Artificial Intelligence and Machine Learning

Barbara Caputo

EXAM MODALITY -FINAL

This information substitutes ANY OTHER PRIOR INFORMATION, written or oral

Homeworks (lab experiences) +written exam

EXAM MODALITY -FINAL

Homeworks:

-Homeworks during the lectures: 3, on three key topics (PCA, svm, cnn). Reports to be submitted with the following schedule:

-Exam 28/01/2019 → deadline for report 19/01/2019
-Exam 19/02/2019 → deadline for report 10/02/2019

- Every experience graded (2 points), if overall points <4 no admission to written exam.

- The vote of the Homeworks does not sum with the vote of the written exam

-

EXAM MODALITY -FINAL

- Once you pass the Homeworks, the pass holds for all the four exam sessions

- How to submit: send your report by email to
  barbara.caputo@polito.it
  paolo.russo@iit.it

-In the subject please write
'AIML Homeworks Report, Name_Surname'

- How you will know your grade: personal email within 3 working days by receipt + file posted on Google Drive within 5 working days from official deadline

EXAM MODALITY -FINAL

Written exam:

- 6 questions, three exercises and three theory questions

- each question will consist of two parts, one easier and one more advanced.

- The easier part is graded up to 3 points. The more difficult part is also graded up to 3 points.

- To pass the exam you must have minimum 18, minimum 3 points for each question

EXAM MODALITY -FINAL

- The maximum grade is 30/30 cum laude: you get a 'laude' from 34 (included)

- The exam grade will be announced with a file in the Google Drive folder

-You can ask for an oral exam, that will give you the chance to change the grade of max 3 points –**up or down**

- When the exam grades are announced, it will also be given a window of time for the oral exams (usually 2-3 days, usually max 1 week after the announcement

EXAM –EXAMPLE OF THEORY QUESTION

Define the perceptron and discuss its properties. Is it a discriminative or generative method?
State and demonstrate the convergence theorem for the perceptron.

EXAM –EXAMPLE OF EXERCISE

Given the points ….., belonging to …. classes, determine to which class the queries …. belong, using an L2 norm.

How would the result change using an exponential kernel? Explicitly derive the new classification algorithm, compute the new results and discuss them.

10. Risk Minimization

What have we seen so far?

Several algorithms that seem to work fine on training datasets:

- Naïve Bayes classifier
- Perceptron
- Support Vector Machines

What have we seen so far?

Several algorithms that seem to work fine on training datasets:

- Naïve Bayes classifier
- Perceptron
- Support Vector Machines

❑How good are these algorithms on unknown test sets?
❑How many training samples do we need to achieve small error?
❑What is the smallest possible error we can achieve?

What have we seen so far?

Several algorithms that seem to work fine on training datasets:

- Naïve Bayes classifier
- Perceptron
- Support Vector Machines

❑How good are these algorithms on unknown test sets?
❑How many training samples do we need to achieve small error?
❑What is the smallest possible error we can achieve?

$\Rightarrow$ Learning Theory

Outline

- Risk and loss
  - Loss functions
  - Risk
  - Empirical risk vs True risk
  - Empirical Risk minimization

