Foundation of Intelligent Systems, Part I

Statistical Learning Theory

mcuturi@i.kyoto-u.ac.jp
• Classification: mapping objects onto $S$ where $|S| < \infty$.

• Binary classification: answers to yes/no questions

• Linear classification algorithms

  o **Logistic Regression**
  o **Perceptron rule**
  o brief introduction to **Support Vector Machine**
Today

• Some theory about the steps in green below:

• Usual steps when using ML algorithms for classification/regression
  ○ Gather data
  ○ Choose representation
  ▪ **Choose algorithm**
  ▪ **Choose parameters**
  ○ Run algorithm, collect results
  ○ **Have second thoughts on overfitting, generalization**

• These steps are arguably the most challenging.  

• To understand better all of this, some theory is useful.

some interesting practical advice
Statistical Learning Theory
• Couples of observations, \((x, y)\) appear in nature.

• These observations are

\[ x \in \mathbb{R}^d, \quad y \in S \]

• \(S \subset \mathbb{R}\), that is \(S\) could be \(\mathbb{R}, \mathbb{R}^+, \{1, 2, 3, \ldots, L\}, \{0, 1\}\)

• Sometimes only \(x\) is visible. We want to guess the most likely \(y\) for that \(x\).

• **Example 1**  
  \(x\): Height \(\in \mathbb{R}\), \(y\): Gender \(\in \{M, F\}\)

• **Example 2**  
  \(x\): Height \(\in \mathbb{R}\), \(y\): Weight \(\in \mathbb{R}\).
Example

- To provide a guess $\leftrightarrow$ estimate a function $f : \mathbb{R}^d \to S$ such that

  $$f(x) \approx y,$$

  for most couples $(x, y)$ we have observed and ideally will observe
Probabilistic Framework

- We **assume** that each observations \((x, y)\) arise as an
  - **independent**,  
  - **identically distributed**, random sample (from the same probability law).
- This probability \(P\) on \(\mathbb{R}^d \times \mathcal{S}\) has a density,
  \[
p(X = x, Y = y).
  \]
- We assume that such a probability **exists** but,  
  **in practice, we will never** know \(p\).
- For illustration purposes, let’s study what would happen **if we knew it**.
Example 1: $S = \{M, F\}$, Height vs Gender
Example 2: $\mathcal{S} = \mathbb{R}^+$, Height vs Weight
Building Blocks: Loss (1)

- A loss is a function $S \times \mathbb{R} \rightarrow \mathbb{R}_+$ designed to quantify mistakes,

how good is the prediction $f(x)$ given that the true answer is $y$?

\[ l(y, f(x)) \]

Examples

- $S = \{0, 1\}$
  - 0/1 loss: $l(a, b) = \delta_{a \neq b} = \begin{cases} 1 & \text{if } a \neq b \\ 0 & \text{if } a = b \end{cases}$

- $S = \mathbb{R}$
  - Squared euclidian distance $l(a, b) = (a - b)^2$
  - norm $l(a, b) = \|a - b\|_q$, $0 \leq q \leq \infty$
Building Blocks: Risk (2)

- The **Risk** of a predictor $f$ with respect to loss $l$ is

$$R_l(f) = \mathbb{E}_p[l(Y, f(X))] = \int_{\mathbb{R}^d \times \mathcal{S}} l(y, f(x)) p(x, y) dxdy$$

- Risk = average loss of $f$ on all possible couples $(x, y)$, **weighted by the probability density**.

- **Remark:** a function $f$ with low risk might could very well make very big mistakes for some $x$ as long as the probability of $x$ is small.
A lower bound on the Risk? Bayes Risk

• Since \( l \geq 0 \), \( R(f) \geq 0 \).

• Consider all possible functions \( \mathbb{R}^d \rightarrow S \), usually written \( (\mathbb{R}^d)^S \).

