MIT 9.520/6.7910
Statistical Learning Theory and Applications

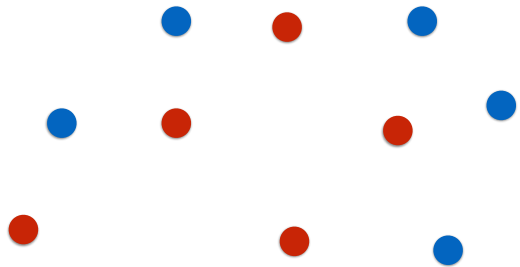
Class 02: Statistical Learning Theory

Lorenzo Rosasco

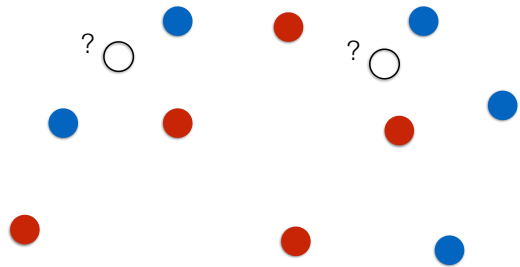
Supervised ML

- **Supervised**: $\{(x_1, y_1), \ldots, (x_n, y_n)\}$.
- Unsupervised: $\{x_1, \ldots, x_m\}$.
- Semi-supervised: $\{(x_1, y_1), \ldots, (x_n, y_n)\} \cup \{x_1, \ldots, x_m\}$.

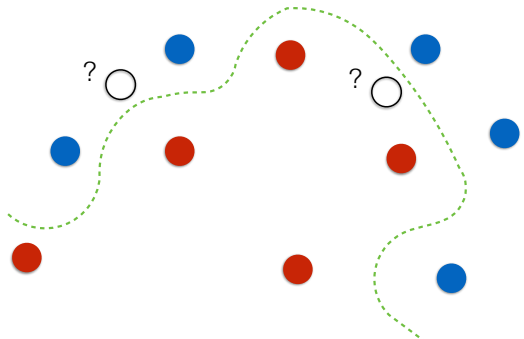
Supervised learning



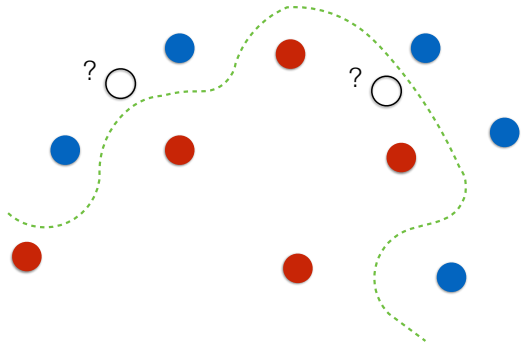
Supervised learning



Supervised learning



Supervised learning



Problem: **given** $S_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ **find** $f(x_{\text{new}}) \sim y_{\text{new}}$

Training and test

The distinction between training and test data is crucial.

In statistical learning, the test set is assumed to be *infinite*, and data drawn from a fixed, unknown distribution.

Outline

Statistical learning

Statistical learning theory

The supervised learning problem

- $X \times Y$ probability space, with distribution P.
- $\ell : Y \times Y \to [0, \infty)$ *loss function*.

The supervised learning problem

- $X \times Y$ probability space, with distribution P.
- $\ell : Y \times Y \to [0, \infty)$ *loss function*.

Problem: Solve

$$\min_{f: X \to Y} L(f) = \mathbb{E}_{(x,y) \sim P}[\ell(y, f(x))],$$

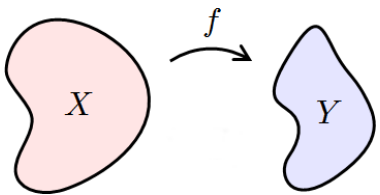
given

$$S_n = (x_1, y_1), \ldots, (x_n, y_n) \sim P^n.$$

Ingredients

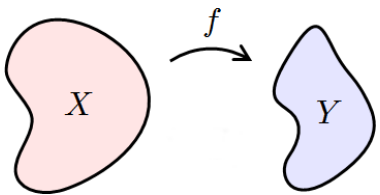
- Data space and distribution.
- Loss functions.
- Expected risk risk and target function.
- Excess risk, sample complexity and no free lunch.

