Foundation of Intelligent Systems, Part I

Statistical Learning Theory

mcuturi@i.kyoto-u.ac.jp

Previous Lecture : Classification

- Classification: mapping objects onto $\mathcal{S}$ where $|\mathcal{S}| < \infty$.
- Binary classification: answers to **yes/no** questions
- Linear classification algorithms: *split* the **yes/no** zones with a **hyperplane**

$$\text{Yes} = \{\mathbf{c}^T x + \boldsymbol{b} \geq 0\}\ ,\ \text{No} = \{\mathbf{c}^T x + \boldsymbol{b} < 0\}$$

- How to select $\mathbf{c}, \boldsymbol{b}$ given a dataset?
  - **Logistic Regression** (classification from a lienar regression viewpoint)
  - **Perceptron rule** (iterative, random update rule)
  - brief introduction to **Support Vector Machine** (optimal margin classifier)
  - a few words on **Linear Discriminant Analysis** (multivariate Gaussians)

Today

- Usual steps when using ML algorithms
  - Define problem (*classification? regression? multi-class?*)
  - Gather data
  - Choose representation for data to build a database
  - Choose method/algorithm based on training set
  - Choose/estimate parameters
  - Run algorithm on new points, collect results

Today

- Usual steps when using ML algorithms
  - Define problem (*classification? regression? multi-class?*)
  - Gather data
  - Choose representation for data to build a database
  - **Choose method/algorithm**
  - **Choose/estimate parameters based on training set**
  - Run algorithm on new points, collect results

  - ... ***did I overfit?***

Today

- Usual steps when using ML algorithms
  - Define problem (*classification? regression? multi-class?*)
  - Gather data
  - Choose representation for data to build a database
  - **Choose method/algorithm**
  - **Choose/estimate parameters based on training set**
  - Run algorithm on new points, collect results

  - ... ***did I overvit?***
- These steps are arguably the most challenging. [link to interesting practical advice]
- To understand better all of this, some theory is useful.

Statistical Learning Theory

General Framework

- Couples of observations, $(\mathbf{x}, y)$ appear in nature.
- These observations are

$$\mathbf{x} \in \mathbb{R}^d, \quad y \in \mathcal{S}$$

- $\mathcal{S} \subset \mathbb{R}$, that is $\mathcal{S}$ could be $\mathbb{R}$, $\mathbb{R}_+$, $\{1, 2, 3, \ldots, L\}$, $\{0, 1\}$
- Sometimes only $\mathbf{x}$ is visible. We want to guess the most likely y for that $\mathbf{x}$.

- **Example 1** $\mathbf{x}$: Height $\in \mathbb{R}$, y: Gender $\in \{M, F\}$
- **Example 2** $\mathbf{x}$: Height $\in \mathbb{R}$, y: Weight $\in \mathbb{R}$.

Estimating the relationship between x and y

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

$$f(\mathbf{x}) \approx y.$$

Estimating the relationship between x and y

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

$$f(\mathbf{x}) \approx y.$$

- Ideally, $f(\mathbf{x}) \approx y$ should apply **both** to
  - couples $(\mathbf{x}, y)$ we **have observed** in the training set
  - couples $(\mathbf{x}, y)$ we **will observe**... (only see $\mathbf{x}$ in practice to guess y)

Estimating the relationship between x and y

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

$$f(\mathbf{x}) \approx y.$$

- Ideally, $f(\mathbf{x}) \approx y$ should apply **both** to
  - couples $(\mathbf{x}, y)$ we **have observed** in the training set
  - couples $(\mathbf{x}, y)$ we **will observe**... (only see $\mathbf{x}$ in practice to guess y)

Let's suppose for a few slides that we have access to **much more**...

