Learning curve in modern ML
The "double descent" behavior

Oren Yuval

Department of Statistics and Operations Research

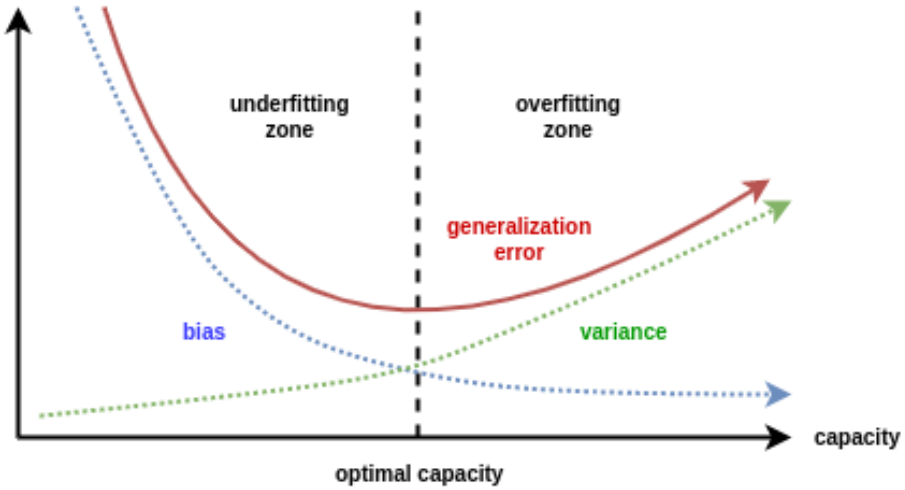
Tel-Aviv University

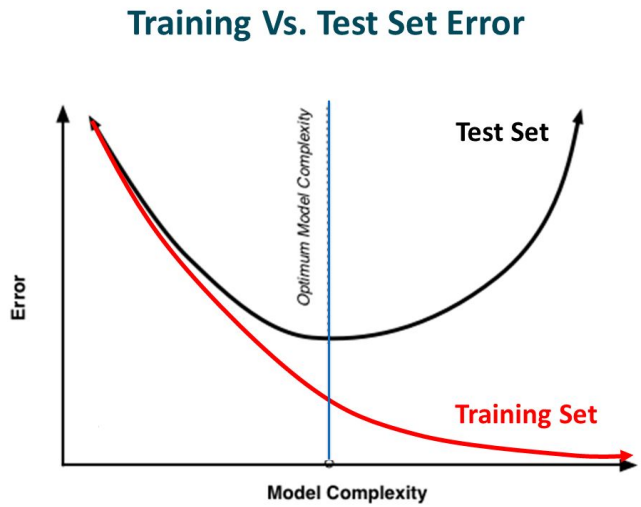
December 23, 2019

Outline

The classical learning curve

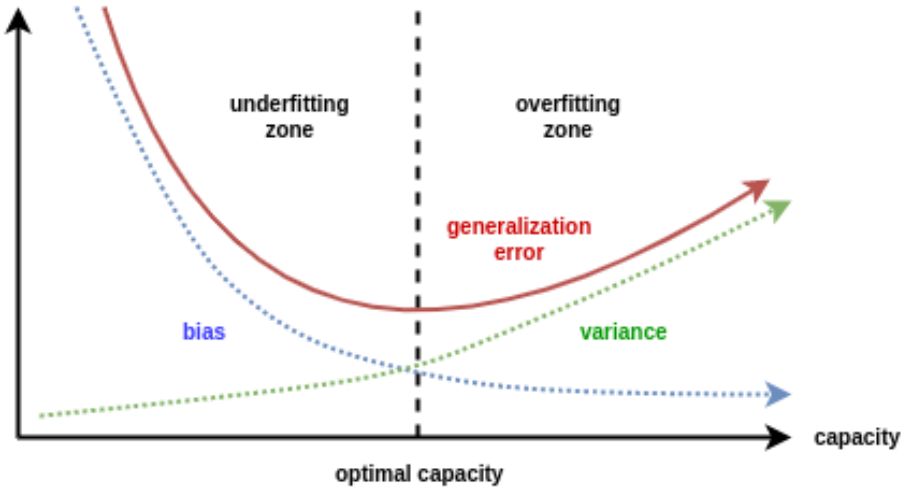
We all know the Bias-Variance Trade-Off:

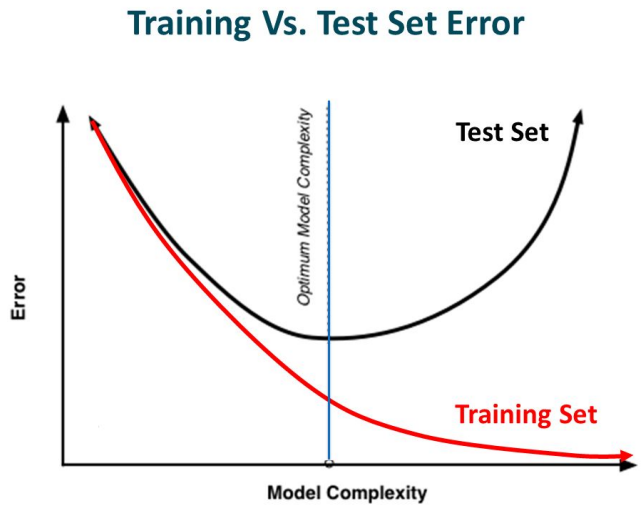




The classical learning curve

We all know the Bias-Variance Trade-Off:

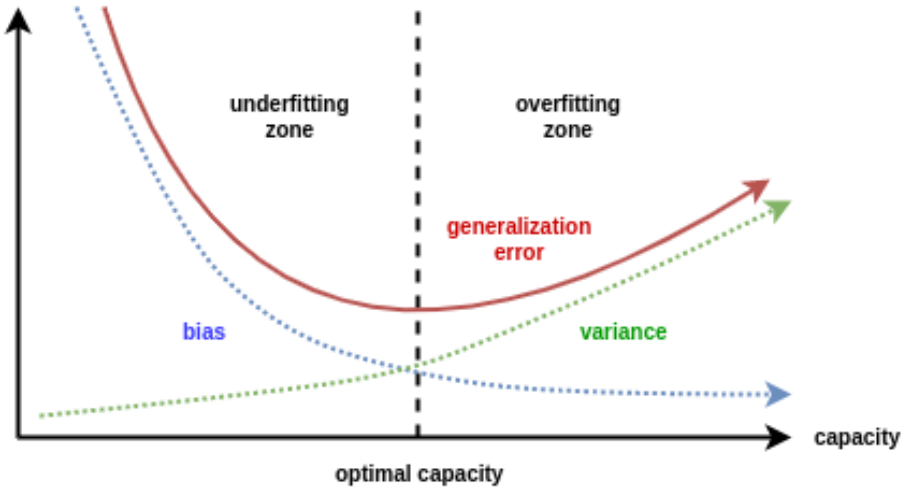


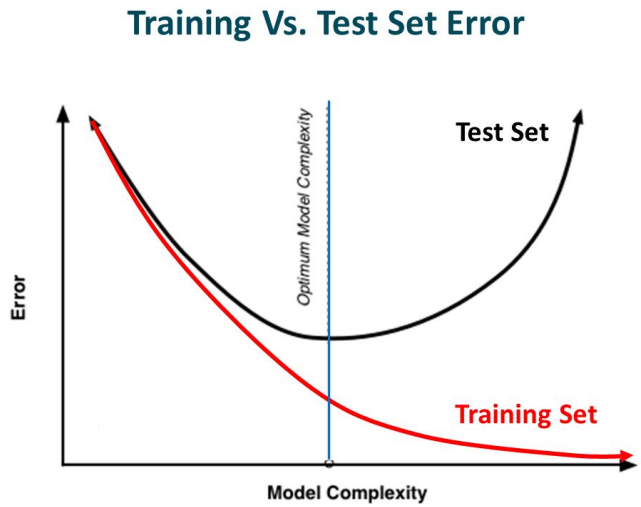


The common knowledge - very low training error $\rightarrow$ very high variance

The classical learning curve

We all know the Bias-Variance Trade-Off:



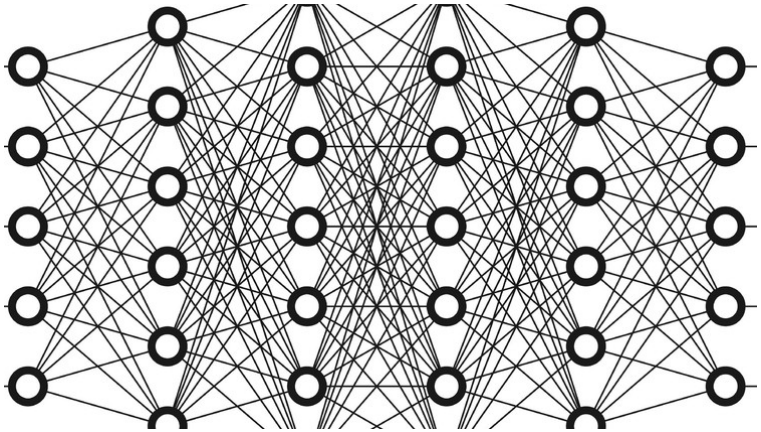


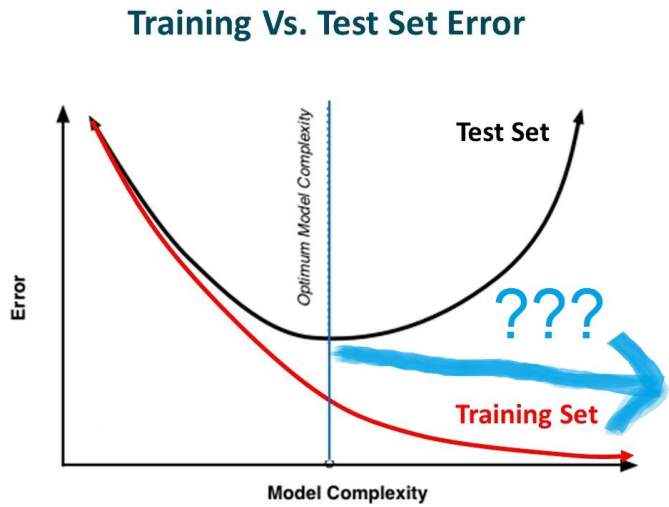
The common knowledge - very low training error $\rightarrow$ very high variance

One may think of some criteria for finding the optimal model

Is NN immune to variance?

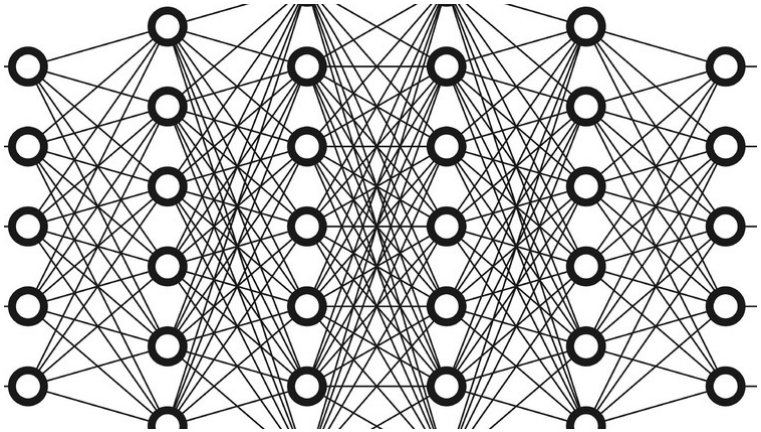
On the other hand - very rich models such as NN are trained to exactly fit the train data, and yet they obtain high accuracy on test data

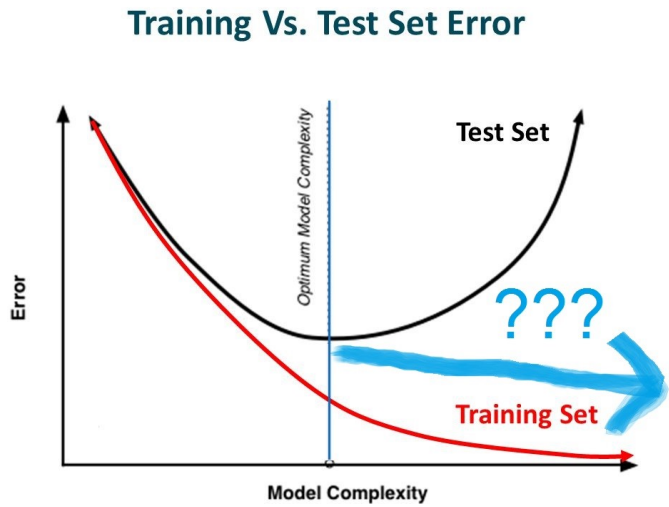




Is NN immune to variance?

On the other hand - very rich models such as NN are trained to exactly fit the train data, and yet they obtain high accuracy on test data

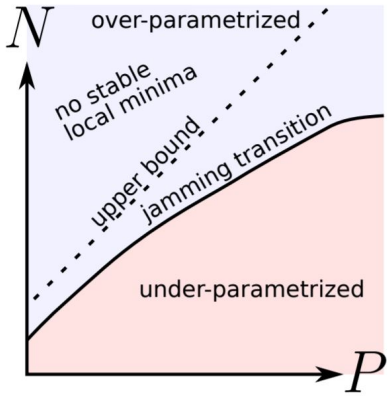




How can we reconcile the modern practice with the classical bias-variance trade-off?

A "jamming transition"

Spigler in mid 2018, argued that in fully-connected networks, a phase transition delimits the over and under-parametrized regimes where fitting can or cannot be achieved



N: degrees of freedom, P: training examples.