Data space



X, Y input/output spaces.

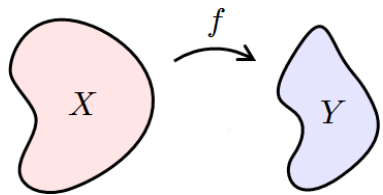
Data space



X, Y input/output spaces.

Vector spaces or more general spaces e.g. strings, graphs, probabilities...

Data space



X, Y input/output spaces.

Vector spaces or more general spaces e.g. strings, graphs, probabilities...

Problems with different output spaces get different names

- $Y = \mathbb{R}$ or $Y = \mathbb{R}^T$ regression,
- Y Hilbert space, functional regression.
- $Y = \{-1, 1\}$, classification,
- $Y = \{1, \ldots, T\}$, multicategory classification,
- structured learning: strings, graphs, probabilities...

Probability distribution

$$S_n = (x_1, y_1), \ldots, (x_n, y_n) \sim P^n.$$

Data identically and independently distributed (i.i.d.) w.r.t. P fixed, but unknown.

Probability distribution

$$S_n = (x_1, y_1), \ldots, (x_n, y_n) \sim P^n.$$

Data identically and independently distributed (i.i.d.) w.r.t. P fixed, but unknown.

P reflects *uncertainty* and *stochasticity* in the learning problem.

Probability distribution

$$S_n = (x_1, y_1), \ldots, (x_n, y_n) \sim P^n.$$

Data identically and independently distributed (i.i.d.) w.r.t. P fixed, but unknown.

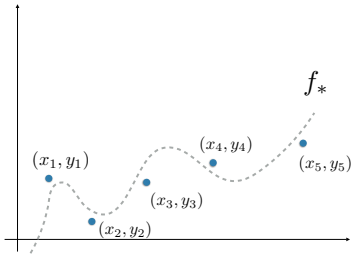
P reflects *uncertainty* and *stochasticity* in the learning problem.

Main assumption:

$$P(x, y) = P_X(x)P(y|x),$$

- P_X marginal distribution on X,
- $P(y|x)$ conditional distribution on Y given $x \in X$.

Conditional distribution and noise



Regression

$$y_i = f_*(x_i) + \epsilon_i.$$

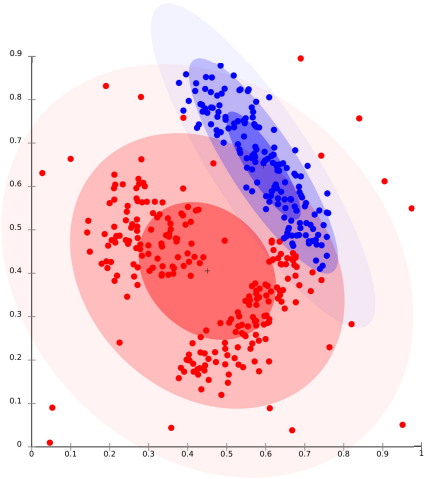
- Let $f_* : X \to Y$, fixed function,
- $\epsilon_1, \ldots, \epsilon_n$ zero mean random variables, $\epsilon_i \sim N(0, \sigma)$,
- $x_1, \ldots, x_n$ random,

$$P(y|x) = N(f_*(x), \sigma).$$

Conditional distribution and misclassification

Classification

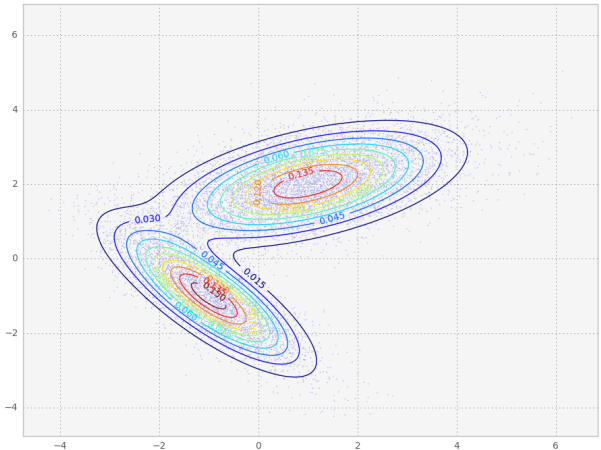
$$P(y|x) = \{P(1|x), P(-1|x)\}.$$



Noise in classification: overlap between the classes

Marginal distribution and sampling

P_X takes into account *uneven* sampling of the input space.

