The Learning Problem

Fabio A. González

Machine Learning 2024-1

What is a machine learning model?

- A function (Linear function, neural network, etc)

- A probability distribution.
  - Mixture of Gaussians distribution
  - Naive Bayes

- An algorithm
  - Decision tree
  - K-nearest neighbor classifier

Models as Functions

Assume a supervised learning model

Predictor: $f: X \longrightarrow Y$ $\qquad f: \mathbb{R}^n \longrightarrow \{0, 1\}$

Hypothesis space: Set of functions from where the predictor is chosen

Example: Linear regression

$$H = \{ f_{w,b}(x) = wx + b \mid w \in \mathbb{R}^n, \, b \in \mathbb{R} \}$$

$\theta = \{w, b\} \longrightarrow$ Parameters that determine f_θ

Learning: finding θ $\qquad$ how to find it?

Models as probability distributions

- Model the probability distribution of the data
  - $P(x, y)$
  - $P(y | x)$
- Use the probability to make a decision

e.g.

$$f(x) = \begin{cases} 1 & \text{if } P(y=1 | x) \geq 1/2 \\ 0 & \text{otherwise} \end{cases}$$

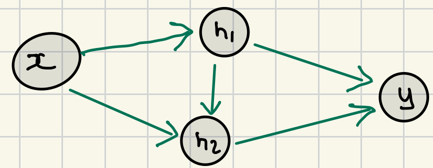
Types of probabilistic models

- Parametric estimation $P(x, y) = D_\theta$ distribution with parameters θ

- Bayesian estimation $\theta \sim D'_{\theta'}$

  Use Bayesian inference to estimate θ, θ'

