Introduction to Statistical Learning and Kernel Machines

Hichem SAHBI

CNRS UPMC

June 2018

Outline

Introduction to Statistical Learning

- Definitions
- Probability Tools
- Generalization Bounds
- Machine Learning Algorithms

Kernel Machines : Supervised and Unsupervised Learning

- The Representer Theorem
- Supervised Learning (Support vector machines and regression)
- Kernel Design (kernel combination, cdk kernels,...)
- Unsupervised Learning (kernel PCA and CCA)

Section 1

Introduction to Statistical Learning

What is Machine Learning

- Observe a phenomenon : images, weather, genes, etc.
- The inductive inference
  - Construct a model of the phenomenon : include a general rule from a set of observed (training) instances.
  - Make predictions.
- The transductive inference
  - Construct a model and make predictions from observed (training) instances to specific (test) ones.
- The goal of machine learning is to automate the inference.

Probabilistic Sampling & Notation

- Let $\mathcal{X}$ be an input space and $\mathcal{Y}$ an output space (in binary classification $\mathcal{Y} = \{-1, +1\}$).
- Data $((X, Y) \in \mathcal{X} \times \mathcal{Y})$: are instances with labels i.i.d according to P. (The distribution P is unknown.)
- A learning algorithm builds a function $g : \mathcal{X} \to \mathcal{Y}$ which assigns for a given observation X a label Y.
- [Un/Semi] Supervised learning means the labels (ground truth) is [Un/Partially] known.
- The overall goal is : how to make few mistakes on unseen instances.

Probabilistic Sampling & Notation

- Let $\mathcal{X}$ be an input space and $\mathcal{Y}$ an output space (in binary classification $\mathcal{Y} = \{-1, +1\}$).
- Data $((X, Y) \in \mathcal{X} \times \mathcal{Y})$: are instances with labels i.i.d according to P. (The distribution P is unknown.)
- A learning algorithm builds a function $g : \mathcal{X} \rightarrow \mathcal{Y}$ which assigns for a given observation X a label Y.
- [Un/Semi] Supervised learning means the labels (ground truth) is [Un/Partially] known.
- The overall goal is : how to make few mistakes on unseen instances.

Probabilistic Sampling & Notation

- Let $\mathcal{X}$ be an input space and $\mathcal{Y}$ an output space (in binary classification $\mathcal{Y} = \{-1, +1\}$).
- Data $((X, Y) \in \mathcal{X} \times \mathcal{Y})$: are instances with labels i.i.d according to P. (The distribution P is unknown.)
- A learning algorithm builds a function $g : \mathcal{X} \to \mathcal{Y}$ which assigns for a given observation X a label Y.
- [Un/Semi] Supervised learning means the labels (ground truth) is [Un/Partially] known.
- The overall goal is : how to make few mistakes on unseen instances.

Error Functions

- The classification function g is chosen to minimize the probability of error.
- This error is referred to as the expected risk or generalization error.

$$R(g) = P(g(X) \neq Y) = \mathbb{E}\left[1_{\{g(X)\neq Y\}}\right] \qquad \text{(Classification)}$$

$$R(g) = P\left(\mathbb{1}_{\{g(X)=g(X')\}} \neq \mathbb{1}_{\{Y=Y'\}}\right) \qquad \text{(Clustering)}$$

- Since P is unknown, we cannot measure directly this risk.
- This measure can only be estimated on a finite set.
- Empirical risk :

$$R_n(g) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{g(X_i) \neq Y_i\}}$$

Empirical Risk Minimization

- Let $\mathcal{G}$ be a set of possible functions with an a priori probability distribution.
- Choose g^* such that $g^* = \arg\min_{g \in \mathcal{G}} R_n(g)$.
- Is that enough !

Empirical Risk Minimization

- Let $\mathcal{G}$ be a set of possible functions with an a priori probability distribution.

- Choose g^* such that $g^* = \arg \min_{g \in \mathcal{G}} R_n(g)$.

- Is that enough !

Empirical Risk Minimization

- Let $\mathcal{G}$ be a set of possible functions with an a priori probability distribution.
- Choose g^* such that $g^* = \arg\min_{g \in \mathcal{G}} R_n(g)$.
- Is that enough !

Over-fitting/Under-fitting

- Data can be misleading.
- Over-fitting : good agreement with the training data but not with the test data. *It is always possible to build a function which fits exactly the data.*
- Under-fitting : model is too small to fit the data.
- Extra-validation can be used to detect such problems.
  For example : cross-validation, n-fold cross validation, etc.

Over-fitting/Under-fitting

- Data can be misleading.
- Over-fitting : good agreement with the training data but not with the test data. *It is always possible to build a function which fits exactly the data.*
- Under-fitting : model is too small to fit the data.
- Extra-validation can be used to detect such problems.
  For example : cross-validation, n-fold cross validation, etc.

Over-fitting/Under-fitting

- Data can be misleading.
- Over-fitting : good agreement with the training data but not with the test data. *It is always possible to build a function which fits exactly the data.*
- Under-fitting : model is too small to fit the data.
- Extra-validation can be used to detect such problems. For example : cross-validation, n-fold cross validation, etc.

Over-fitting/Under-fitting

- Data can be misleading.
- Over-fitting : good agreement with the training data but not with the test data. *It is always possible to build a function which fits exactly the data.*
- Under-fitting : model is too small to fit the data.
- Extra-validation can be used to detect such problems.
  For example : cross-validation, n-fold cross validation, etc.

Structural risk minimization

- Let a collection of models $\{G_d, d = 1...\}$ with an increasing complexity.
- Minimize the empirical risk in each model.
- Minimize the penalized empirical risk.

$$\min_{d} \left[\left(\min_{g \in \mathcal{G}_d} R_n(g) \right) + pen(d) \right]$$

- $pen(d)$ gives preference to models where the estimation error is small.
- $pen(d)$ measures the complexity or capacity of the model.