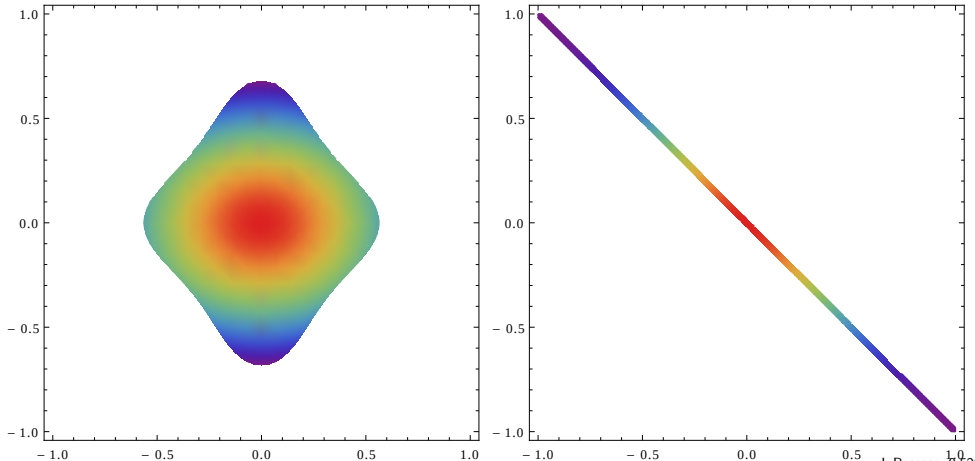


Marginal distribution, densities and manifolds

$$p(x) = \frac{dP_X(x)}{dx}$$

Marginal distribution, densities and manifolds

$$p(x) = \frac{dP_X(x)}{dx} \Rightarrow p(x) = \frac{dP_X(x)}{d\mathrm{vol}(x)}$$



IID Data

$$S_n = (x_1, y_1), \ldots, (x_n, y_n) \sim P^n$$

The assumption of i.i.d. data is hardly ever satisfied!
Nonetheless, the model proved to be useful even if the assumption is violated.

Are other models possible?

Indpendence. The independence assumption can be relaxed without changing substantially the framework (e.g. introducing mixing processes).

Identical. Dropping the identically distributed assumption require substantially different frameworks (e.g. sequential prediction, active learning, reinforcement learning).

Ingredients

- Data space and distribution.
- Loss functions.
- Expected risk risk and target function.
- Excess risk, sample complexity and no free lunch.

Loss functions

$$\ell : Y \times Y \to [0, \infty)$$

- Cost of predicting $f(x)$ in place of y.
- Measures the *pointwise error* $\ell(y, f(x))$.
- Part of the problem definition since $L(f) = \int_{X \times Y} \ell(y, f(x)) dP(x, y)$.

Note: sometimes it is useful to consider a loss of the form

$$\ell : Y \times \mathcal{G} \to [0, \infty)$$

for some space $\mathcal{G}$, e.g. $\mathcal{G} = \mathbb{R}$, $Y = \{\pm 1\}$.

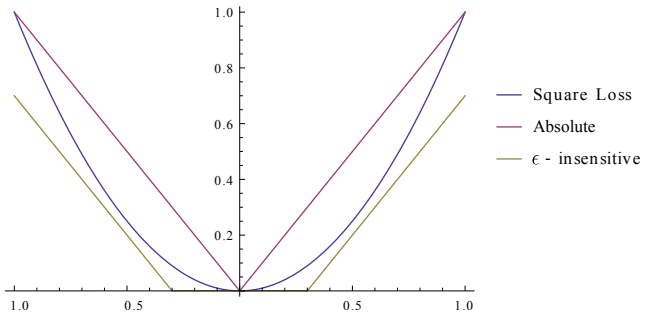
Loss for regression

$$\ell(y, y') = V(y - y'), \quad V : \mathbb{R} \to [0, \infty).$$

Loss for regression

$$\ell(y, y') = V(y - y'), \quad V : \mathbb{R} \to [0, \infty).$$

- Square loss $\ell(y, y') = (y - y')^2$.
- Absolute loss $\ell(y, y') = |y - y'|$.
- ϵ-insensitive $\ell(y, y') = \max(|y - y'| - \epsilon, 0)$.