22 Oct 2018

A jamming transition from under- to over-parametrization affects loss landscape and generalization

Stefano Spigler [1] stefano.spigler@epfl.ch
Mario Geiger [1] mario.geiger@epfl.ch
Stéphane d'Ascoli [2] stephane.dascoli@ens.fr
Levent Sagun [1] levent.sagun@epfl.ch
Giulio Biroli [2] giulio.biroli@ens.fr
Matthieu Wyart [1] matthieu.wyart@epfl.ch

[1] Institute of Physics, École Polytechnique Fédérale de Lausanne
1015 Lausanne, Switzerland

Reconciling the discrepancy

Just one year ago - Belkin et al first analyzed the behavior of rich models around the interpolation point

Reconciling modern machine learning practice and the bias-variance trade-off

Mikhail Belkin[a], Daniel Hsu[b], Siyuan Ma[a], and Soumik Mandal[a]

[a]*The Ohio State University, Columbus, OH*
[b]*Columbia University, New York, NY*

September 12, 2019

Abstract

Breakthroughs in machine learning are rapidly changing science and society, yet our fun-

Surprises in High-Dimensional Ridgeless Least Squares Interpolation

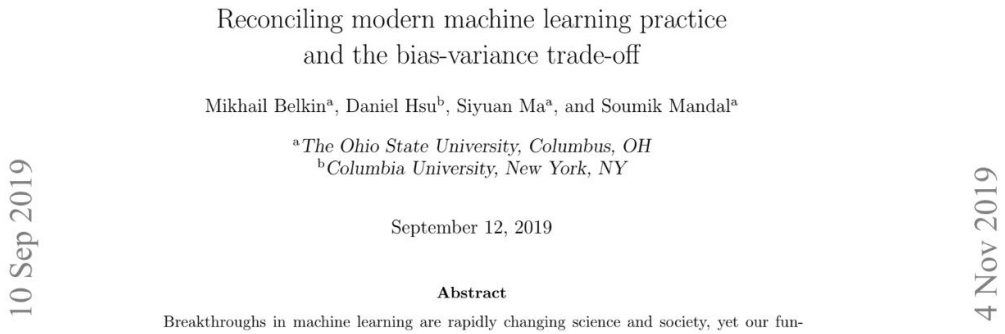
Trevor Hastie Andrea Montanari* Saharon Rosset Ryan J. Tibshirani*

Abstract

Interpolators—estimators that achieve zero training error—have attracted growing attention in machine learning, mainly because state-of-the art neural networks appear to be models of this type. In this paper, we study minimum ℓ_2 norm ("ridgeless") interpolation in high-dimensional least squares regression. We consider two different models for the feature distribution: a linear model, where the feature vectors $x_i \in \mathbb{R}^p$ are obtained by applying a linear transform to a vector of i.i.d. entries, $x_i = \Sigma^{1/2} z_i$ (with $z_i \in \mathbb{R}^p$); and a nonlinear model, where the feature vectors are obtained by passing the input through a random one-layer neural network, $x_i = \varphi(W z_i)$ (with $z_i \in \mathbb{R}^d$, $W \in \mathbb{R}^{p \times d}$ a matrix of i.i.d. entries, and φ an activation function acting componentwise on $W z_i$). We recover—in a precise quantitative

Reconciling the discrepancy

Just one year ago - Belkin et al first analyzed the behavior of rich models around the interpolation point



10 Sep 2019

Reconciling modern machine learning practice and the bias-variance trade-off

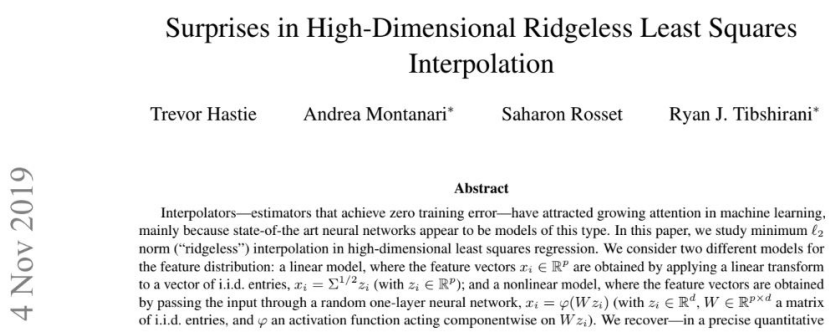
Mikhail Belkin[a], Daniel Hsu[b], Siyuan Ma[a], and Soumik Mandal[a]

[a]*The Ohio State University, Columbus, OH*
[b]*Columbia University, New York, NY*

September 12, 2019

Abstract

Breakthroughs in machine learning are rapidly changing science and society, yet our fun-



4 Nov 2019

Surprises in High-Dimensional Ridgeless Least Squares Interpolation

Trevor Hastie Andrea Montanari* Saharon Rosset Ryan J. Tibshirani*

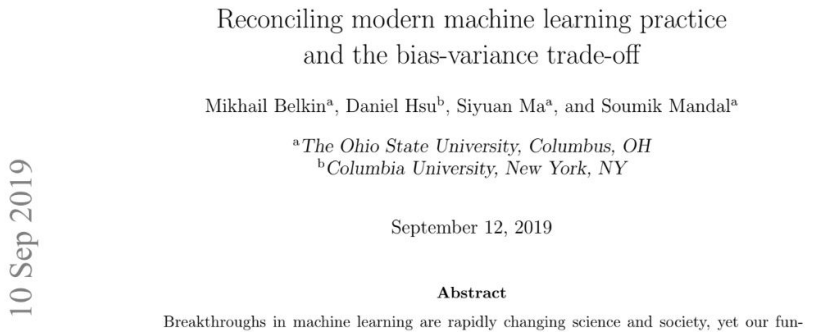
Abstract

Interpolators—estimators that achieve zero training error—have attracted growing attention in machine learning, mainly because state-of-the art neural networks appear to be models of this type. In this paper, we study minimum ℓ_2 norm ("ridgeless") interpolation in high-dimensional least squares regression. We consider two different models for the feature distribution: a linear model, where the feature vectors $x_i \in \mathbb{R}^p$ are obtained by applying a linear transform to a vector of i.i.d. entries, $x_i = \Sigma^{1/2} z_i$ (with $z_i \in \mathbb{R}^p$); and a nonlinear model, where the feature vectors are obtained by passing the input through a random one-layer neural network, $x_i = \varphi(W z_i)$ (with $z_i \in \mathbb{R}^d$, $W \in \mathbb{R}^{p \times d}$ a matrix of i.i.d. entries, and φ an activation function acting componentwise on $W z_i$). We recover—in a precise quantitative

Since then many authors published results, that justified the innovative approach

Reconciling the discrepancy

Just one year ago - Belkin et al first analyzed the behavior of rich models around the interpolation point



10 Sep 2019

Reconciling modern machine learning practice and the bias-variance trade-off

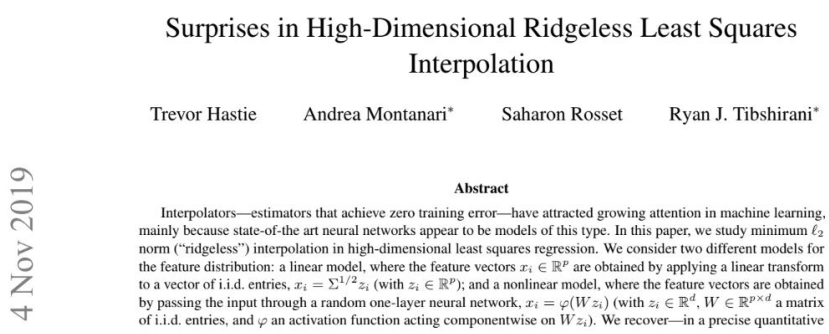
Mikhail Belkin[a], Daniel Hsu[b], Siyuan Ma[a], and Soumik Mandal[a]

[a]*The Ohio State University, Columbus, OH*
[b]*Columbia University, New York, NY*

September 12, 2019

Abstract

Breakthroughs in machine learning are rapidly changing science and society, yet our fun-



4 Nov 2019

Surprises in High-Dimensional Ridgeless Least Squares Interpolation

Trevor Hastie Andrea Montanari* Saharon Rosset Ryan J. Tibshirani*

Abstract

Interpolators—estimators that achieve zero training error—have attracted growing attention in machine learning, mainly because state-of-the art neural networks appear to be models of this type. In this paper, we study minimum ℓ_2 norm ("ridgeless") interpolation in high-dimensional least squares regression. We consider two different models for the feature distribution: a linear model, where the feature vectors $x_i \in \mathbb{R}^p$ are obtained by applying a linear transform to a vector of i.i.d. entries, $x_i = \Sigma^{1/2} z_i$ (with $z_i \in \mathbb{R}^p$); and a nonlinear model, where the feature vectors are obtained by passing the input through a random one-layer neural network, $x_i = \varphi(W z_i)$ (with $z_i \in \mathbb{R}^d$, $W \in \mathbb{R}^{p \times d}$ a matrix of i.i.d. entries, and φ an activation function acting componentwise on $W z_i$). We recover—in a precise quantitative

Since then many authors published results, that justified the innovative approach

Hastie, Rosset, and Tibshirani provided a precise quantitative explanation for the potential benefits of over-parametrization in linear regression

Empirical risk minimization

Given a training sample $(x_1, y_1), \cdots, (x_n, y_n)$, where $(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, we learn a predictor $h_n : \mathbb{R}^d \to \mathbb{R}$.

In ERM, the predictor is taken to be

$$h_n = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \left\{ \frac{1}{n} \sum_{i=1}^{n} l\left(y_i, h(x_i)\right) \right\}$$

Empirical risk minimization

Given a training sample $(x_1, y_1), \cdots, (x_n, y_n)$, where $(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, we learn a predictor $h_n : \mathbb{R}^d \to \mathbb{R}$.

In ERM, the predictor is taken to be

$$h_n = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \left\{ \frac{1}{n} \sum_{i=1}^{n} l\left(y_i, h(x_i)\right) \right\}$$

- Empirical risk (train error): $\frac{1}{n} \sum_{i=1}^{n} l\left(y_i, h_n(x_i)\right)$
  - *Interpolation*: $l\left(y_i, h_n(x_i)\right) = 0 \quad \forall i$
- True risk (test error): $\mathbb{E}_{x,y}\left[l\left(y_i, h_n(x_i)\right)\right]$

Controlling $\mathcal{H}$

Conventional wisdom in machine learning suggests controlling the capacity of $\mathcal{H}$:

- $\mathcal{H}$ too small $\rightarrow$ under-fitting (large empirical and true risk)
- $\mathcal{H}$ too large $\rightarrow$ over-fitting (small empirical risk but large true risk)

Controlling $\mathcal{H}$

Conventional wisdom in machine learning suggests controlling the capacity of $\mathcal{H}$:

- $\mathcal{H}$ too small $\rightarrow$ under-fitting (large empirical and true risk)
- $\mathcal{H}$ too large $\rightarrow$ over-fitting (small empirical risk but large true risk)

Example (OLS)

$$\hat{\beta}_n = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(y_i - x_i^T \beta \right)^2 \right\}$$

$$\text{True risk} \propto \frac{p}{n-p}$$

Controlling $\mathcal{H}$

Conventional wisdom in machine learning suggests controlling the capacity of $\mathcal{H}$:

- $\mathcal{H}$ too small $\rightarrow$ under-fitting (large empirical and true risk)
- $\mathcal{H}$ too large $\rightarrow$ over-fitting (small empirical risk but large true risk)

Example (OLS)

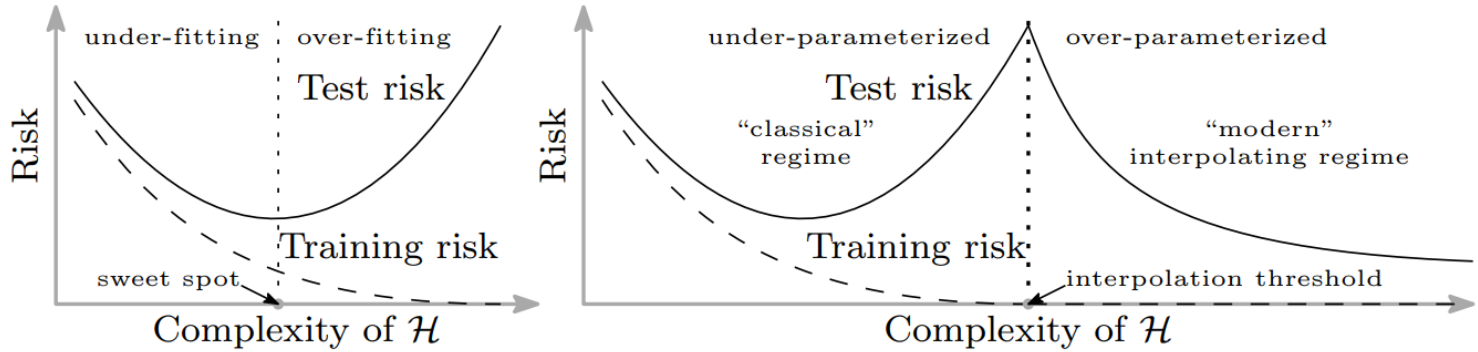
$$\hat{\beta}_n = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(y_i - x_i^T \beta \right)^2 \right\}$$

$$\text{True risk} \propto \frac{p}{n-p}$$

Yet, best practice in DL: network should be large enough to permit effortless zero train-loss

The "double descent" risk curve

The main finding of Belkin's work is summarized in the "double descent" risk curve:

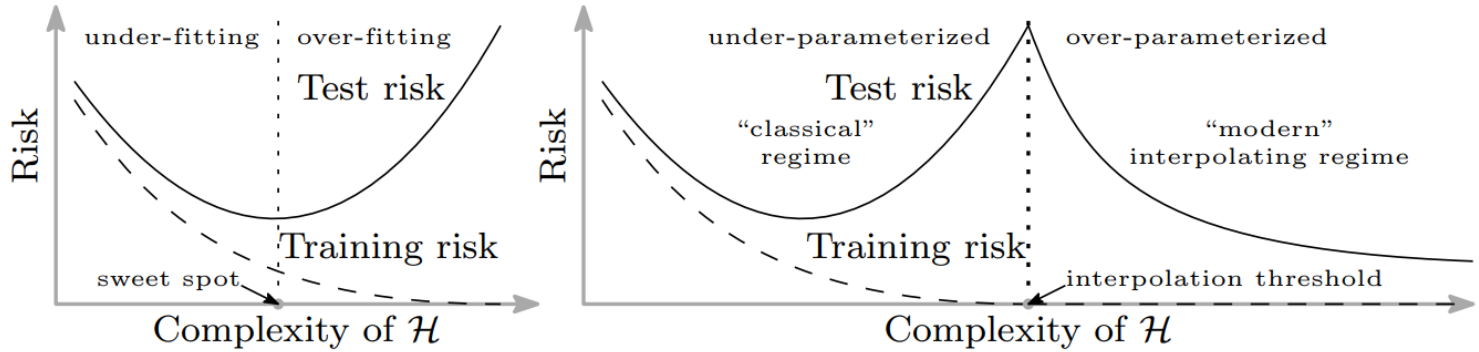


(a) U-shaped "bias-variance" risk curve

(b) "double descent" risk curve

The "double descent" risk curve

The main finding of Belkin's work is summarized in the "double descent" risk curve:



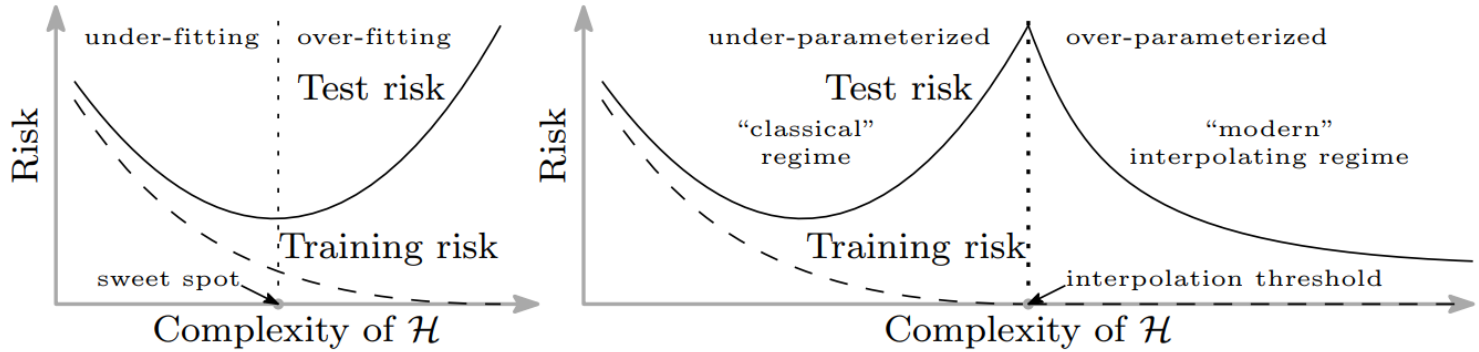
(a) U-shaped "bias-variance" risk curve

(b) "double descent" risk curve

This is demonstrated on important model classes including neural networks and a range of real data sets

The "double descent" risk curve

The main finding of Belkin's work is summarized in the "double descent" risk curve:



(a) U-shaped "bias-variance" risk curve

(b) "double descent" risk curve

This is demonstrated on important model classes including neural networks and a range of real data sets

The capacity of $\mathcal{H}$ is identified with the number of parameters needed to specify the function h_n