Probabilistic Framework

- We **assume** that **each** observations $(\mathbf{x}, y)$ 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,

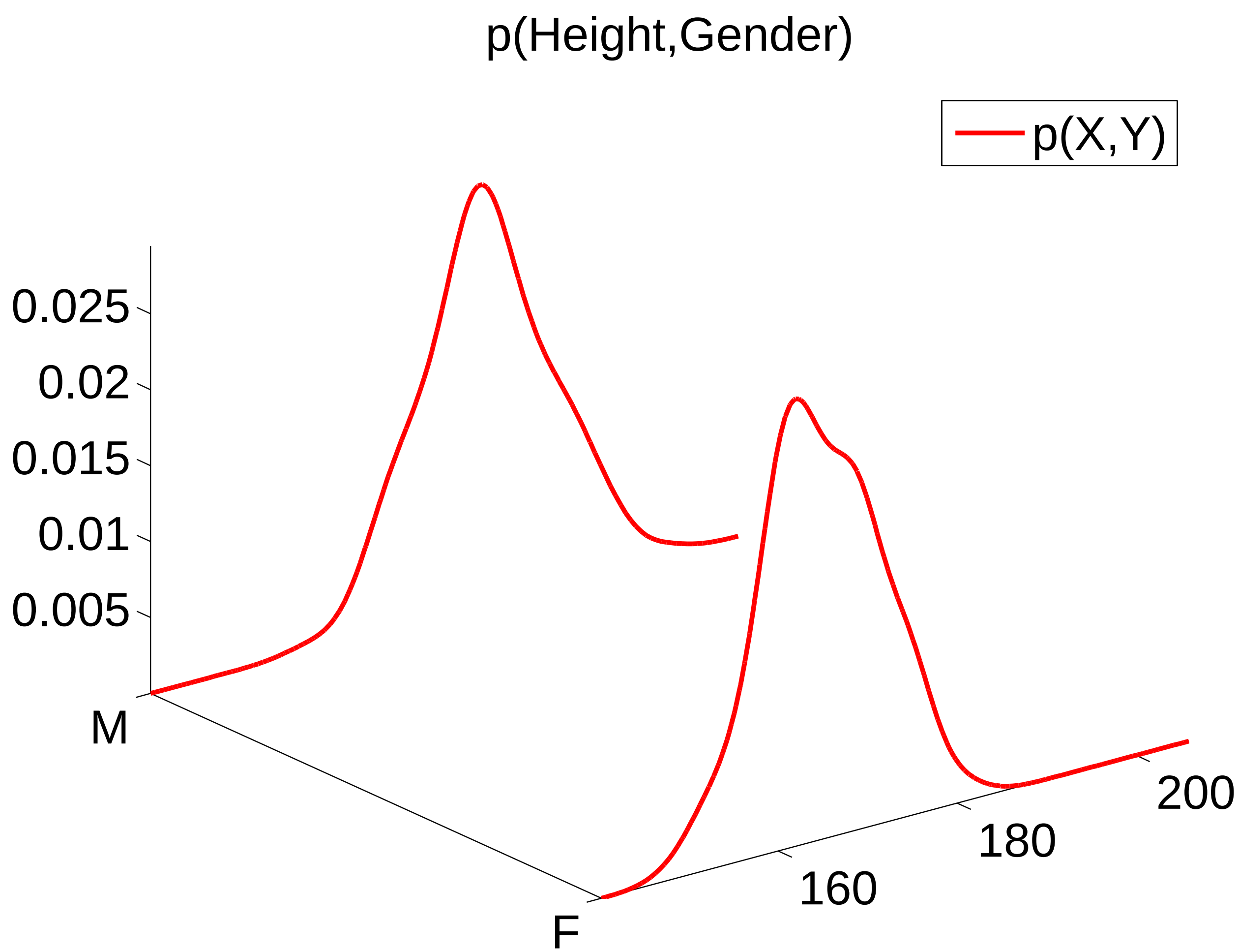
$$p(X = \mathbf{x}, Y = y).$$

- Assume that such a probability **exists**.. (**in practice**, we will **never** know p).