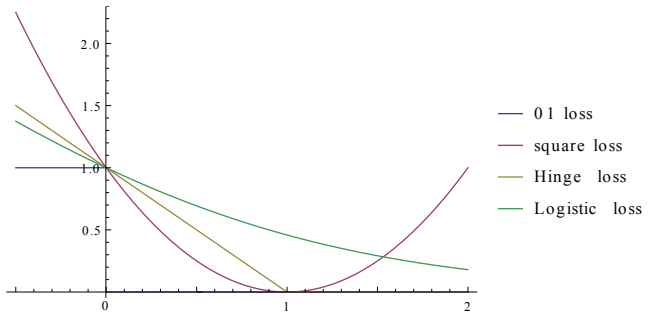
Loss for classification

$$\ell(y, y') = V(-yy'), \quad V : \mathbb{R} \to [0, \infty).$$

Loss for classification

$$\ell(y, y') = V(-yy'), \quad V : \mathbb{R} \to [0, \infty).$$

- 0-1 loss $\ell(y, y') = \Theta(-yy'), \quad \Theta(a) = 1$, if $a \geq 0$ and 0 otherwise.
- Square loss $\ell(y, y') = (1 - yy')^2$.
- Hinge-loss $\ell(y, y') = \max(1 - yy', 0)$.
- Logistic loss $\ell(y, y') = \log(1 + \exp(-yy'))$.



Loss function for structured prediction

Loss specific for each learning task, e.g.

- Multiclass: square loss, weighted square loss, logistic loss, ...
- Multitask: weighted square loss, absolute, ...
- ...

Ingredients

- Data space and distribution.
- Loss functions.
- Expected risk risk and target function.
- Excess risk, sample complexity and no free lunch.

Expected risk

$$L(f) = \mathbb{E}_{(x,y)\sim P}[\ell(y, f(x))] = \int_{X\times Y} \ell(y, f(x)) dP(x, y),$$

with

$$f \in \mathcal{F}, \quad \mathcal{F} = \{f : X \to Y \mid f \text{ measurable}\}.$$

[1] $\Theta(a) = 1$, if $a \geq 0$ and 0 otherwise.

Expected risk

$$L(f) = \mathbb{E}_{(x,y)\sim P}[\ell(y, f(x))] = \int_{X\times Y} \ell(y, f(x)) dP(x, y),$$

with

$$f \in \mathcal{F}, \quad \mathcal{F} = \{f : X \to Y \mid f \text{ measurable}\}.$$

Example: Classification

$$Y = \{-1, +1\}, \quad \ell(y, f(x)) = \Theta(-yf(x)) \ [1]$$
$$L(f) = \mathbb{P}(\{(x, y) \in X \times Y \mid f(x) \neq y\}).$$

[1] $\Theta(a) = 1$, if $a \geq 0$ and 0 otherwise.

Target function

$$f_P \quad = \quad \arg\min_{f \in \mathcal{F}} \; L(f),$$

can be derived for many loss functions.

Target function

$$f_P \quad = \quad \arg\min_{f \in \mathcal{F}} \; L(f),$$

can be derived for many loss functions.

Inner risk

$$L(f) = \int dP(x,y)\ell(y, f(x)) = \int \left(\underbrace{\int \ell(y, f(x)) dP(y|x)}_{L_x(f(x))} \right) dP_X(x).$$

Target function

$$f_P \quad = \quad \arg\min_{f \in \mathcal{F}} \; L(f),$$

can be derived for many loss functions.

Inner risk

$$L(f) = \int dP(x,y)\ell(y, f(x)) = \int \left(\underbrace{\int \ell(y, f(x)) dP(y|x)}_{L_x(f(x))} \right) dP_X(x).$$

It is easy to show that:

$$f_P(x) = \arg\min_{a \in \mathbb{R}} L_x(a),$$

(measurability can be ensured).

