Machine Learning

Marc Sebban & Amaury Habrard

Hubert Curien lab, UMR CNRS 5516
University of Jean Monnet Saint-Étienne (France)

Academic year 2016-2017

Outline of the Course

1. Lecture 1: Introduction to ML - Supervised Learning (MS)
2. Lecture 2: Linear/Polynomial/Logistic Regression (MS)
3. Lecture 3: Sparsity in Convex Optimization - K Nearest-Neighbors (MS)
4. Lecture 4: Support Vector Machines (AH)
5. Lecture 5: Neural Networks and Deep-Learning (AH)
6. Lecture 6: Theory of Boosting (MS)
7. Lecture 7: Unsupervised Learning (PCA and k-Means), Metric Learning (MS)
8. Lecture 8: Domain Adaptation (AH)

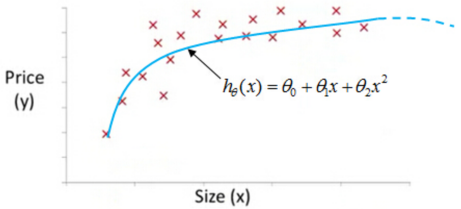
Practical Sessions and Tutorials (AH)

scikit-learn (`http://scikit-learn.org/`) and Tensor Flow (`https://www.tensorflow.org/`)

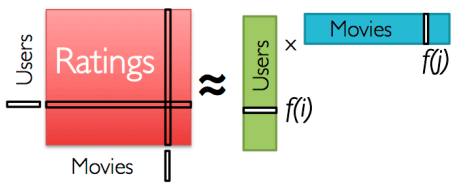
What is Machine Learning?

Machine Learning is about the construction and study of algorithms that can automatically learn from data. It covers two main settings:

- **Supervised learning** to deal with *classification*, *regression* or *ranking* tasks $\rightarrow$ **prediction from labeled data**

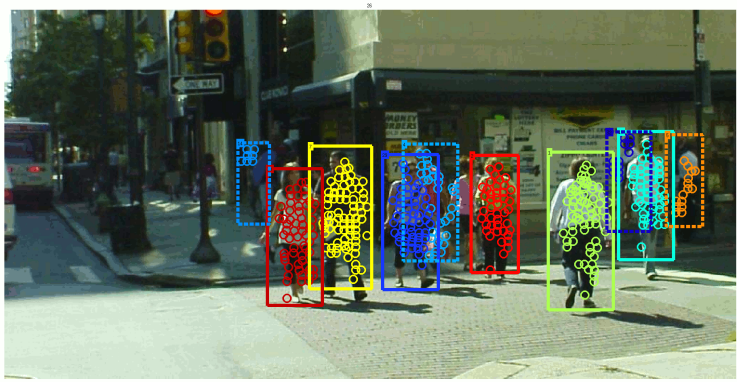


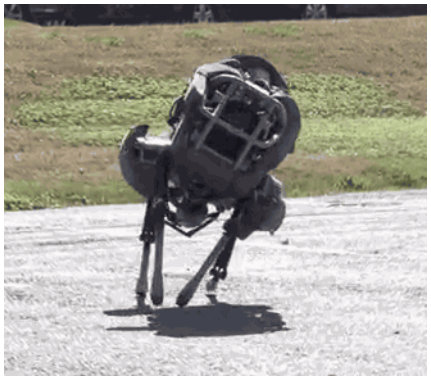
- **Unsupervised learning** to address *clustering* or *dimensionality reduction* tasks $\rightarrow$ **find the underlying structure of unlabeled data**

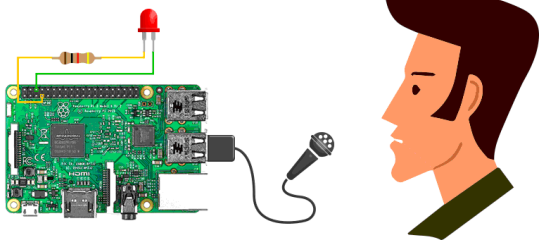


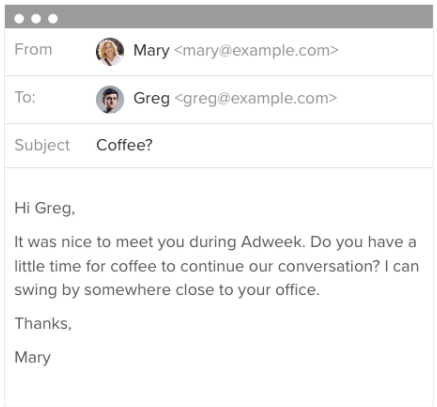
Some Machine Learning applications

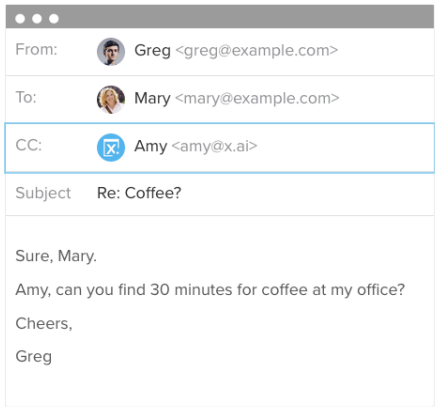
Successful domains of application of Machine Learning include **computer vision, robotics, speech recognition, natural language processing**, etc.