- Underfitting and Overfitting

Supervised Learning Setup

Supervised Learning Setup

$\mathcal{D} = \{(X_1, Y_1), \ldots (X_n, Y_n)\}$ training data

$\{(X_{n+1}, Y_{n+1}), \ldots (X_m, Y_m)\}$ test data

Supervised Learning Setup

$\mathcal{D} = \{(X_1, Y_1), \ldots (X_n, Y_n)\}$ training data

$\{(X_{n+1}, Y_{n+1}), \ldots (X_m, Y_m)\}$ test data

Features: $X \in \mathcal{X} \subset \mathbb{R}^d$

Labels: $Y \in \mathcal{Y} \subset \mathbb{R}$

Supervised Learning Setup

$\mathcal{D} = \{(X_1, Y_1), \ldots (X_n, Y_n)\}$ training data

$\{(X_{n+1}, Y_{n+1}), \ldots (X_m, Y_m)\}$ test data

Features: $X \in \mathcal{X} \subset \mathbb{R}^d$

Labels: $Y \in \mathcal{Y} \subset \mathbb{R}$

Generative model of the data: $X \sim \mu,\ \mu(A) = \Pr(X \in A)$

(train and test data) $Y \sim p(\cdot|X)$

Supervised Learning Setup

$\mathcal{D} = \{(X_1, Y_1), \ldots (X_n, Y_n)\}$ training data

$\{(X_{n+1}, Y_{n+1}), \ldots (X_m, Y_m)\}$ test data

Features: $X \in \mathcal{X} \subset \mathbb{R}^d$

Labels: $Y \in \mathcal{Y} \subset \mathbb{R}$

Generative model of the data: $X \sim \mu,\ \mu(A) = \Pr(X \in A)$

(train and test data) $Y \sim p(\cdot|X)$

Classification: Labels: $\mathcal{Y} = \{0, 1\}$

Loss

$\mathcal{D} = \{(X_1, Y_1), \ldots (X_n, Y_n)\}$ training data

$\{(X_{n+1}, Y_{n+1}), \ldots (X_m, Y_m)\}$ test data

Loss

$\mathcal{D} = \{(X_1, Y_1), \ldots (X_n, Y_n)\}$ training data

$\{(X_{n+1}, Y_{n+1}), \ldots (X_m, Y_m)\}$ test data

Loss function: $L(x, y, f(x))$

where $L : \mathcal{X} \times \mathcal{Y} \times \mathbb{R} \to [0, \infty]$

It measures how good we are on a particular (x,y) pair.

Loss

$\mathcal{D} = \{(X_1, Y_1), \ldots (X_n, Y_n)\}$ training data

$\{(X_{n+1}, Y_{n+1}), \ldots (X_m, Y_m)\}$ test data

Loss function: $L(x, y, f(x))$

where $L : \mathcal{X} \times \mathcal{Y} \times \mathbb{R} \to [0, \infty]$

It measures how good we are on a particular (x,y) pair.

We want the loss $L(X_t, Y_t, f(X_t))$ to be small for many (X_t, Y_t) pairs in the test data.