Structural risk minimization

- Let a collection of models $\{G_d, d = 1 \ldots\}$ with an increasing complexity.
- Minimize the empirical risk in each model.
- Minimize the penalized empirical risk.

$$\min_{d} \left[\left(\min_{g \in \mathcal{G}_d} R_n(g) \right) + pen(d) \right]$$

- $pen(d)$ gives preference to models where the estimation error is small.
- $pen(d)$ measures the complexity or capacity of the model.

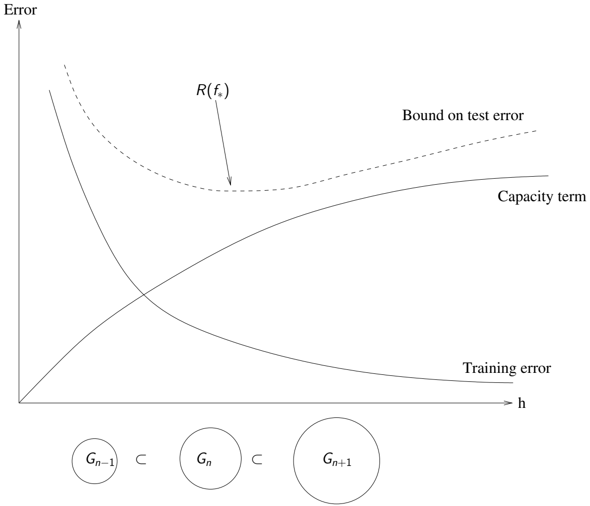
Structural risk minimization

- Let a collection of models $\{G_d, d = 1...\}$ with an increasing complexity.
- Minimize the empirical risk in each model.
- Minimize the penalized empirical risk.

$$\min_{d} \left[\left(\min_{g \in \mathcal{G}_d} R_n(g) \right) + pen(d) \right]$$

- $pen(d)$ gives preference to models where the estimation error is small.
- $pen(d)$ measures the complexity or capacity of the model.

Structural risk minimization



Regularization

- Choose a model $\mathcal{G}$.
- Choose a regulizer $\Omega(g)$ (e.g., $\Omega(g)$ can be ℓ_0, ℓ_1, ℓ_2, etc.)
- Minimize a regularized empirical risk (Tikhonov 1977) :

$$\min_{g \in \mathcal{G}} R_n(g) + \lambda\Omega(g), \qquad \lambda \geq 0$$

- Equivalent problems

1. Morozov (1984) : $\min_{g \in \mathcal{G} : R_n(g) \leq e} \Omega(g)$
2. Ivanov(1976) : $\min_{g \in \mathcal{G} : \Omega(g) \leq R} R_n(g)$

Regularization

- Choose a model $\mathcal{G}$.
- Choose a regulizer $\Omega(g)$ (e.g., $\Omega(g)$ can be ℓ_0, ℓ_1, ℓ_2, etc.)
- Minimize a regularized empirical risk (Tikhonov 1977) :

$$\min_{g \in \mathcal{G}} R_n(g) + \lambda\Omega(g), \qquad \lambda \geq 0$$

- Equivalent problems

1. Morozov (1984) : $\min_{g \in \mathcal{G} : R_n(g) \leq e} \Omega(g)$
2. Ivanov(1976) : $\min_{g \in \mathcal{G} : \Omega(g) \leq R} R_n(g)$

Regularization

- Choose a model $\mathcal{G}$.
- Choose a regulizer $\Omega(g)$ (e.g., $\Omega(g)$ can be ℓ_0, ℓ_1, ℓ_2, etc.)
- Minimize a regularized empirical risk (Tikhonov 1977) :

$$\min_{g \in \mathcal{G}} R_n(g) + \lambda\Omega(g), \qquad \lambda \geq 0$$

- Equivalent problems

1. Morozov (1984) : $\min_{g \in \mathcal{G} : R_n(g) \leq e} \Omega(g)$
2. Ivanov(1976) : $\min_{g \in \mathcal{G} : \Omega(g) \leq R} R_n(g)$

Regularization

- Choose a model $\mathcal{G}$.
- Choose a regulizer $\Omega(g)$ (e.g., $\Omega(g)$ can be ℓ_0, ℓ_1, ℓ_2, etc.)
- Minimize a regularized empirical risk (Tikhonov 1977) :

$$\min_{g \in \mathcal{G}} R_n(g) \; + \; \lambda \Omega(g), \qquad \lambda \geq 0$$

- Equivalent problems

1. Morozov (1984) : $\min_{g \in \mathcal{G} \, : \, R_n(g) \leq e} \Omega(g)$
2. Ivanov(1976) : $\min_{g \in \mathcal{G} \, : \, \Omega(g) \leq R} R_n(g)$

Approximation/Estimation

- What if the Bayes classifier is not in the model ?
- Risks

$$\begin{aligned} R^* &= \inf_g R(g) \quad \text{(Bayes risk)} \\ R(g^*) &= \inf_{g \in \mathcal{G}} R(g) \quad \text{(Best in a class)} \end{aligned}$$

- Decomposition.

$$R(g_n) - R^* = \underbrace{R(g^*) - R^*}_{approximation} + \underbrace{R(g_n) - R(g^*)}_{estimation}$$

- Only the estimation error is random (i.e., depends on the data).

Approximation/Estimation

- What if the Bayes classifier is not in the model ?
- Risks

$$\begin{aligned} R^* &= \inf_{g} R(g) \quad \text{(Bayes risk)} \\ R(g^*) &= \inf_{g \in \mathcal{G}} R(g) \quad \text{(Best in a class)} \end{aligned}$$

- Decomposition.

$$R(g_n) - R^* = \underbrace{R(g^*) - R^*}_{approximation} + \underbrace{R(g_n) - R(g^*)}_{estimation}$$

- Only the estimation error is random (i.e., depends on the data).