Target functions in regression

$$f_P(x) = \arg\min_{a \in \mathbb{R}} L_x(a).$$

Square loss

$$f_P(x) = \int_Y y dP(y|x).$$

Target functions in regression

$$f_P(x) = \arg\min_{a \in \mathbb{R}} L_x(a).$$

Square loss

$$f_P(x) = \int_Y y dP(y|x).$$

Absolute loss

$$f_P(x) = \textbf{median}(P(y|x)),$$

$$\textbf{median}(p(\cdot)) = y \quad \text{s.t.} \quad \int_{-\infty}^{y} t dp(t) = \int_{y}^{+\infty} t dp(t).$$

Target functions in classification

Misclassification loss

$$f_P(x) = \textbf{sign}(P(1|x) - P(-1|x)).$$

Target functions in classification

Misclassification loss

$$f_P(x) = \textbf{sign}(P(1|x) - P(-1|x)).$$

Square loss

$$f_P(x) = P(1|x) - P(-1|x).$$

Target functions in classification

Misclassification loss

$$f_P(x) = \textbf{sign}(P(1|x) - P(-1|x)).$$

Square loss

$$f_P(x) = P(1|x) - P(-1|x).$$

Logistic loss

$$f_P(x) = \log \frac{P(1|x)}{P(-1|x)}.$$

Target functions in classification

Misclassification loss

$$f_P(x) = \textbf{sign}(P(1|x) - P(-1|x)).$$

Square loss

$$f_P(x) = P(1|x) - P(-1|x).$$

Logistic loss

$$f_P(x) = \log \frac{P(1|x)}{P(-1|x)}.$$

Hinge-loss

$$f_P(x) = \textbf{sign}(P(1|x) - P(-1|x)).$$

Different loss, different target

- Each loss functions defines a different target function. Learning enters the picture when the latter is impossible or hard to compute (as in simulations).

- As we see in the following, loss functions also differ in terms of computations.

Ingredients

- Data space and distribution.
- Loss functions.
- Expected risk risk and target function.
- Excess risk, sample complexity and no free lunch.

Outline

Statistical learning

Statistical learning theory

How good is a solution?

$$S_n \to \widehat{f} = \widehat{f}_{S_n}.$$

$\widehat{f}$ estimates f_P given the observed examples S_n.

How good is a solution?

$$S_n \rightarrow \widehat{f} = \widehat{f}_{S_n}.$$

$\widehat{f}$ estimates f_P given the observed examples S_n.

How to measure the quality of a solution?

How good is a solution?

$$S_n \to \widehat{f} = \widehat{f}_{S_n}.$$

$\widehat{f}$ estimates f_P given the observed examples S_n.

How to measure the quality of a solution?

Excess risk:

$$L(\widehat{f}) - L(f_P).$$

Excess risk is a random variable

Excess risk:

$$L(\widehat{f}) - L(f_P).$$

It is random!

Excess risk is a random variable

Excess risk:

$$L(\widehat{f}) - L(f_P).$$

It is random!

We could consider its expectation

$$\mathbb{E}[L(\widehat{f}) - L(f_P)],$$

or it cumulative function

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) > \epsilon\right).$$

Consistency

Consistency in Expectation: For any $\epsilon > 0$,

$$\lim_{n \to \infty} \mathbb{E}[L(\widehat{f}) - L(f_P)] = 0.$$

Other forms of consistency

Weak consistency: For any $\epsilon > 0$,

$$\lim_{n \to \infty} \mathbb{P}\left(L(\widehat{f}) - L(f_P) > \epsilon \right) = 0.$$

Strong consistency: For any $\epsilon > 0$,

$$\mathbb{P}\left(\lim_{n \to \infty} L(\widehat{f}) - L(f_P) = 0 \right) = 1.$$

Consistency and convergence of random variables

Different notions of consistency correspond to different notions of convergence for random variables: respectively, in expectation (L^1), in probability and almost sure (a.s.).

Consistency and convergence of random variables

Different notions of consistency correspond to different notions of convergence for random variables: respectively, in expectation (L^1), in probability and almost sure (a.s.).

The following implications hold:

$$\text{a.s.} \implies \text{in probability}, \qquad L^1 \implies \text{in probability}.$$

Consistency and convergence of random variables

Different notions of consistency correspond to different notions of convergence for random variables: respectively, in expectation (L^1), in probability and almost sure (a.s.).