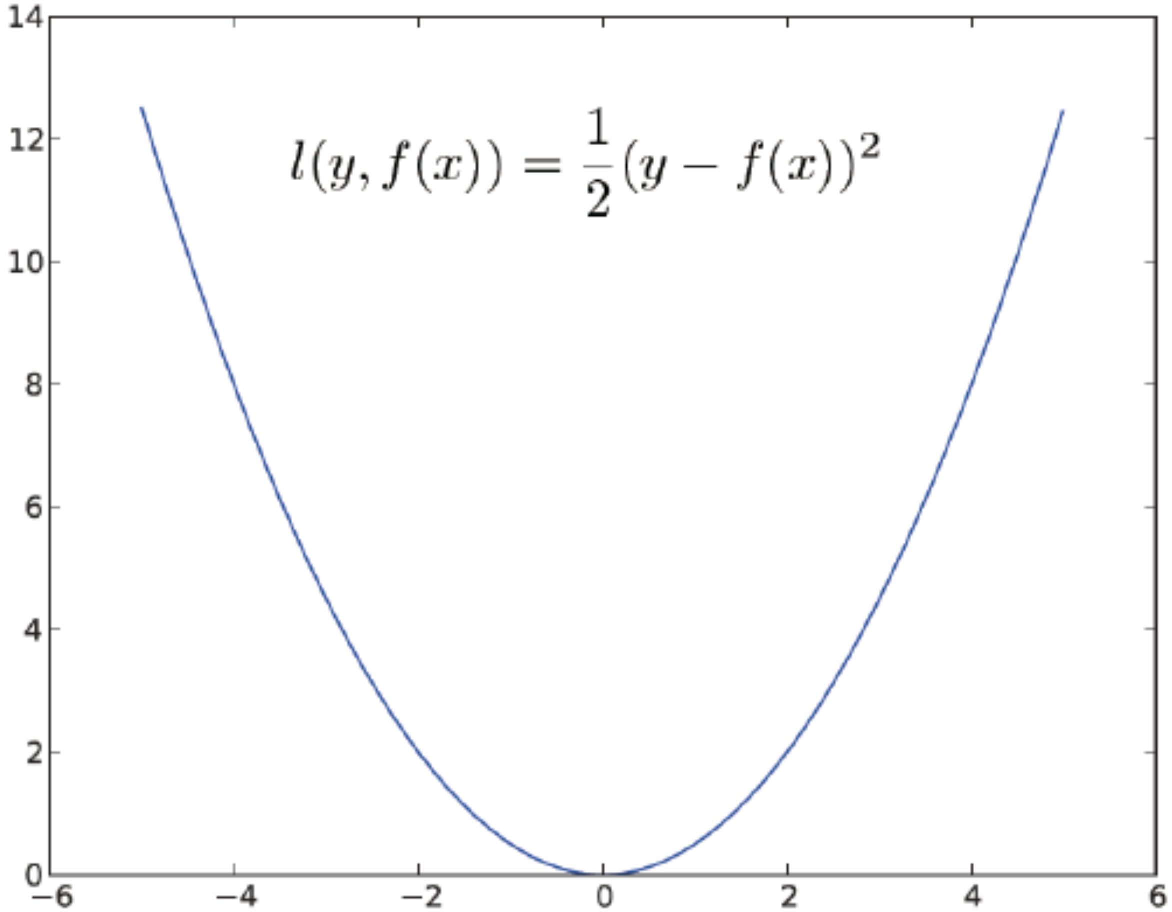
Loss Examples

Loss Examples

Classification loss:

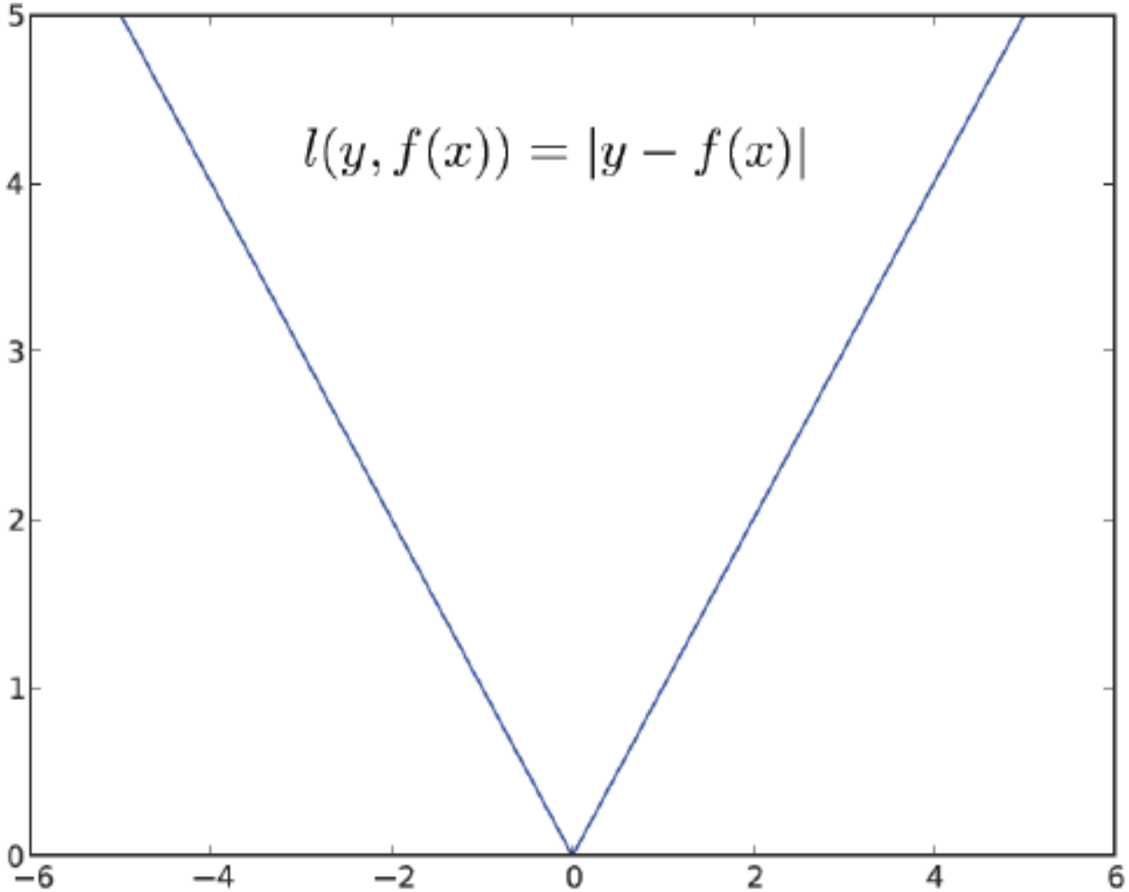
$$L(x, y, f(x)) = \begin{cases} 1 & y \neq f(x) \\ 0 & y = f(x) \end{cases}$$