Approximation/Estimation

- What if the Bayes classifier is not in the model ?

- Risks

$$\begin{aligned} R^* &= \inf_g R(g) \quad \text{(Bayes risk)} \\ R(g^*) &= \inf_{g \in \mathcal{G}} R(g) \quad \text{(Best in a class)} \end{aligned}$$

- Decomposition.

$$R(g_n) - R^* = \underbrace{R(g^*) - R^*}_{\textit{approximation}} + \underbrace{R(g_n) - R(g^*)}_{\textit{estimation}}$$

- Only the estimation error is random (i.e., depends on the data).

Generalization bounds

- Given a dataset $(X_1, Y_1), ..., (X_n, Y_n)$ drawn from a probability distribution $P(X, Y)$. We want to build a function g_n (a classifier).
- The risk of g_n is a random quantity which depends on the data (it can be bounded).

From an empirical quantity	$R(g_n)$	$\leq$	$R_n(g_n)$	$+$	B
Best in a class	$R(g_n)$	$\leq$	$R(g^*)$	$+$	B
Bayes risk	$R(g_n)$	$\leq$	R^*	$+$	B

Generalization bounds

- Vapnik & Chervonenkis

  With a probability at least $1-\delta$; we have $\forall g_n \in \mathcal{G}$

$$R(g_n) \leq R_n(g_n) + \frac{2[h \ \log \frac{2n}{h} \ + \ \log \frac{2}{\delta}]}{n}$$

- h is related to the capacity of the model (VC dimension).

Section 2

Probability Tools

Some Probability Tools

- Union bound
  $P(A \cup B) = P(A) + P(B) - P(A \cap B) \leq P(A) + P(B)$.

- Inclusion if $A \Rightarrow B$, $P(A) \leq P(B)$.

- Inversion $P(X \geq t) \leq F(t) \;\Rightarrow\; P(X \leq F^{-1}(\delta)) \geq 1 - \delta$, with $\delta = F(t)$.

- Expectation if $X \geq 0$, we have $\mathbb{E}[X] = \int_0^{\infty} P(X \geq t)\, dt$

Probability Tools : Basic Inequalities

- Jensen : if f is convex $f(\mathbb{E}(X)) \ \leq \ \mathbb{E}(f(X))$.
- Markov : if $X \geq 0$ then $\forall t > 0,\ P(X \geq t) \ \leq \ \mathbb{E}(X)/t$.
- Chebyshev : $\forall t > 0,\ P(|X - \mathbb{E}(X)| \geq t) \ \leq \ Var(X)/t^2$.

Jensen (sketch of the proof)

- Let X be a Bernoulli RV which takes its value in $\{X_1, X_2\}$ and $\{p, 1-p\}$ its probability distribution.

- We have :

$$\begin{aligned}
\mathbb{E}(X) &= X_1\ p\ +\ X_2\ (1-p) \\
f(\mathbb{E}(X)) &= f\,(X_1\ p\ +\ X_2\ (1-p)) \\
&\leq p\ f(X_1)\ +\ (1-p)\ f(X_2) \\
&\leq \mathbb{E}\,(f(X))\ \square
\end{aligned}$$

Markov (the proof)

$$
\begin{aligned}
\mathbb{E}(X) &= \int_0^\infty x\ f(x) \\
&= \int_0^t x\ f(x) + \int_t^\infty x\ f(x) \\
&\geq \int_t^\infty x\ f(x) \\
&\geq \int_t^\infty t\ f(x) \\
&\geq t \int_t^\infty f(x)
\end{aligned}
$$

We have $\mathbb{E}(X) \geq t\ P(X \geq t)$, hence :

$$P(X \geq t) \leq \mathbb{E}(X) \ / \ t \ \square$$

Chebyshev (the proof)

- Chebyshev :
  Using Markov, we have :

$$
\begin{aligned}
\forall t > 0, P\left(X \geq t^2\right) &\leq \mathbb{E}(X)/t^2 \\
\forall t > 0, P\left((X - \mathbb{E}[X])^2 \geq t^2\right) &\leq \mathbb{E}\left[(X - \mathbb{E}(X))^2\right]/t^2 \\
\Rightarrow \quad P\left(|X - \mathbb{E}[X]| \geq t\right) &\leq Var(X)/t^2 \quad \square
\end{aligned}
$$

Section 3

Generalization Bounds

What we need ?

- We need to bound $P\big(R(g) - R_n(g) \; \geq \; \epsilon\big)$, with $g \in \mathcal{G}$.
- Loss class : for a given class of functions $\mathcal{G}$

$$\mathcal{F}(\mathcal{G}) = \mathcal{F} \; = \; \{f : (X, Y) \mapsto 1_{\{g(X) \neq Y\}} : g \in \mathcal{G}\}$$

- There is a bijection between $\mathcal{F}$ and $\mathcal{G}$.
- The quantity of interest $R(f) - R_n(f)$.

What we need ?

- We need to bound $P\big(R(g) - R_n(g) \ \geq \ \epsilon\big)$, with $g \in \mathcal{G}$.

- Loss class : for a given class of functions $\mathcal{G}$

$$\mathcal{F}(\mathcal{G}) = \mathcal{F} \ = \ \big\{f : (X, Y) \mapsto 1_{\{g(X) \neq Y\}} : g \in \mathcal{G}\big\}$$

- There is a bijection between $\mathcal{F}$ and $\mathcal{G}$.

- The quantity of interest $R(f) - R_n(f)$.

What we need ?