The following implications hold:

$$\text{a.s.} \implies \text{in probability}, \qquad L^1 \implies \text{in probability}.$$

In general, no implication between a.s. and L^1.
If the random variables are uniformly bounded (or more generally uniformly integrable), then

$$\text{a.s.} \implies L^1.$$

Finite sample bounds

Error given n points,

$$\mathbb{E}[L(\widehat{f}) - L(f_P)] \leq \epsilon_{P,\mathcal{F}}(n).$$

Finite sample bounds

Error given n points,

$$\mathbb{E}[L(\widehat{f}) - L(f_P)] \leq \epsilon_{P,\mathcal{F}}(n).$$

Equivalently, sample complexity $n_{P,\mathcal{F}}(\epsilon)$ for an error $\epsilon > 0$ is

$$\mathbb{E}[L(\widehat{f}) - L(f_P)] \leq \epsilon$$

Sample complexity, tail bounds and error bounds

- *Error bounds*: For any $\delta \in (0, 1]$, $n \in \mathbb{N}$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \leq \epsilon_{P,\mathcal{F}}(n, \delta)\right) \geq 1 - \delta.$$

Sample complexity, tail bounds and error bounds

- *Error bounds*: For any $\delta \in (0, 1]$, $n \in \mathbb{N}$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \leq \epsilon_{P,\mathcal{F}}(n, \delta)\right) \geq 1 - \delta.$$

- Sample complexity (PAC): For any $\epsilon > 0, \delta \in (0, 1]$, when $n \geq n_{P,\mathcal{F}}(\epsilon, \delta)$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \geq \epsilon\right) \leq \delta.$$

Sample complexity, tail bounds and error bounds

- *Error bounds*: For any $\delta \in (0, 1]$, $n \in \mathbb{N}$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \leq \epsilon_{P,\mathcal{F}}(n, \delta)\right) \geq 1 - \delta.$$

- Sample complexity (PAC): For any $\epsilon > 0, \delta \in (0, 1]$, when $n \geq n_{P,\mathcal{F}}(\epsilon, \delta)$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \geq \epsilon\right) \leq \delta.$$

- *Tail bounds*: For any $\epsilon > 0, n \in \mathbb{N}$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \geq \epsilon\right) \leq \delta_{P,\mathcal{F}}(n, \epsilon).$$

Sample complexity, tail bounds and error bounds

- *Error bounds*: For any $\delta \in (0, 1]$, $n \in \mathbb{N}$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \leq \epsilon_{P,\mathcal{F}}(n, \delta)\right) \geq 1 - \delta.$$

- Sample complexity (PAC): For any $\epsilon > 0, \delta \in (0, 1]$, when $n \geq n_{P,\mathcal{F}}(\epsilon, \delta)$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \geq \epsilon\right) \leq \delta.$$

- *Tail bounds*: For any $\epsilon > 0, n \in \mathbb{N}$,

$$\mathbb{P}\left(L(\widehat{f}) - L(f_P) \geq \epsilon\right) \leq \delta_{P,\mathcal{F}}(n, \epsilon).$$

Finite sample bounds and consistency

$$\mathbb{P}\Big(L(\widehat{f}) - L(f_P) \geq \epsilon\Big) \leq \delta_{P,\mathcal{F}}(n,\epsilon).$$

A bound of the above form implies:

- Convergence in probability (by definition).
- Convergence in expectation (by Markov's inequality).
- Almost sure convergence, provided

$$\sum_{n=1}^{\infty} \delta_{P,\mathcal{F}}(n,\epsilon) < \infty$$

(by the Borel–Cantelli lemma).

Sample bounds and generalization gap

Empirical risk

$$\widehat{L}(f) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, f(x_i))$$

Sample bounds and generalization gap

Empirical risk

$$\widehat{L}(f) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i))$$

A useful decomposition

$$L(\widehat{f}) - L(f_P) = \underbrace{L(\widehat{f}) - \widehat{L}(f)}_{\text{Generalization gap}} + \widehat{L}(f) - L(f_P)$$

No free-lunch theorem

A good algorithm should have small sample complexity for many distributions P.

No free-lunch theorem

A good algorithm should have small sample complexity for many distributions P.

No free-lunch (informal)

Is it not possible to have an algorithm with small (finite) sample complexity for **all** problems!

For any fixed algorithm there's a problem with arbitrary high sample complexity.

The rest of this course

- Models needed to learn!
- Computational and modeling principles= algorithms.
- Explore modern machine (deep) learning from first principles!