Squared loss, L_2 loss



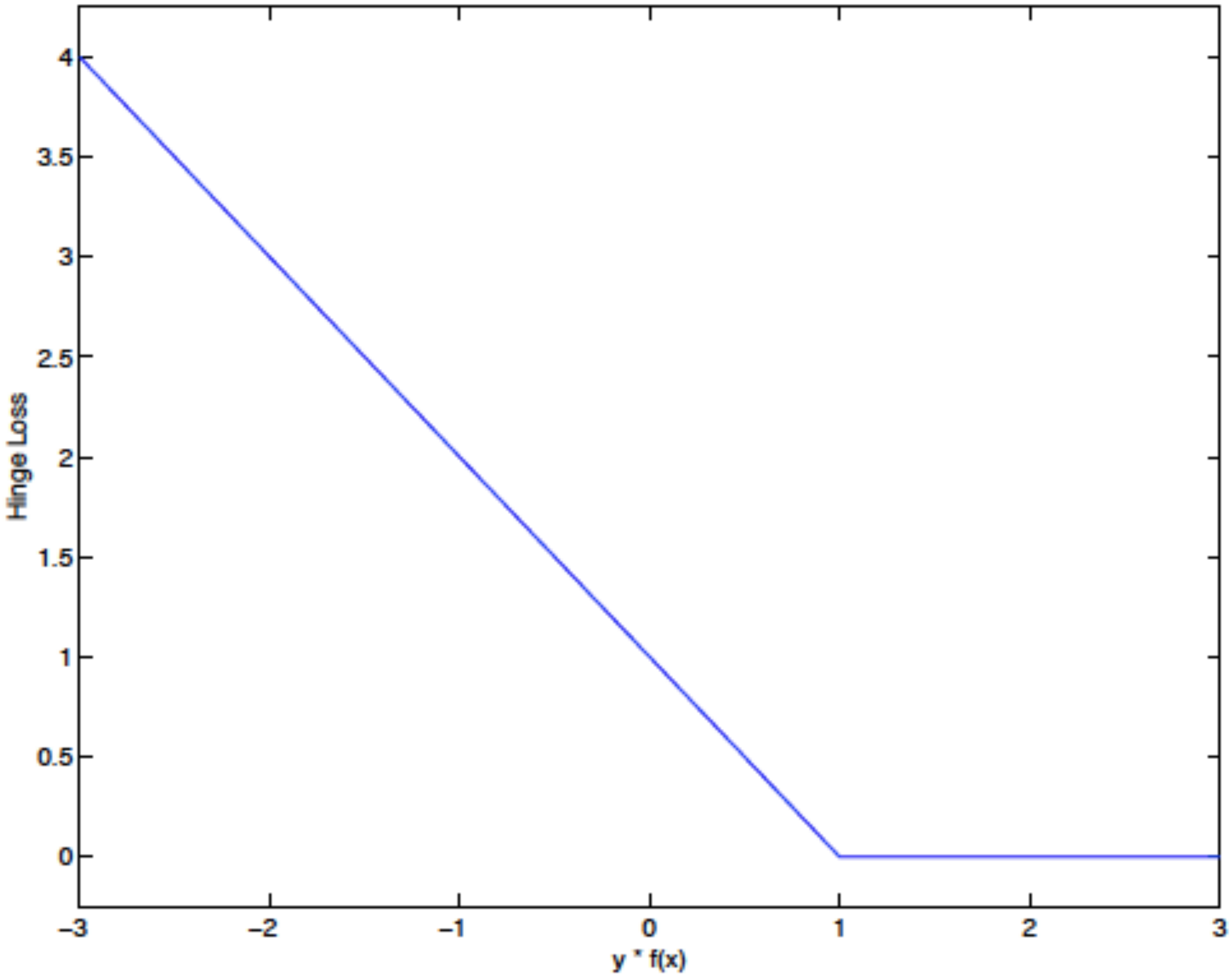
Picture form Alex

L_1 loss



Picture form Alex

The Hinge Loss



Risk

Risk

Risk of *f* classification/regression function:

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y) \quad \text{= The expected loss}$$

$$= \mathbb{E}[L(X,Y,f(X))]$$

$p(y,x)dydx$

Risk

Risk of *f* classification/regression function:

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x, y, f(x))dP(x, y) \quad \text{= The expected loss}$$
$$= \mathbb{E}[L(X, Y, f(X))]$$

$p(y, x)dydx$

$L(x, y, f(x))$: Loss function

$P(x, y)$: Distribution of the data.