- We need to bound $P\big(R(g) - R_n(g) \ \geq \ \epsilon\big)$, with $g \in \mathcal{G}$.
- Loss class : for a given class of functions $\mathcal{G}$

$$\mathcal{F}(\mathcal{G}) = \mathcal{F} \ = \ \big\{f : (X, Y) \mapsto 1_{\{g(X) \neq Y\}} : g \in \mathcal{G}\big\}$$

- There is a bijection between $\mathcal{F}$ and $\mathcal{G}$.

$$\begin{aligned} R(g) &= R(f) &&= \mathbb{E}\left[f(X, Y)\right] = \mathbb{E}\left[f(Z)\right] \\ R_n(g) &= R_n(f) &&= \frac{1}{n}\sum_{i=1}^{n} f(X_i, Y_i) = \frac{1}{n}\sum_{i=1}^{n} f(Z_i) \end{aligned}$$

with $Z = (X, Y)$ and $Z_i = (X_i, Y_i)$

- The quantity of interest $R(f) - R_n(f)$.

What we need ?

- We need to bound $P\big(R(g) - R_n(g) \ \geq \ \epsilon\big)$, with $g \in \mathcal{G}$.

- Loss class : for a given class of functions $\mathcal{G}$

$$\mathcal{F}(\mathcal{G}) = \mathcal{F} \ = \ \big\{f : (X, Y) \mapsto 1_{\{g(X) \neq Y\}} : g \in \mathcal{G}\big\}$$

- There is a bijection between $\mathcal{F}$ and $\mathcal{G}$.

$$\begin{aligned}
R(g) \quad &= \quad R(f) \quad = \mathbb{E}\left[f(X, Y)\right] = \mathbb{E}\left[f(Z)\right] \\
R_n(g) \quad &= \quad R_n(f) \quad = \frac{1}{n}\sum_{i=1}^{n} f(X_i, Y_i) \ = \ \frac{1}{n}\sum_{i=1}^{n} f(Z_i)
\end{aligned}$$

with $Z = (X, Y)$ and $Z_i = (X_i, Y_i)$

- The quantity of interest $R(f) - R_n(f)$.

The law of large numbers

Definition : the average of the results obtained from a large number of trials should be close to the expected value.

Example :

- Suppose we toss a coin n times (n is very large).
- The expected number of heads (m) will be approximately $n/2$.
- As m gets far from $n/2$, the probability to have m heads is small (and vice-versa).

In our case : for any $\epsilon > 0$,

$$P\left(\left|\mathbb{E}[f(Z)] - \frac{1}{n}\sum_{i=1}^{n} f(Z_i)\right| \geq \epsilon\right) \to 0 \quad \text{as } n \to \infty$$

The law of large numbers (Proof)

Using Chebyshev inequality, we have :

$$P\left(\left| \, \mathbb{E}[f(Z)] \; - \; \frac{f(Z_1)+\cdots+f(Z_n)}{n}\right| \; \geq \; \epsilon\right) \;\; \leq \;\; \frac{Var\left(\frac{f(Z_1)+\ldots+f(Z_n)}{n}\right)}{\epsilon^2}$$

$$\begin{aligned} Var\left(\frac{f(Z_1)+\ldots+f(Z_n)}{n}\right) &= Var\left(\frac{f(Z_1)}{n}\right)+\ldots+Var\left(\frac{f(Z_n)}{n}\right) \\ &= \left(\frac{\sigma^2}{n^2}+\ldots+\frac{\sigma^2}{n^2}\right)=\frac{\sigma^2}{n} \end{aligned}$$

$$P\left(\left|\mathbb{E}[f(Z)] \; - \; \frac{f(Z_1)+\cdots+f(Z_n)}{n}\right| \; \geq \; \epsilon\right) \leq \frac{\sigma^2}{n\epsilon^2} \;\; \rightarrow \;\; 0 \quad \text{as} \quad n\rightarrow\infty$$

Hoeffding's inequality

- **Theorem :** *Let* $Z_1, ..., Z_n$ *n i.i.d random variables. If* $f(Z) \in [a, b]$ *then* $\forall \epsilon > 0$, *we have :*

$$P\left(\mathbb{E}\left[f(Z)\right] \;-\; \frac{1}{n}\sum_{i=1}^{n} f(Z_i) \;\geq\; \epsilon\right) \;\leq\; 2\; exp\; \left(-\frac{2\;n\;\epsilon^2}{(b-a)^2}\right)$$

Simple G.B. using Hoeffding's inequality

- Using Inversion and Hoeffding's

$$P(X \geq \epsilon) \leq F(\epsilon) \;\Rightarrow\; P(X \leq F^{-1}(\delta)) \geq 1 - \delta$$

$$P\left(\mathbb{E}\left[f(Z)\right] \;-\; \frac{1}{n}\sum_{i=1}^{n} f(Z_i) \;\geq\; \epsilon\right) \;\leq\; 2\ exp\left(-\frac{2\ n\ \epsilon^2}{(b-a)^2}\right)$$

- Let $\delta = 2\ exp\left(-\frac{2\ n\ \epsilon^2}{(b-a)^2}\right)$.

$$P\left(\mathbb{E}\left[f(Z)\right] \;-\; \frac{1}{n}\sum_{i=1}^{n} f(Z_i) \;\geq\; (b-a)\sqrt{\frac{\log\frac{2}{\delta}}{2n}}\right) \;\leq\; \delta$$

$$P\left(R(f) - R_n(f) \;\leq\; (b-a)\sqrt{\frac{\log\frac{2}{\delta}}{2n}}\right) \;\geq\; 1 - \delta$$

Simple G.B. using Hoeffding's inequality

- Using Inversion and Hoeffding's

$$P(X \geq \epsilon) \leq F(\epsilon) \;\Rightarrow\; P(X \leq F^{-1}(\delta)) \geq 1-\delta$$

$$P\left(\mathbb{E}\left[f(Z)\right] - \frac{1}{n}\sum_{i=1}^{n} f(Z_i) \geq \epsilon\right) \leq 2\ exp\left(-\frac{2\ n\ \epsilon^2}{(b-a)^2}\right)$$