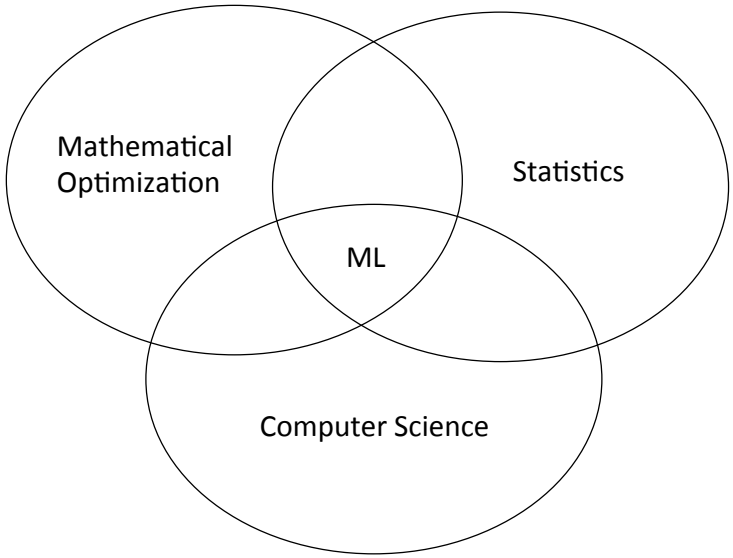








Required skills in Machine Learning



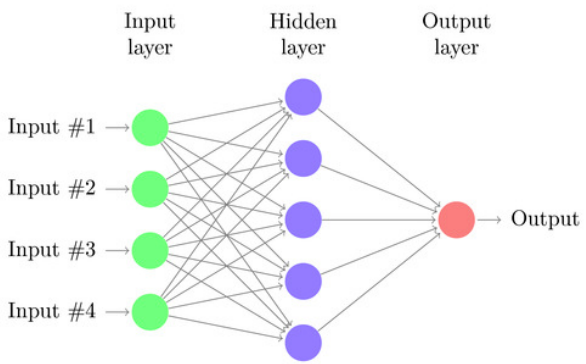
We can deal with Machine Learning from two complementary points of view:

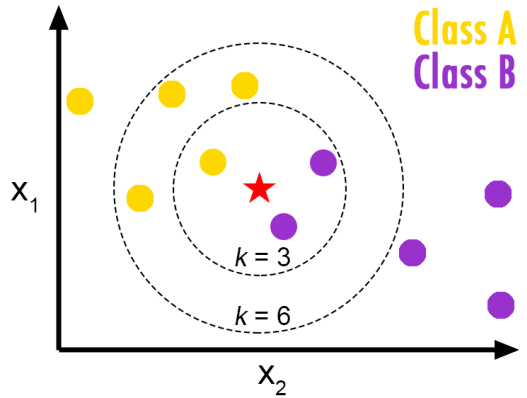
1. **Algorithmic**: How to design new efficient algorithms able to address new learning problems (big data, noisy data, imbalancy, etc.)?
2. **Theoretical**: How to derive theoretical guarantees about the (at least, asymptotic) behavior of the algorithms? What are the conditions to learn well?

Popular Supervised Learning Algorithms

See some GUI demos from Standford University:
`http://cs.stanford.edu/people/karpathy/svmjs/demo/`

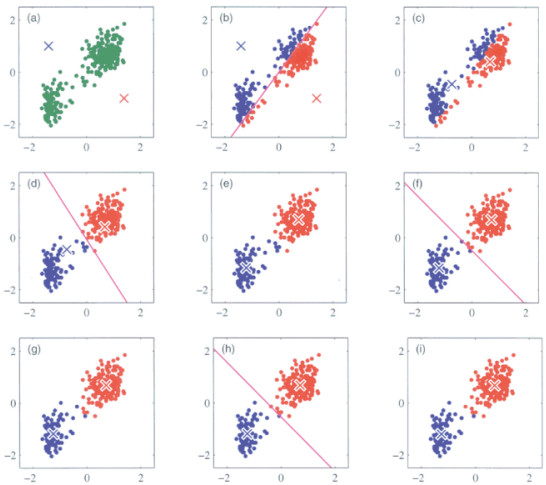
Learning Algorithm	**Learned model**
Regression	Polynomial
Support Vector Machines	Hyperplane
Neural networks	Weights of the architecture
k-Nearest Neighbors	Lazy algorithm

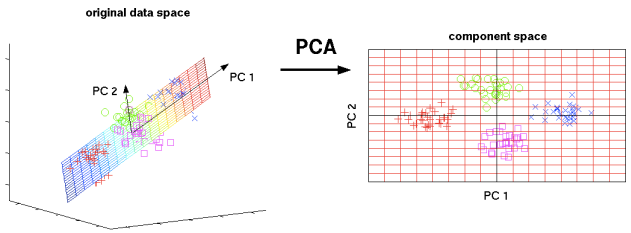


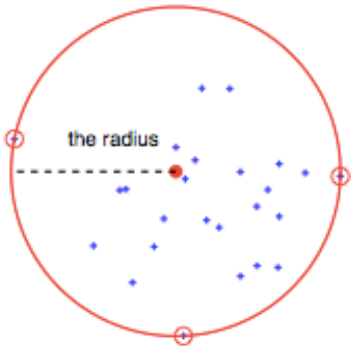


Popular Unsupervised Learning Algorithms

Learning Algorithm	**Learned model**
k-Means	centroids, clusters
Principal Component Analysis	latent variables (embedding space)
Maximum Excluding Balls	radius, center







Outline

Some references

- **Machine Learning**, *Tom Mitchell*, MacGraw Hill, 1997
- **Statistical Learning Theory**, *V. Vapnik*, 1989
- **Pattern Recognition and Machine Learning**, *M. Bishop*, 2013
- **Pattern Recognition**, *S.Theodoridis, K. Koutroumbas*, 2015.
- **Convex Optimization**, *Stephen Boyd & Lieven Vandenberghe*, Cambridge University Press, 2012.
- **On-line courses**: Coursera `https://www.coursera.org/`, Andrew NG's courses

Supervised learning problem

Notations

- *Let* $S = \{z_i = (\mathbf{x_i}, y_i)\}_{i=1}^m$ *be a set of* m *training examples independently and identically distributed (i.i.d.) from an unknown joint distribution* $\mathcal{D}_\mathcal{Z}$ *over a space* $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$.
- *The* $\mathbf{x_i}$ *values* ($\mathbf{x_i} \in \mathcal{X}$) *are typically vectors in* $\mathbb{R}^d$ *whose components are usually called features.*
- *The* y *values* ($y \in \mathcal{Y}$) *are drawn from a discrete set of classes/labels (typically* $\mathcal{Y} = \{-1, +1\}$ *in binary classification) or are continuous values (regression).*
- *We assume that there exists a* **target function** f *such that* $y = f(\mathbf{x}), \forall (\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$.