Why do we care about this?

Why do we care about risk?

Risk of *f* classification/regression function:

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x, y, f(x))\,dP(x, y) \quad \text{=The expected loss}$$

$$= \mathbb{E}[L(X, Y, f(X))]$$

$p(y, x)dydx$

Why do we care about risk?

Risk of *f* classification/regression function:

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x, y, f(x))dP(x, y)$$

$$= \mathbb{E}[L(X, Y, f(X))]$$

=The expected loss

$p(y, x)dydx$

Our true goal is to minimize the loss of the test points!

$$f^* = \arg\min_f \frac{1}{m-n} \sum_{i=n+1}^{m} L(X_i, Y_i, f(X_i))$$

Why do we care about risk?

Risk of *f* classification/regression function:

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y) \quad \text{=The expected loss}$$
$$= \mathbb{E}[L(X,Y,f(X))]$$

$p(y,x)dydx$

Our true goal is to minimize the loss of the test points!

$$f^* = \arg\min_f \frac{1}{m-n}\sum_{i=n+1}^{m} L(X_i, Y_i, f(X_i))$$

Usually we don't know the test points and their labels in advance..., but

$$\frac{1}{m-n}\sum_{i=n+1}^{m} L(X_i, Y_i, f(X_i)) \xrightarrow{m\to\infty} R_{L,P}(f) \quad \text{(LLN)}$$

That is why our goal is to minimize the risk.

Risk Examples

Risk Examples

Risk: $R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x, y, f(x))dP(x, y)$ The expected loss

Risk Examples

Risk: $R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y)$ The expected loss

Classification loss: $L(x,y,f(x)) = \begin{cases} 1 & y \neq f(x) \\ 0 & y = f(x) \end{cases}$

Risk of classification loss:

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y) = \mathbb{E}[\mathbf{1}_{\{f(X)\neq Y\}}] = \Pr(f(X) \neq Y)$$

Bayes Risk

Bayes Risk

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y)$$

The expected loss

Bayes Risk

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y)$$ The expected loss

Definition: Bayes Risk

$$R^*_{L,P} = \inf_{f:\mathcal{X}\to\mathbb{R}} \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y)$$

We consider all possible function f here

Bayes Risk

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x, y, f(x))dP(x, y)$$ The expected loss

Definition: Bayes Risk

$$R^*_{L,P} = \inf_{f:\mathcal{X}\to\mathbb{R}} \int_{\mathcal{X}\times\mathcal{Y}} L(x, y, f(x))dP(x, y)$$

We consider all possible function f here

We don't know P, but we have i.i.d. training data sampled from P!

Bayes Risk

$$R_{L,P}(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y)$$ The expected loss

Definition: Bayes Risk

$$R^*_{L,P} = \inf_{f:\mathcal{X}\to\mathbb{R}} \int_{\mathcal{X}\times\mathcal{Y}} L(x,y,f(x))dP(x,y)$$

We consider all possible function f here

We don't know P, but we have i.i.d. training data sampled from P!

Goal of Learning:

Build a function f_D (using data D) whose risk $R_{L,P}(f_D)$ will be close to the Bayes risk $R^*_{L,P}$

The learning algorithm constructs this function f_D from the training data.

Consistency of learning methods

Consistency of learning methods

Risk is a random variable: $R_{L,P}(f_D) = \mathbb{E}[L(X,Y,f_D(X)|D]$

Consistency of learning methods

Risk is a random variable: $R_{L,P}(f_D) = \mathbb{E}[L(X, Y, f_D(X)|D]$

Definition:

A learning method is **univerally consistent** if **for all** $P(X, Y)$ distributions the risk converges to the Bayes risk when we increas the sample size

$$R_{L,P}(f_D) \xrightarrow{p} R^*_{L,P} \text{ as } n \to \infty.$$

Consistency of learning methods

Risk is a random variable: $R_{L,P}(f_D) = \mathbb{E}[L(X,Y,f_D(X)|D]$

Definition:
A learning method is **univerally consistent** if **for all** $P(X,Y)$ distributions the risk converges to the Bayes risk when we increas the sample size

$$R_{L,P}(f_D) \xrightarrow{p} R^*_{L,P} \text{ as } n \to \infty.$$

Stone's theorem 1977: Many classification, regression algorithms are universally consistent for certain loss functions under certain conditions: kNN, Parzen kernel regression, SVM,…