- Let $\delta = 2\ exp\left(-\frac{2\ n\ \epsilon^2}{(b-a)^2}\right)$.

$$P\left(\mathbb{E}\left[f(Z)\right] - \frac{1}{n}\sum_{i=1}^{n} f(Z_i) \geq (b-a)\sqrt{\frac{\log\frac{2}{\delta}}{2n}}\right) \leq \delta$$

$$P\left(R(f) - R_n(f) \leq (b-a)\sqrt{\frac{\log\frac{2}{\delta}}{2n}}\right) \geq 1-\delta$$

Simple G.B. using Hoeffding's inequality

- Using Inversion and Hoeffding's

$$P(X \geq \epsilon) \leq F(\epsilon) \;\Rightarrow\; P(X \leq F^{-1}(\delta)) \geq 1-\delta$$

$$P\left(\mathbb{E}\left[f(Z)\right] \;-\; \frac{1}{n}\sum_{i=1}^{n} f(Z_i) \;\geq\; \epsilon\right) \;\leq\; 2\ exp\left(-\frac{2\ n\ \epsilon^2}{(b-a)^2}\right)$$

- Let $\delta = 2\ exp\left(-\frac{2\ n\ \epsilon^2}{(b-a)^2}\right)$.

$$P\left(\mathbb{E}\left[f(Z)\right] \;-\; \frac{1}{n}\sum_{i=1}^{n} f(Z_i) \;\geq\; (b-a)\sqrt{\frac{\log\frac{2}{\delta}}{2n}}\right) \;\leq\; \delta$$

$$P\left(R(f) - R_n(f) \;\leq\; (b-a)\sqrt{\frac{\log\frac{2}{\delta}}{2n}}\right) \;\geq\; 1-\delta$$

Hoeffding's inequality

- This bound is for a fixed function f (or g) and the bound is with respect to the sampling of the data.
- If the function changes with the data, this bound is not valid.
- For a given function, only fraction of the data will satisfy the inequality.

Union bound

- Before seeing the data, we do not know which function the algorithm will choose.
- We need a bound which holds for all functions in a class.

$$P\Big(\sup_{f \in \mathcal{F}} R(f) - R_n(f) \geq \epsilon\Big) \leq \sum_{f\in\mathcal{F}} P\Big(R(f) - R_n(f) \geq \epsilon\Big)$$
$$\leq 2N \; exp\left(-2n\epsilon^2\right)$$

here $N = \#\mathcal{F} = \#\mathcal{G}$

Union bound

- Let $\delta \; = \; 2N \; exp\left(-2n\epsilon^2\right)$.
- Using inversion, we can show that $\forall \delta > 0$, with probability at least $1 - \delta$, we have :

$$\forall g \in \mathcal{G}, \;\; R(g) - R_n(g) \;\; \leq \;\; \sqrt{\frac{\log N \;\; + \;\; \log \frac{2}{\delta}}{2n}}$$

- $\log N$ can be thought as the number of bits to specify a function in $\mathcal{G}$.
- N controls the trade-off ($R_n(g)$ decreases with N while the bound increases with N).

Sum Up

- For a fixed function, for most of the samples :

$$R(g) - R_n(g) \;\leq\; O\left(\frac{1}{\sqrt{n}}\right)$$

- For most of the samples, if $|\mathcal{G}| = N$:

$$\sup_{g \in \mathcal{G}} R(g) - R_n(g) \;\leq\; O\left(\sqrt{\frac{\log N}{n}}\right)$$

- Can be improved since :
  1. Union bounds are as bad as if the classifiers are independent.
  2. Supremum is not what the algorithm chooses.
  3. We can extend it to the infinite classes of functions.

VC Theory

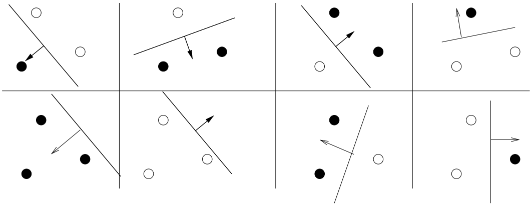
The VC dimension

- This is a measure of the capacity of a class of hypotheses.
- This is the maximal size of a training set that can be separated (whatever the labeling of the data).
- Depends of course on the geometry of a class.
- if :

$$\begin{aligned} \mathcal{G}_1 &= \{\text{set of rectangles}\} \\ \mathcal{G}_2 &= \{\text{set of lines}\} \end{aligned}$$

- $VC(\mathcal{G}_1) \neq VC(\mathcal{G}_2)$.
- Not necessarily related to the number of parameters.

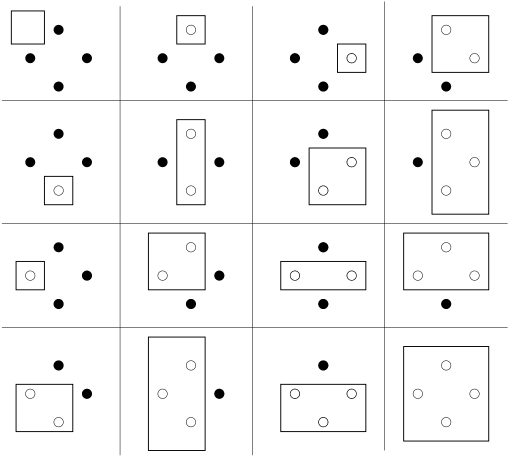
The VC dimension



- The VC dimension in R^2 is 3.
- In R^d the VC dimension of a set of hyper-planes is $d+1$.

The VC dimension

- Rectangles in R^2.

