Inmas Machine Learning Workshop 2023

Internship Network in the Mathematical Sciences

Christian Kümmerle, January 14-15, 2023

Learning Goals

- Gain intuition about what are fundamental problems and concepts to **learn from data**
- Exposure to some **popular models and computational tools** to solve machine learning problems
- Learn about **different** data **types**
- Gain intuition about **challenges & peculiarities** of **high-dimensional** data
- Learn how to use **Python** popular packages to **apply** techniques

Schedule

Today, Saturday, Jan 14

- **10:00 AM - 1:00 PM ET (9:00 AM - 12:00 PM CT):**

  Framework of Statistical Learning, Regularization, High-Dimensional Data

- **90 minutes lunch break**

- **2:30 PM - 5:30 PM ET (1:30 PM - 4:30 PM CT):**

  Classification Problems, Natural Language Processing

Structure of Workshop:

~ <=1 h per Session: Presentation
~ >=2 h per Session: Work in Groups of 5-6
on Python Jupyter Notebooks

Schedule

Tomorrow, Sunday, Jan 15

- **10:00 AM - 1:00 PM ET (9:00 AM - 12:00 PM CT):**

  - Unsupervised Learning: Principal Component Analysis, Clustering

- **90 minutes lunch break**

- **2:30 PM - 5:30 PM ET (1:30 PM - 4:30 PM CT):**
  - **Short presentation:** A Testimonial of an Industrial Internship
  - Neural Networks and Deep Learning

A bit about myself

ckuemmerle.com

Current Position:

- Assistant Professor in Computer Science at University of North Carolina at Charlotte since 2022

Background:

- Ph.D. in Mathematics (Technical University of Munich, Germany)
- Postdoc at Johns Hopkins 2020-2022

Research Interests:

- **Make machine learning & AI more powerful, more resource-efficient, more data-efficient**

Optimization for machine learning, development of scalable algorithms, few-shot learning, recommender systems, high-dimensional probability

Our TA Team

- Emily Shinkle (Illinois)
- Yuxuan Li (Illinois)
- Derek Kielty (Illinois)
- Yashil Sukurdeep (JHU)
- Tim Wang (JHU)
- Ben Brindle (JHU)

What is Data Science?

- [Tukey '62 "The Future of Data Analysis]:
  "Data Analysis" as an **empirical science**:

  - Procedures for gathering data, for interpreting data
  - Uses mathematical statistics
  - **"reliance upon the test of experience as ultimate standard of validity"**

  Focus in this workshop: Prediction instead of Inference

Why has Data Science/ ML become so big?

In last 15-20 Years: Massive technological advances in
Pattern & image recognition, machine translation,
targeted advertisement, semiautonomous cars

- [Liberman 2010; Donoho 2015]: One crucial ingredient:
  Common Task Framework

  - Public “training” data set: List of observations with labels
  - Competitive participants with **common task** to infer label
  prediction rule from training data, **submit to**
  -> Referee mechanism which reports accuracy of prediction rule
  when applied to (hidden) test dataset.

  Examples:
  - $1M Netflix Prize: Recommender Systems
  - ImageNet: Image Classification

What is Machine Learning?

Tom Mitchell (CMU), 1997:

“A computer program is said to ***learn from experience E*** *with respect to some* ***class of tasks T****, and* ***performance measure P****, if its performance at tasks in T, as measured by P, improves with experience E.”*

Goal: Learn from data

a) **Supervised Learning**

Ex.: - Detect spam based on large set of spam/non-spam emails

- Predict salary of professor based on employment data

b) **Unsupervised Learning**

"Learning without teacher", find meaningful data representation / summary

Ex.: - Find categories among pictures on phone

- Visualize complex data

The Framework of Statistical Learning

- $\mathcal{X} \subset \mathbb{R}^k$: domain set (e.g. space of images with a certain number of pixels)
- $\mathcal{Y} \subset \mathbb{R}^q$: ~~target~~ set / set of labels (e.g. $\mathcal{Y} \subseteq \{0, 1\}$)
- let $\mathcal{D}$ be a probability distribution on $\mathcal{X} \times \mathcal{Y}$

<u>Assume:</u> Given ~~training~~ set $S := (x_i, y_i)_{i=1}^{n} \overset{i.i.d.}{\sim} \mathcal{D}$

<u>Goal:</u> Find a **predictor / classifier** $h: \mathcal{X} \to \mathcal{Y}$

that minimizes the expected risk $L_{\mathcal{D}}(h) = \mathbb{E}_{\mathcal{D}}[\ell(y, h(x))]$

where $\ell: \mathcal{Y} \times \mathcal{Y} \to \mathbb{Z}$ is a given loss/error function

<u>Learning Algorithm:</u> Specific algorithm that maps S to a specific member function $h_n \in \mathcal{F}$ of a **hypothesis space $\mathcal{F}$** $\subseteq \{h: \mathcal{X} \to \mathcal{Y}\}$

based on information of S.