Yayyy!!! ☺

Consistency of learning methods

Risk is a random variable: $R_{L,P}(f_D) = \mathbb{E}[L(X, Y, f_D(X)|D]$

Definition:

A learning method is **univerally consistent** if **for all** $P(X, Y)$ distributions the risk converges to the Bayes risk when we increas the sample size

$$R_{L,P}(f_D) \xrightarrow{p} R^*_{L,P} \text{ as } n \to \infty.$$

Stone's theorem 1977: Many classification, regression algorithms are universally consistent for certain loss functions under certain conditions: kNN, Parzen kernel regression, SVM,…

Yayyy!!! ☺

Wait! This doesn't tell us anything about the rates…

No Free Lunch!

No Free Lunch!

Devroy 1982: For every consistent learning method and for every fixed convergence rate a_n, $\exists$ P(X,Y) distribution such that the convergence rate of this learning method on P(X,Y) distributed data is slower than a_n

No Free Lunch!

Devroy 1982: For every consistent learning method and for every fixed convergence rate a_n, ∃ P(X,Y) distribution such that the convergence rate of this learning method on P(X,Y) distributed data is slower than a_n

$$\boxed{R_{L,P}(f_D) \xrightarrow{p} R^*_{L,P} \text{ as } n \to \infty \text{ with slower rate than } a_n}$$

☹

What can we do now?

5 minutes break

Empirical Risk and True Risk

Empirical Risk

Empirical Risk

For simplicity, let $L(x, y, f(x)) = L(y, f(x))$

Shorthand:

True risk of f (deterministic): $R(f) = R_{L,P}(f) = \mathbb{E}[L(Y, f(X))]$

Bayes risk: $R^* = R^*_{L,P} = \inf_{f:\mathcal{X}\to\mathbb{R}} R(f)$

Empirical Risk

For simplicity, let $L(x, y, f(x)) = L(y, f(x))$

Shorthand:

True risk of f (deterministic): $R(f) = R_{L,P}(f) = \mathbb{E}[L(Y, f(X))]$

Bayes risk: $R^* = R^*_{L,P} = \inf_{f:\mathcal{X}\to\mathbb{R}} R(f)$

We dont know P, and hence we don't know $R(f)$ either.

Let us use the empirical counter part:

Empirical risk: $\widehat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i))$

Empirical Risk Minimization

Empirical Risk Minimization

$$R(f) = R_{L,P}(f) = \mathbb{E}[L(Y, f(X)] \qquad R^* = R^*_{L,P} = \inf_{f:\mathcal{X}\to\mathbb{R}} R(f)$$

$$\widehat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i))$$

Empirical Risk Minimization

$$R(f) = R_{L,P}(f) = \mathbb{E}[L(Y, f(X)]$$

$$R^* = R^*_{L,P} = \inf_{f:\mathcal{X}\to\mathbb{R}} R(f)$$

$$\widehat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i))$$

Law of Large Numbers:

For each fixed f, $\widehat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i)) \xrightarrow{n\to\infty} R(f)$

Empirical risk is converging to the Bayes risk

Empirical Risk Minimization

$$R(f) = R_{L,P}(f) = \mathbb{E}[L(Y, f(X)] \qquad R^* = R^*_{L,P} = \inf_{f:\mathcal{X}\to\mathbb{R}} R(f)$$

$$\widehat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i))$$

Law of Large Numbers:

For each fixed f, $\widehat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i)) \xrightarrow{n\to\infty} R(f)$

Empirical risk is converging to the Bayes risk

We need $\inf_{f:\mathcal{X}\to\mathbb{R}} R(f)$, so let us calculate $\inf_{f:\mathcal{X}\to\mathbb{R}} \widehat{R}_n(f)$!

$$\inf_{f:\mathcal{X}\to\mathbb{R}} \widehat{R}_n(f) = \inf_{f:\mathcal{X}\to\mathbb{R}} \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i))$$

This is a **terrible idea** to optimize over all possible $f : \mathcal{X} \to \mathbb{R}$ functions! [Extreme overfitting]

Solutions to Overfitting

Solutions to Overfitting

Terrible idea to optimize over all possible $f : \mathcal{X} \to \mathbb{R}$ functions! [Extreme overfitting]

$\Rightarrow$ minimze over a smaller function set $\mathcal{F}$.