• The **Bayes** risk is the quantity

\[
R^* = \inf_{f \in (\mathbb{R}^d)^S} R(f) = \inf_{f \in (\mathbb{R}^d)^S} \mathbb{E}_p[l(Y, f(X))]
\]

• Ideal classifier would have Bayes risk.
Bayes Classifier: \( \mathcal{S} = \{0, 1\} \), \( l \) is the 0/1 loss.

- Define the following rule:

\[
  f_B(x) = \begin{cases} 
  1, & \text{if } \eta(x) \geq \frac{1}{2}, \\
  0, & \text{otherwise}. 
  \end{cases}
\]

where

\[
  \eta(x) = p(Y = 1 | X = x).
\]

The **Bayes classifier** achieves the **Bayes Risk**.

**Theorem 1.** \( R(f_B) = R^* \).
Bayes Classifier: $\mathcal{S} = \{0, 1\}$, $l$ is the 0/1 loss.

- Chain rule of conditional probability $p(A, B) = p(B)p(A|B)$
- Bayes rule
  $$p(A|B) = \frac{p(B|A)p(A)}{p(B)}$$
- A simple way to compute $\eta$:
  $$\eta(x) = p(Y = 1|X = x) = \frac{p(Y = 1, X = x)}{p(X = x)}$$
  $$= \frac{p(X = x|Y = 1)p(Y = 1)}{p(X = x)}$$
  $$= \frac{p(X = x|Y = 1)p(Y = 1)}{p(X = x|Y = 1)p(Y = 1) + p(X = x|Y = 0)p(Y = 0)}.$$
Bayes Classifier: $\mathcal{S} = \{0, 1\}$, $l$ is the 0/1 loss.

In addition, $p(Y = 1) = 0.4871$. As a consequence, $p(Y = 0) = 1 - 0.4871 = 0.5129$.
Bayes Classifier: $\mathcal{S} = \{0, 1\}$, $l$ is the 0/1 loss.
Consider the following rule:

\[ f_B(x) = \mathbb{E}[Y|X = x] = \int_{\mathbb{R}} y p(Y = y, X = x) \, dy \]

Here again, the Bayes estimator achieves the Bayes Risk.

**Theorem 2.** \( R(f_B) = R^* \).
Bayes Estimator: $S = \mathbb{R}$, $l$ is the 2-norm

- Using Bayes rule again,

$$f^*(x) = \mathbb{E}[Y|X = x] = \int_{\mathbb{R}} y p(Y = y|X = x) dy$$

$$= \int_{\mathbb{R}} y \frac{p(X = x|Y = y)p(Y = y)}{p(X = x)} dy$$

$$= \int_{\mathbb{R}} y \frac{p(X = x|Y = y)p(Y = y)}{\int_{\mathbb{R}} p(X = x|Y = u)p(Y = u) du} dy$$

$$= \int_{\mathbb{R}} \frac{y p(X = x|Y = y)p(Y = y) dy}{\int_{\mathbb{R}} p(X = x|Y = y)p(Y = y) dy}$$
In practice
What can we do?

- If we had access to the real probability, Bayes estimator would be fine.
- In practice, the only thing we can use is a training set,

\[ \{(x_j, y_j)\}_{i=1, \ldots, n}. \]

- For instance, a set of Heights, gender

<table>
<thead>
<tr>
<th>Height</th>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>163.000</td>
<td>0</td>
</tr>
<tr>
<td>170.000</td>
<td>0</td>
</tr>
<tr>
<td>175.300</td>
<td>1</td>
</tr>
<tr>
<td>184.000</td>
<td>1</td>
</tr>
<tr>
<td>175.000</td>
<td>1</td>
</tr>
</tbody>
</table>
Approximating Risk

• For any function, instead of considering $R$, we introduce the empirical Risk $R_{emp}^n$, defined as

$$R_{emp}^n(f) = \frac{1}{n} \sum_{i=1}^{n} l(y_i, f(x_i))$$

• The law of large numbers tells us that for any given $f$

$$R_{emp}^n(f) \rightarrow R(f).$$

• Convergence can be characterized with strong or weak versions of the law.
A flawed intuition

As sample size grows, the empirical behaves like the real risk

• It may thus seem like a good idea to **minimize directly** the empirical risk.