- Graphical



- Non-parametric

  Model the probability distribution using the data

Models based on algorithms

- Decision trees

  Hierarchical model
  Information gain

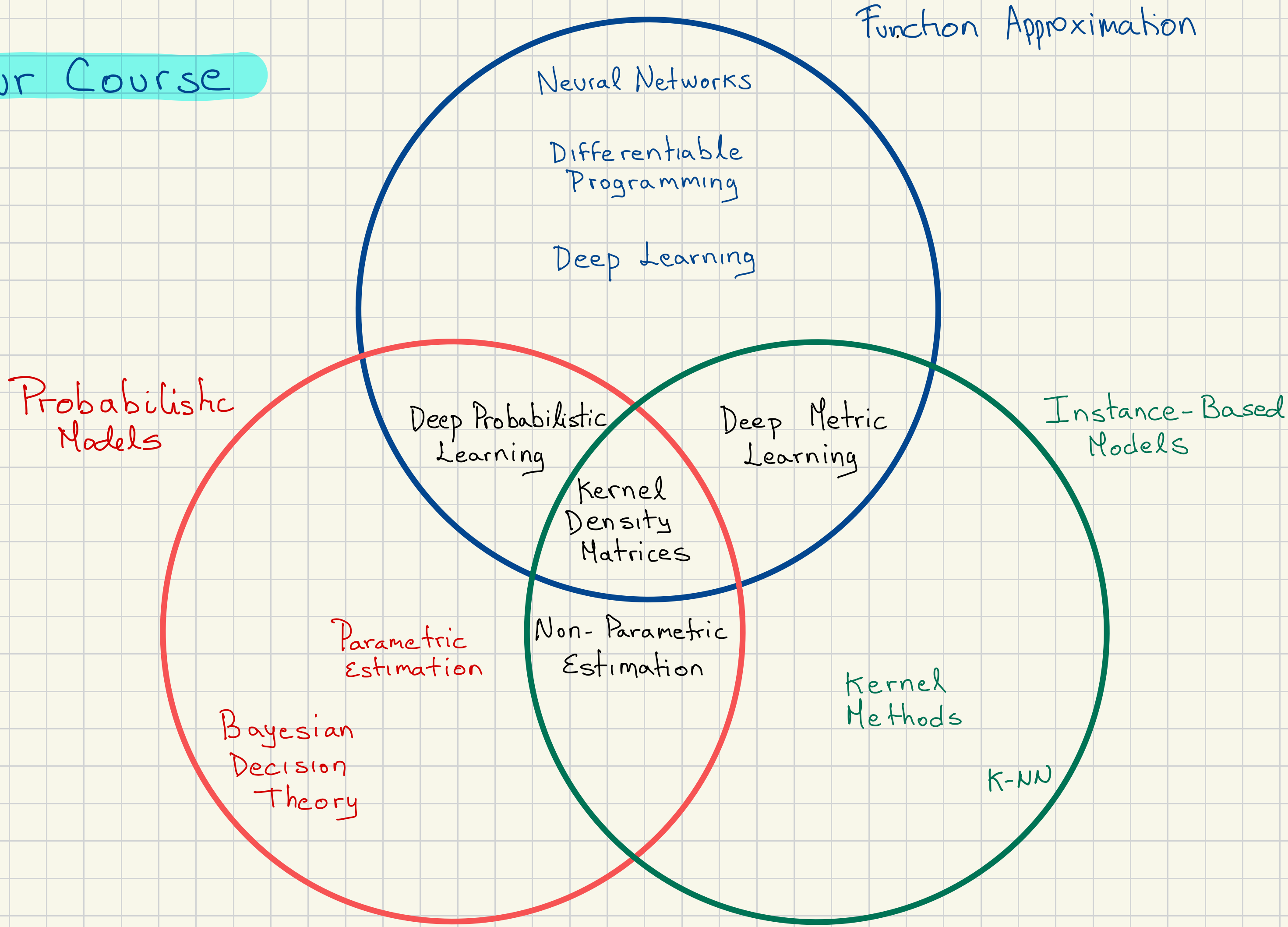
- Random Forest

  Ensamble model {Bagging, Boosting}

- K-nearest neighbors

  Instance based learning

Our Course



Function Approximation
Neural Networks
Differentiable Programming
Deep Learning
Probabilistic Models
Deep Probabilistic Learning
Kernel Density Matrices
Deep Metric Learning
Instance-Based Models
Parametric Estimation
Non-Parametric Estimation
Bayesian Decision Theory
Kernel Methods
K-NN

Empirical Risk and True risk

Training data set: $D = \{(x_i, y_i)\}_{i=1..N} \quad x_i \in X, \; y_i \in Y$

Empirical Risk: $R_{emp}(f_\theta, D) = \frac{1}{N}\sum_{i=1}^{N} l(y_i, f_\theta(x_i))$

Loss function: $l(y_i, \hat{y}_i) = \begin{cases} \sim 0 & \text{if } y_i \sim \hat{y}_i \\ \text{high value} & \text{otherwise} \end{cases}$

True Risk: $R(f_\theta) = E_{x,y}[l(y, f_\theta(x))]$

Minimizing the empirical risk.

$$\theta^* = \arg\min_{\theta} R_{emp}(f_\theta, D)$$

f_{θ^*} best predictor on training data.

$$\theta^*_{True} = \arg\min_{\theta} R(f_\theta)$$

$f_{\theta^*_{True}}$ best predictor for all possible data $\begin{cases} \text{seen} \\ \text{unseen} \end{cases}$

Is it really the BEST predictor?

yes, if the Hypothesis space include all possible function

Bayes optimal predictor (BOP)

Assumption: We know $P(x,y) \longrightarrow P(y|x)$

$$f_{Bayes}(x) = \begin{cases} 1 & \text{if } P(y=1|x) \geq 1/2 \\ 0 & \text{otherwise} \end{cases}$$

Optimality: $\forall g : x \longrightarrow y$

$$\forall_{x,y} \quad P(g(x) = y \mid x) \leq P(f_{Bayes}(x) = y \mid x)$$

$$R(g(x)) \geq R(f_{Bayes})$$

with $\ell(y, \hat{y}) = \mathbb{1}(y \neq \hat{y})$

Overfitting

The model adjusts too much the training data

$$R_{emp} \downarrow \quad vs \quad R \uparrow$$

How to prevent it?

- Control $R(f)$

Problem: I cannot calculate it

Solution: Estimate it $\longrightarrow$ Cross Validation

$$R(f) \approx \hat{R}(f) = R_{emp}(f, Test)$$ Test: Test dataset

- Regularization

Regularization

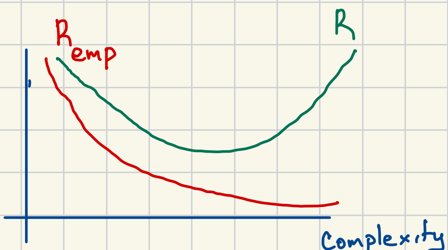
Measure of complexity

Add a penalization term to R_{emp}.

$$\theta^* = \arg\min_{\theta} R_{emp}(f_\theta, D) + \alpha \underbrace{Reg(\theta)}_{\text{Regularization term.}}$$

Example: regularized least squares

$$w^* = \arg\min_{w} \frac{1}{N} \sum_{i=1}^{N} (f_w(x_i) - y_i)^2 + \alpha \|w\|^2$$



Lasso $\alpha \|w\|_1$

$$\|w\|_1 = \sum_{i=1}^{d} |w_i|$$

Vapnik-Chervonenkis Bound

$$R(\alpha) \leq R_{emp}(\alpha) + \sqrt{\left(\frac{h(\log(2l/h)+1) - \log(\eta/4)}{l}\right)}$$

References