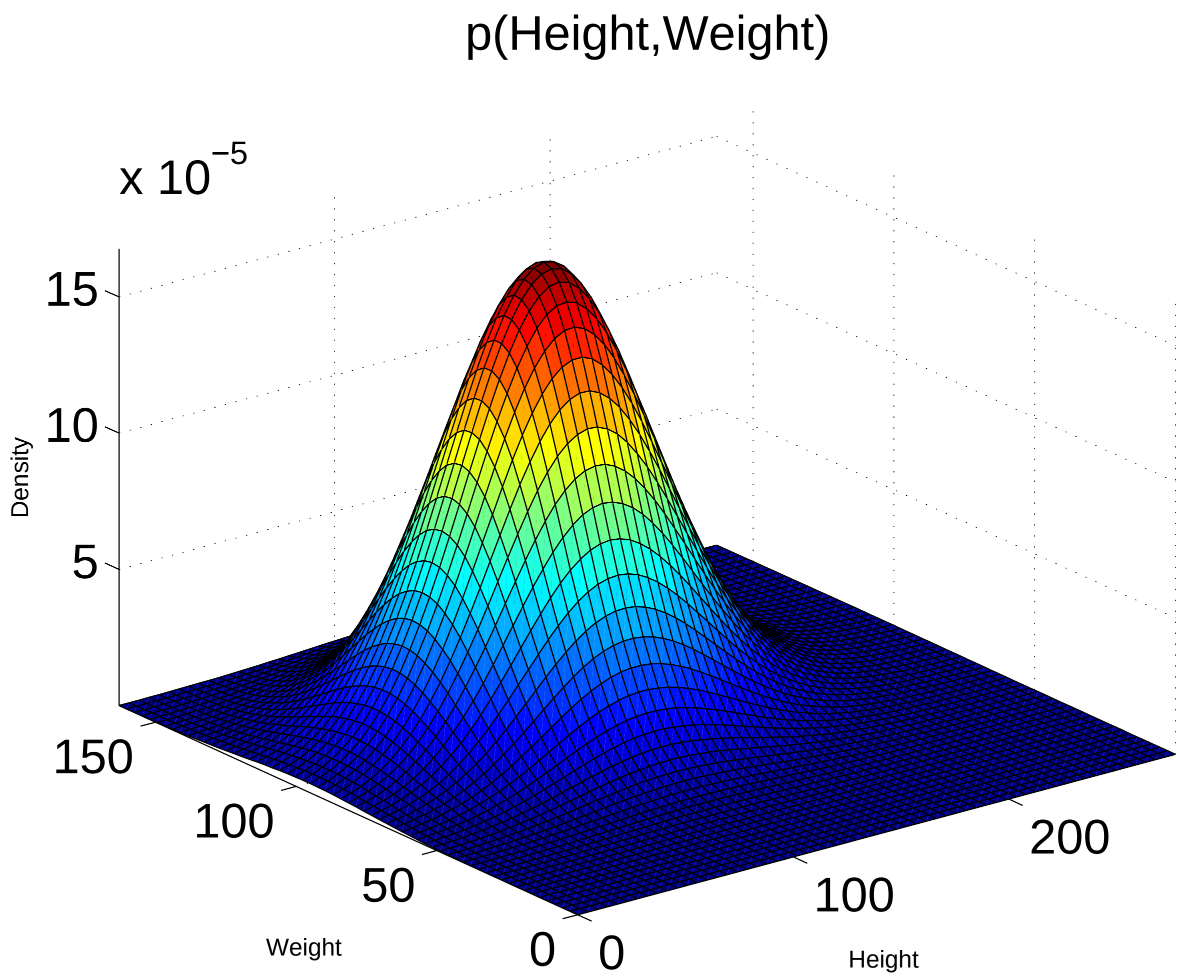
For illustration purposes, let's study what would happen **if we knew it**.

($\approx$ playing Trivial Pursuit with wikipedia, $\approx$ using Cheat codes on a video game)

Example 1: $\mathcal{S} = \{M, F\}$, Height vs Gender



Example 2: $\mathcal{S} = \mathbb{R}^+$, Height vs Weight



Building Blocks: Loss (1)

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

how good is the prediction $f(\mathbf{x})$ given that the true answer is y? $\Updownarrow$ How small is $l(y, f(\mathbf{x}))$?