• The intuition is that
  
  ◦ since a function \( f \) such that \( R(f) \) is low is desirable,
  
  ◦ since \( R_{\text{emp}}^n(f) \) converges to \( R(f) \) as \( n \to \infty \),

  why not look directly for any function \( f \) such that \( R_{\text{emp}}^n(f) \) is low?

• Typically, in the context of classification with 0/1 loss, find a function such that

\[
R_{\text{emp}}^n(f) = \frac{1}{n} \sum_{i=1}^{n} \delta_{y_i \neq f(x_i)}
\]

...is low.
A flawed intuition

- Focusing only on $R_{\text{emp}}^n$ is not viable:
- Consider the function defined as

$$h(x) = \begin{cases} 
 y_1, & \text{if } x = x_1, \\
 y_2, & \text{if } x = x_2, \\
 \vdots \\
 y_n, & \text{if } x = x_n, \\
 0 & \text{otherwise}
\end{cases}$$

- Since, $R_{\text{emp}}^n(h) = \frac{1}{n} \sum_{i=1}^{n} \delta_{y_i \neq h(x_i)} = \frac{1}{n} \sum_{i=1}^{n} \delta_{y_i \neq y_i} = 0$, $h$ minimizes $R_{\text{emp}}^n$.
- However, $h$ always answers 0, except for a few points.
- In practice, we can expect $R(h)$ to be much higher, equal to $P(Y = 1)$ in fact.
Here is what this function would predict on the Height/Gender Problem

Overfitting is probably the **most frequent mistake** made by ML practitioners.
Ideas to Avoid Overfitting

- Our criterion $R_n^{\text{emp}}(g)$ only considers a **finite** set of points.
- A function $g$ defined on $\mathbb{R}^d$ is defined on an **infinite** set of points.

A few approaches to control overfitting

- Restrict the set of candidates
  \[
  \min_{g \in \mathcal{G}} R_n^{\text{emp}}(g).
  \]

- Penalize “undesirable” functions
  \[
  \min_{g \in \mathcal{G}} R_n^{\text{emp}}(g) + \lambda \|g\|^2
  \]

- Penalize properly sets of functions $\mathcal{G}_d$ of increasing complexity
  \[
  \min_{d \in \mathbb{N}, g \in \mathcal{G}_d} R_n^{\text{emp}}(g) + \lambda \text{pen}(d, \mathcal{G}_d)\]
Overfitting Illustration

k-NN Classification
Bounds
Flow of a learning process in Machine Learning

- Assumption 1. existence of a probability density $p$ for $X, Y$.
- Assumption 2. points are observed i.i.d. following this probability density.

**typical flow would be**

- Get a random training sample $\{(x_j, y_j)\}_{i=1,\ldots,n}$ is random.
- Choose a function $g_n$ within a class $\mathcal{G}$ using any algorithm.
- We are very likely to have selected $g_n$ such that $R_{\text{emp}}(g_n)$ is low.

**In the end, the important question we hope to have a clue about...**

How good would be $f_n$ if we knew $p$? namely what about $R(g_n)$? Also interesting: how big is $R(g_n) - R(f_B)$
Excess Risk

• By constraining our search in $\mathcal{G}$, we want to
  ○ avoid overfitting
  ○ obtain a function that has suitable properties

• Of course, there is no reason that $f_B \in \mathcal{G}$.

• Hence, by introducing $g^*$ as a function achieving the lowest risk in $\mathcal{G}$,

$$R(g^*) = \inf_{g \in \mathcal{G}} R(g),$$

we decompose

$$R(g_n) - R(f_B) = [R(g_n) - R(g^*)] + [R(g^*) - R(f_B)].$$
Excess Risk Decomposed

• By constraining our search in $\mathcal{G}$, we want to
  ○ avoid overfitting
  ○ obtain a function that has suitable properties

• Of course, there is no reason that $f_B \in \mathcal{G}$.