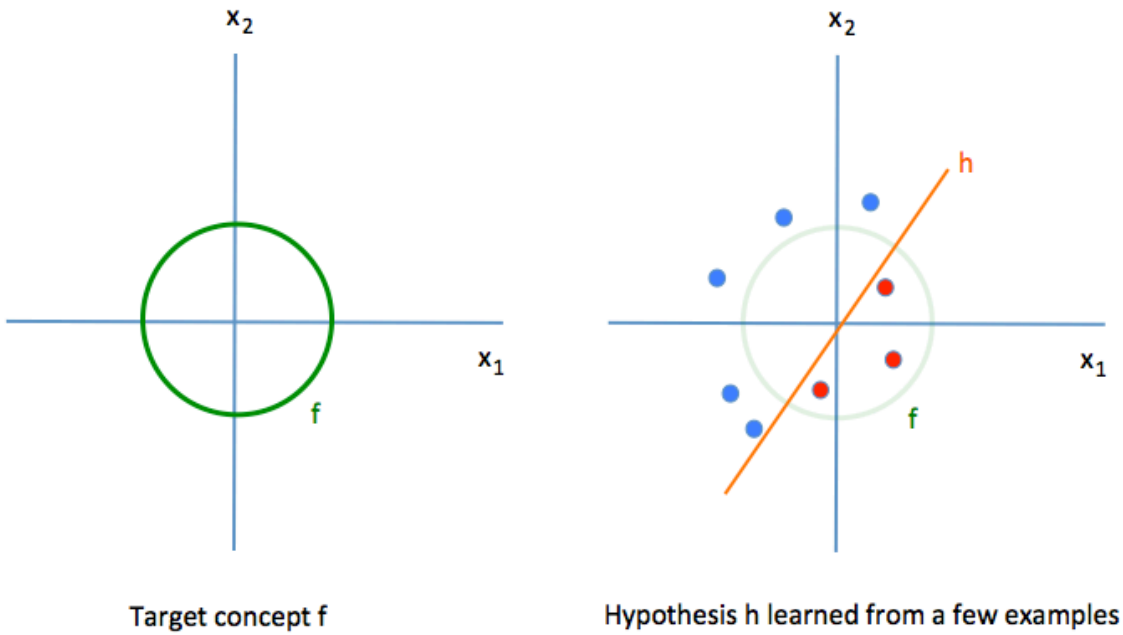
Definition

A supervised learning algorithm L automatically outputs from S a model or a classifier (or a hypothesis) $h \in \mathcal{H}$ as close to f as possible.

Impact of the number m of training examples on h

Let's assume that the underlying concept f is the following:

If $x_1^2 + x_2^2 < R^2$ then $y = +1$ otherwise $y = -1$



Target concept f

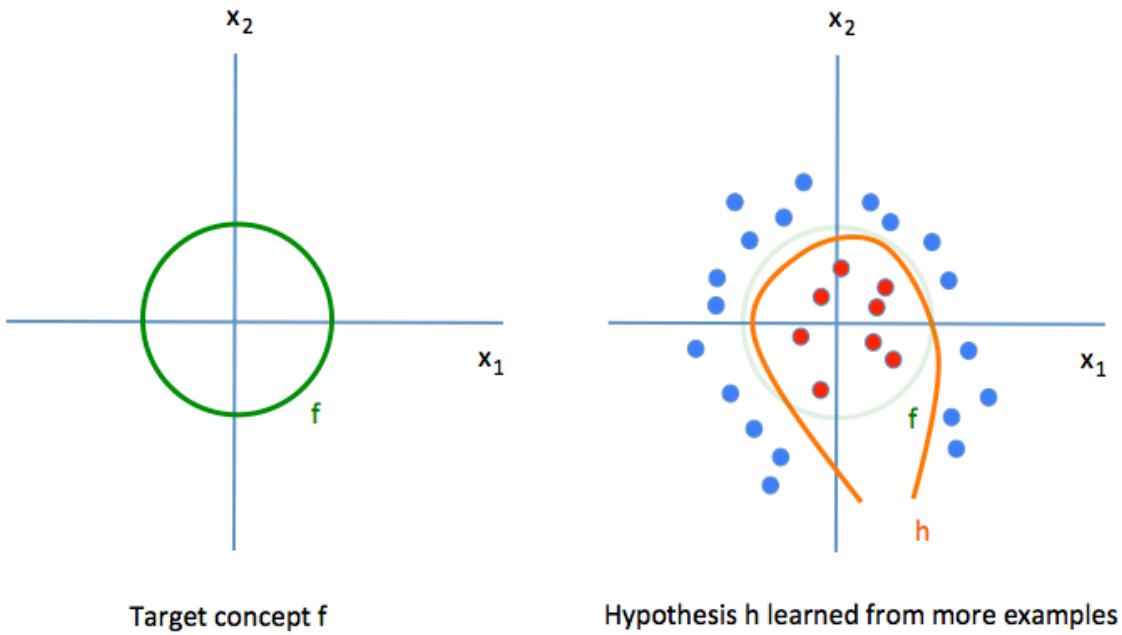
Hypothesis h learned from a few examples

h is perfect on the training set but is very weak a test time.