Fitting beyond the interpolation point

When zero train error can be achieved, we choose h_n as follows:

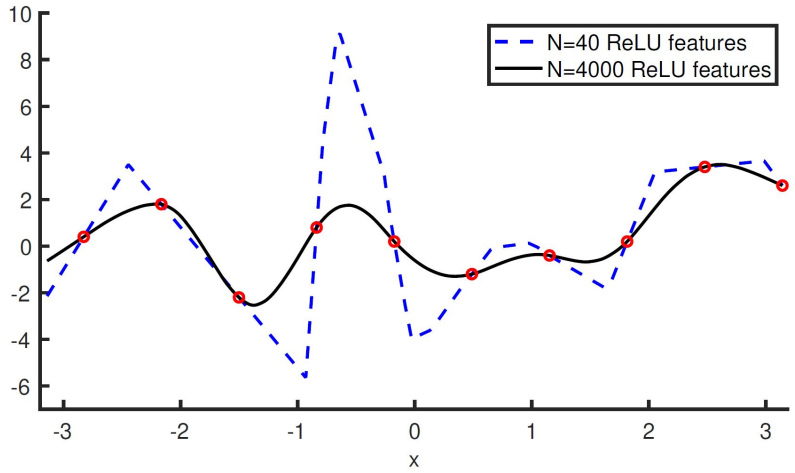
$$h_n = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \left\{ ||h_n|| \;\; s.t : \frac{1}{n} \sum_{i=1}^{n} l\left(y_i, h(x_i)\right) = 0 \right\}$$

Fitting beyond the interpolation point

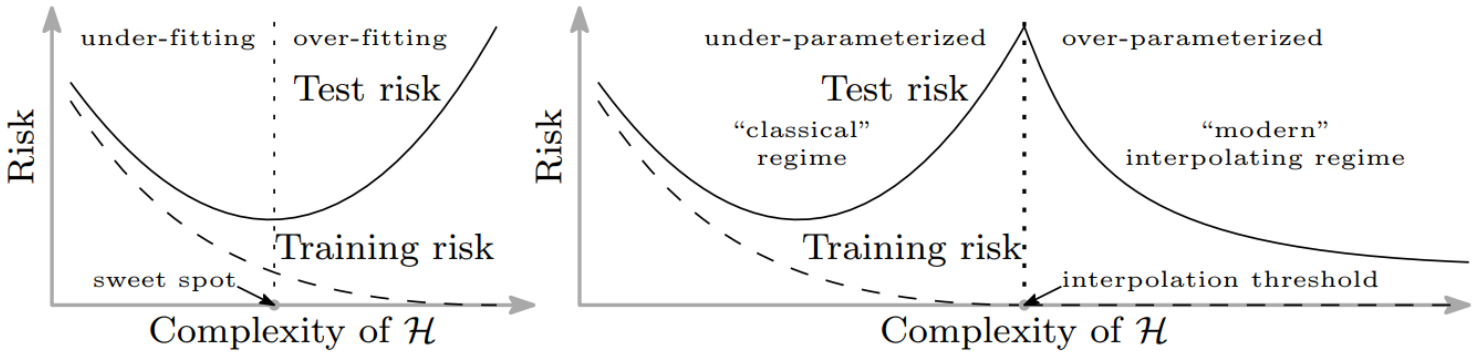
When zero train error can be achieved, we choose h_n as follows:

$$h_n = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \left\{ ||h_n|| \;\; s.t : \frac{1}{n} \sum_{i=1}^{n} l\left(y_i, h(x_i)\right) = 0 \right\}$$

- Looking for the simplest/smoothest function that explain the data
- By increasing the capacity of $\mathcal{H}$, we are able to find interpolating functions that have smaller norm



Why is the "double descent" important?

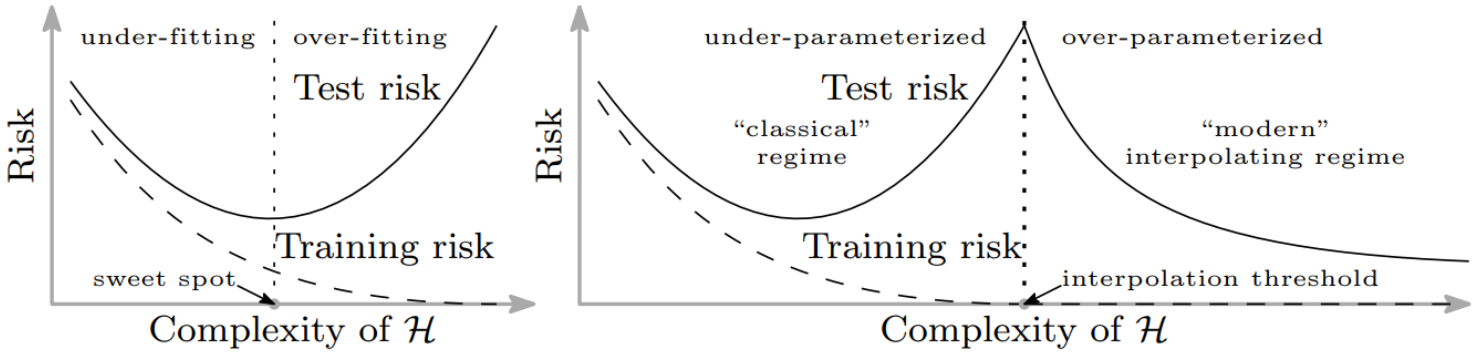


(a) U-shaped "bias-variance" risk curve

(b) "double descent" risk curve

Why is the "double descent" important?

- Stating that interpolation does not necessarily lead to poor generalization, as long as you "deep" enough in the interpolation regime

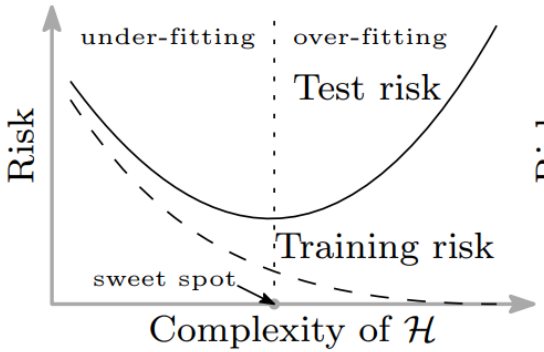


(a) U-shaped "bias-variance" risk curve

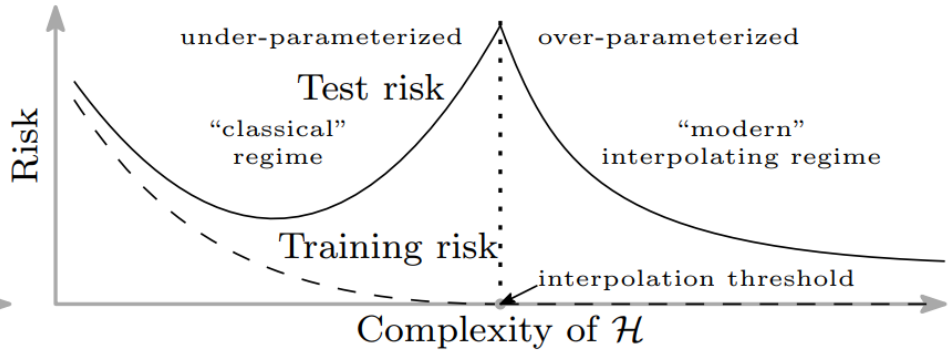
(b) "double descent" risk curve

Why is the "double descent" important?

- Stating that interpolation does not necessarily lead to poor generalization, as long as you "deep" enough in the interpolation regime
- Reconciling the modern practice with a statistical point-of-view



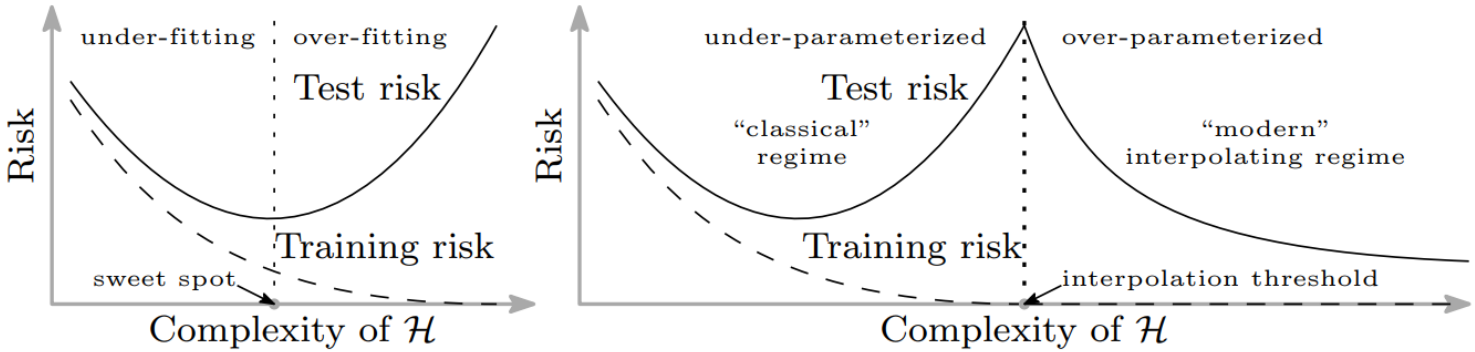
(a) U-shaped "bias-variance" risk curve



(b) "double descent" risk curve

Why is the "double descent" important?

- Stating that interpolation does not necessarily lead to poor generalization, as long as you "deep" enough in the interpolation regime
- Reconciling the modern practice with a statistical point-of-view
- Explicit analysis for Linear Models

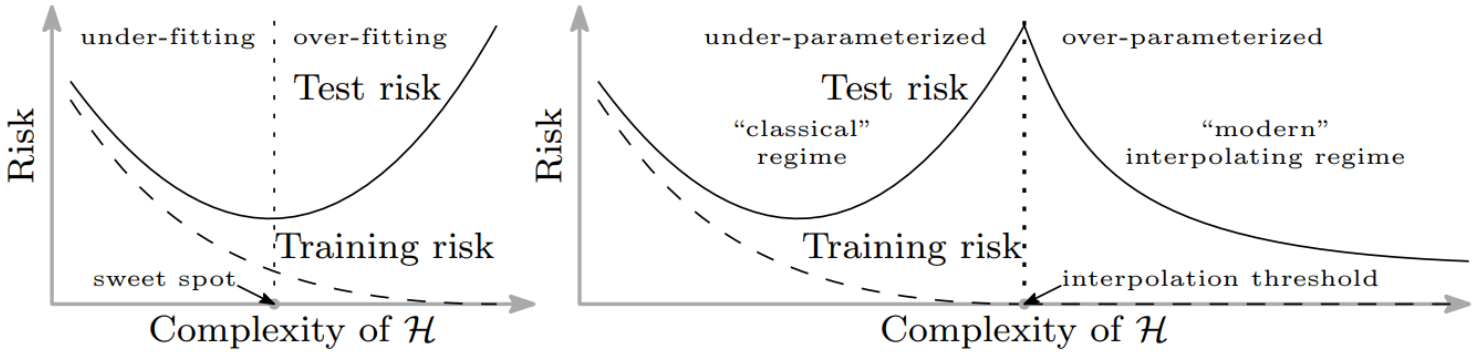


(a) U-shaped "bias-variance" risk curve

(b) "double descent" risk curve

Why is the "double descent" important?

- Stating that interpolation does not necessarily lead to poor generalization, as long as you "deep" enough in the interpolation regime
- Reconciling the modern practice with a statistical point-of-view
- Explicit analysis for Linear Models
- The true risk in the over-parameterized regime is typically lower!

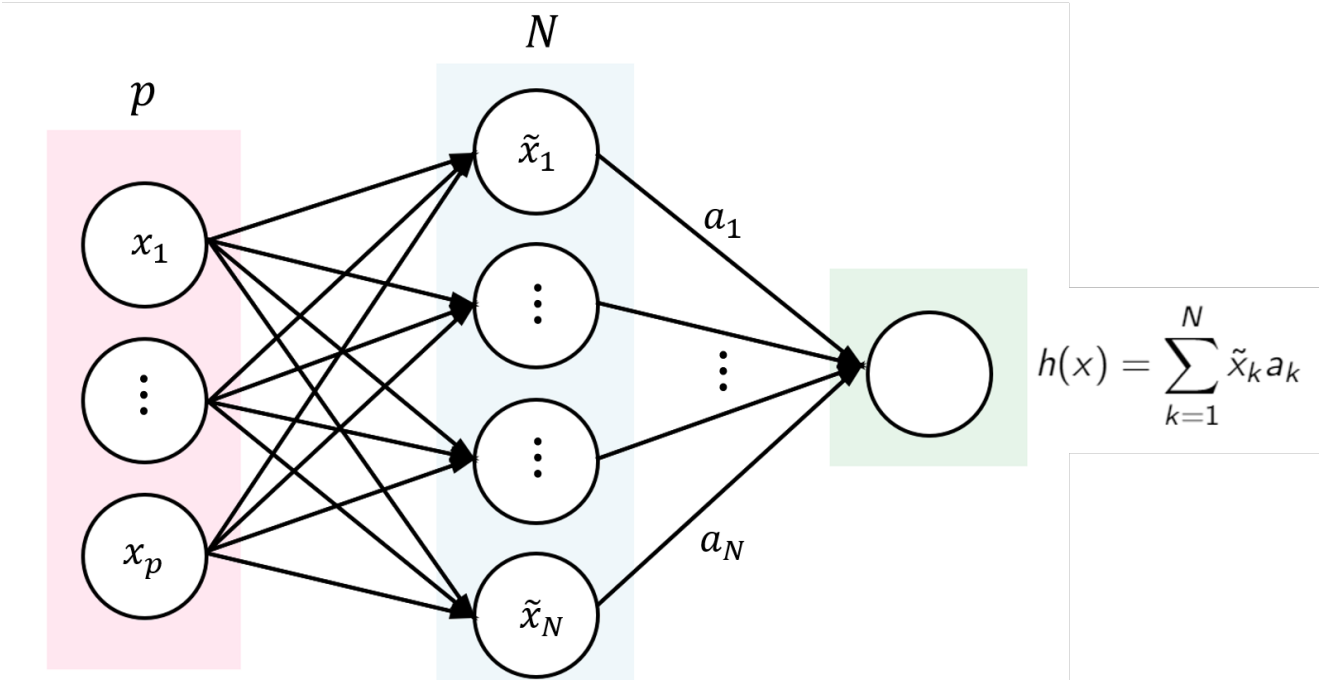


(a) U-shaped "bias-variance" risk curve

(b) "double descent" risk curve

Neural networks

- One Hidden layer with N random features
- Minimizing squared loss or $||a||_2$ when $N \geq n$