• Hence, by introducing $g^*$ as a function achieving the lowest risk in $\mathcal{G}$,
  \[
  R(g^*) = \inf_{g \in \mathcal{F}} R(g),
  \]

  we decompose
  \[
  R(g_n) - R(f_B) = \left[ R(g_n) - R(g^*) \right] + \left[ R(g^*) - R(f_B) \right]
  \]

  • Estimation error is random, Approximation error is fixed.

• In the following we focus on the estimation error.
Types of Bounds of Interest

Error Bounds

\[ R(g_n) \leq R_{\text{emp}}(g_n) + C(n, G). \]

Error Bounds Relative to Best in Class

\[ R(g_n) \leq R(g^\star) + C(n, G). \]

Error Bounds Relative to the Bayes Risk

\[ R(g_n) \leq R(f_B) + C(n, G). \]
Error Bounds / Generalization Bounds

\[ R(g_n) - R_{\text{emp}}(g_n) \]
What is Overfitting?

- Overfitting is the idea that,
  - given \( n \) points sampled randomly,
  - given a function \( g_n \) estimated from these points,

\[
R(g_n) \gg R_{\text{emp}}(g_n).
\]

- Question of interest:

\[
P[R(g_n) - R_{\text{emp}}(g_n) > \varepsilon] = ?
\]

- From now on, we consider the **classification** case, namely \( \mathcal{G} : \mathbb{R}^d \rightarrow \{0, 1\} \).
Alleviating Notations

- More convenient to see a couple \((x, y)\) as a realization of \(Z\), namely

\[ z_i = (x_i, y_i), Z = (X, Y). \]

- We define the \textit{loss class}

\[ \mathcal{F} = \{ f : z = (x, y) \rightarrow \delta_{g(x) \neq y}, g \in \mathcal{G}\}, \]

- with the additional notations

\[ P f = \mathbb{E}[f(X, Y)], P_n f = \frac{1}{n} \sum_{i=1}^{n} f(x_i, y_i), \]

where we recover

\[ P_n f = R_n^{\text{emp}}(g), \quad P f = R(g)] \]
Empirical Processes

For each $f \in \mathcal{F}$, $P_{nf}$ is a random variable which depends on $n$ realizations of $Z$.

- If we consider all possible functions $f \in \mathcal{F}$, we obtain

  The set of random variables $\{P_{nf}\}_{f \in \mathcal{F}}$ is called an **Empirical measure** indexed by $\mathcal{F}$.

- A branch of mathematics studies explicitly the convergence of $\{Pf - P_{nf}\}_{f \in \mathcal{F}}$,

  This branch is known as **Empirical process theory**.
Hoeffding’s Inequality

• Recall that for a given $g$ and corresponding $f$,

\[ R(g) - R^{\text{emp}}(g) = Pf - P_n f = \mathbb{E}[f(Z)] - \frac{1}{n} \sum_{i=1}^{n} f(z_i), \]

which is simply the difference between the **expectation** and the empirical average of $f(Z)$.

• The **strong** law of large numbers says that

\[
P \left( \lim_{n \to \infty} \mathbb{E}[f(Z)] - \frac{1}{n} \sum_{i=1}^{n} f(z_i) = 0 \right) = 1.
\]
Hoeffding’s Inequality

• A more detailed result is

**Theorem 3** (Hoeffding). Let $Z_1, \cdots, Z_n$ be $n$ i.i.d random variables with $f(Z) \in [a, b]$. Then, $\forall \epsilon$,

$$P \left( \left| P_n f - Pf \right| > \epsilon \right) \leq 2e^{-\frac{2n\epsilon^2}{(b-a)^2}}.$$