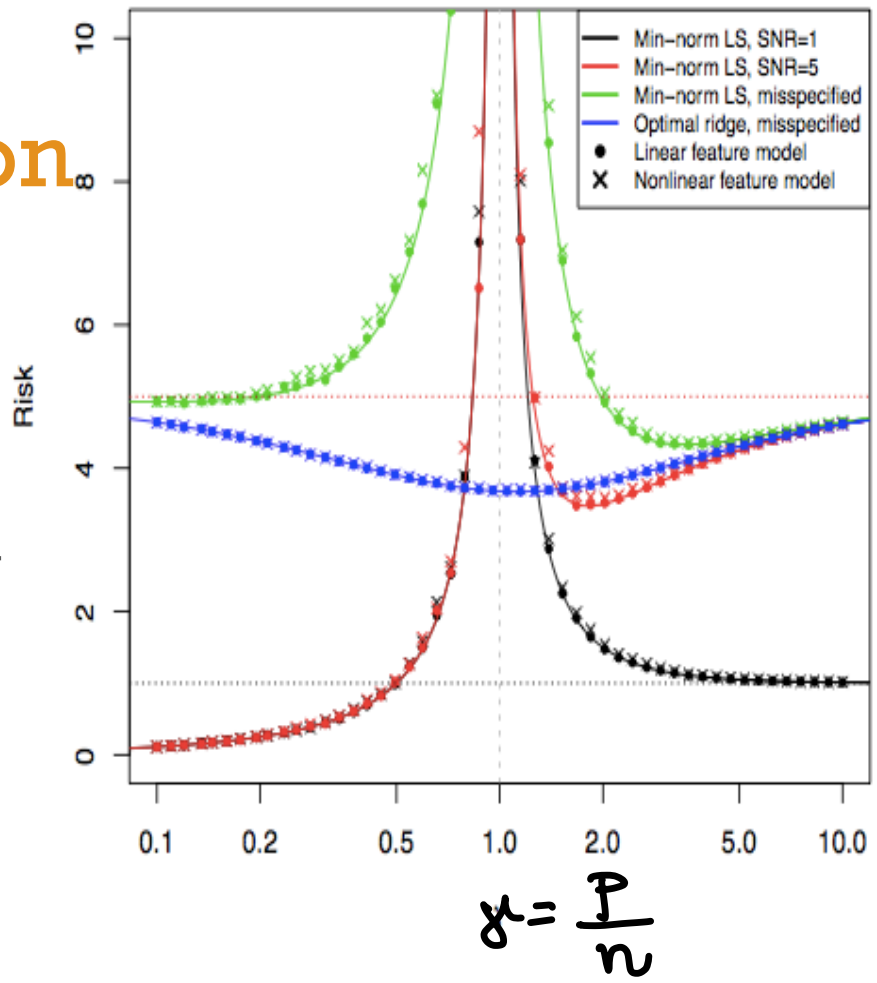
Boosted decision trees



Linear regression

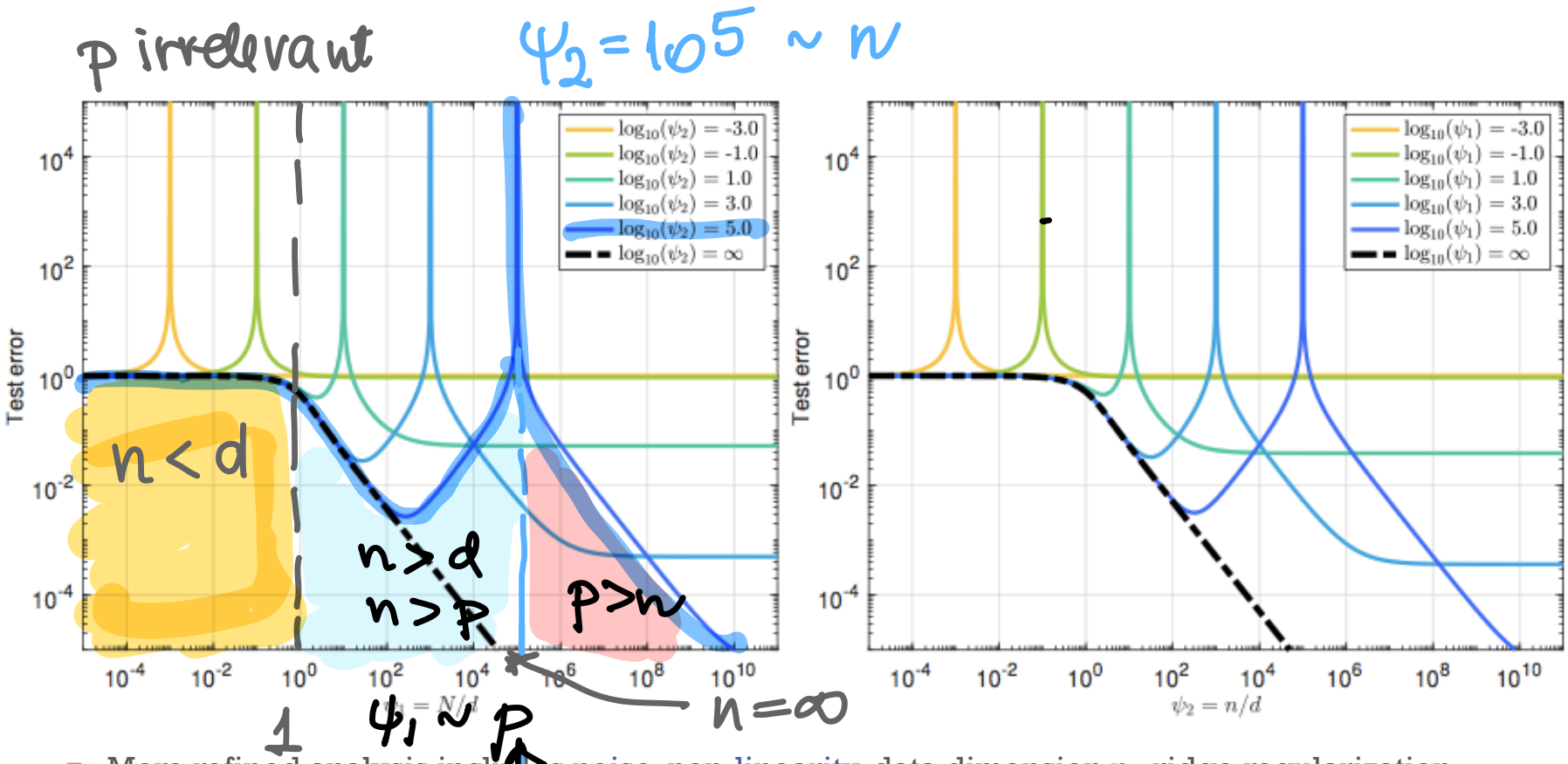
[Hastie, Montanari, Rosset, Tibshirani 2019]

- Linear, nonlinear features behave the same way
- Model correct, misspecified
- Noise level sigma affects asymptotic error
- and optimal N/n



- Double descent is not regularization

Figure 1: Asymptotic risk curves for the linear feature model, as a function of the limiting aspect ratio γ. The risks for min-norm least squares, when SNR $= 1$ and SNR $= 5$, are plotted in black and red, respectively. These two match for $\gamma < 1$ but differ for $\gamma > 1$. The null risks for SNR $= 1$ and SNR $= 5$ are marked by the dotted black and red lines, respectively. The risk for the case of a misspecified model (with significant approximation bias, $a = 1.5$ in (13)), when SNR $= 5$, is plotted in green. Optimally-tuned (equivalently, CV-tuned) ridge regression, in the same misspecified setup, has risk plotted in blue. The points denote finite-sample risks, with $n = 200$, $p = [\gamma n]$, across various values of γ, computed from features X having i.i.d. $N(0, 1)$ entries. Meanwhile, the "x" points mark finite-sample risks for a nonlinear feature model, with $n = 200$, $p = [\gamma n]$, $d = 100$, and $X = \varphi(ZW^T)$, where Z has i.i.d. $N(0, 1)$ entries, W has i.i.d. $N(0, 1/d)$ entries, and $\varphi(t) = a(|t| - b)$ is a "purely nonlinear" activation function, for constants a, b. The theory predicts that this nonlinear risk should converge to the linear risk with p features (regardless of d). The empirical agreement between these two—and the agreement in finite-sample and asymptotic risks—is striking.