Ex: For ℓ: $\ell(y, \tilde{y}) = \frac{1}{2}(y - \tilde{y})^2$

Ex for Learning Algorithm:

Empirical Risk Minimization:

$$\hat{h}_n = \underset{h \in \mathcal{F}}{\operatorname{argmin}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i)) \right\}$$

Ex: · Linear Regression:

- $\ell(y, z) = \frac{1}{2}(y - z)^2$
- $\mathcal{F} := \{ x \mapsto \beta_0 + \langle x, \beta \rangle, \beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^k \}$

Bias-Variance Tradeoff:

$$\underbrace{L_{\mathcal{D}}(\hat{h}_n) - \min_h (L_{\mathcal{D}}(h))}_{\text{generalization error}} = \underbrace{(L_{\mathcal{D}}(\hat{h}_n) - L_{\mathcal{D}}(h_{\mathcal{F}}))}_{\text{estimation error}} + \underbrace{(L_{\mathcal{D}}(h_{\mathcal{F}}) - \min_h L_{\mathcal{D}}(h))}_{\text{approximation error}}$$

$h_{\mathcal{F}} = \underset{h \in \mathcal{F}}{\operatorname{argmin}}\, L_{\mathcal{D}}(h)$

- estimation error: for fixed sample size $|S|$, larger if $\mathcal{F}$ large
- approximation error: smaller if $\mathcal{F}$ large

Ridge Regression: $R(h) := \|\beta\|_2^2$ $\lambda > 0$

- $\mathcal{F}$: space of linear functions

$$\hat{\beta}_n = \beta(\hat{h}_n) = \underset{\beta \in \mathbb{R}^k}{\operatorname{argmin}} \left\{ \frac{1}{n} \|X\beta - y\|_2^2 + \lambda \|\beta\|_2^2 \right\}$$

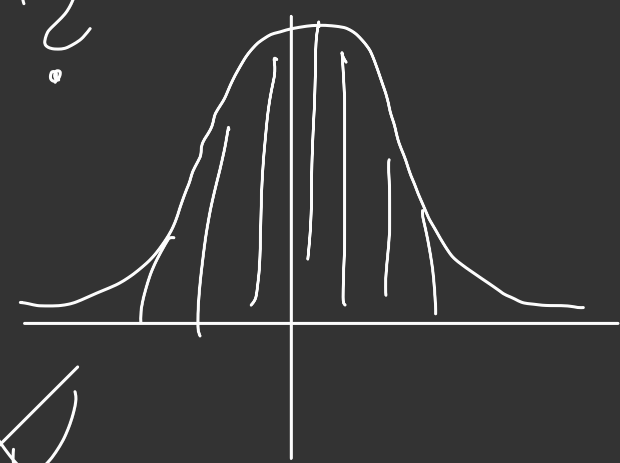
$$= (X^T X + \lambda I)^{-1} X^T (y)$$

- If $\lambda = 0$: $\Longrightarrow$ linear regression
- If $\lambda \to \infty$: coeffecient $\hat{\beta}_n \longrightarrow 0$
- If λ in between: balance fit of linear model and size of coeffecient

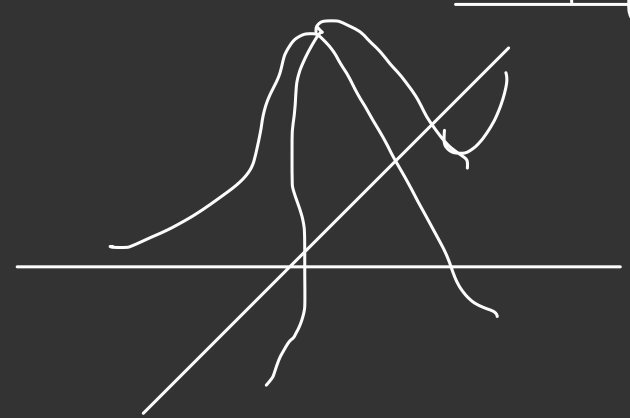
High Dimensional Geometry

Q: How do $x_1, x_2 \in \mathbb{R}^d$, $d >> 1$ relate to each other when "generic"?

- 1-dim Gaussian:



d=2



- d-dim Gaussian:

$$\|X\| \sim \sqrt{d} \pm O(1)$$

$\Rightarrow$ All very far from origin!

- If $x_1, x_2 \in \mathbb{R}^d$ indep. d-dim Gaussion

$\Rightarrow \|x_1 - x_2\|_2 \sim$ close to const. & large.

<u>Preprocessing:</u> If domain set s.t. $X \subset \mathbb{R}^k$, we can define a feature map $\phi : X \to \tilde{X} \subset \mathbb{R}^\ell$ (often with $k \ll \ell$) such that $S = (\phi(x_i), y_i)_{i=1}^n$ is used as training set.

<u>Ex:</u> Polynomial features. E.g, if $k=1$, $\ell=5$:

$$\phi(x) = (x, x^2, x^3, x^4, x^5).$$

$\longrightarrow$ Often <u>improves</u> expressive power of a learning model!

2. Sparse Regression

- If features are designed to "explain" the target variable as a linear combination of few features (e.g., $k \ll n$), we can use

$$\mathcal{F}_k^{sparse} := \{ h : \mathbb{R} \longrightarrow \mathbb{R} : h(x) = \langle \beta, x \rangle \quad \text{s.t. } \|\beta\|_0 \le k \},$$

where $\|\beta\|_0 = \sum_{i=1}^{n} \mathbb{1}_{\{|\beta_i| \neq 0\}}$ is the number of non-zero coefficients of β.

- Problem: ERM on $\mathcal{F}_k^{sparse}$ is NP-hard

  ⟶ computational challenges!

- Possible approach: Lasso Regression:

$$\hat{\beta}_n = \underset{\beta \in \mathbb{R}^2}{\arg\min} \; \|A\beta - y\|_2^2 + \lambda \|\beta\|_1 \qquad (*)$$

- Unlike linear/ridge regression, (*) has no closed form solution, but convex optimization problem ⟵ well-established theory/methods exist.

- With respect to original class $\mathcal{F}_k^{sparse}$:

Generalization error = Optimization error + estimation error + approximation error