Examples

- $\mathcal{S} = \{0, 1\}$
  - 0/1 loss: $l(a, b) = \delta_{a \neq b} = \begin{cases} 1 \text{ if } a \neq b \\ 0 \text{ if } a = b \end{cases}$
- $\mathcal{S} = \mathbb{R}$
  - 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 $\boldsymbol{f}$ with respect to **loss** l is

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

- Risk $=$ average loss of $\boldsymbol{f}$ on **all possible couples** $(\mathbf{x}, y)$,

weighted by the probability density.

Risk($\boldsymbol{f}$) measures the performance of f w.r.t. l and p.

- Remark: a function $\boldsymbol{f}$ with low risk might could very well make very big mistakes for some $\mathbf{x}$ as long as the probability of $\mathbf{x}$ is small.

A lower bound on the Risk? Bayes Risk

- Since $l \geq 0$, $R(\boldsymbol{f}) \geq 0$.
- Consider all possible functions $\mathbb{R}^d \to \mathcal{S}$, usually written $(\mathbb{R}^d)^{\mathcal{S}}$.
- The **Bayes** risk is the quantity

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

- Ideal classifier would have Bayes risk.

Bayes Classifier : $\mathcal{S} = \{0, 1\}$, l is the 0/1 loss.

- Define the following rule:

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

where

$$\eta(\mathbf{x}) = p(Y = 1 | X = \mathbf{x}).$$

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

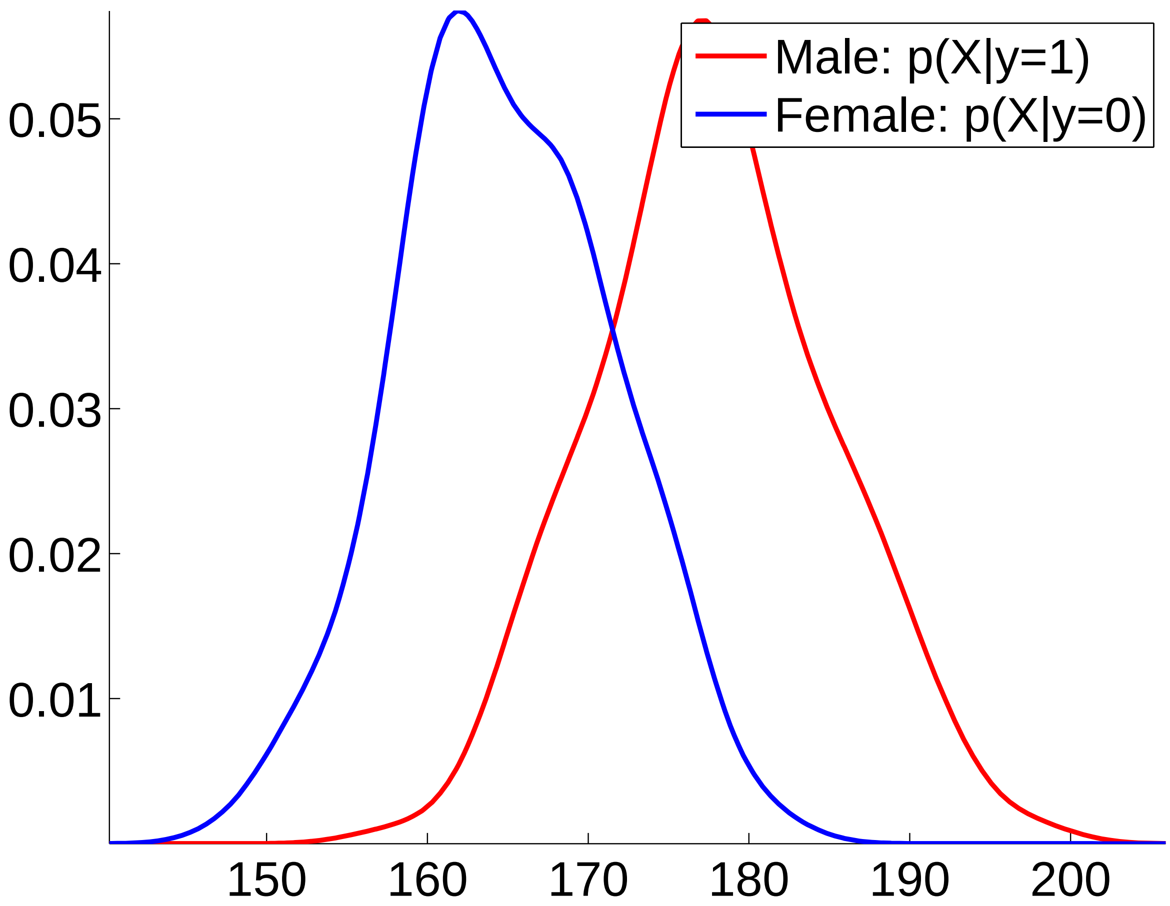
Theorem 1. $R(f_B) = R^*$.