- More refined analysis includes noise, non-linearity, data dimension n, ridge regularization lambda [Mei, Montanari 20[illegible]9]
- When is global minimum in overparametrized regime?
- Enough data N/n > 1
- lambda → 0 (or min-norm LS)
- p >> N
- SNR || beta ||/noise > 1
- Bias, Variance strictly decreasing with p/N to > 0 limit

$\psi_1 = \frac{p}{d}$ $\psi_2 = \frac{n}{d} =$ data dim

hidden / input dim

p = hidden units

n

d = dim x

$\gamma = \frac{p}{d} = \frac{\psi_1}{\psi_2}$

Lecture Notes IV.I.2 – Simple analysis of gradient descent

Marina Meilă
mmp@stat.washington.edu

Department of Statistics
University of Washington

November, 2022

so far : smooth is good !

now : GD ⇒ smooth

Rate of linear convergence ✗

Newton-Raphson "rounds" the surface of f around minimum ✗

Implicit bias of Gradient Descent ✓

Reading HTF Ch.: –, Murphy Ch.: –, Bach Ch.: , Bach Chapter 5.2, 10.1

Useful facts

Assume that our function f is quadratic, i.e

$$f(x) = \frac{1}{2}x^T Hx + g^T x + c \text{ with } H \succ 0. \quad (1)$$

Then,

$$\nabla f(x) = Hx + g = H(x - x^*) \quad (2)$$
$$\nabla^2 f(x) = H \quad (3)$$
$$x^* = -H^{-1}g, \quad \text{and} \quad Hx^* = -g \quad (4)$$
$$(5)$$

Gradient descent $x^{t+1} = x^t - \eta \nabla f(x^t)$

Rate of linear convergence

$$x^{t+1} - x^* = (x^t - \eta H(x^t - x^*)) - x^* \tag{6}$$

$$= [I - \eta H](x^t - x^*) = (I - \eta H)^t (x^0 - x^*) \tag{7}$$

$$e^{t+1} \leq \|I - \eta H\|^t e^0 \quad \text{with } e^t = \|x^t - x^*\| \tag{8}$$

$$f(x) - f(x^*) = \frac{1}{2}(x - x^*)^T H(x - x^*) \quad \text{for any } x \tag{9}$$

Proof

$$\frac{1}{2}(x - x^*)^T H(x - x^*) = \frac{1}{2}x^T Hx + \frac{1}{2}(x^*)^T Hx^* - \underbrace{x^T Hx^*}_{-x^T g} \quad \text{recall } Hx^* = -g \tag{10}$$

$$= f(x) - \left(\frac{1}{2}(x^*)^T Hx^* + g^T x^*\right) \tag{11}$$

Hence,

$$f(x) - f(x^*) = \frac{1}{2}(x^0 - x^*)^T (I - \eta H)^{2t} H(x^0 - x^*) \tag{12}$$

because

$$H(I - \eta H) = (I - \eta H)H \tag{13}$$

Choice of η

For convergence, we want to control the maximum eigenvalue of $(I - \eta H)$. Let m, M the min, max singular values of H.

$$\text{minimize}_\eta \; max_{\lambda \in [m,M]} |1 - \eta\lambda| \tag{14}$$

We obtain $\frac{1}{\eta^*} = \frac{M+m}{2}$ or

$$\eta^* = \frac{2}{M+m} \tag{15}$$

For this η^* we obtain

$$\beta^* \equiv \sigma_{max}(I - \eta H) \; = \; \frac{M-m}{M+m} \tag{16}$$

This value is always in $[0, 1]$. Denote by $\kappa = \frac{M}{m}$ the **condition number** of H; β^* approaches 1 when κ is large.

Newton-Raphson "rounds" the surface of f around minimum

- If we take $H = I$, then $\beta = 0$, meaning that the first order convergence is infinitely fast (super-linear convergence).
- How can we make $H = I$? We transform the variable x by

$$x = H^{-1/2}z, \quad z = H^{1/2}x \tag{17}$$

Then $f(z) = \frac{1}{2}\|z\|^2 + g^T H^{-1/2} z + c$ and the new Hessian is I.
Let us look at the gradient descent in z.

$$\begin{aligned}
\nabla_z f(z) &= z + (H^{-1/2})^T g && (18)\\
z^{t+1} &= z^t - \eta(z^t + (H^{-1/2})^T g) && (19)\\
x^{t+1} &= H^{-1/2} z^{t+1} = (1-\eta) H^{-1/2} z^t - \eta H^{-1} g && (20)\\
&= (1-\eta)x^t - \eta \underbrace{\nabla_x^2 f(x^t) \nabla_x f(x^t)}_{\text{Newtonstep}} && (21)
\end{aligned}$$

- Hence the Newton step is a gradient step in the transformed coordinates z.

For a symmetric $A \succ 0$, $B = A^{1/2}$ is a matrix for which $B^T B = A$ holds; $A^{1/2}$ is not unique. We have also $A^{-1} = (B^T B)^{-1} = B^{-1}(B^T)^{-1}$. Exercise Prove that B is non-singular when A is non-singular; find the equivalence class of all B which are the square root of some A.