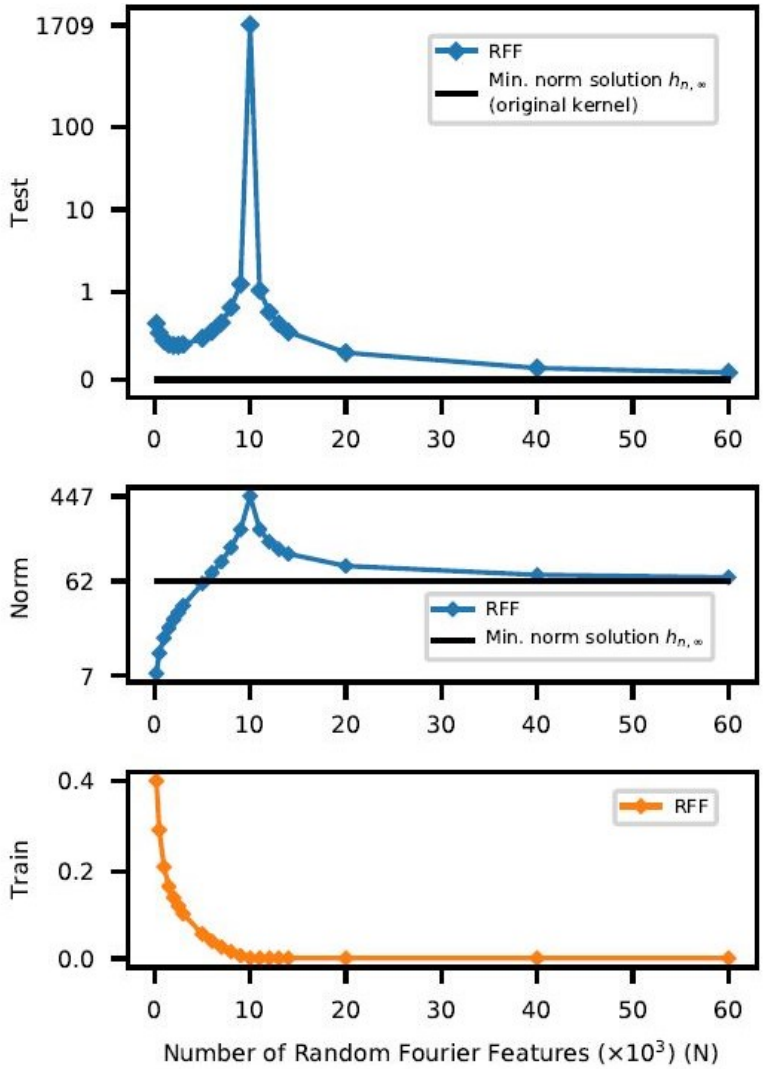


$$\tilde{x}_k = \varphi(x; v_k) = \varphi(< x, v_k >) \ , \quad v_k \sim MN(\mathbf{0}, I_p)$$

Neural networks

Risk curve for RFF model on MNIST

- Near interpolation - parameters are "forced" to fit the training data
- Increasing N results in decreasing the l_2 norm of the predictors



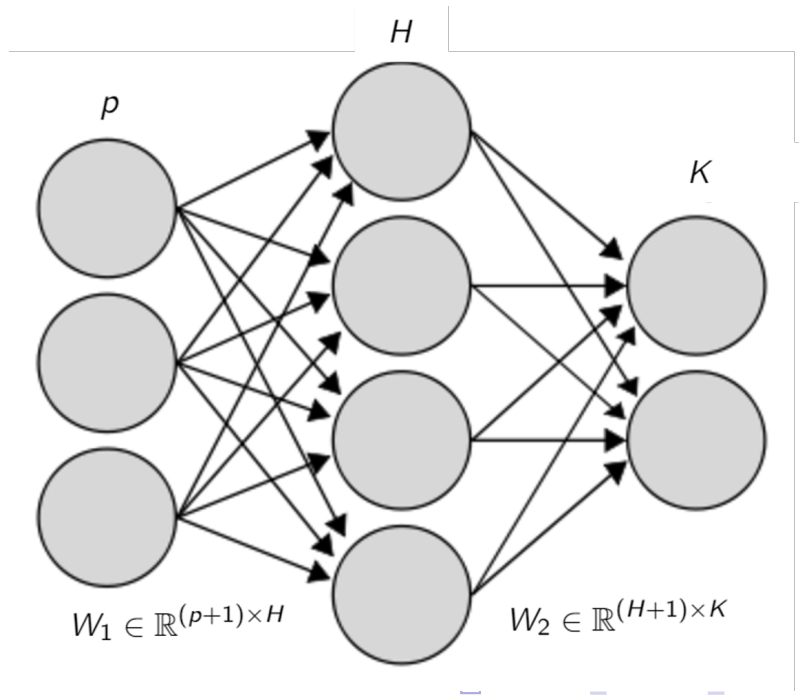
Neural networks

Fully connected two-layers network with H hidden units

Optimizing the weight using SGD with up to $6 \cdot 10^3$ iterations:

- Interpolation is not assured even in the over-parameterized regime
- Automatically prefers minimal-norm solution

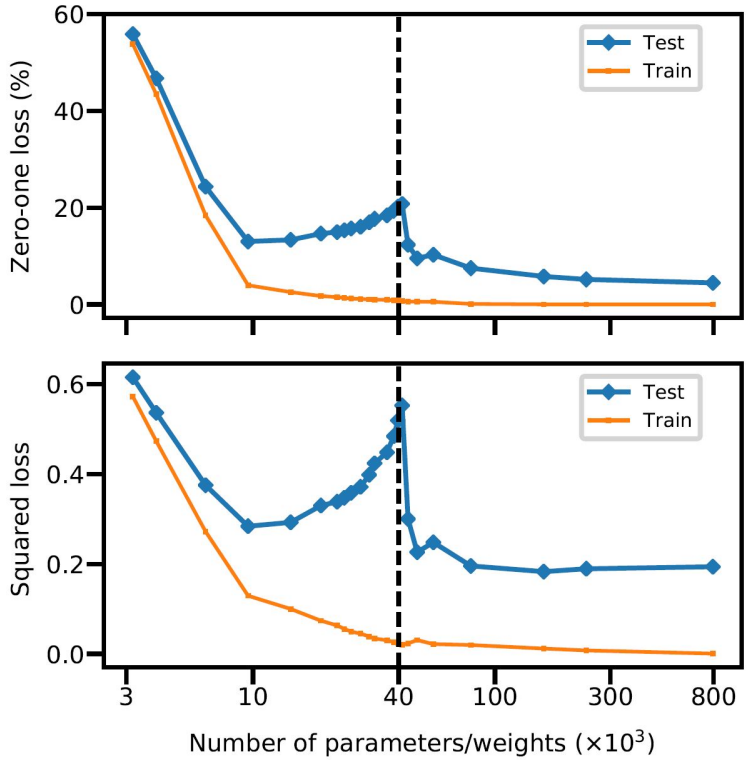
Sub-optimal behavior can lead to high variability in both the training and test risks that masks the double descent curve



Neural networks

Risk curves for two-Layers fully connected NN on MNIST

- Train risk may increase with increasing number of parameters

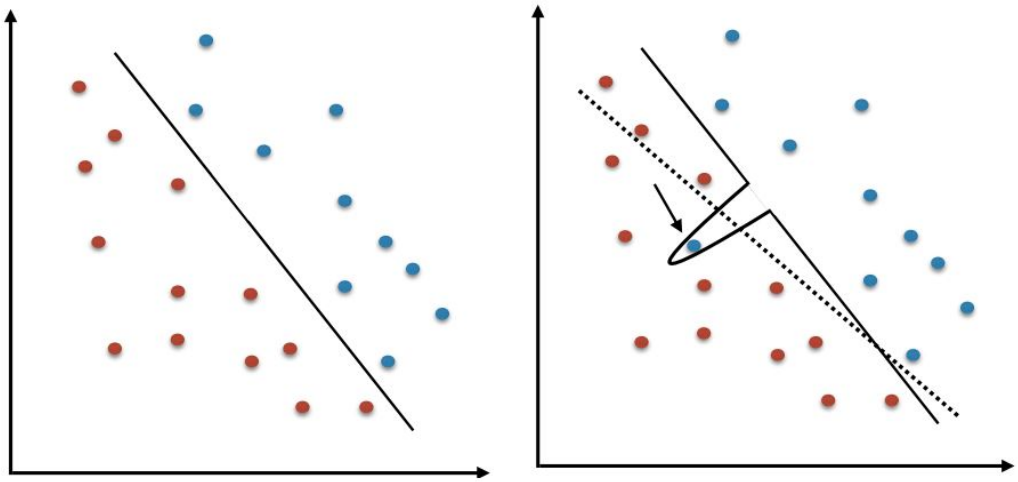


Decision trees

It was shown by Wyner et al (2017) that AdaBoost and Random-Forests perform better with large (interpolating) decision trees and are more robust to noise in the training data

They questioned the conventional wisdom that suggests that boosting algorithms for classification requires regularization/early stopping/low complexity

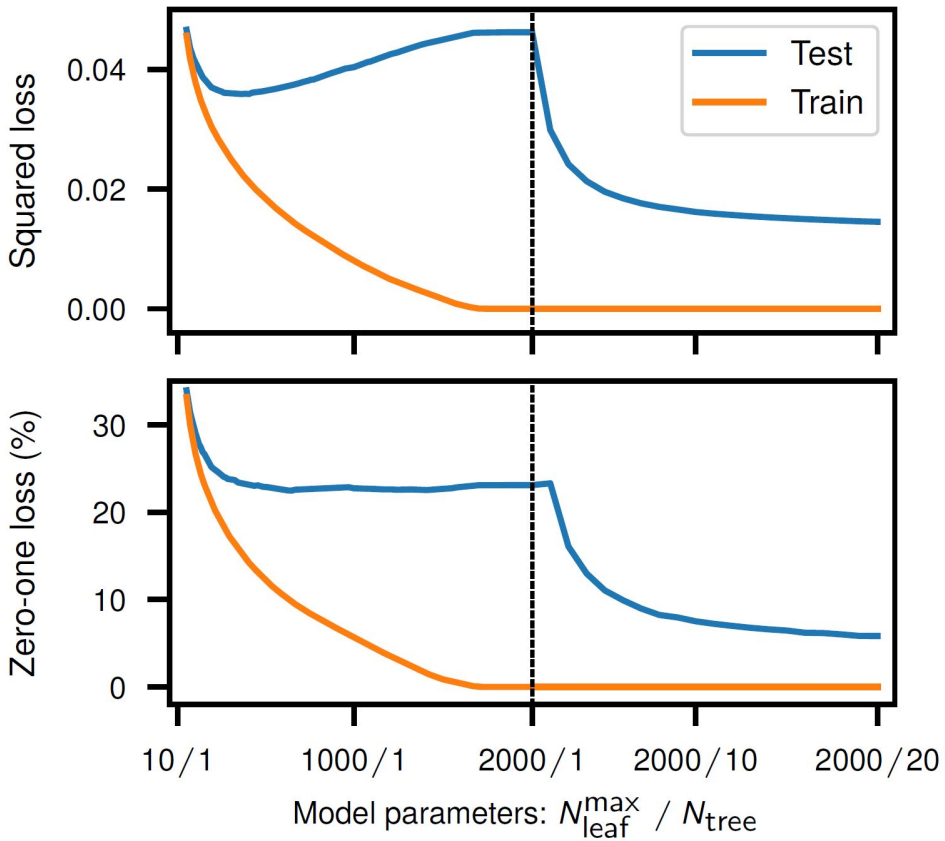
The effect of noise-point on a classifier: interpolating Vs non-interpolating



Decision trees

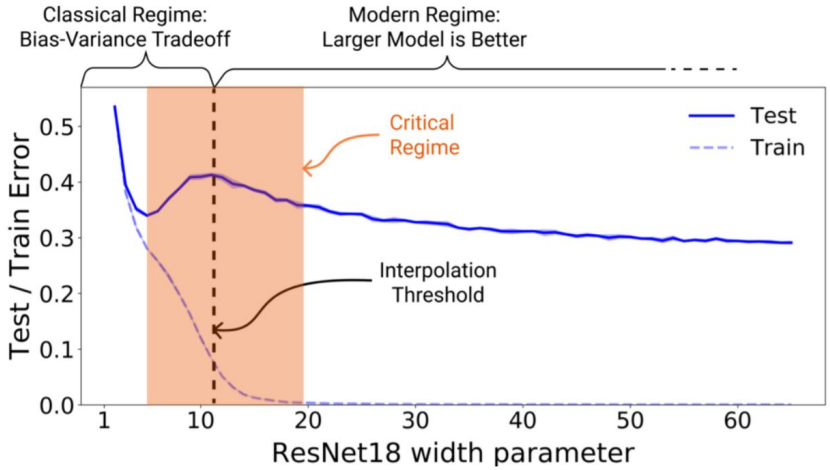
Risk curves for Random-Forests on MNIST

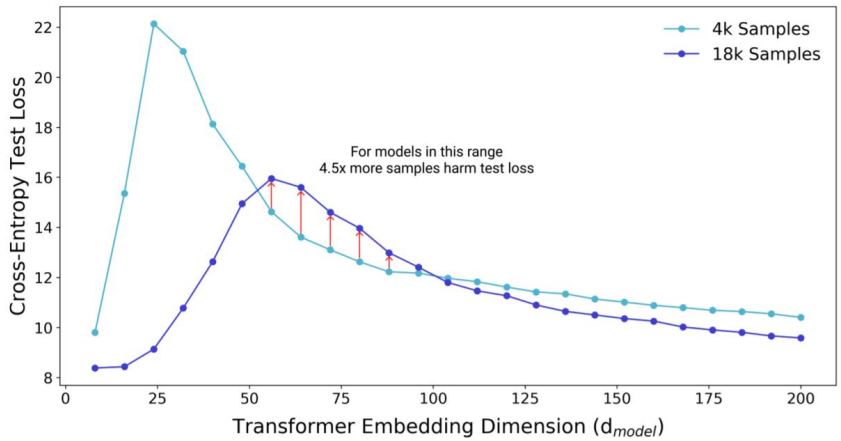
- The complexity is controlled by the size of a decision tree, and the number of trees
- Averaging of interpolating trees ensures substantially smoother function



Thinking about it...

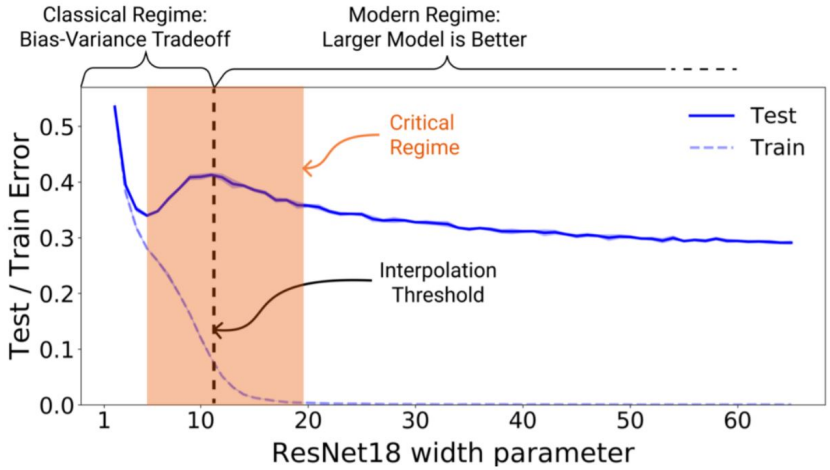
The peak at the interpolation threshold is observed within a narrow range of parameters - sampling parameter-space out of that range may lead to the misleading (but conventional) conclusions

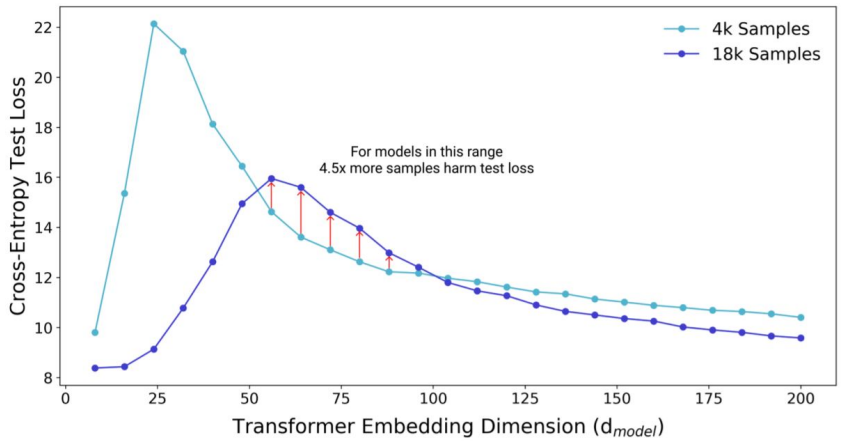




Thinking about it...