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 η:

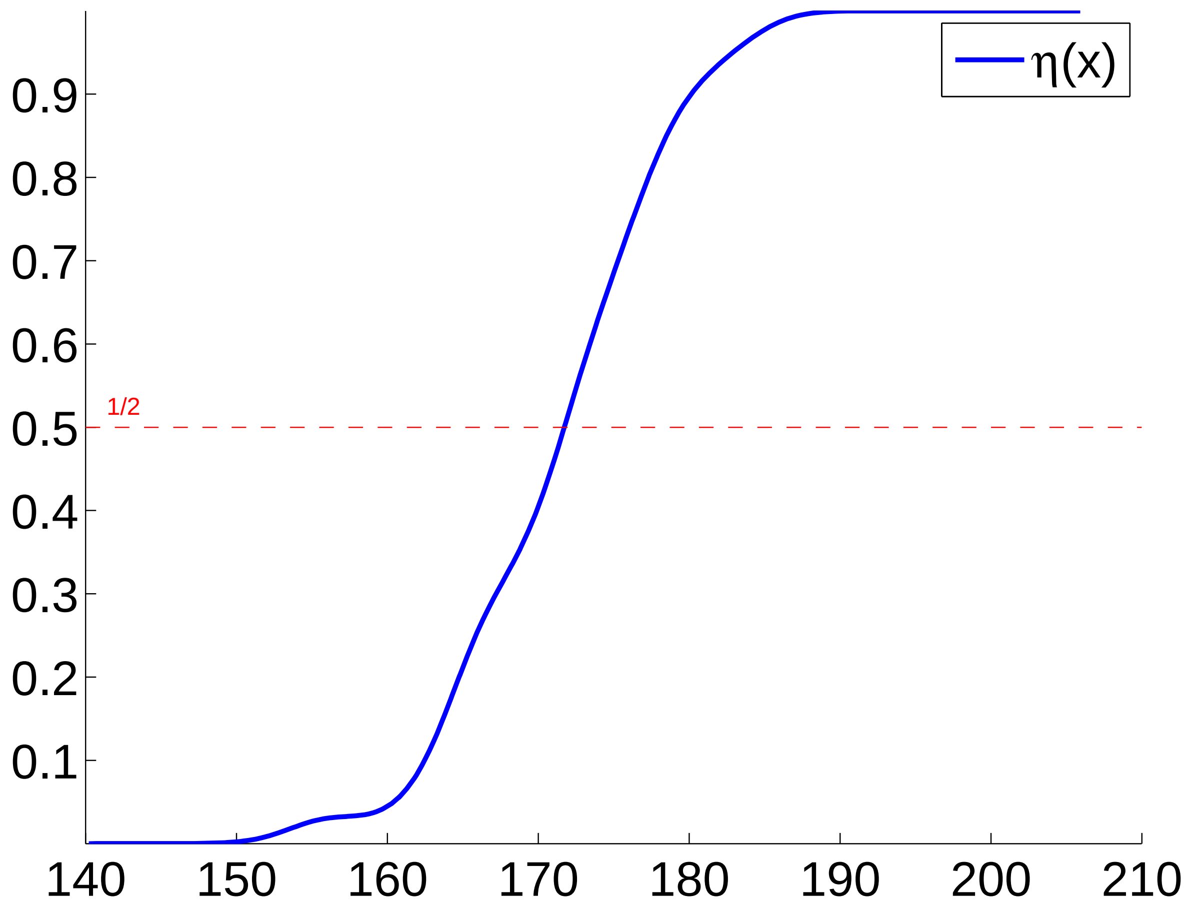
$$\begin{aligned}\eta(\mathbf{x}) = p(Y = 1|X = \mathbf{x}) &= \frac{p(Y = 1, X = \mathbf{x})}{p(X = \mathbf{x})} \\ &= \frac{p(X = \mathbf{x}|Y = 1)p(Y = 1)}{p(X = \mathbf{x})} \\ &= \frac{p(X = \mathbf{x}|Y = 1)p(Y = 1)}{p(X = \mathbf{x}|Y = 1)p(Y = 1) + p(X = \mathbf{x}|Y = 0)p(Y = 0)}.\end{aligned}$$