Impact of the number m of training examples on h

Let's assume that the underlying concept f is the following:

If $x_1^2 + x_2^2 < R^2$ then $y = +1$ otherwise $y = -1$



Target concept f

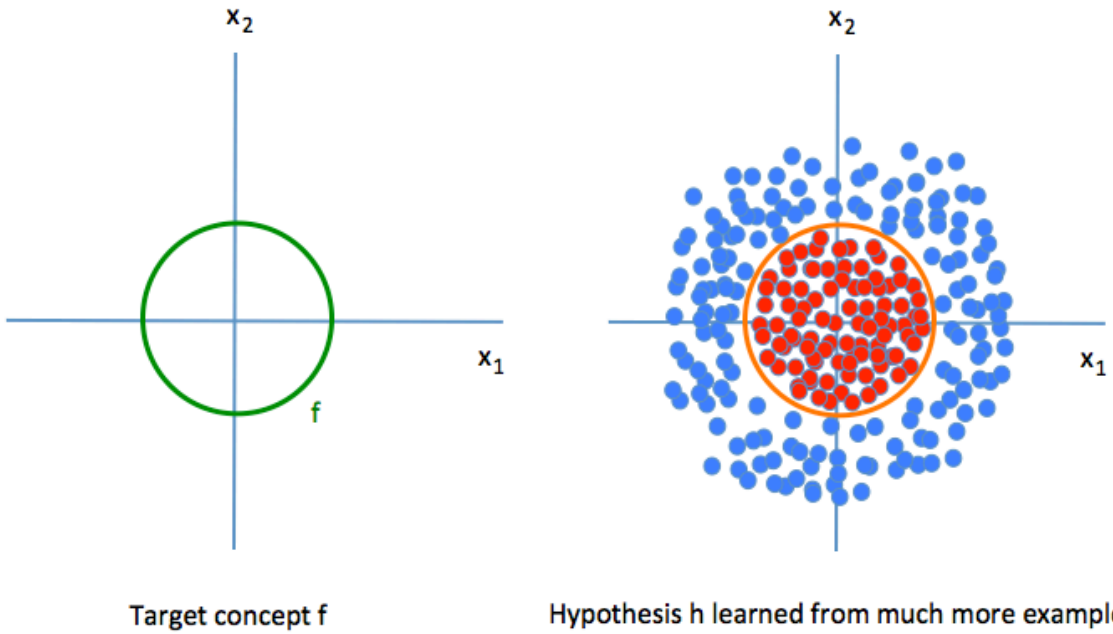
Hypothesis h learned from more examples

h is perfect on the training set but is still not good enough a test time.

Impact of the number m of training examples on h

Let's assume that the underlying concept f is the following:

If $x_1^2 + x_2^2 < R^2$ then $y = +1$ otherwise $y = -1$



Target concept f

Hypothesis h learned from much more examples

h is perfect on the training set AND accurate at test time (i.e. $h \approx f$).

Impact of the number m of training examples on h

Conjecture

The larger the training set, the higher the probability to get an hypothesis h whose behavior at training time on **labeled data** is close to that of at test time on new **unlabeled data**.

We will prove this conjecture later.

Curse of dimensionality

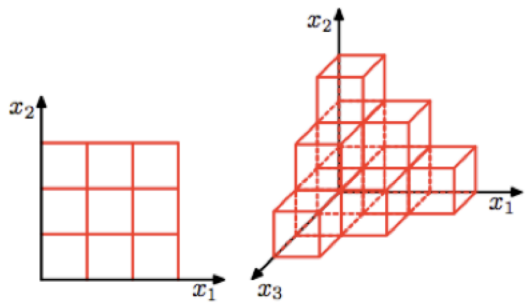
Learning from m data in $\mathbb{R}^d$ with $m > d$ is cool, not the other way around.

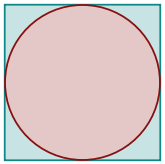
Curse of dimensionality

As the number of features or dimensions grows, the amount of data (at training time) we need to generalize accurately (i.e. at test time) grows exponentially.

Example 1

$10^2 = 100$ evenly-spaced sample points suffice to sample a unit interval (a "1-dimensional cube") with a 0.01 distance between points; a 10-dimensional unit hypercube that has the same spacing would require 10^{20} sample points.



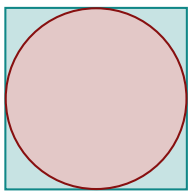


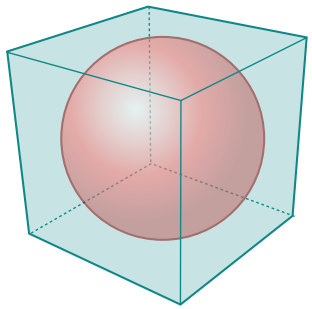
Example 2

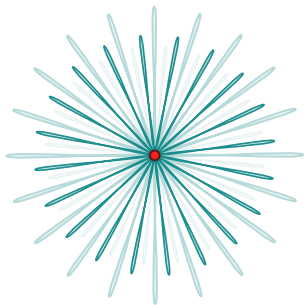
Let us compare the proportion of an inscribed hypersphere with radius r and dimension d, to that of a hypercube with edges of length $2r$.

- The volume V_s of the hypersphere is $V_s = \frac{2r^d \pi^{d/2}}{d\Gamma(d/2)}$.
- The volume V_c of the hypercube is $V_c = (2r)^d$.
- As the dimension d increases, we get:

$$lim_{d \to \infty} \frac{V_s}{V_c} = 0$$







Curse of dimensionality

How to overcome the curse of dimensionality?

When facing the curse of dimensionality, we can:

- **pre-process the data** into a lower-dimensional space (feature selection, principal component analysis, etc.)
- or **regularize** the underlying optimization problem at training time.

Overfitting

The curse of dimensionallity is closely related to the notion of overfitting.

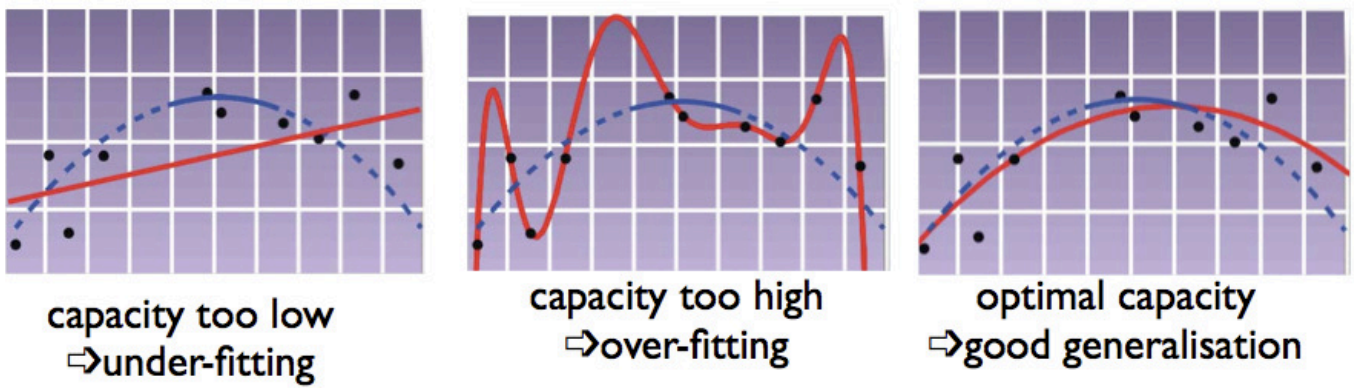
Overfitting - Underfitting

Definition

In statistics, **overfitting** occurs when a model is **excessively complex**, such as having too many degrees of freedom (e.g. polynomial of high order) w.r.t. the amount of data available $\rightarrow$ use of a **regularization**.