The peak at the interpolation threshold is observed within a narrow range of parameters - sampling parameter-space out of that range may lead to the misleading (but conventional) conclusions





The understanding of the "double descent" behavior is important for practitioners to choose between models for optimal performance

Why Least-Squares?

Why Least-Squares?

Easy to analyze

Why Least-Squares?

Easy to analyze

Easy to explain

Why Least-Squares?

Easy to analyze

Easy to explain

Easy to simulate

Why Least-Squares?

Easy to analyze

Easy to explain

Easy to simulate

Any non-linear model can be approximated by a linear one with large number of random features

$$\mathbb{E}[y|z] = f(z;\theta) \approx \nabla_\theta f(z;\theta_0)^T \beta$$

Why Least-Squares?

Easy to analyze

Easy to explain

Easy to simulate

Any non-linear model can be approximated by a linear one with large number of random features

$$\mathbb{E}[y|z] = f(z;\theta) \approx \nabla_\theta f(z;\theta_0)^T \beta$$

There is a well known connection between the gradient descent and the minimun-norm Least-Squares solution

The Linear model

$$y_i = x_i^T \beta + \epsilon_i \;\; ; \;\; i = 1, \cdots, n$$

$$x_i \in \mathbb{R}^p, \; \mathbb{E}[x_i] = \mathbf{0}, \; Cov(x_i) = \Sigma$$

$$\mathbb{E}[\epsilon_i] = 0, \; Var(\epsilon_i) = \sigma^2$$

We consider an asymptotic setup where $n, p \to \infty$ and $p/n \to \gamma \in (0, \infty)$

We also assume that $||\beta||_2^2 = r^2$ - constant "signal"

The Linear model

$$y_i = x_i^T \beta + \epsilon_i \;\; ; \;\; i = 1, \cdots, n$$

$$x_i \in \mathbb{R}^p, \; \mathbb{E}[x_i] = \mathbf{0}, \; Cov(x_i) = \Sigma$$

$$\mathbb{E}[\epsilon_i] = 0, \; Var(\epsilon_i) = \sigma^2$$

We consider an asymptotic setup where $n, p \to \infty$ and $p/n \to \gamma \in (0, \infty)$

We also assume that $||\beta||_2^2 = r^2$ - constant "signal"

Some assumptions over the distribution of x may be taken:

- $x \sim MN(\mathbf{0}, \Sigma)$
- $x = \Sigma^{1/2} z, \; z_j \sim (0, 1)$
  - Isotropic features: $\Sigma = I_p$
- $x = \varphi(Wz)$, where $W \in \mathbb{R}^{p \times d}$ a random matrix with i.i.d. entries

The Linear model

Assuming that the model is well-specified, the out-of-sample prediction risk is:

$$R(\hat{\beta};\beta) = \mathbb{E}_{XYx_0}\left(x_0^T\hat{\beta} - x_0^T\beta\right)^2$$

The Linear model

Assuming that the model is well-specified, the out-of-sample prediction risk is:

$$R(\hat{\beta};\beta) = \mathbb{E}_{XYx_0}\left(x_0^T\hat{\beta} - x_0^T\beta\right)^2$$

Assuming isotropic features, we can decompose the risk to bias and variance terms:

$$R(\hat{\beta};\beta) = \mathbb{E}_X\left[||\mathbb{E}[\hat{\beta}|X] - \beta||_2^2\right] + \mathbb{E}_X\left[tr[Cov(\hat{\beta}|X)]\right] := B(\hat{\beta};\beta) + V(\hat{\beta})$$

The Linear model

Assuming that the model is well-specified, the out-of-sample prediction risk is:

$$R(\hat{\beta};\beta) = \mathbb{E}_{XYx_0}\left(x_0^T\hat{\beta} - x_0^T\beta\right)^2$$

Assuming isotropic features, we can decompose the risk to bias and variance terms:

$$R(\hat{\beta};\beta) = \mathbb{E}_X\left[||\mathbb{E}[\hat{\beta}|X] - \beta||_2^2\right] + \mathbb{E}_X\left[tr[Cov(\hat{\beta}|X)]\right] := B(\hat{\beta};\beta) + V(\hat{\beta})$$

Taking $\hat{\beta} = (X^TX)^{-1}X^TY$, we get:

$$\mathbb{E}[\hat{\beta}|X] = \beta \;\; ; \;\; Cov(\hat{\beta}|X) = \sigma^2(X^TX)^{-1} \rightarrow \frac{\sigma^2}{n-p}I_p$$

and therefore $R(\hat{\beta};\beta) \rightarrow \frac{\sigma^2\gamma}{1-\gamma}$

High dimensional Least-Squares

When $\gamma > 1$, the empirical risk $||Y - X\beta||_2^2$ can be eliminated, and we are looking for the *minimum l_2 norm estimator*:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ ||\beta||_2 \;\; s.t : ||Y - X\beta||_2^2 = 0 \right\}$$

High dimensional Least-Squares

When $\gamma > 1$, the empirical risk $||Y - X\beta||_2^2$ can be eliminated, and we are looking for the *minimum l_2 norm estimator*:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ ||\beta||_2 \;\; s.t : ||Y - X\beta||_2^2 = 0 \right\}$$

Solving with Lagrange multipliers:

$$\underset{\beta,\lambda}{\operatorname{argmin}} \left\{ \beta^T \beta + \lambda^T (Y - X\beta) \right\}$$

We get:

$$\hat{\beta} = X^T \hat{\lambda} \;\; ; \;\; \hat{\lambda} = (XX^T)^{-1} Y \implies \hat{\beta} = X^T (XX^T)^{-1} Y$$

High dimensional Least-Squares

When $\gamma > 1$, the empirical risk $||Y - X\beta||_2^2$ can be eliminated, and we are looking for the *minimum l_2 norm estimator*:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ ||\beta||_2 \;\; s.t : ||Y - X\beta||_2^2 = 0 \right\}$$

Solving with Lagrange multipliers:

$$\underset{\beta,\lambda}{\operatorname{argmin}} \left\{ \beta^T \beta + \lambda^T (Y - X\beta) \right\}$$

We get:

$$\hat{\beta} = X^T \hat{\lambda} \;\; ; \;\; \hat{\lambda} = (XX^T)^{-1} Y \implies \hat{\beta} = X^T (XX^T)^{-1} Y$$

One can write: $\hat{\beta} = (X^T X)^{+} X^T Y$

Computing the bias term

Now we have:

$$\mathbb{E}[\hat{\beta}|X] = X^T(XX^T)^{-1}X\beta \neq \beta$$

and therefore the bias term is:

$$B(\hat{\beta};\beta) = \mathbb{E}_X\left[||\mathbb{E}[\hat{\beta}|X] - \beta||_2^2\right] = \beta^T(I_p - \mathbb{E}_X[X^T(XX^T)^{-1}X])\beta$$

Computing the bias term

Now we have:

$$\mathbb{E}[\hat{\beta}|X] = X^T(XX^T)^{-1}X\beta \neq \beta$$

and therefore the bias term is:

$$B(\hat{\beta};\beta) = \mathbb{E}_X\left[||\mathbb{E}[\hat{\beta}|X] - \beta||_2^2\right] = \beta^T(I_p - \mathbb{E}_X[X^T(XX^T)^{-1}X])\beta$$

We can show that $\mathbb{E}_X[X^T(XX^T)^{-1}X] \to \frac{n}{p}I_p$, and therefore:

$$B(\hat{\beta};\beta) = \beta^T\beta - \frac{n}{p}\beta^T\beta = r^2(1 - \frac{1}{\gamma})$$

Computing the bias term

Now we have:

$$\mathbb{E}[\hat{\beta}|X] = X^T(XX^T)^{-1}X\beta \neq \beta$$

and therefore the bias term is:

$$B(\hat{\beta};\beta) = \mathbb{E}_X\left[||\mathbb{E}[\hat{\beta}|X] - \beta||_2^2\right] = \beta^T(I_p - \mathbb{E}_X[X^T(XX^T)^{-1}X])\beta$$

We can show that $\mathbb{E}_X[X^T(XX^T)^{-1}X] \to \frac{n}{p}I_p$, and therefore:

$$B(\hat{\beta};\beta) = \beta^T\beta - \frac{n}{p}\beta^T\beta = r^2(1 - \frac{1}{\gamma})$$

Note that:

$$\mathbb{E}_X\left[||\mathbb{E}[\hat{\beta}|X]||_2^2\right] \to \frac{n}{p}r^2 = \frac{r^2}{\gamma}$$

Computing the variance term

Recall that:

$$V(\hat{\beta}) = \mathbb{E}_X \left[tr[Cov(\hat{\beta}|X)] \right]$$

Now we have:

$$Cov(\hat{\beta}|X) = \sigma^2 (XX^T)^{-1} XX^T (XX^T)^{-1} = \sigma^2 (XX^T)^{-1} \to \frac{\sigma^2}{p-n} I_n$$

and therefore:

$$V(\hat{\beta}) \to \frac{\sigma^2}{\gamma - 1}$$

Limiting Risk

For the asymptotic setting, we can obtain the following formula:

$$R(\gamma) = \begin{cases} \sigma^2 \frac{\gamma}{1-\gamma} & \text{for } \gamma < 1 \\ r^2\left(1 - \frac{1}{\gamma}\right) + \sigma^2 \frac{1}{\gamma - 1} & \text{for } \gamma > 1 \end{cases}$$

- A new bias-variance trade-off in the over-parameterized regime
- The behavior can be controlled by the $SNR = r^2/\sigma^2$
- If $SNR > 1$, there is a local min at $\gamma = \frac{\sqrt{SNR}}{\sqrt{SNR}-1}$
- As $\gamma \to \infty$, the estimator $\hat{\beta}$ converge to the null estimator $\tilde{\beta} = 0$, and the total risk is r^2

Empirical results

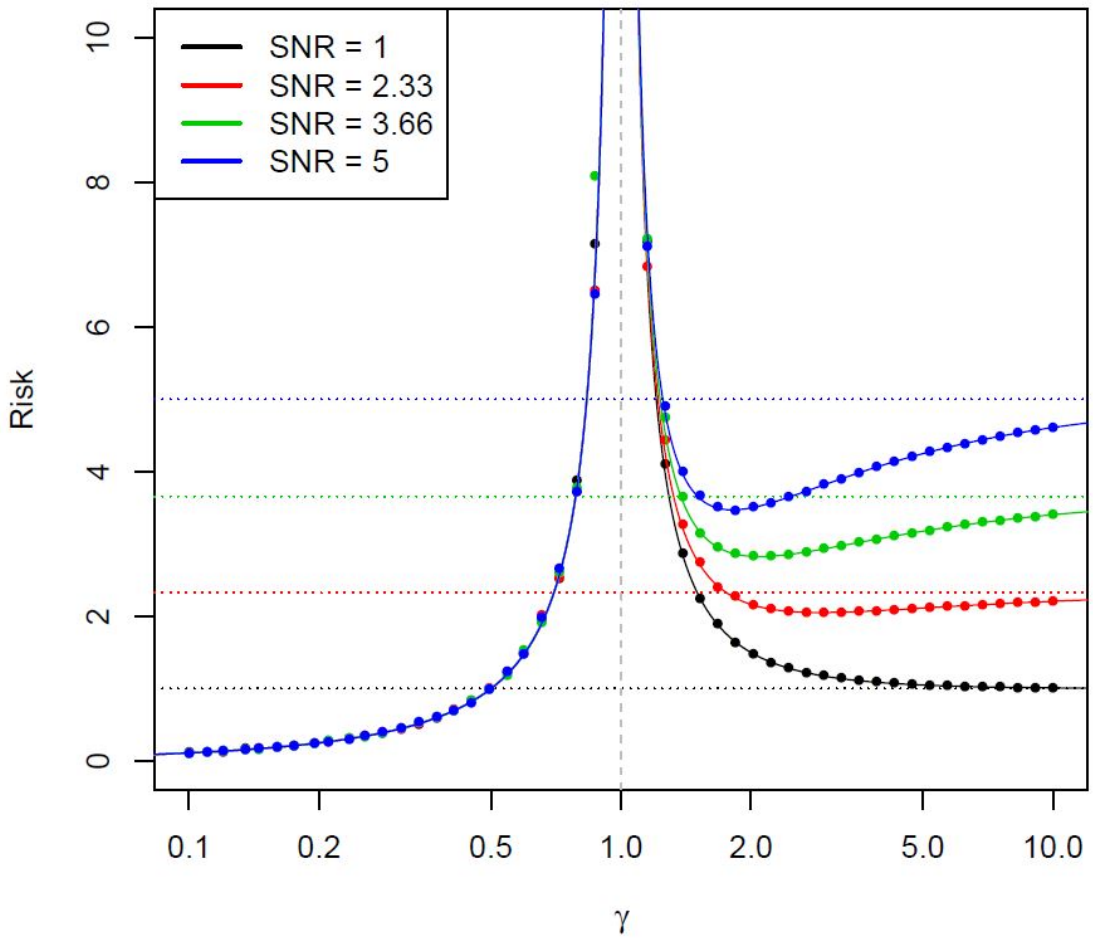


Figure: $\sigma^2 = 1$, r^2 varies from 1 to 5

Misspecified model

$$y_i = x_i^T \beta + \omega_i^T \theta + \epsilon_i \ ; \ \ i = 1, \cdots, n$$

$$x_i \in \mathbb{R}^p, \ \omega_i \in \mathbb{R}^d, \ \mathbb{E}[(x_i, \omega_i)] = \mathbf{0}, \ \mathbb{E}[\epsilon_i] = 0, \ Var(\epsilon_i) = \sigma^2$$

For simplicity, we assume that $Cov((x_i, \omega_i)) = I_{p+d}$

Misspecified model

$$y_i = x_i^T \beta + \omega_i^T \theta + \epsilon_i \;\; ; \;\; i = 1, \cdots, n$$

$$x_i \in \mathbb{R}^p, \; \omega_i \in \mathbb{R}^d, \; \mathbb{E}[(x_i, \omega_i)] = \mathbf{0}, \; \mathbb{E}[\epsilon_i] = 0, \; Var(\epsilon_i) = \sigma^2$$

For simplicity, we assume that $Cov((x_i, \omega_i)) = I_{p+d}$

In this case we can write:

$$y_i = x_i^T \beta + \delta_i \;\; ; \;\; i = 1, \cdots, n$$

$$\mathbb{E}[\delta_i] = 0, \; Var(\delta_i) = \sigma^2 + ||\theta||_2^2$$

Misspecified model

$$y_i = x_i^T \beta + \omega_i^T \theta + \epsilon_i \;\; ; \;\; i = 1, \cdots, n$$

$$x_i \in \mathbb{R}^p, \; \omega_i \in \mathbb{R}^d, \; \mathbb{E}[(x_i, \omega_i)] = \mathbf{0}, \; \mathbb{E}[\epsilon_i] = 0, \; Var(\epsilon_i) = \sigma^2$$