Gradient descent for Least Squares Loss

Consider linear regression, with ~~$f(\theta)$~~ $\equiv L_{LS}(\theta) = \frac{1}{2n}\|y - X\theta\|^2$ with $d > n$. Let $XX^T \in \mathbb{R}^{n\times n}$ be the **kernel matrix** and $H = \frac{1}{n}X^TX$ the **covariance** matrix.

GD on L_{LS} →

$$L_f(\theta) = \frac{1}{2}\theta^T H\theta - \underbrace{\frac{1}{n}y^T X\theta}_{g} + \frac{1}{2n}y^T y \quad (22)$$

quadratic

$$X = \begin{bmatrix} — x^i — \end{bmatrix} \qquad y = \begin{bmatrix} y^i \end{bmatrix}$$

- We start from $\theta^0 = 0$.
- We don't assume the solution is unique. In other words, H may be singular.
- In particular, note that for $d > n$, H is singular, but K is invertible w.l.o.g. when the system $X\theta = y$ has a solution (and the system has an infinite number of solutions).
- For any θ^* satisfying $y = X\theta^*$ and for some iterate θ^t we have

$$\theta^t - \theta^* = (I - \eta H)^t(\theta^0 - \theta^*) \quad (23)$$

$$\theta^t = [I - (I - \eta H)^t]\theta^* \quad (24)$$

nonsingular

Pb: Linear regression

$\hat{y} = \theta^T x$

$L = L_{LS}$

$G \equiv K = XX^T$ nxn (Kernel) Gram matrix

singular

$H = \frac{1}{n}X^TX$ dxd Covariance

assume $\sum x^i = 0$ ↗

The GD path

min L by GD

$\nabla L = H\theta + \frac{1}{n} X^T y$

- Now on the GD path (which is deterministic given X)

$\theta = 0 \implies \nabla f(0) = g = \frac{1}{n} X^T y$ (25)

$\theta^1 = 0 - \eta \nabla f(0) = -\eta \frac{1}{n} X^T y$ (26) ← linear combination of $X^{1:n}$

Thus θ^1 is a linear combination of the rows of X (i.e. of the data points).

- By induction, θ^t for any t is a linear combination of the rows of X, hence

$\theta^t = X^T \alpha^t$, with $\alpha^t \in \mathbb{R}^n$ (27)

α_i = coef of X^i

- Since the gradient is non-zero whenever $y \neq X\theta$, the GD algorithm converges to a point[1] where $y = X\theta = XX^T\alpha$.
- When K is invertible, let $\alpha^* = K^{-1}y$; then $\theta^* = X^T\alpha^*$ is the limit of GD.

$\theta^* = X^T\alpha^*$ at convergence

$X\theta^* = y = XX^T\alpha^* = K\alpha^* \implies \alpha^* = K^{-1}y$

$\theta^* = XK^{-1}y$

solution of GD

[1] This is informal. What we can say that when t is sufficiently large, $X\theta^t = XX^T\alpha^t$ is arbitrarily close to y.

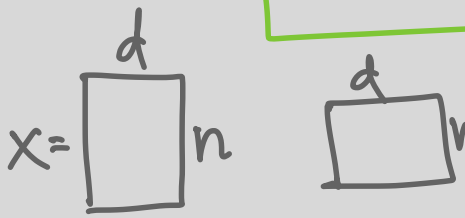
θ^* is the minimum norm solution of $X\theta = y$

- To prove this, we must use convex duality.

$$\text{Primal: } \inf_{\theta} \frac{1}{2}\|\theta\|^2 \text{ s.t. } X\theta = y \quad \Leftrightarrow \quad \text{Dual: } \sup_{\alpha} \inf_{\theta} \frac{1}{2}\|\theta\|^2 + \alpha^T(y - X\theta) \tag{28}$$

- Solving the optimization over θ as a function of the parameter α we obtain $\theta = X^T\alpha$.
- We replace θ in (28) to obtain

$$\sup_{\alpha} \alpha^T y - \frac{1}{2}\alpha^T K \alpha \tag{29}$$



This is a concave function with optimum $\alpha^* = K^{-1}y$ **Yes, we get the same α^* from the previous page!**

- Finally, the solution to the Primal problem is $\theta^* = X^T\alpha^* = X^TK^{-1}y$, the solution obtained by Gradient Descent!

Note that θ^* above is not the OLS solution. In OLS, we minimize residuals norm, here we minimize the θ norm.

