Seminars: Overparametrized Machine Learning

Surprises in High-Dimensional Least Squares Interpolation

Antônio Horta Ribeiro
Dave Zacharias
Per Mattersson

UPPSALA
UNIVERSITET

Division of Systems and Control
Department of Information Technology
Uppsala University

it-seminar-overparameterized-ml@lists.uu.se

UPPSALA UNIVERSITET

Next seminar

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 way—several phenomena that have been observed in large-scale neural networks and kernel machines, including the "double descent" behavior of the prediction risk, and the potential benefits of overparametrization.

1 Introduction

Modern deep learning models involve a huge number of parameters. In many applications, current practice suggests that we should design the network to be sufficiently complex so that the model (as trained, typically, by gradient descent) interpolates the data, i.e., achieves zero training error. Indeed, in a thought-provoking experiment, Zhang et al. (2016) showed that state-of-the-art deep neural network architectures are complex enough that they can be trained to interpolate the data even when the actual labels are replaced by entirely random ones.

Despite their enormous complexity, deep neural networks are frequently observed to generalize well in practice. At first sight, this seems to defy conventional statistical wisdom: interpolation (vanishing training error) is commonly taken to be a proxy for overfitting, poor generalization (large gap between training and test error), and hence large test error. In an insightful series of papers, Belkin et al. (2018b,c,a) pointed out that these concepts are in general distinct, and interpolation does not contradict generalization. For example, recent work has investigated interpolation —via kernel ridge regression— in reproducing kernel Hilbert spaces (Liang et al., 2020; Ghorbani et al., 2019). While in low dimension a positive regularization is needed to achieve good interpolation, in certain high dimensional settings interpolation can be nearly optimal.

In this paper, we investigate these phenomena in the context of simple linear models. We assume to be given i.i.d. data (y_i, x_i), $i \le n$, with $x_i \in \mathbb{R}^p$ a feature vector and $y_i \in \mathbb{R}$ a response variable. These are distributed according to the model (see Section 2 for further definitions)

$$(x_i, \epsilon_i) \sim P_x \times P_\epsilon, \quad i = 1, \ldots, n, \tag{1}$$

$$y_i = x_i^T \beta + \epsilon_i, \quad i = 1, \ldots, n, \tag{2}$$

Highlights

Connection to neural networks (Section 1.2)

- Let $\theta \in \mathbb{R}^m$ be the parameter vector.
- $\theta = \theta_0 + \beta$:
- The number of parameters is so large that training effectively only changes the parameter by a small amount. Them β is small and:

$$f(z;\theta) \approx f(z;\theta_0) + \nabla_\theta f(z;\theta_0)^\mathsf{T}\beta,$$

Chizat, L., Oyallon, E., and Bach, F. *On Lazy Training in Differentiable Programming.* Neural Information Processing Systems (NeurIPS), 2019

Connection to neural networks (Section 1.2)

- Let $\theta \in \mathbb{R}^m$ be the parameter vector.
- $\theta = \theta_0 + \beta$:
- The number of parameters is so large that training effectively only changes the parameter by a small amount. Them β is small and:

$$f(z;\theta) \approx \cancel{f(z;\theta_0)} + \overbrace{\nabla_\theta f(z;\theta_0)^\mathsf{T}}^{x^\mathsf{T}} \beta,$$

Chizat, L., Oyallon, E., and Bach, F. *On Lazy Training in Differentiable Programming.* Neural Information Processing Systems (NeurIPS), 2019

Connection to neural networks (Section 1.2)

- Nonlinear map to feature space: $z \mapsto \nabla_\theta f(z; \theta_0) = x$
- Regression on the feature space:
$$y = x^{\mathsf{T}} \beta$$
- More about that line in Seminar S4:

Jacot, A., Gabriel, F., and Hongler, C. *Neural Tangent Kernel: Convergence and Generalization in Neural Networks.* Neural Information Processing Systems (NeurIPS), 2018

Ridgeless least squares (Section 2.2)

Estimated parameter: using train dataset (x_t, y_t), $t = 1, \cdots, n$:

Ridgeless least squares (Section 2.2)

Estimated parameter: using train dataset $(x_t, y_t), t = 1, \cdots, n$:

- Underparametrized:

$$\min_{\theta} \sum_t (y_i - \theta^\mathsf{T} x_t)^2$$

Ridgeless least squares (Section 2.2)

Estimated parameter: using train dataset (x_t, y_t), $t = 1, \cdots, n$:

- Underparametrized:

$$\min_{\theta} \sum_{t} (y_i - \theta^{\mathsf{T}} x_t)^2$$

- Overparametrized:

$$\min_{\theta} \|\theta\|_2^2$$

$$\text{subject to } y_t = \theta^{\mathsf{T}} x_t$$

$$\text{for every } t = 1, \cdots, n$$

- Connection with gradient descent (*Proposition 1*).