For simplicity, we assume that $Cov((x_i, \omega_i)) = I_{p+d}$

In this case we can write:

$$y_i = x_i^T \beta + \delta_i \;\; ; \;\; i = 1, \cdots, n$$

$$\mathbb{E}[\delta_i] = 0, \; Var(\delta_i) = \sigma^2 + ||\theta||_2^2$$

We also assume that $||\beta||_2^2 + ||\theta||_2^2 = r^2$ - constant "signal"

Misspecified model

The risk is:

$$R(\hat{\beta};\beta,\theta) = \mathbb{E}_X\left[||\mathbb{E}[\hat{\beta}|X] - \beta||_2^2\right] + \mathbb{E}_X\left[tr[Cov(\hat{\beta}|X)]\right] + ||\theta||_2^2$$

$$:= B(\hat{\beta};\beta) + V(\hat{\beta};\theta) + M(\beta,\theta)$$

Misspecified model

The risk is:

$$R(\hat{\beta};\beta,\theta) = \mathbb{E}_X\left[||\mathbb{E}[\hat{\beta}|X] - \beta||_2^2\right] + \mathbb{E}_X\left[tr[Cov(\hat{\beta}|X)]\right] + ||\theta||_2^2$$

$$:= B(\hat{\beta};\beta) + V(\hat{\beta};\theta) + M(\beta,\theta)$$

For the variance term we have:

- $V(\hat{\beta};\theta) = (\sigma^2 + ||\theta||_2^2)\frac{\gamma}{1-\gamma}$, for $\gamma < 1$
- $V(\hat{\beta};\theta) = (\sigma^2 + ||\theta||_2^2)\frac{1}{\gamma-1}$, for $\gamma > 1$

Misspecified model

The total risk for $\gamma < 1$:

$$||\theta||_2^2 + (\sigma^2 + ||\theta||_2^2)\frac{\gamma}{1-\gamma}$$

The total risk for $\gamma > 1$:

$$||\theta||_2^2 + ||\beta||_2^2(1 - \frac{1}{\gamma}) + (\sigma^2 + ||\theta||_2^2)\frac{1}{\gamma - 1}$$

Misspecified model

The total risk for $\gamma < 1$:

$$||\theta||_2^2 + (\sigma^2 + ||\theta||_2^2)\frac{\gamma}{1-\gamma}$$

The total risk for $\gamma > 1$:

$$||\theta||_2^2 + ||\beta||_2^2(1 - \frac{1}{\gamma}) + (\sigma^2 + ||\theta||_2^2)\frac{1}{\gamma - 1}$$

What can be a conventional connection between γ and $||\theta||_2^2$?

How the signal is distributed over γ?

Polynomial decay of the signal

We now assume that:

$$||\theta||_2^2 = r^2(1+\gamma)^{-a}$$

$$||\beta||_2^2 = r^2(1-(1+\gamma)^{-a})$$

for some $a > 0$

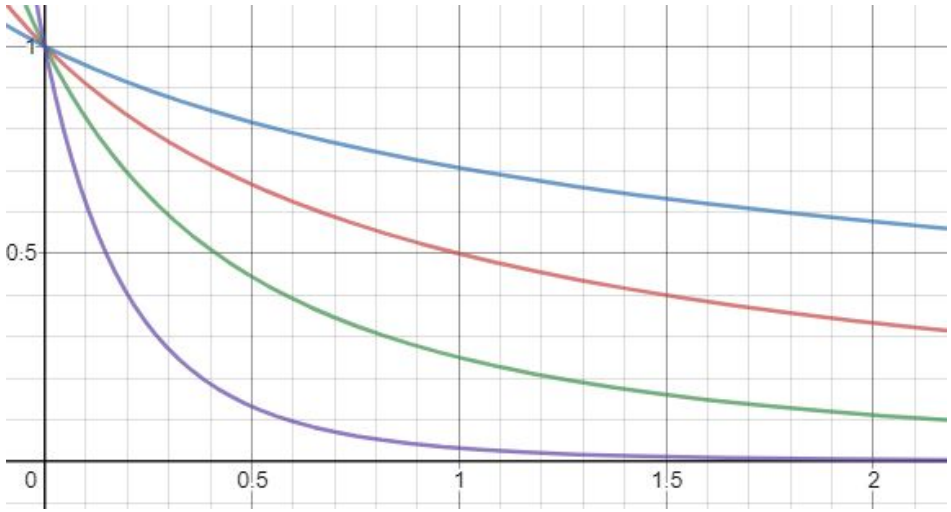


Figure: $||\theta||_2^2 = ((1+\gamma)^{-a},\ a \in \{0.5, 1, 2, 5\}$

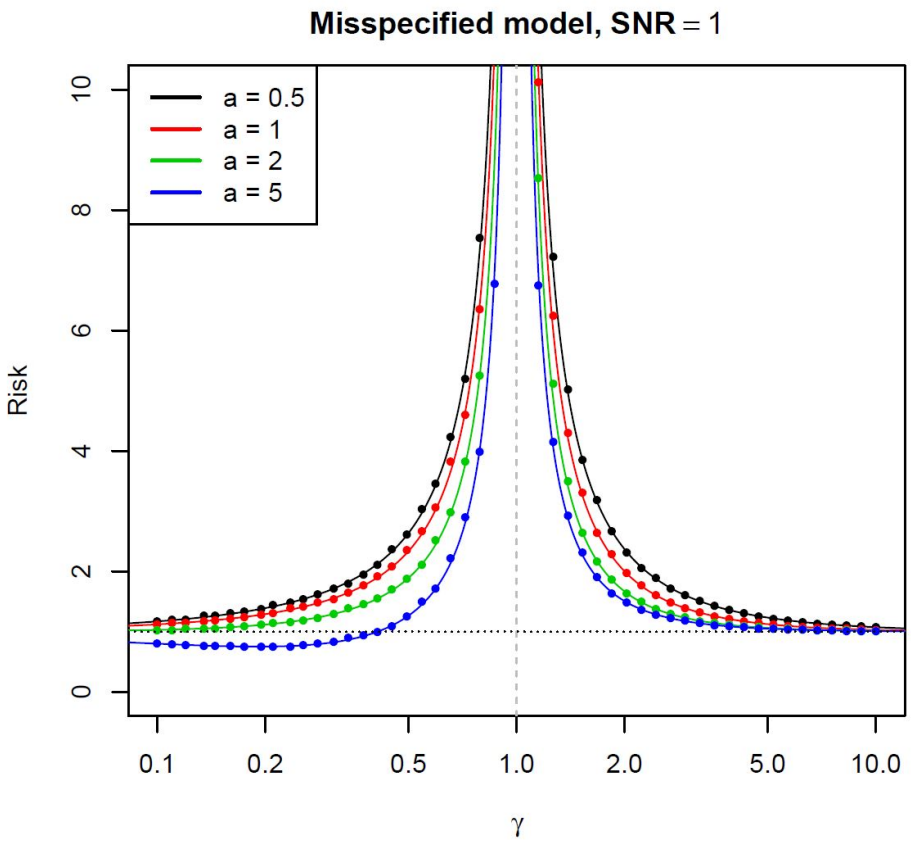
Polynomial decay of the signal

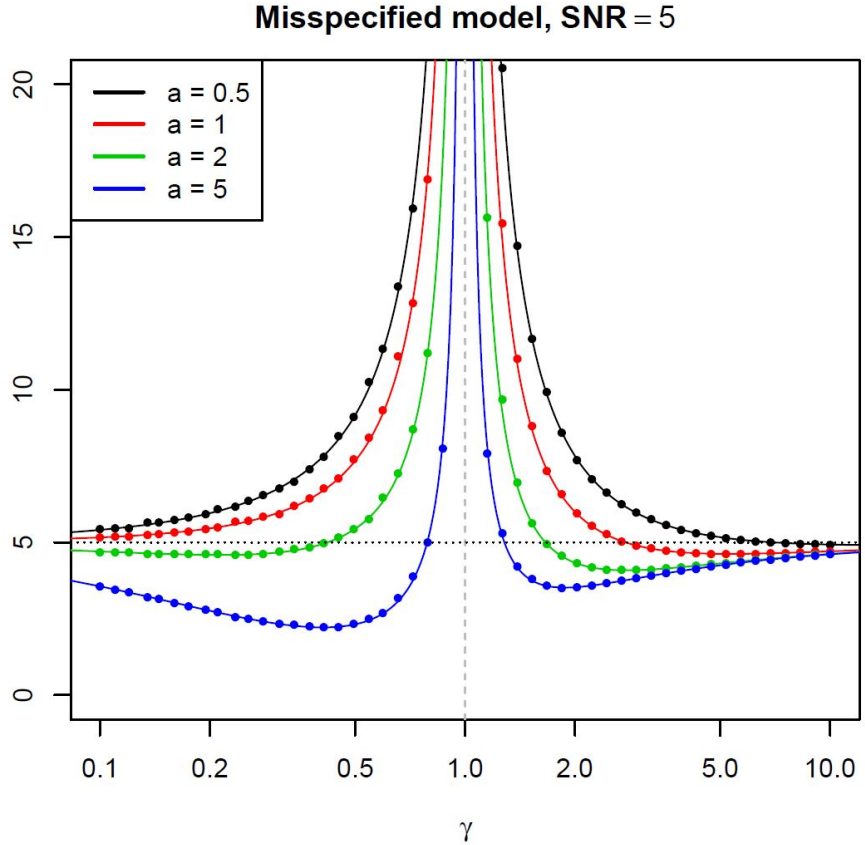
We now obtain the following formula:

$$R_a(\gamma) = \begin{cases} r^2(1+\gamma)^{-a} + (r^2(1+\gamma)^{-a} + \sigma^2)\frac{\gamma}{1-\gamma} & \text{for } \gamma < 1 \\ r^2(1+\gamma)^{-a} + r^2\left(1-(1+\gamma)^{-a}\right)\left(1-\frac{1}{\gamma}\right) + (r^2(1+\gamma)^{-a} + \sigma^2)\frac{1}{\gamma-1} & \text{for } \gamma > 1 \end{cases}$$

- We can see that $R(\gamma = 0) = R(\gamma = \infty) = r^2$ (the null risk)
- For $a \leq 1 + \frac{1}{SNR}$, $R_a(\gamma)$ is a monotonically increasing function in the under-parameterized regime
- If $SNR \leq 1$, the risk in the over-parameterized regime always worse than the null risk
- If $SNR > 1$, there is a local minimum in the over-parameterized regime, and it is **global** for small enough a

Empirical results





The "double descent" behavior achieved...

Thinking about it...

Yet, we did not see the same behavior as in the NN simulations...

The reason may be - the distribution of the signal over the parameters space

Thinking about it...

Yet, we did not see the same behavior as in the NN simulations...

The reason may be - the distribution of the signal over the parameters space

What if - the majority of the signal is located within some range in the over-parameterized regime?

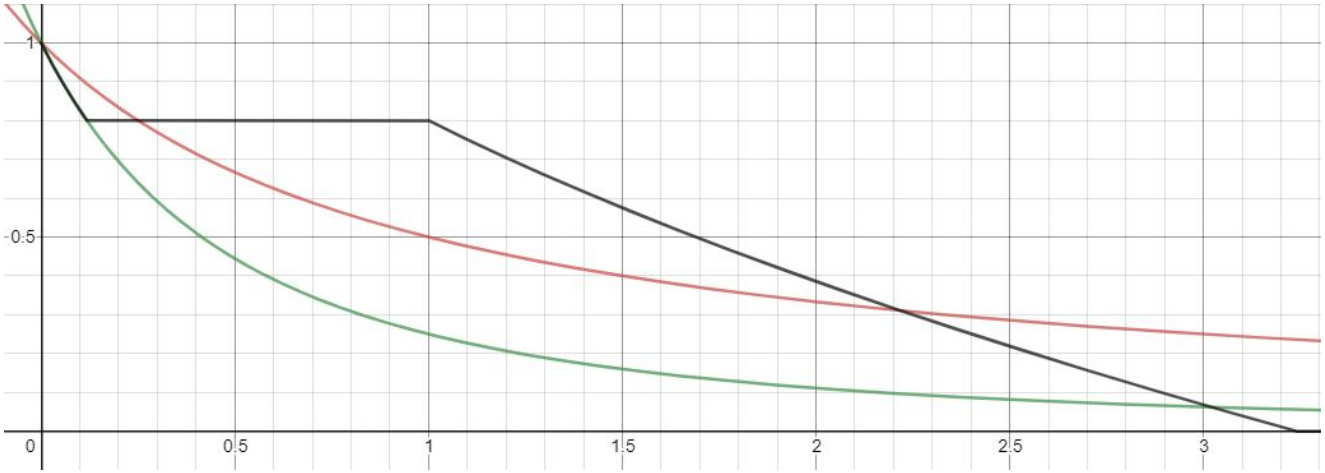


Figure: $||\theta||_2^2 = g(\gamma)$

Model evaluation

For the task of models evaluation and selection we may use the *leave-one-out cross-validation* estimator (CV for short):

$$CV_n = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}_n^{-i}(x_i)\right)^2$$

Model evaluation

For the task of models evaluation and selection we may use the *leave-one-out cross-validation* estimator (CV for short):

$$CV_n = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}_n^{-i}(x_i)\right)^2$$

We may also want use the "shortcut formula":

$$CV_n = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{f}_n(x_i)}{1 - S_{ii}}\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - [SY]_i}{1 - S_{ii}}\right)^2$$

where S is the linear smoother matrix

Model evaluation

For any linear interpolator:

$$SY = Y \implies S = I_n \implies \frac{y_i - [SY]_i}{1 - S_{ii}} = \frac{0}{0}$$

in particular for the min-norm interpolator: $S = XX^T(XX^T)^{-1} = I_n$

Model evaluation

For any linear interpolator:

$$SY = Y \implies S = I_n \implies \frac{y_i - [SY]_i}{1 - S_{ii}} = \frac{0}{0}$$

in particular for the min-norm interpolator: $S = XX^T(XX^T)^{-1} = I_n$

Fortunately, we can solve this problem! Rewrite S to be:

$$S = XX^T(XX^T + \lambda I_n)^{-1} \quad , \quad \lambda \to 0^+$$

Model evaluation

Now we can apply L'Hopital's rule by with derivative at $\lambda = 0$

$$\frac{(y_i - [SY]_i)'}{(1 - S_{ii})'} = \frac{\left[XX^T(XX^T + \lambda I_n)^{-2}Y\right]_i}{[XX^T(XX^T + \lambda I_n)^{-2}]_{ii}} \mid_{\lambda=0} = \frac{\left[(XX^T)^{-1}Y\right]_i}{[(XX^T)^{-1}]_{ii}}$$

Model evaluation

Now we can apply L'Hopital's rule by with derivative at $\lambda = 0$

$$\frac{(y_i - [SY]_i)'}{(1 - S_{ii})'} = \frac{\left[XX^T(XX^T + \lambda I_n)^{-2}Y\right]_i}{[XX^T(XX^T + \lambda I_n)^{-2}]_{ii}} \mid_{\lambda=0} = \frac{\left[(XX^T)^{-1}Y\right]_i}{[(XX^T)^{-1}]_{ii}}$$

Finally, the CV estimator can be calculated with the following formula:

$$CV_n = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\left[(XX^T)^{-1}Y\right]_i}{[(XX^T)^{-1}]_{ii}}\right)^2$$

Ridge regression

The min-norm estimator is related to the Ridge regression estimator as follows:

$$\hat{\beta} = lim_{\lambda \to 0^+} \hat{\beta}_\lambda$$

where $\hat{\beta}_\lambda$ is the Ridge regression estimator:

$$\hat{\beta}_\lambda = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \frac{1}{n} ||Y - X\beta||_2^2 + \lambda ||\beta||_2 \right\} = (X^T X + n\lambda I_p)^{-1} X^T Y$$