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.



Bayes Estimator : $\mathcal{S} = \mathbb{R}$, l is the 2-norm

- Consider the following rule:

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

Here again, the **Bayes estimator** achieves the **Bayes Risk**.

Theorem 2. $R(f_B) = R^*$.

Bayes Estimator : $\mathcal{S} = \mathbb{R}$, l is the 2-norm

- Using Bayes rule again,

$$
\begin{aligned}
f^\star(\mathbf{x}) = \mathbb{E}[Y|X=\mathbf{x}] &= \int_{\mathbb{R}} \boldsymbol{y}\, p(Y=y|X=\mathbf{x})dy \\
&= \int_{\mathbb{R}} \boldsymbol{y}\, \frac{p(X=\mathbf{x}|Y=y)p(Y=y)}{p(X=\mathbf{x})}dy \\
&= \int_{\mathbb{R}} \boldsymbol{y}\, \frac{p(X=\mathbf{x}|Y=y)p(Y=y)}{\int_{\mathbb{R}} p(X=\mathbf{x}|Y=u)p(Y=u)du}dy \\
&= \frac{\int_{\mathbb{R}} \boldsymbol{y}\, p(X=\mathbf{x}|Y=y)p(Y=y)dy}{\int_{\mathbb{R}} p(X=\mathbf{x}|Y=y)p(Y=y)dy}
\end{aligned}
$$

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,

$$\{(\mathbf{x}_j, y_j)\}_{i=1,\cdots,n}.$$

- For instance, a set of Heights, gender

163.0000	0
170.0000	0
175.3000	1
184.0000	1
175.0000	1

Approximating Risk

- For any function, instead of considering R, we introduce

$$\text{the } \textbf{empirical} \text{ Risk } \boldsymbol{R_n^{\text{emp}}},$$

  defined as

$$\boldsymbol{R_n^{\text{emp}}}(\boldsymbol{f}) = \frac{1}{n}\sum_{i=1}^{n} l(y_i, \boldsymbol{f}(\mathbf{x}_i))$$

- The law of large numbers tells us that for any given $\boldsymbol{f}$

$$\boldsymbol{R_n^{\text{emp}}}(\boldsymbol{f}) \rightarrow R(f).$$

- 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 $R_n^{\text{emp}}(f)$ converges to $R(f)$ as $n \to \infty$,

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

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

$$R_n^{\text{emp}}(f) = \frac{1}{n} \sum_{i=1}^{n} \delta_{y_i \neq f(\mathbf{x}_i)}$$

...is low.