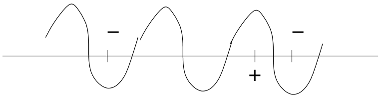


The VC dimension

- The VC dimension does not reflect always the number of parameters as :

$$\mathcal{G} = \left\{ \, sgn\left[sin(\omega x)\right] , \; \omega \in \mathbb{R}^+ \right\}$$

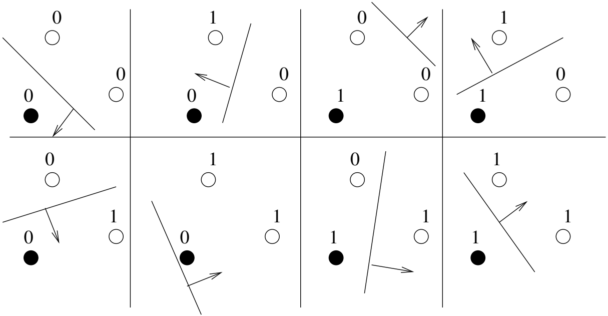
has an infinite VC dimension. (We can always choose ω as small as possible to guarantee the separation of the data.)



- VC-dimension is distribution independent and may be infinite.
- The class of hyperplanes in R^∞ has infinite VC dimension.

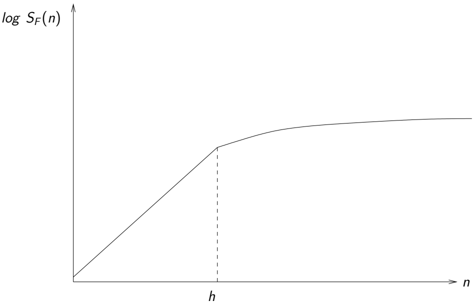
Function class

- How to measure size of an infinite class of functions $\mathcal{F}$ (or $\mathcal{G}$) ?
- Function class : restriction of $\mathcal{F}$ on a finite subset $\{Z_1, \ldots, Z_n\}$ denoted $\mathcal{F}_{Z_1,\ldots,Z_n} = \{(f(Z_1), \ldots, f(Z_n)) : f \in \mathcal{F}\}$.
- This set corresponds to different ways the function f responds on the set $\{Z_1, \ldots, Z_n\}$.



Growth Function

- This is defined as the max size of the function class $S_F(n) = \max_{Z_1,\dots,Z_n} |\mathcal{F}_{Z_1,\dots,Z_n}|$.
- $S_F(n) = 2^n$ if $(n \leq h)$ or equivalently $\log S_F(n) = n$
- $S_F(n) \leq 2^n$ if $(n \geq h)$ or equivalently $\log S_F(n) \leq n$



Proof

- If $n \leq h$: $S_F(n) = 2^n$

  $$(n \leq h) \quad \Rightarrow \quad \exists\ (X_1, Y_1), ..., (X_i, Y_i), ..., (X_n, Y_n)$$

  $$\exists f \ \in \ \mathcal{F} \quad \text{s.t.} \quad f(X_1, Y_1) = 0, ..., f(X_i, Y_i) = 0, ..., f(X_n, Y_n) = 0$$

  If we need $f(X_i, Y_i) = 1$, this is equivalent to switch Y_i and find another $f^{'} \in \mathcal{F}$ such that $f^{'}(X_i, 1 - Y_i) = 0$.

- If $n > h$: $S_F(n) \leq 2^n$ (by enumeration when $n = 4$ in 2D).

- The VC-dimension h is equal to the largest n such that $S_F(n) = 2^n$.

Infinite Class Generalization Bounds

VC-Bound

With a probability at least $1 - \delta$:

$$\forall g \in \mathcal{G}, \quad R(g) - R_n(g) \leq \sqrt{\frac{\log \ S_F(2n) + \log \ \frac{4}{\delta}}{8 \ n}}$$

Symmetrization

- **Lemma** : Let Z_1, ..., Z_n, (resp. $Z_1^{'}$, ..., $Z_n^{'}$) an independent sample and R_n (resp. $R_n^{'}$) the underlying empirical measure.

Provided that $n\epsilon^2 \geq 2$, $\forall \epsilon$

$$P\left(\sup_{f \in \mathcal{F}} R(f) - R_n(f) \geq \epsilon\right) \quad \leq \quad 2P\left(\sup_{f \in \mathcal{F}} R_n^{'}(f) - R_n(f) \geq \epsilon/2\right)$$

Symmetrization (proof)

$$P(R_n'(f) - R_n(f) > \epsilon/2) \geq P(R(f) - R_n(f) > \epsilon).P(R(f) - R_n'(f) < \epsilon/2)$$

$$\{R_n'(f) - R_n(f) > \epsilon/2\} \Leftarrow \{R(f) - R_n(f) > \epsilon \ \wedge \ R(f) - R_n'(f) < \epsilon/2\}$$

$$\begin{aligned} & P\left(R_n'(f) - R_n(f) > \epsilon/2\right) \\ \geq \quad & P\left(R(f) - R_n(f) > \epsilon\right).\left(1 - P\left(R(f) - R_n'(f) \geq \epsilon/2\right)\right) \end{aligned}$$

Using Chebychev

$$\begin{aligned} P\left(R(f) - R_n'(f) \geq \epsilon/2\right) \quad & \leq \quad \frac{Var\left[\frac{1}{n}\sum \ f(Z_i')\ \right]}{(\epsilon/2)^2} = \frac{\frac{1}{n^2}\sum Var\left[f(Z_i')\right]}{\epsilon^2/4} \\ & = \quad \frac{\frac{n}{n^2}Var\left[f(Z_*)\right]}{\epsilon^2/4} = 4\frac{Var\left[f(Z_*)\right]}{n\epsilon^2} \end{aligned}$$

Symmetrization (proof)

- We have : $\frac{4Var\left[f(Z_*)\right]}{n\ \epsilon^2} \leq \frac{1}{n\epsilon^2}$ since $f(Z_*)$ is a Bernoulli random variable with a variance bounded by $1/4$.