Solutions to Overfitting

Terrible idea to optimize over all possible $f : \mathcal{X} \to \mathbb{R}$ functions! [Extreme overfitting]

$\Rightarrow$ minimze over a smaller function set $\mathcal{F}$.

Empirical risk minimization over the function set $\mathcal{F}$.

$$\boxed{f_n^* = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))}$$

Solutions to Overfitting
Structural Risk Minimization

Solutions to Overfitting
Structural Risk Minimization

Empirical risk minimization over the function set $\mathcal{F}$.

$$\boxed{f_n^* = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))}$$

Solutions to Overfitting
Structural Risk Minimization

Empirical risk minimization over the function set $\mathcal{F}$.

$$f_n^* = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$$

Notation: $R_{\mathcal{F}}^* = \inf_{f \in \mathcal{F}} \mathbb{E}[L(Y, f(X))]$ $\widehat{R}_{n,\mathcal{F}}^* = \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$

Risk

Empirical risk

Solutions to Overfitting
Structural Risk Minimization

Empirical risk minimization over the function set $\mathcal{F}$.

$$f_n^* = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$$

Notation: $R_{\mathcal{F}}^* = \inf_{f \in \mathcal{F}} \mathbb{E}[L(Y, f(X))]$ $\widehat{R}_{n,\mathcal{F}}^* = \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$

Risk Empirical risk

1st issue: $R_{\mathcal{F}}^* - R^* \geq 0$ needs to be small.

(Model error, Approximation error)

Risk in $\mathcal{F}$ - Bayes risk

Solutions to Overfitting
Structural Risk Minimization

Empirical risk minimization over the function set $\mathcal{F}$.

$$f_n^* = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$$

Notation: $R^*_{\mathcal{F}} = \inf_{f \in \mathcal{F}} \mathbb{E}[L(Y, f(X))]$ $\widehat{R}^*_{n,\mathcal{F}} = \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$

Risk

Empirical risk

1st issue: $R^*_{\mathcal{F}} - R^* \geq 0$ needs to be small.

(Model error, Approximation error)

Risk in $\mathcal{F}$ - Bayes risk

Solution: **Structural Risk Minimzation (SRM)**

Solution to Overfitting

Solution to Overfitting

$$R^*_{\mathcal{F}} = \inf_{f \in \mathcal{F}} R(f) = \inf_{f \in \mathcal{F}} \mathbb{E}[L(Y, f(X))]$$

ERM on $\mathcal{F}$: $\quad \widehat{R}^*_{n,\mathcal{F}} = \inf_{f \in \mathcal{F}} \widehat{R}_n(f) = \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$

Solution to Overfitting

$$R^*_{\mathcal{F}} = \inf_{f \in \mathcal{F}} R(f) = \inf_{f \in \mathcal{F}} \mathbb{E}[L(Y, f(X))]$$

ERM on $\mathcal{F}$: $\quad \widehat{R}^*_{n,\mathcal{F}} = \inf_{f \in \mathcal{F}} \widehat{R}_n(f) = \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$

2nd issue: $\quad \widehat{R}^*_{n,\mathcal{F}} = \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$

$\inf_{f \in \mathcal{F}} \widehat{R}_n(f)$ might be a very difficult optimization problem in f

It might be not even convex in f

Solution to Overfitting

$$R^*_{\mathcal{F}} = \inf_{f \in \mathcal{F}} R(f) = \inf_{f \in \mathcal{F}} \mathbb{E}[L(Y, f(X))]$$

ERM on $\mathcal{F}$: $\quad \widehat{R}^*_{n,\mathcal{F}} = \inf_{f \in \mathcal{F}} \widehat{R}_n(f) = \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$

2nd issue: $\quad \widehat{R}^*_{n,\mathcal{F}} = \inf_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(Y_i, f(X_i))$

$\inf_{f \in \mathcal{F}} \widehat{R}_n(f)$ might be a very difficult optimization problem in f

It might be not even convex in f

Solution:

Choose loss function L such that $\widehat{R}_n(f)$ will be convex in f

$L(y, f(x)) = \begin{cases} 1 & y \neq f(x) \\ 0 & y = f(x) \end{cases} \quad \Rightarrow$ not convex $\widehat{R}_n(f)$

Hinge loss $\Rightarrow$ convex $\widehat{R}_n(f)$

Quadratic loss $\Rightarrow$ convex $\widehat{R}_n(f)$

That’s all!