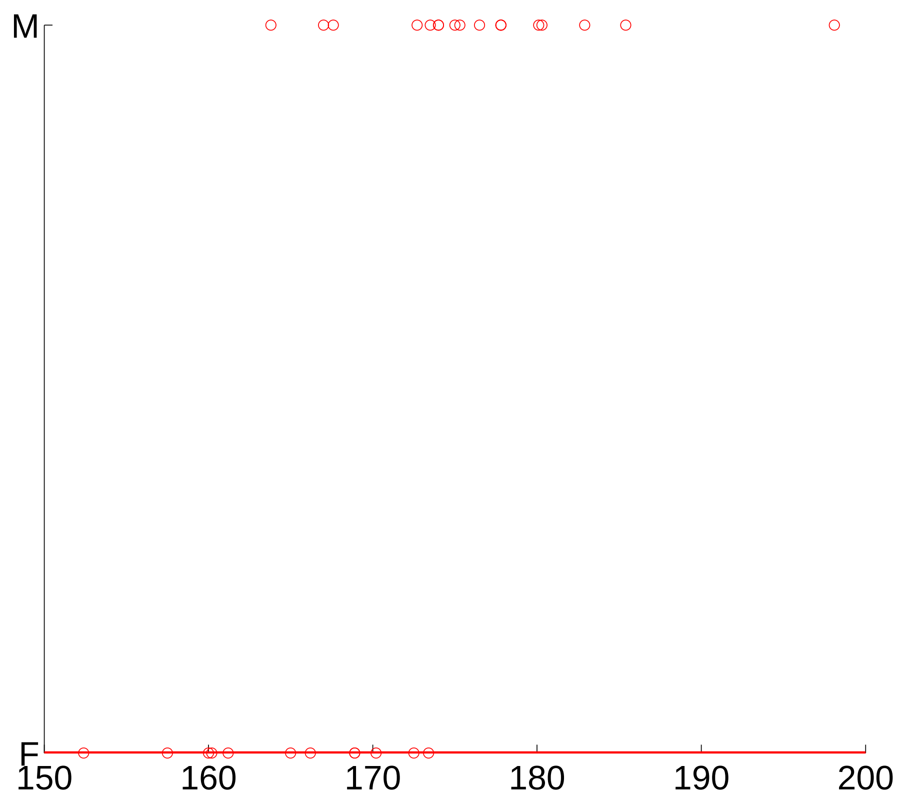
A flawed intuition

- Focusing only on R_n^{emp} is not viable:
- Consider the function defined as

$$h(\mathbf{x}) = \begin{cases} y_1, & \text{if } \mathbf{x} = \mathbf{x}_1, \\ y_2, & \text{if } \mathbf{x} = \mathbf{x}_2, \\ \vdots & \\ y_n, & \text{if } \mathbf{x} = \mathbf{x}_n, \\ 0 \text{ otherwise..} & \end{cases}$$

- Since, $R_n^{\mathrm{emp}}(h) = \frac{1}{n}\sum_{i=1}^{n} \delta_{y_i \neq h(\mathbf{x}_i)} = \frac{1}{n}\sum_{i=1}^{n} \delta_{y_i \neq y_i} = 0$, h minimizes R_n^{emp}.
- 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**

$$\min_{g \in \boldsymbol{\mathcal{G}}} R_n^{\text{emp}}(g).$$

- **Penalize "undesirable" functions**

$$\min_{g \in \boldsymbol{\mathcal{G}}} R_n^{\text{emp}}(g) + \lambda \|\boldsymbol{g}\|^{\mathbf{2}}$$

- **Penalize properly sets of functions $\mathcal{G}_d$ of increasing complexity**

$$\min_{d \in \mathbb{N}, g \in \boldsymbol{\mathcal{G}_d}} R_n^{\text{emp}}(g) + \lambda \text{pen}(d, \mathcal{G}_d)$$

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.

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.

Roadmap

- Get a random training sample $\{(\mathbf{x}_j, y_j)\}_{i=1,\cdots,n}$ (*training set*)
- Choose a class of functions $\mathcal{G}$ (*method* or *model*)
- Choose g_n in $\mathcal{G}$ such that $R_n^{\text{emp}}(g_n)$ is **low** (*estimation algorithm*)

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.

Roadmap

- Get a random training sample $\{(\mathbf{x}_j, y_j)\}_{i=1,\cdots,n}$ (*training set*)
- Choose a class of functions $\mathcal{G}$ (*method* or *model*)
- Choose g_n in $\mathcal{G}$ such that $R_n^{\text{emp}}(g_n)$ is **low** (*estimation algorithm*)

Next... use g_n in practice

Flow of a learning process in Machine Learning

Yet, you may want to have a partial answer to these questions

- How good would be $\boldsymbol{f_B}$ if we knew the real probability p?
- what about $\boldsymbol{R(g_n)}$?
- What's the gap between them, $\boldsymbol{R(g_n)} - \boldsymbol{R}(\boldsymbol{f_B})$?
- Is the *estimation* algorithm reliable? how big is $\boldsymbol{R}^{\text{emp}}\boldsymbol{(g_n)} - \inf_{g\in\mathcal{G}} \boldsymbol{R}^{\text{emp}}(g)$?
- how big is $\boldsymbol{R}^{\text{emp}}\boldsymbol{(g_n)} - \inf_{g\in\mathcal{G}} \boldsymbol{R}(g)$?