$$P\left(R(f) - R_n^{'}(f) \geq \epsilon/2\right) \ \leq \ \frac{1}{n\epsilon^2}$$

- Hence

$$\begin{aligned}
& P\left(R_n^{'}(f) - R_n(f) > \epsilon/2\right) \\
\geq \quad & P\left(R(f) - R_n(f) > \epsilon\right).\left(1 - \frac{1}{n\epsilon^2}\right) \\
\geq \quad & \frac{1}{2}\,P\left(R(f) - R_n(f) > \epsilon\right) \qquad \text{Since} \quad n\epsilon^2 \ \geq \ 2 \\
\Rightarrow \quad & P\left(R(f) - R_n(f) > \epsilon\right) \leq 2\ P\left(R_n^{'}(f) - R_n(f) > \epsilon/2\right) \ \square
\end{aligned}$$

VC-Bound

Using symmetrization, we have :

$$
\begin{aligned}
P\left(\sup_{f\in\mathcal{F}} R(f) - R_n(f) \geq \epsilon\right) &\leq 2P\left(\sup_{f\in\mathcal{F}} R_n^{'}(f) - R_n(f) \geq \epsilon/2\right) \\
&= 2\ P\left(\sup_{f\in\mathcal{F}_{z_1,\dots,z_n,z_1^{'},\dots,z_n^{'}}} R_n^{'}(f) - R_n(f) \geq \epsilon/2\right) \\
&\leq 4\ S_F(2n)\ e^{-\frac{n\epsilon^2}{8}} \ \text{(Using Hoeffdings's ineq.)}
\end{aligned}
$$

By inversion, we have with a probability at least $1-\delta$:

$$
\forall g\ \in\ \mathcal{G},\quad R(g) - R_n(g)\ \leq\ \sqrt{\frac{\log\ S_F(2n) + \log\ \frac{4}{\delta}}{8\ n}}
$$

VC-Entropy

- VC-dimension is distribution independent. (The same bound holds for any distribution).
- It is loose for many distributions.
- The VC-entropy is a measure which is always finite.

Section 4

Machine Learning Algorithms

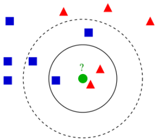
Non-Parametric vs. Parametric Learning

- Parametric : a learning model that summarizes data with a set of parameters of fixed size (independent of the number of training examples) is called a parametric model.
  Examples : Linear Discriminant Analysis, Perceptron, Naive Bayes, Neural Networks, etc.

- Non-Parametric : algorithms that do not make strong assumptions about the form of the mapping function are called non-parametric machine learning algorithms.
  Examples : k-Nearest Neighbors, Support Vector Machines, etc.

Non-Parametric Classifiers : k-Nearest Neighbors

- Given $\mathcal{T} = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, a given test sample X is assigned to the most common class Y using majority vote

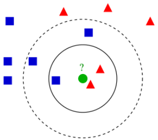


$$Y = \textbf{argmax}_{y \in \{1,\ldots,C\}} \sum_{X_j \in \mathcal{N}_k(X)} 1_{\{Y_j = y\}}$$

- The distance can be Euclidean for continuous variables or other metrics (as Hamming) for discrete variables (e.g. text). Distance can also be learned.

Non-Parametric Classifiers : k-Nearest Neighbors

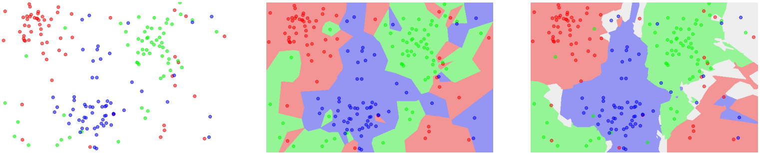
- Given $\mathcal{T} = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, a given test sample X is assigned to the most common class Y using majority vote



$$Y = \mathbf{argmax}_{y \in \{1,\ldots,C\}} \sum_{X_j \in \mathcal{N}_k(X)} 1_{\{Y_j = y\}}$$

- The distance can be Euclidean for continuous variables or other metrics (as Hamming) for discrete variables (e.g. text). Distance can also be learned.

Non-Parametric Classifiers : k-Nearest Neighbor

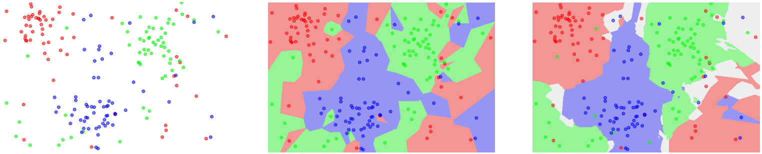


- For small k, the "majority voting" can be severely degraded by noise. Another drawback occurs when the class distribution is imbalanced (frequent class tends to dominate the prediction).
- One way to overcome this problem is to use weights.

$$Y = \mathbf{argmax}_{y \in \{1,\dots,C\}} \ \frac{1}{N_y} \sum_{X_j \in \mathcal{N}_k(X)} 1_{\{Y_j = y\}}$$

- For very-high-dimensional datasets (videos), running a fast approximate k-NN search (e.g., locality sensitive hashing, random projections, etc.) is necessary.

Non-Parametric Classifiers : k-Nearest Neighbor

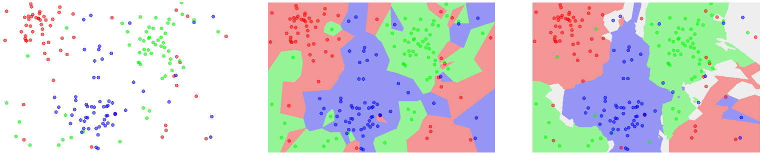


- For small k, the "majority voting" can be severely degraded by noise. Another drawback occurs when the class distribution is imbalanced (frequent class tends to dominate the prediction).
- One way to overcome this problem is to use weights.

$$Y = \textbf{argmax}_{y \in \{1,\dots,C\}} \ \frac{1}{N_y} \sum_{X_j \in \mathcal{N}_k(X)} 1_{\{Y_j = y\}}$$

- For very-high-dimensional datasets (videos), running a fast approximate k-NN search (e.g., locality sensitive hashing, random projections, etc.) is necessary.