  See also: https://math.stackexchange.com/q/3499305

Isotropic features (Section 3)

Theorem 1. *Assume the model (1), (2), where $x \sim P_x$ has i.i.d. entries with zero mean, unit variance, and a finite moment of order $4+\eta$, for some $\eta > 0$. Also assume that $\|\beta\|_2^2 = r^2$ for all n, p. Then for the min-norm least squares estimator $\hat{\beta}$ in (4), as $n, p \to \infty$, such that $p/n \to \gamma \in (0, \infty)$, it holds almost surely that*

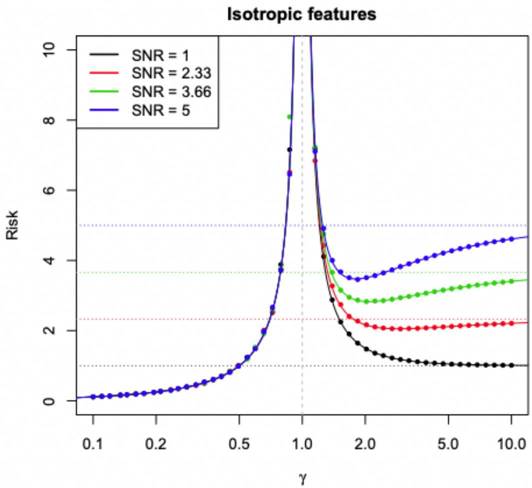
$$B_X(\hat{\beta}; \beta) \to r^2\Big(1 - \frac{1}{\gamma}\Big), \tag{6}$$

$$V_X(\hat{\beta}; \beta) \to \sigma^2 \frac{1}{\gamma - 1}. \tag{7}$$

Hence, summarizing with Proposition 2, we have

$$R_X(\hat{\beta}; \beta) \to \begin{cases} \sigma^2 \frac{\gamma}{1-\gamma} & \text{for } \gamma < 1, \\ r^2\big(1 - \frac{1}{\gamma}\big) + \sigma^2 \frac{1}{\gamma - 1} & \text{for } \gamma > 1. \end{cases} \tag{8}$$

Isotropic features (Section 3)



UPPSALA UNIVERSITET

NOTE!

We will only cover sections 1, 2 and 3 of this paper (first 11 pages)!

Background

Wigner matrix and semicircle distribution

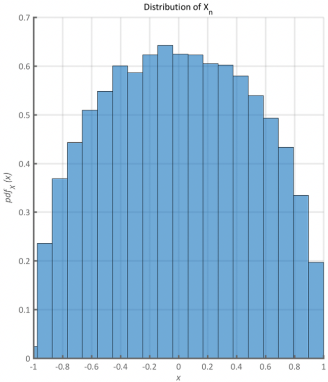
- Assume a symmetric matrix: $(W \sim W^\mathsf{T})$

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} \\ w_{2,1} & w_{2,2} & w_{2,3} \\ w_{3,1} & w_{3,2} & w_{3,3} \end{bmatrix}$$

  with the lower diagonal entries i.i.d. sampled from a random distribution.
- What can be said about the eigenvalues of such a matrix.
- The idea was introduced by Eugene Wigner when working with heavy nuclei atoms.

E. P. Wigner. Characteristic vectors of bordered matrices with infinite dimensions. Annals Math., 62:548–564, 1955.

Distribution of eigenvalues.



Wigner PDF (Kris Buchanan). License: CC BY-SA 4.0.
`en.wikipedia.org/wiki/Wigner_semicircle_distribution`

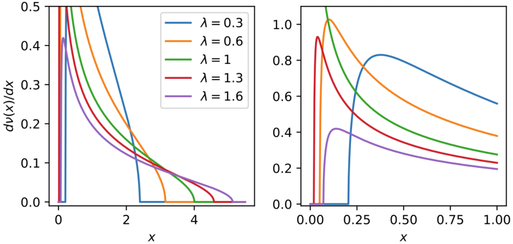
Wishart matrix and the Marchenko–Pastur distribution

- Let now A be

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \end{bmatrix}$$

  with the lower diagonal entries i.i.d. sampled from a random distribution.
- What is the distribution of the eigen values of:

$$A^{\mathsf{T}} A?$$



Marchenko-Pastur distribution (Mario Geiger). License: CC BY-SA 4.0.
`en.wikipedia.org/wiki/Marchenko-Pastur_distribution`

Random matrix theory

References:

- G. W. Anderson, A. Guionnet, and O. Zeitouni. *An Introduction to Random Matrices*, 2009.
- Z. Bai and J. W. Silverstein. *Spectral analysis of large dimensional random matrices*, volume 20 of *Springer Series in Statistics*. Springer, 2010
- T. Tao. *Topics in random matrix theory*, volume 132 of *Graduate Studies in Mathematics*. American Mathematical Society, 2012.

Other sources:

- ICML 2021 tutorial: https://random-matrix-learning.github.io/