Thus, an optimal tune $\hat{\beta}_\lambda$ should be better than $\hat{\beta}$

Ridge regression

The min-norm estimator is related to the Ridge regression estimator as follows:

$$\hat{\beta} = lim_{\lambda \to 0^+} \hat{\beta}_\lambda$$

where $\hat{\beta}_\lambda$ is the Ridge regression estimator:

$$\hat{\beta}_\lambda = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \frac{1}{n} ||Y - X\beta||_2^2 + \lambda ||\beta||_2 \right\} = (X^T X + n\lambda I_p)^{-1} X^T Y$$

Thus, an optimal tune $\hat{\beta}_\lambda$ should be better than $\hat{\beta}$

The limiting risk for the optimal $\hat{\beta}_\lambda$ can be written explicitly as:

$$\sigma^2 \frac{-(1-(1+\sigma^2/r^2)\gamma) + \sqrt{(1-(1+\sigma^2/r^2)\gamma)^2 - 4\sigma^2\gamma^2/r^2}}{2\gamma}$$

Ridge regression

In overview looking we can simplify the optimal risk into:

$$R(\hat{\beta}_{\lambda^*}; \beta, \theta) \approx ||\theta||_2^2 + f(\sigma^2; \gamma) + g(||\beta||_2^2; \gamma)$$

where $f(z; \gamma) \to 0$, $g(z; \gamma) \to z$ as $\gamma \to \infty$

and $g(z; 0) = f(z; 0) = 0$

Ridge regression

In overview looking we can simplify the optimal risk into:

$$R(\hat{\beta}_{\lambda^*}; \beta, \theta) \approx ||\theta||_2^2 + f(\sigma^2; \gamma) + g(||\beta||_2^2; \gamma)$$

where $f(z; \gamma) \to 0$, $g(z; \gamma) \to z$ as $\gamma \to \infty$

and $g(z; 0) = f(z; 0) = 0$

Looks like trade-off between observed and unobserved signals

Ridge regression

In overview looking we can simplify the optimal risk into:

$$R(\hat{\beta}_{\lambda^*}; \beta, \theta) \approx ||\theta||_2^2 + f(\sigma^2; \gamma) + g(||\beta||_2^2; \gamma)$$

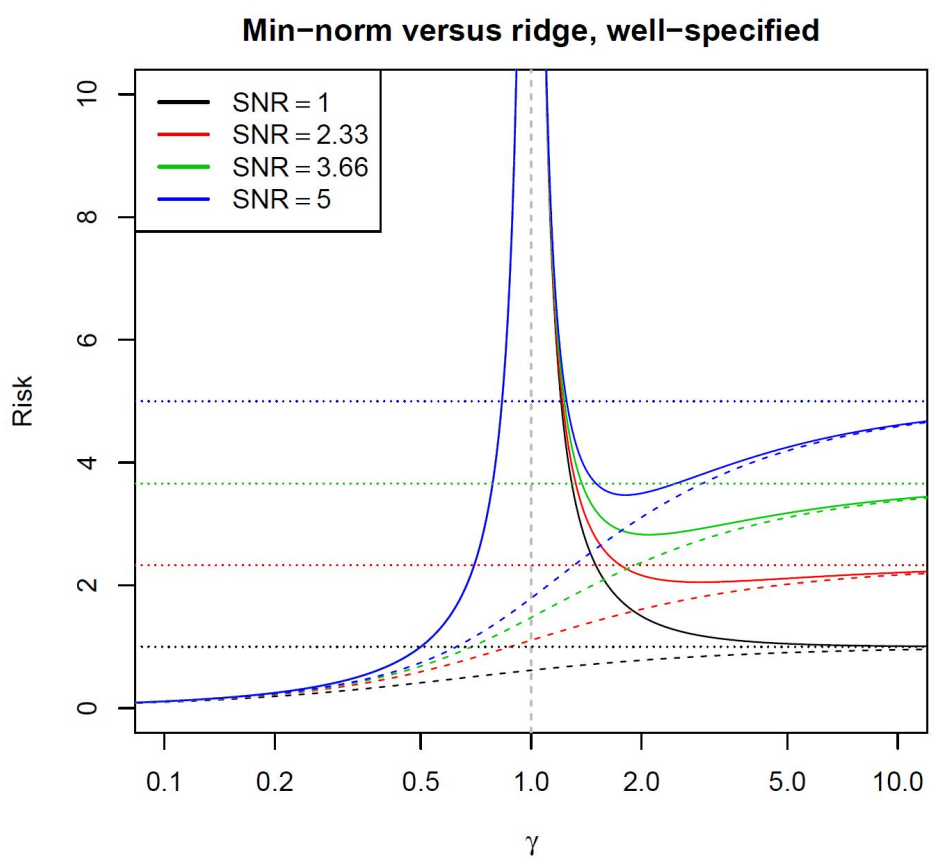
where $f(z; \gamma) \to 0$, $g(z; \gamma) \to z$ as $\gamma \to \infty$

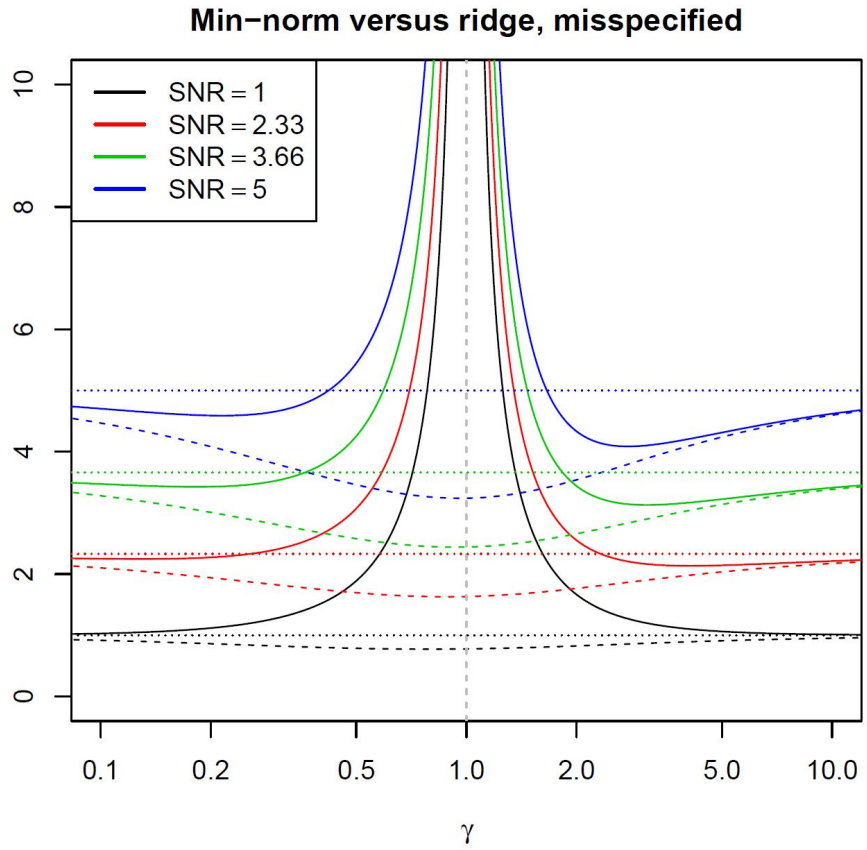
and $g(z; 0) = f(z; 0) = 0$

Looks like trade-off between observed and unobserved signals

Again, the distribution of the signal over γ may play a role...

Ridge regression - optimal risk curves





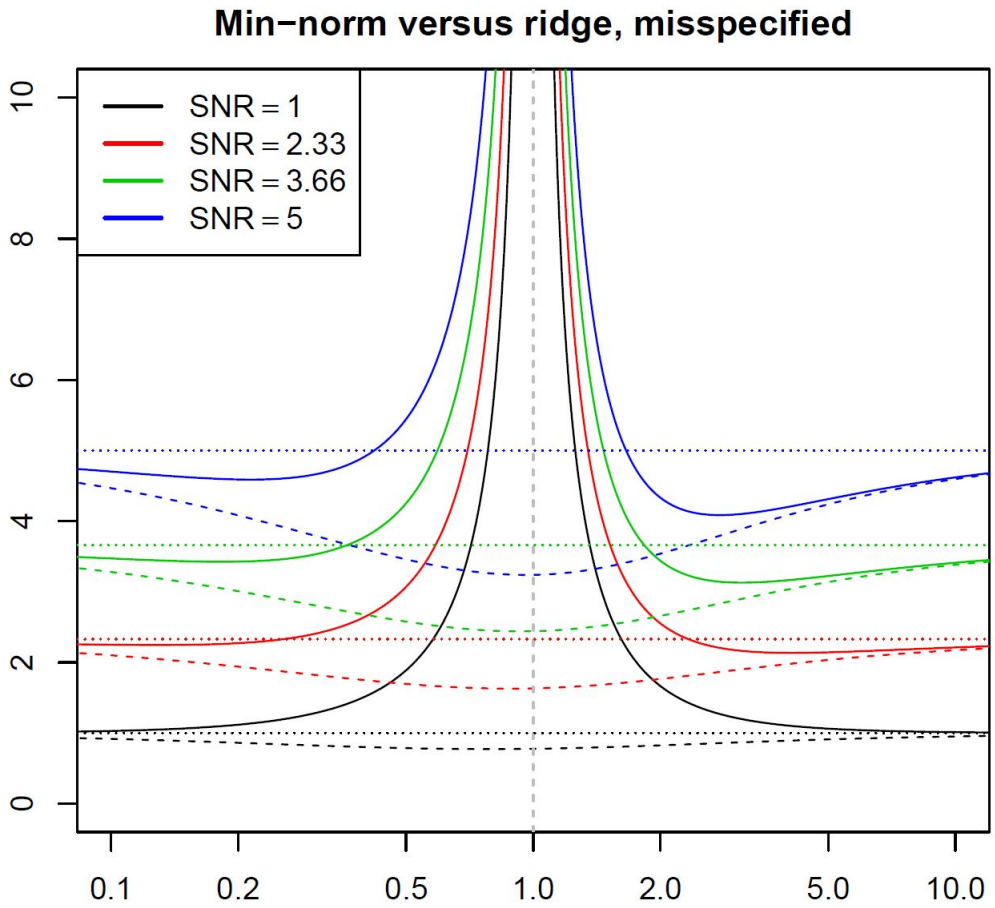
Optimal risk curves - misspecified model

$$R(\hat{\beta}_{\lambda^*}; \beta, \theta) \approx ||\theta||_2^2 + f(\sigma^2; \gamma) + g(||\beta||_2^2; \gamma)$$

Optimal risk curves with $||\theta||_2^2 = (1+\gamma)^{-a}$, $a = 2$

Why is the minimum risk around $\gamma = 1$?

"... it seems we want the complexity of the feature space to put us as close to the interpolation boundary as possible..."



Optimal risk curves - misspecified model

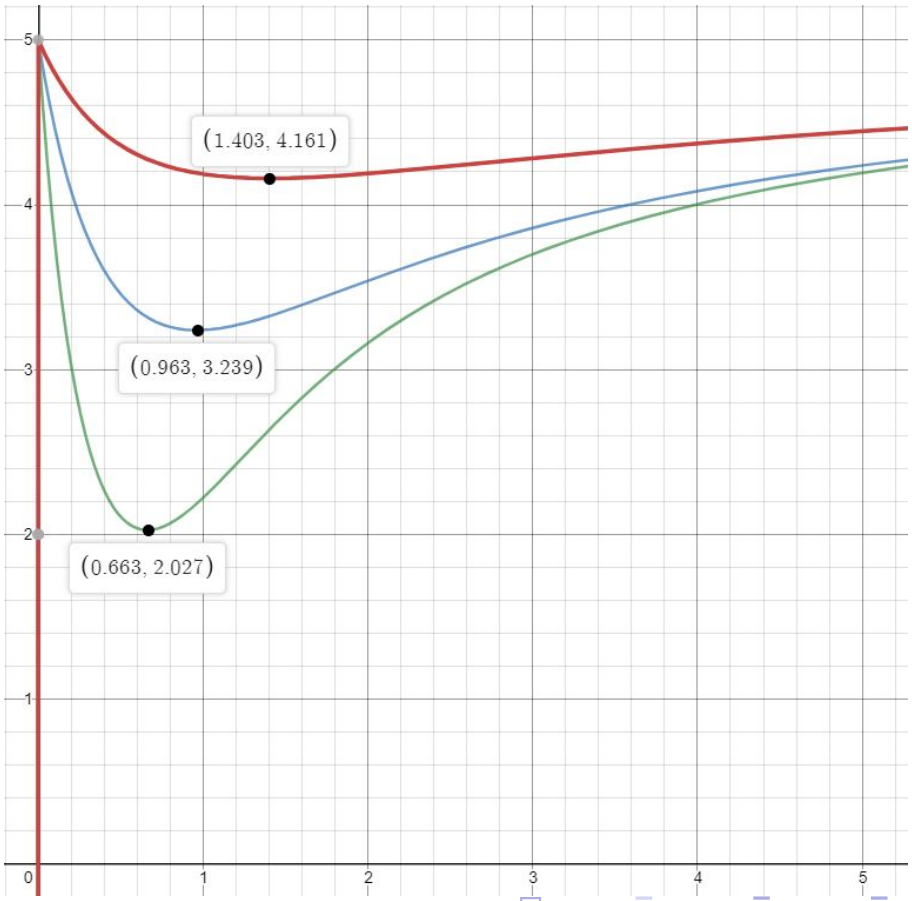
$$R(\hat{\beta}_{\lambda^*}; \beta, \theta) \approx ||\theta||_2^2 + f(\sigma^2; \gamma) + g(||\beta||_2^2; \gamma)$$

Optimal risk curves with:

- $\sigma^2 = 1, r^2 = 5$
- $||\theta||_2^2 = (1+\gamma)^{-a}$, $a \in \{1, 2, 4\}$

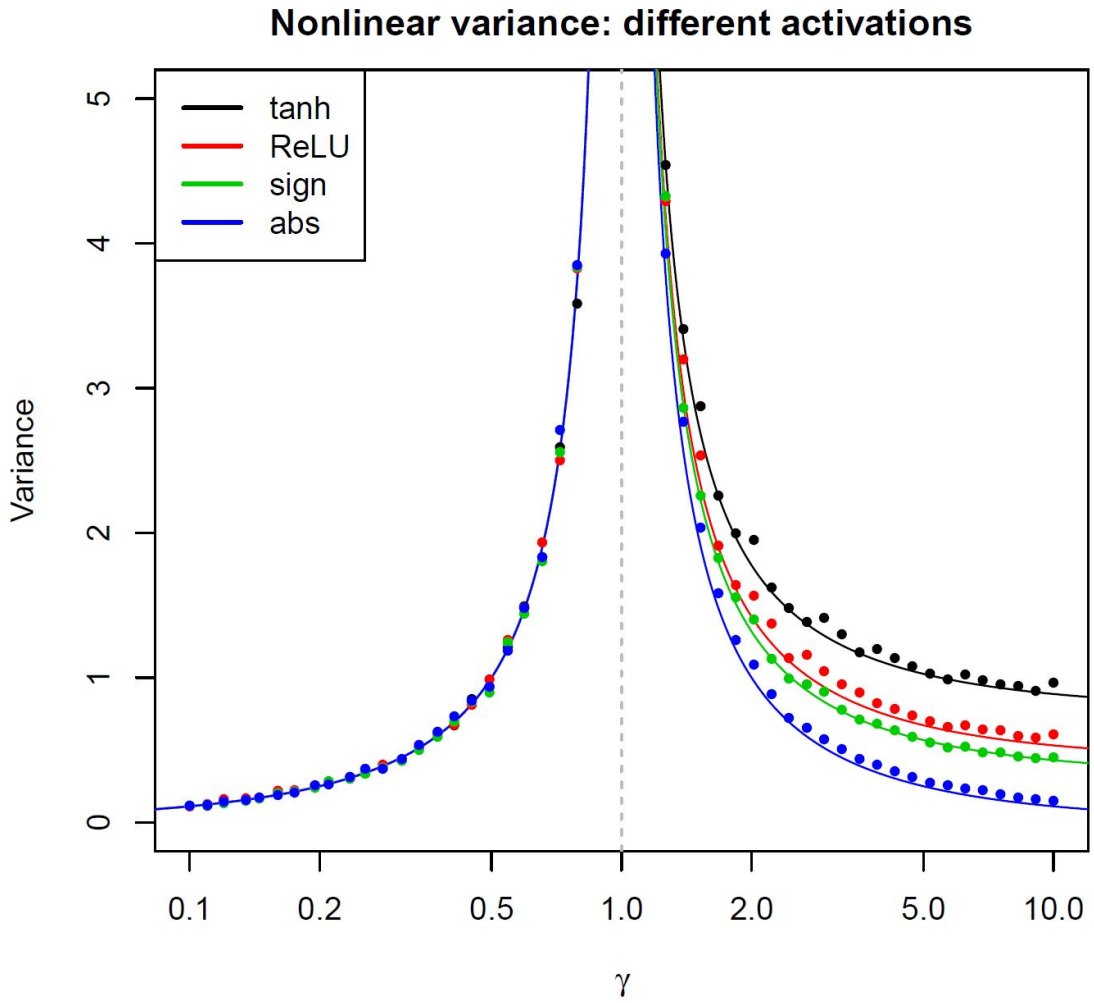
Curves get steeper as r^2 grows

What conclusions can we draw regarding Neural Networks?