Definition

Underfitting occurs when a statistical model or machine learning algorithm cannot capture the underlying trend of the data.



True risk and empirical risk

Reminder

A supervised learning algorithm L automatically outputs from S a model or a classifier (or a hypothesis) $h \in \mathcal{H}$ as close to f as possible.

To pick the best hypothesis h^* , we need a criterion to assess the quality of h. Given a nonnegative loss function $\ell : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}^+$ measuring the degree of agreement between $h(\mathbf{x})$ and y, we can define the **true risk**.

True risk and empirical risk

Definition (True Risk)

The true risk $\mathcal{R}^{\ell}(h)$ (also called **generalization error**) of a hypothesis h with respect to a loss function ℓ corresponds to the expected loss suffered by h over the distribution $\mathcal{D}_{\mathcal{Z}}$.

$$\mathcal{R}^{\ell}(h) = \mathbb{E}_{z \sim \mathcal{D}_{\mathcal{Z}}} \ell(h, z).$$

Unfortunately, $\mathcal{R}^{\ell}(h)$ cannot be computed because $\mathcal{D}_{\mathcal{Z}}$ is unknown. We can only measure it on the training sample S. This is called the **empirical risk** $\hat{\mathcal{R}}^{\ell}(h)$.

True risk and empirical risk (ctd)

Definition (True Risk)

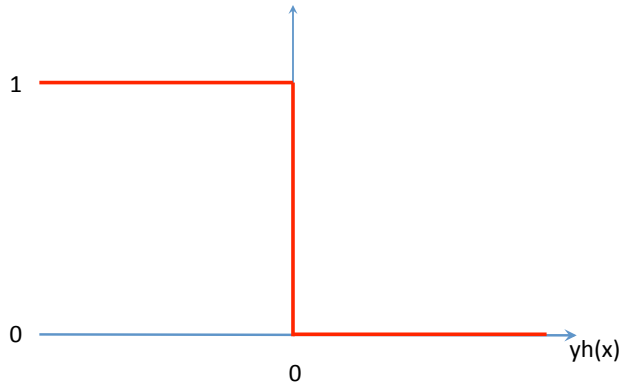
$$\mathcal{R}^{\ell}(h) = \mathbb{E}_{z \sim \mathcal{D}_{\mathcal{Z}}} \ell(h, z),$$

Definition (Empirical Risk)

Let $S = \{z_i = (\mathbf{x_i}, y_i))\}_{i=1}^{m}$ be a training sample. The empirical risk $\hat{\mathcal{R}}^{\ell}(h)$ (also called empirical error) of a hypothesis $h \in \mathcal{H}$ with respect to a loss function ℓ corresponds to the expected loss suffered by h on S.

$$\hat{\mathcal{R}}^{\ell}(h) = \frac{1}{m} \sum_{i=1}^{m} \ell(h, z_i)$$

Loss functions



Definition (0/1 loss)

The **most natural loss function** for binary classification is the 0/1 loss (also called classification error):

$$\ell_{0/1}(h, z) = 1 \text{ if } yh(x) < 0 \text{ and } 0 \text{ otherwise.}$$

$\mathcal{R}^{\ell_{0/1}}(h)$ then corresponds to the proportion of correct predictions.

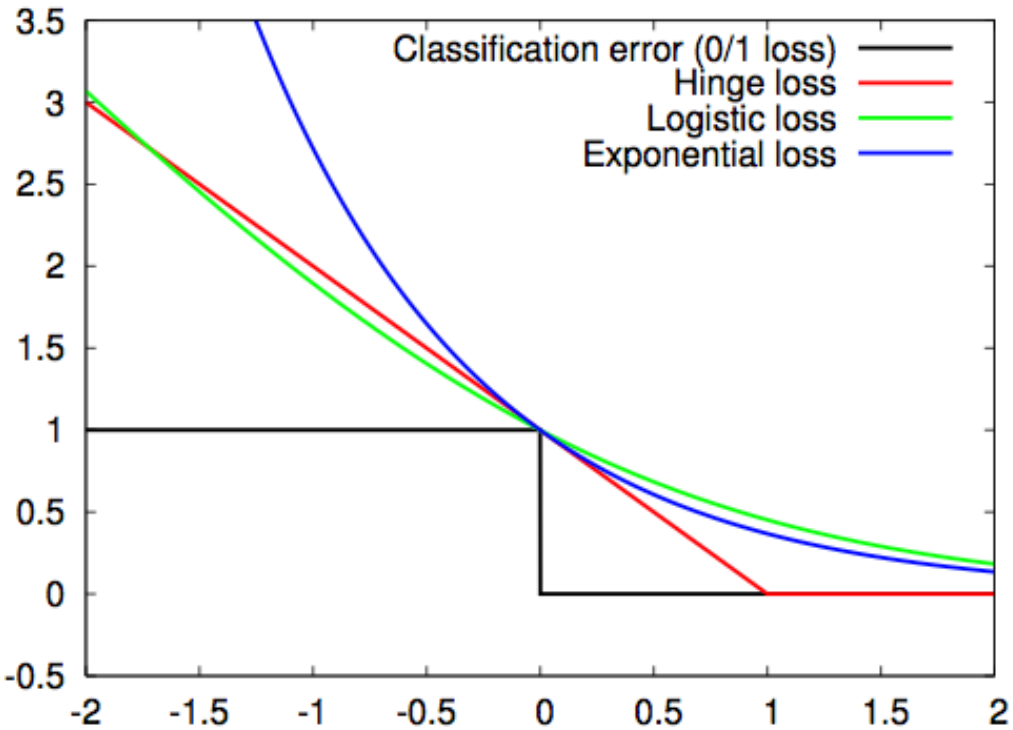
Warning

Due to the nonconvexity and non differentiability of the 0/1 loss, minimizing (or approximately minimizing) $\mathcal{R}^{\ell_{0/1}}(h)$ is known to be NP-hard.