Excess Risk

- In the general case $f_B \notin \mathcal{G}$.
- Hence, by introducing $g^\star$ as a function achieving the lowest risk in $\mathcal{G}$,

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

we decompose

$$R(g_n) - R(f_B) = [R(g_n) - R(g^\star)] + [R(g^\star) - R(f_B)]$$

Excess Risk

- In the general case $f_B \notin \mathcal{G}$.
- Hence, by introducing $g^\star$ as a function achieving the lowest risk in $\mathcal{G}$,

$$R(g^\star) = \inf_{g \in \mathcal{F}} R(g),$$

we decompose

$$R(g_n) - R(f_B) = \underbrace{\left[\boldsymbol{R(g_n) - R(g^\star)}\right]}_{\textbf{Estimation Error}} + \underbrace{\left[\boldsymbol{R(g^\star) - R(f_B)}\right]}_{\textbf{Approximation Error}}$$

- Estimation error is **random**, Approximation error is **fixed**.
- In the following we focus on the estimation error.

Types of Bounds

Error Bounds

$$R(g_n) \leq R_n^{\mathrm{emp}}(g_n) + C(n, \mathcal{G}).$$

Types of Bounds

Error Bounds

$$R(g_n) \leq R_n^{\mathrm{emp}}(g_n) + C(n, \mathcal{G}).$$

Error Bounds Relative to Best in Class

$$R(g_n) \leq R(g^\star) + C(n, \mathcal{G}).$$

Types of Bounds

Error Bounds

$$R(g_n) \leq R_n^{\text{emp}}(g_n) + C(n, \mathcal{G}).$$

Error Bounds Relative to Best in Class

$$R(g_n) \leq R(g^\star) + C(n, \mathcal{G}).$$

Error Bounds Relative to the Bayes Risk

$$R(g_n) \leq R(f_B) + C(n, \mathcal{G}).$$

Error Bounds / Generalization Bounds

$$R(g_n) - R_n^{\mathrm{emp}}(g_n)$$

What is Overfitting?

- Overfitting is the idea that,
  - given n training points sampled randomly,
  - given a function g_n estimated from these points,
  - we may have...

$$R(g_n) \gg R_n^{\text{emp}}(g_n).$$

What is Overfitting?

- Overfitting is the idea that,
  - given n training points sampled randomly,
  - given a function g_n estimated from these points,
  - we may have...

$$R(g_n) \gg R_n^{\text{emp}}(g_n).$$

- Question of interest:

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

- From now on, we consider the **classification** case, namely $\mathcal{G} : \mathbb{R}^d \to \{0, 1\}$.

Alleviating Notations

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

$$\mathbf{z}_i = (\mathbf{x}_i, y_i), Z = (X, Y).$$

- We define the *loss class*

$$\mathcal{F} = \{f : \mathbf{z} = (\mathbf{x}, y) \to \delta_{g(\mathbf{x}) \neq y},\ g \in \mathcal{G}\},$$

- with the additional notations

$$Pf = \mathbb{E}[f(X, Y)], P_n f = \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{x}_i, y_i),$$

where we recover

$$P_n f = R_n^{\mathrm{emp}}(g), \quad Pf = R(g)]$$

Empirical Processes

For each $f \in \mathcal{F}$, $P_n f$ 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 $\{P_n f\}_{f \in \mathcal{F}}$ is called an Empirical measure indexed by $\mathcal{F}$.

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

This branch is known as Empirical process theory.

Hoeffding's Inequality

- Recall that for a given g and corresponding f,

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

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

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

$$P\left(\lim_{n\to\infty} \mathbb{E}[f(Z)] - \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{z}_i) = 0\right) = 1.$$

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,* $\forall \varepsilon$*,*

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