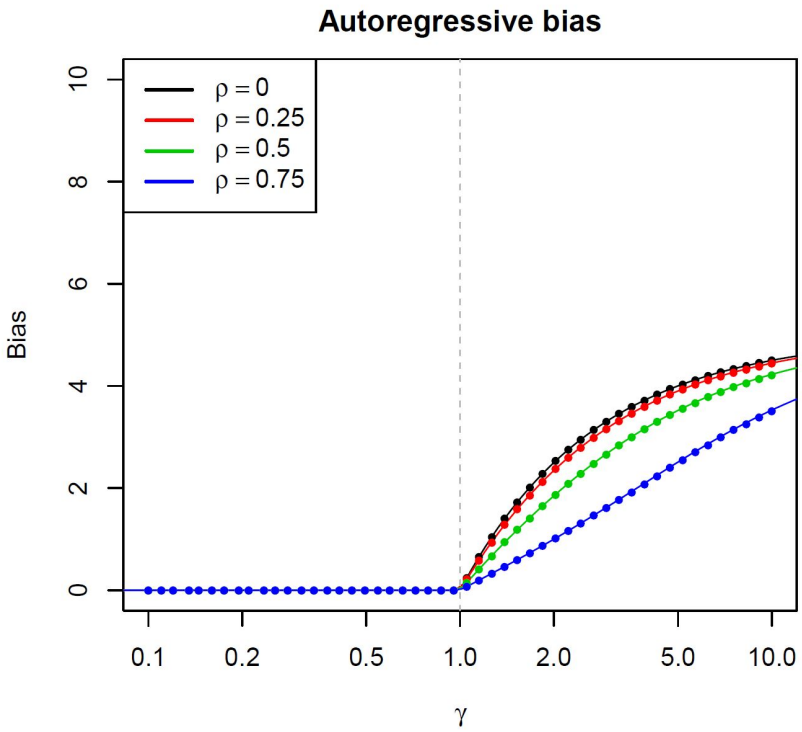
Additional results - nonlinear features

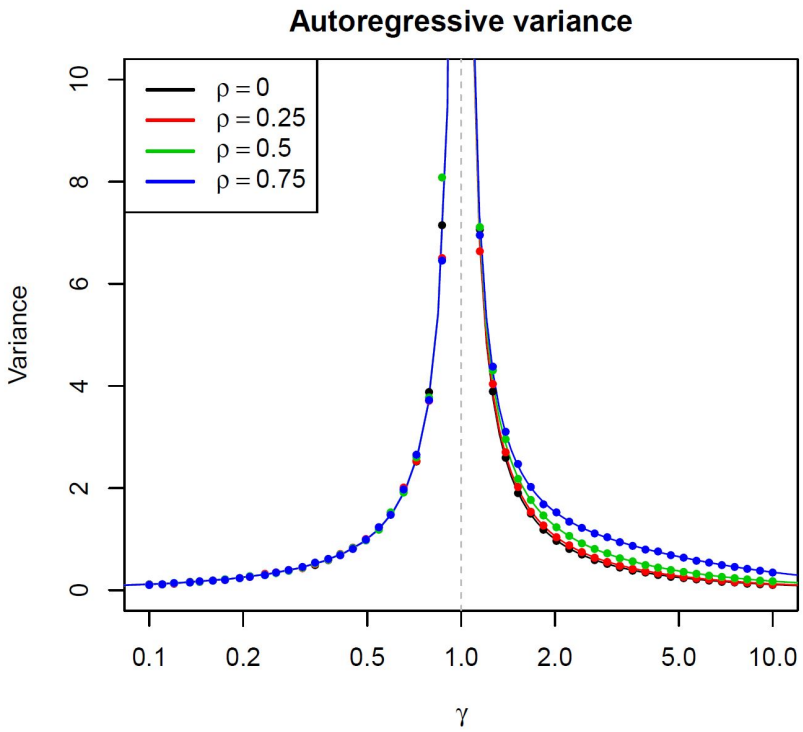
Asymptotic variance in a nonlinear feature model, $x = \varphi(Wz)$



Additional results - correlated features

Asymptotic variance and bias for auto-regressive structure, $\Sigma_{ij} = \rho^{|i-j|}$



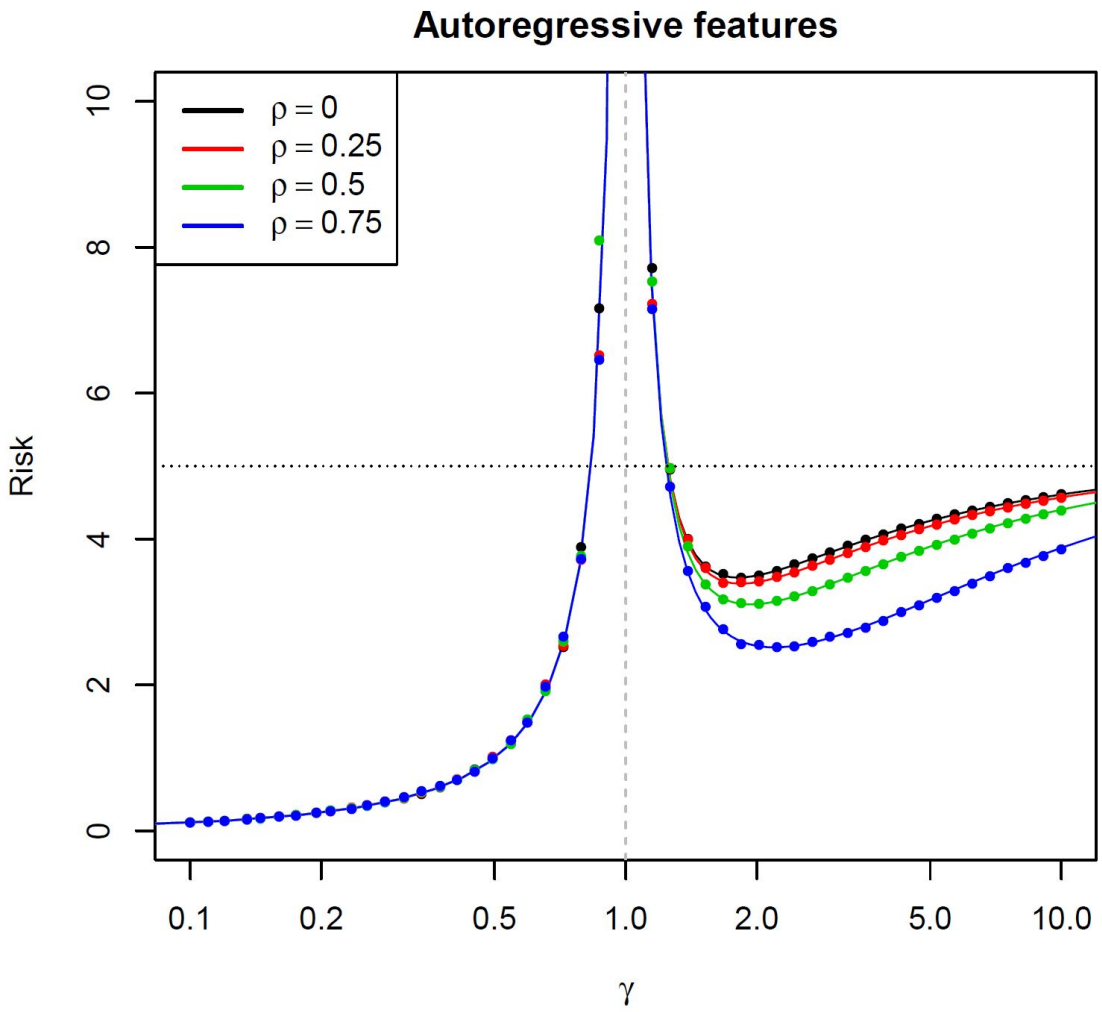


Reminder:

$$B(\hat{\beta};\beta) = \beta^T(I_p - \mathbb{E}_X[X^T(XX^T)^{-1}X])\beta$$

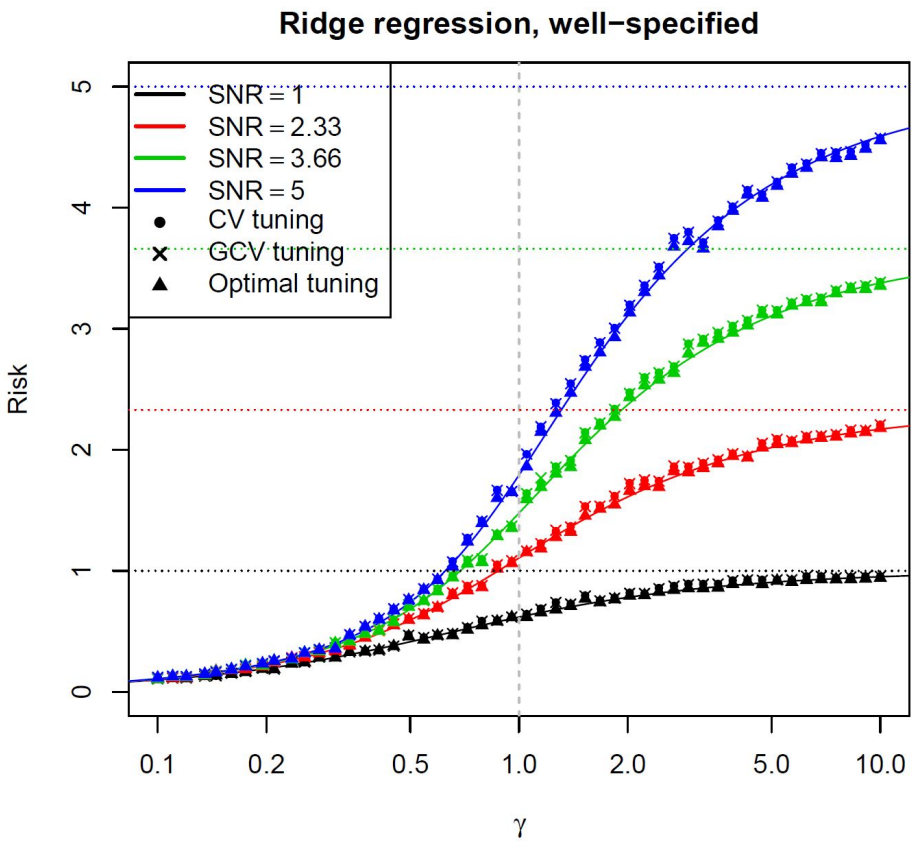
Additional results - correlated features

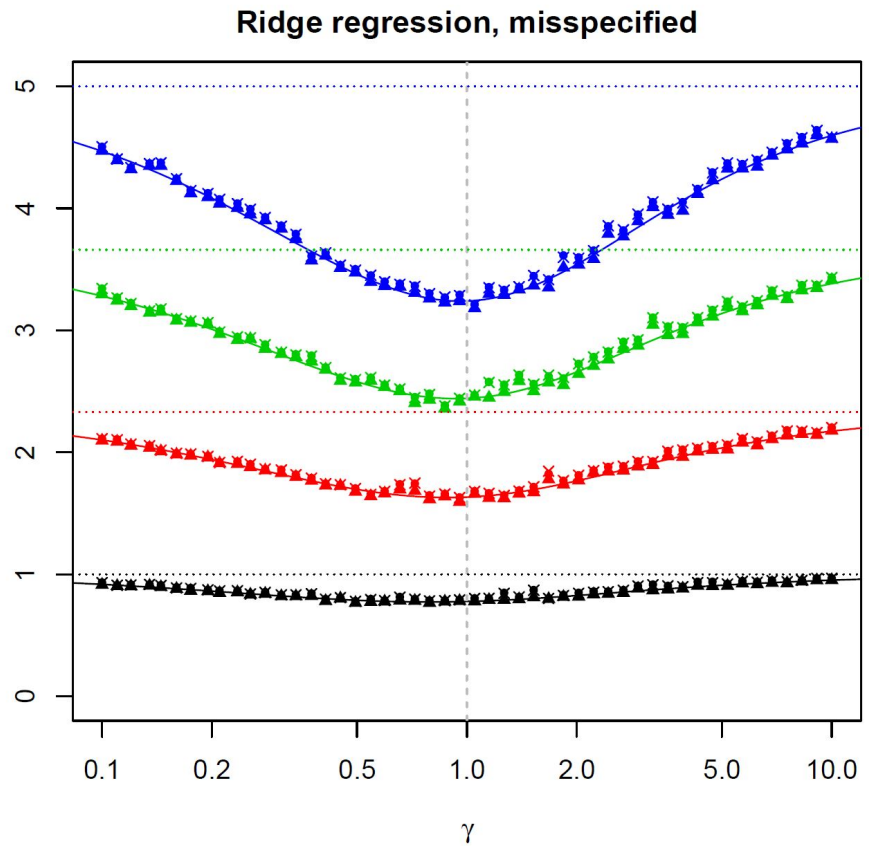
Asymptotic risk for auto-regressive structure, $\Sigma_{ij} = \rho^{|i-j|}$



CV-tuned Ridge regression

Finite-sample risks for CV-tuned ridge regression estimator compered to Asymptotic risk (20 independent training samples)





Summary

Summary

There is a growing interest in *Interpolators* in ML

Summary

There is a growing interest in *Interpolators* in ML

The double descent phenomenon must be well understood and taken into account for model optimization

The linear model analysis explains the bias-variance trade-off in the interpolation regime

Summary

There is a growing interest in *Interpolators* in ML

The double descent phenomenon must be well understood and taken into account for model optimization

The linear model analysis explains the bias-variance trade-off in the interpolation regime

The real-life trade-off:

- Balance between signal_{obs}-bias-variance
- Controlled by complexity-regularization/early stopping


Thank you
for
listening ☺