Surrogate Margin-based Loss Functions

For this reason, surrogate convex loss functions are often used:

- the **hinge loss** (used in SVM): $\ell_{hinge}(h, z) = max(0, 1 - yh(x))$,
- the **exponential loss** (used in boosting): $\ell_{exp}(h, z) = e^{-yh(x)}$,
- the **logistic loss** (used in logistic regression): $\ell_{log}(h, z) = ln(1 + e^{-yh(x)})$.



Regularized Risk Minimization

To prevent the algorithm from overfitting, a supervised learning problem often takes the following regularized form:

$$\min_{h \in \mathcal{H}} \hat{\mathcal{R}}_\ell(h) + \lambda \|h\|_p$$

where:

- $\hat{\mathcal{R}}_\ell$ is the empirical risk w.r.t. a given loss function ℓ,
- λ is a regularization parameter,
- and $\|.\|_p$ is a ℓ_p-norm over the classifier h.

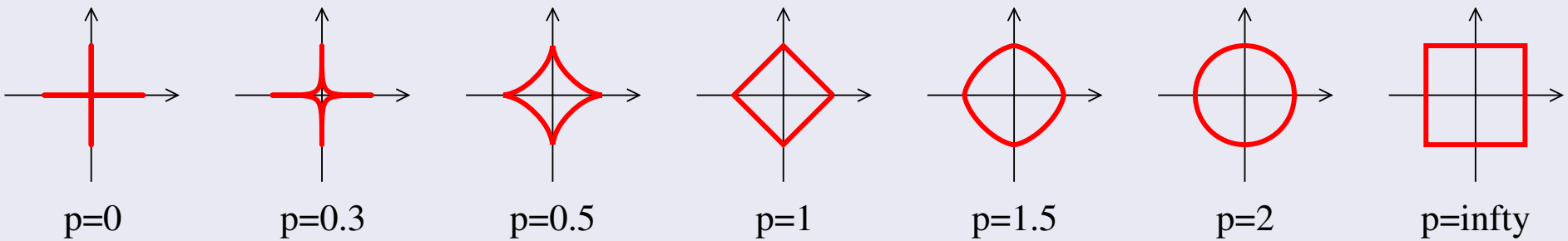
To avoid overfitting, learning boils down to selecting a classifier h that achieves a good trade-off between empirical risk minimization and regularization.

Usual Norms

ℓ_p Norms

The ℓ_p norm of a d-dimensional vector θ is defined as follows:

$$||\theta||_p = \left(\sum_{i=1}^{d} |\theta_i|^p \right)^{\frac{1}{p}}$$



- The ℓ_2 norm is used to reduce the risk of overfitting by decreasing the largest values of the model (example: ridge-regression).
- The ℓ_1 also allows the induction of sparse models - i.e. with less features (example: LASSO or ℓ_1-SVM).

Regularization with ℓ_2-norm

Example: Let us consider the following problem:

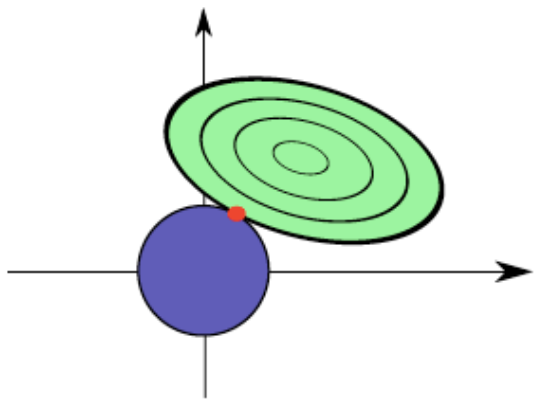
$$\min_{\theta \in \mathbb{R}^d} \frac{1}{2}\theta^T\theta - \theta^T\mathbf{x} + \lambda||\theta||_2^2$$

- if $\lambda = 0$

$$\frac{\partial \frac{1}{2}\theta^T\theta - \theta^T\mathbf{x}}{\partial \theta_j} = 0 \Rightarrow \theta_j - x_j = 0 \Rightarrow \forall j = 1..d, \boxed{\theta_j = x_j}$$

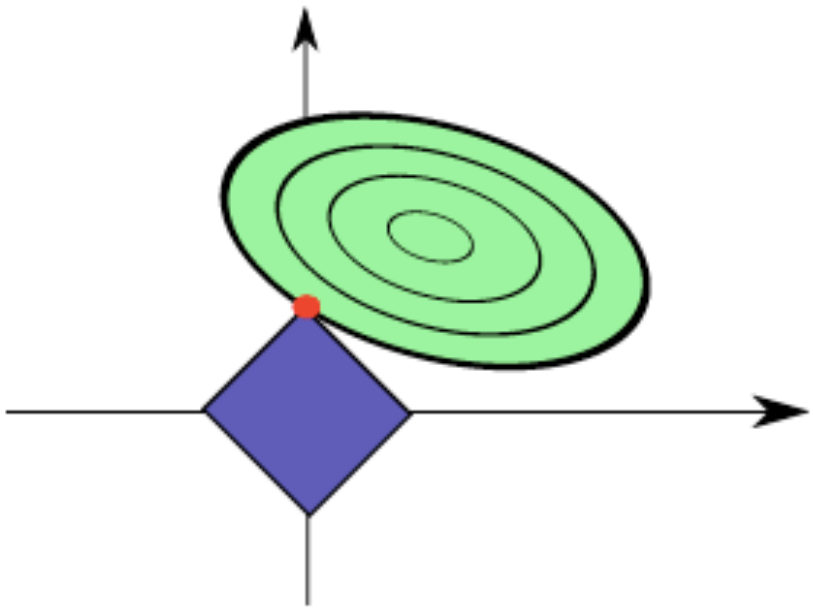
- if $\lambda \neq 0$

$$\frac{\partial \frac{1}{2}\theta^T\theta - \theta^T\mathbf{x} + \lambda||\theta||_2^2}{\partial \theta_j} = 0 \Rightarrow \theta_j - x_j + 2\lambda\theta_j = 0 \Rightarrow \forall j = 1..d, \boxed{\theta_j = \frac{x_j}{1+2\lambda}}$$