Non-Parametric Classifiers : k-Nearest Neighbor



- For small k, the "majority voting" can be severely degraded by noise. Another drawback occurs when the class distribution is imbalanced (frequent class tends to dominate the prediction).
- One way to overcome this problem is to use weights.

$$Y = \textbf{argmax}_{y\in\{1,\dots,C\}} \quad \frac{1}{N_y} \sum_{X_j \in \mathcal{N}_k(X)} 1_{\{Y_j=y\}}$$

- For very-high-dimensional datasets (videos), running a fast approximate k-NN search (e.g., locality sensitive hashing, random projections, etc.) is necessary.

Parametric Classifiers : Naive Bayes

- Naive Bayes classifiers are a family of probabilistic classifiers based on Bayes' theorem with strong (naive) independence assumptions between the variables.
- Given an instance X to be classified, represented by a vector $X = (x_1, \dots, x_d)$ of independent variables

$$Y = \textbf{argmax}_{y \in \{1,\dots,C\}} P(y|X)$$

- For example, a fruit $X = (\textit{red}, \ \textit{round}, \ 10\textit{cm diameter})$ is likely to be $Y = \textit{apple}$ (regardless of possible correlations between its color, shape and its diameter).

Parametric Classifiers : Naive Bayes

- Using Bayes' theorem, the conditional probability can be decomposed as

$$Y = \textbf{argmax}_{y\in\{1,\ldots,C\}} \frac{P(y)P(X|y)}{P(X)} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

- Using the chain rule and independence ;

$$P(X|y) = P(x_1,\ldots,x_d|y) = \prod_{i=1}^{d} P(x_i|y)$$

$$P(y|X) = P(y|x_1,\ldots,x_d) = \frac{1}{P(X)} P(y) \prod_{i=1}^{d} P(x_i|y)$$

$$P(X) = \sum_{\ell=1}^{C} P(y_\ell)P(X|y_\ell)$$

Parametric Classifiers : Naive Bayes

- Example : given a test data $X = (\text{weight}, \text{height}, \text{foot size})$. Y male or female ?

$$P(male|X) = \frac{P(male)P(weight|male)P(height|male)P(footsize|male)}{evidence}$$

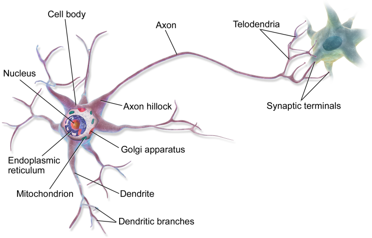
$$P(female|X) = \frac{P(female)P(weight|female)P(height|female)P(footsize|female)}{evidence}$$

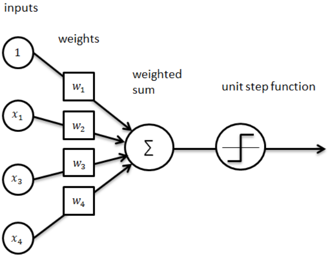
$$\begin{aligned} evidence &= P(male)P(weight|male)P(height|male)P(footsize|male) \\ &+ P(female)P(weight|female)P(height|female)P(footsize|female) \end{aligned}$$

- Let $X = (90, 1.90, 46)$, $P(male|X) > P(female|X)$, so $Y = male$

Parametric Classifiers : Perceptron

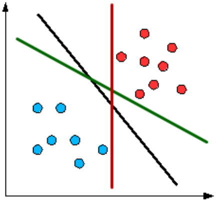
- The perceptron (called also single layer perceptron) is a simplified model of a biological neuron

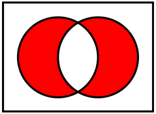


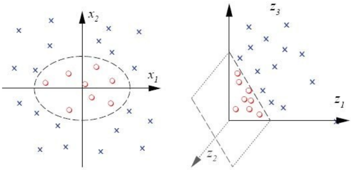


- It is a model for learning a binary classifier : a function that maps its input X (a real-valued vector) to an output value $g(X) = 1_{\{\langle w,X\rangle + b > 0\}} = 1_{\{\sum_i w_i x_i + b > 0\}}$ (a single binary value) :

Parametric Classifiers : Perceptron







- Existing perceptron learning algorithm does not terminate if the learning set is not linearly separable (eg. exclusive or).
- The perceptron of optimal stability/robustness is known as linear SVM.
- The non separable case can be solved using the kernel trick (kernel SVMs).
- Sometimes, the best classifier is not necessarily that separate all the training data perfectly.

General Conclusion

- Generalization bounds in machine learning are useful (at least) :
  - To understand the capacity and the behavior of a family of classifiers (or models in general) under some specific regimes (large/small training data, etc.)
  - To derive new learning algorithms (max margins lead to low VC dimension, better generalization and hence to SVMs, etc.)
  - To derive the best parameters, etc.

Outline

Introduction to Statistical Learning

- Definitions
- Probability Tools
- Generalization Bounds
- Machine Learning Algorithms

Kernel Machines : Supervised and Unsupervised Learning

- The Representer Theorem
- Supervised Learning (Support vector machines and regression)
- Kernel Design (kernel combination, cdk kernels,...)
- Unsupervised Learning (kernel PCA and CCA)

Outline

Introduction to Statistical Learning

- Definitions
- Probability Tools
- Generalization Bounds
- Machine Learning Algorithms

Kernel Machines : Supervised and Unsupervised Learning

- The Representer Theorem
- Supervised Learning (Support vector machines and regression)
- Kernel Design (kernel combination, cdk kernels,...)
- Unsupervised Learning (kernel PCA and CCA)