Lecture VII – Wide multilayer networks and the Neural Tangent Kernel (NTK)

Marina Meilă
mmp@stat.washington.edu

Department of Statistics
University of Washington

October, 2022

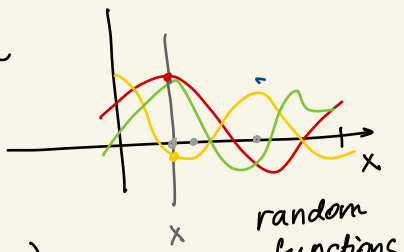
Summary

- DD explained by $\min \|f\|_{\mathcal{H}}$ s.t $f(x^i) = y^i$ (interpolation)
  - $\|f\|_{\mathcal{H}}$: smoothness
  - Proved "universally"
    [also for $\lambda \to$ Ridge regression]
  - GD finds "$\|f\|_{\mathcal{H}}$"

Gaussian Process

$X \ni x$ distribution over functions on X

Prior $GP(0, k)$ Posterior $GP(\mu_{|\mathcal{D}}, k_{|\mathcal{D}})$



- $f(x) \sim N(0, k(x,x))$ for all x
  - ↑ or known
- any set $x^{1:n}$, $f(x^{1:n}) \sim N(0, G_x)$
  - $x^{1:n}$

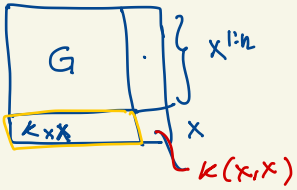
define $k(x, x') = \mathrm{Cov}(f(x), f(x')) \equiv E[f(x)f(x')]$

↑ Mercer kernel

Assume $k(x, x') = K(\|x - x'\|)$ e.g. Gaussian kernel

- Pb
  - Know k
  - Observe $(x^{1:n}, y^{1:n} = f(x^{1:n})) = \mathcal{D}$
  - Bayesian: Prior $f \sim GP(0, k)$
    - Posterior $f \mid \mathcal{D} = ?$

$\Sigma \equiv k_{xx,xx} =$



Same problem:

→ some $x \in \mathcal{X}$: wanted $f(x|\mathcal{D}) \sim N(\mu, \sigma^2)$ posterior

$\mu_{x|\mathcal{D}}$ $\sigma^2_{x|\mathcal{D}}$

Remark $\begin{bmatrix} y \\ y^{1:n} \end{bmatrix} \sim N(0_{n+1}; \Sigma)$ jointly Gaussian → want conditional $y \mid y^{1:n}, x^{1:n} \sim N(\mu_{x|\mathcal{D}}, \sigma^2_{x|\mathcal{D}})$

$$\mu_{x|\mathcal{D}} = \underbrace{[k(x,x') \cdots]}_{k(x,X)} G^{-1} \begin{bmatrix} y^{1:n} \end{bmatrix}$$

$k(x,X)$: $1 \times n$; G^{-1}: $n \times n$; $y^{1:n}$: $n \times 1$

$$\sigma^2_{x|\mathcal{D}} = k(x,x) - k(x,X) G^{-1} k(X,x)$$

$k(x,x)$: 1×1; $k(x,X)$: $1 \times n$; G^{-1}: $n \times n$; $k(X,x)$: $n \times 1$

Notation

- Neural network predictor $f(x;\theta)$, where $x \in \mathbb{R}^d$
- For each layer $l = 1 : L$ of dimension m_l, with $x^0 \equiv x$, and $z^L \equiv f(x)$

$$z^{l+1} = W^{l+1}x^l + b^{l+1} \qquad x^{l+1} = \phi(z^{l+1}) \tag{1}$$

  Here $x^{l,l+1}, z^{l+1}, b^{l+1}$ are column vectors W^{l+1} is a $m_{l+1} \times m_l$ matrix, $\phi()$ is the non-linearity/activation function.
- The weights

$$W^l_{ij} = \sigma_w w^l_{ij}/\sqrt{m_l}, \qquad b^l_j = \sigma_b \beta^l_j, \quad \text{Known as NTK parametrization} \tag{2}$$

- Parameter vector $\theta = \text{vector}\{w^{1:L}, \beta^{1:L}\} \in \mathbb{R}^p$ initialized i.i.d. $\sim N(0,1)$
- $\sigma_{w,b}$ are fixed hyper-parameters, $1/\sqrt{m_l}$ normalizes the expected norm of W^l columns
- Loss $\mathcal{L}(y, f)$

- We want to analize the behavior of this network $f()$ at initialization and during training, when $m_{1:L}$ very large
- Three approximations help analysis
  - (A1) continuous time training, called **gradient flow**
  - (A2) $m_{1:L} \to \infty$ in the wide limit, we can apply the Central Limit Theorem (CLT), and Gaussian Processes
  - (A3) parameters θ do not change much during training, i.e. $\theta_t - \theta_0$ is small