Effect of the ℓ_1 norm

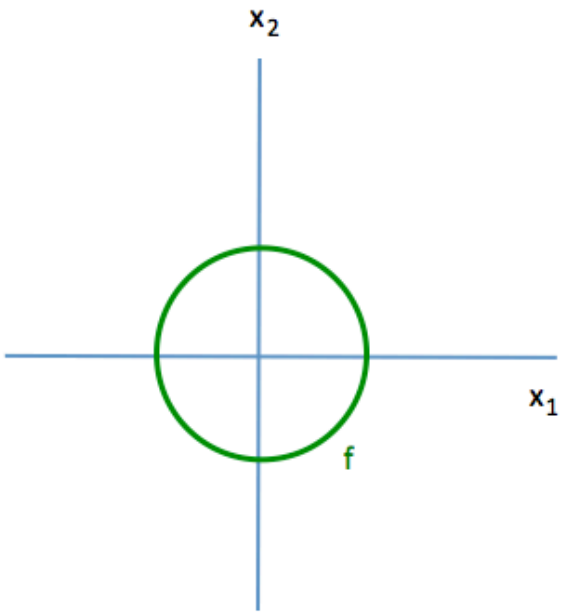
Increasing λ with the ℓ_1 norm causes more and more of the parameters θ_j to be driven to zero. The gradient on the ℓ_1 norm is constant w.r.t. the magnitude of each vector component.



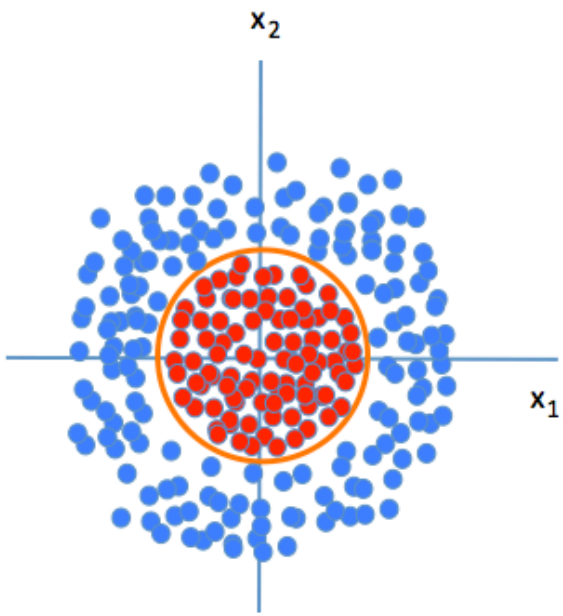
CONS

The ℓ_1 norm is not differentiable.

Never forget that the best regularizer is more data!!



Target concept f



Hypothesis h learned from much more examples

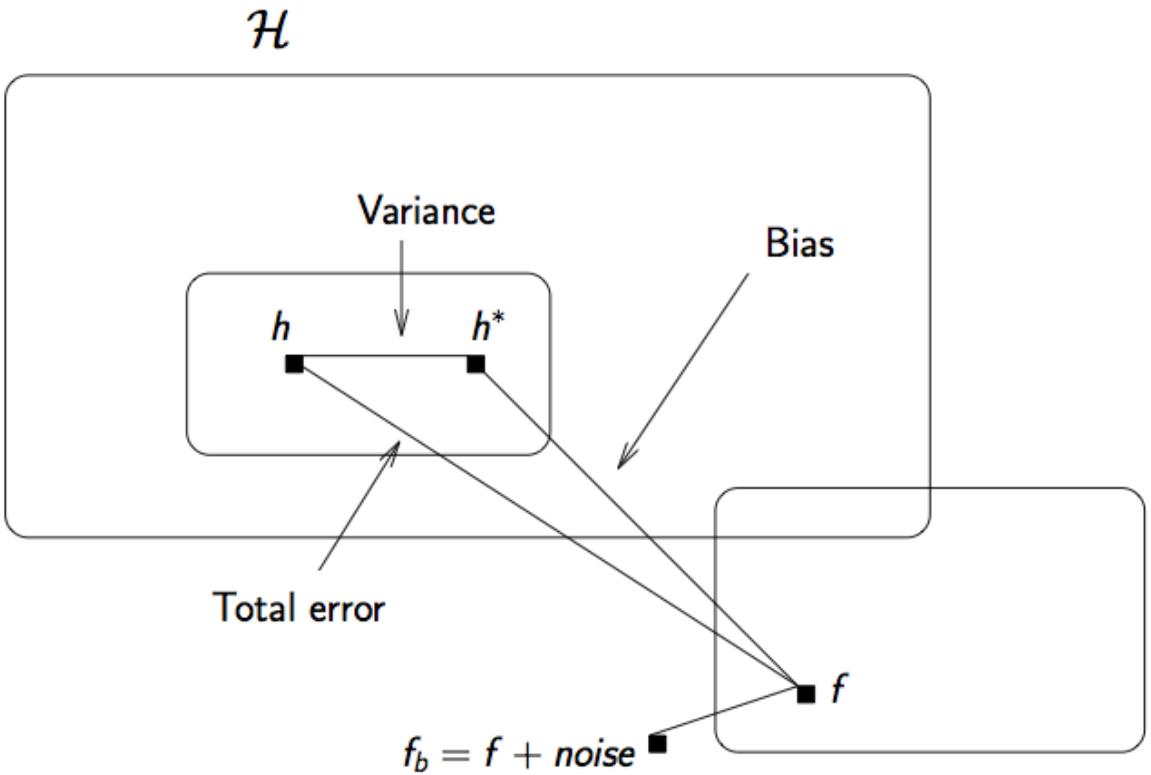
Bias/Variance trade-off

There are three sources of error between $h \in \mathcal{H}$ and the target function $f \in \mathcal{F}$.

Sources of error

1. **The inductive bias**: nothing guarantees the equality between the target concept space $\mathcal{F}$ and the selected class of hypotheses $\mathcal{H}$ even if the learner is able to provide an optimal hypothesis h^* from $\mathcal{H}$.
2. **The variance**: since the training set S is finite and randomly drawn from $D_{\mathcal{Z}}$, the learner usually does not provide the optimal hypothesis h^*.
3. **The presence of noise**: some training examples can be mislabeled. The learner receives a training set of a "noisy"function $f_b = f + noise$.

Bias/Variance trade-off



Bias/Variance trade-off

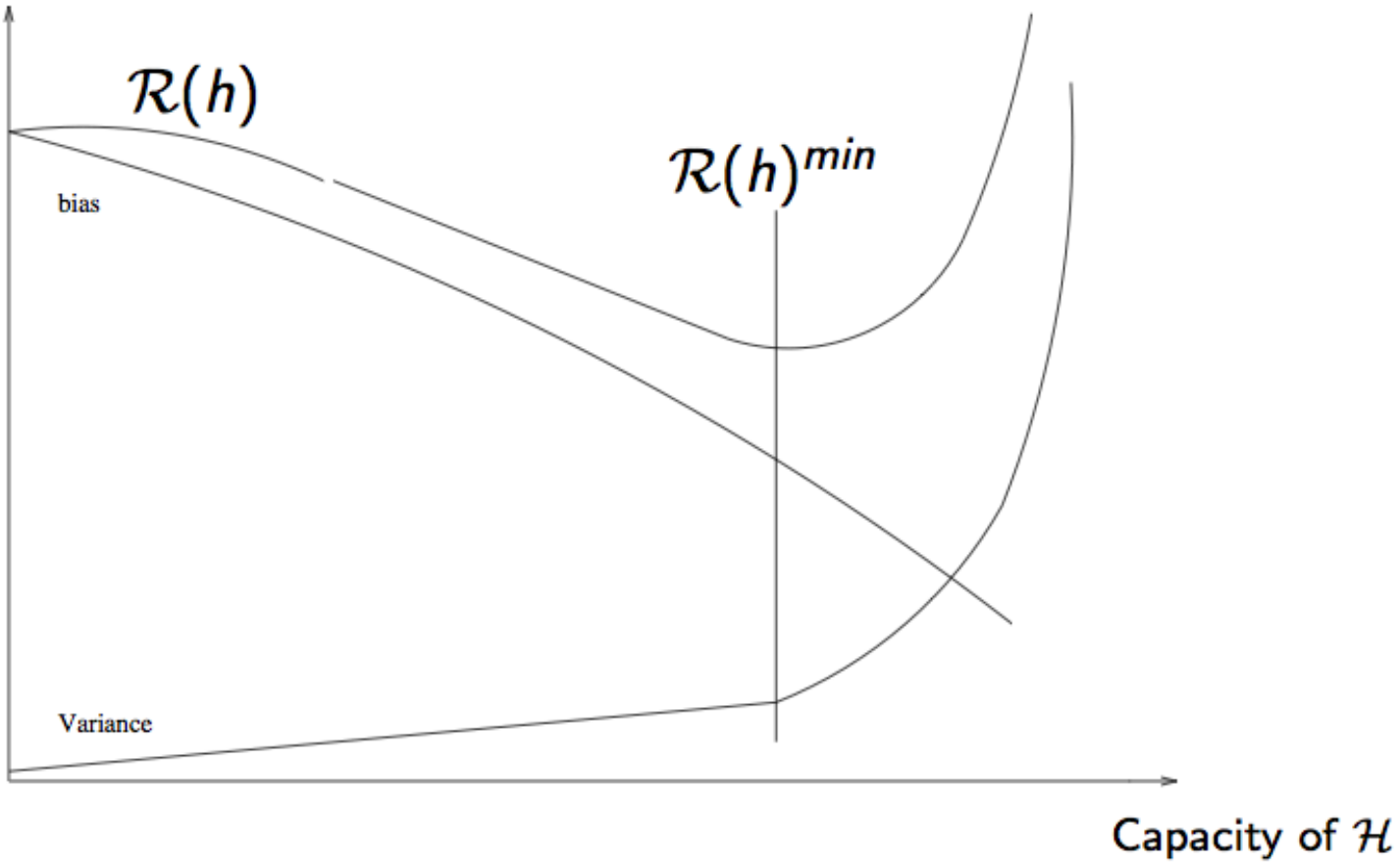
This Bias/Variance trade-off comes from statistics and the notion of mean squared error (MSE).

Definition

Let θ a theoretical parameter ($\mathcal{R}(h)$ in our case) and $\hat{\theta}$ an estimate of θ ($\hat{\mathcal{R}}(h)$ in our case). Let $B = \mathbb{E}[\hat{\theta}] - \theta$ be the bias of $\hat{\theta}$ w.r.t. θ. The MSE assesses the quality of θ in terms of its variation and unbiasedness. It is the expected value of the square loss between $\hat{\theta}$ and θ.

$$
\begin{aligned}
MSE &= \mathbb{E}_z[(\hat{\theta} - \theta)^2] \\
&= \mathbb{E}_z[(\hat{\theta} - E(\hat{\theta}) + E(\hat{\theta}) - \theta)^2] \\
&= \mathbb{E}_z[(\hat{\theta} - E(\hat{\theta}) + B)^2] \\
&= \mathbb{E}_z[(\hat{\theta} - E(\hat{\theta}))^2 + 2B(\hat{\theta} - E(\hat{\theta})) + B^2] \\
&= \mathbb{E}_z[(\hat{\theta} - E(\hat{\theta}))^2] + 2BE[\hat{\theta} - E(\hat{\theta})] + B^2 \\
&= Var(\hat{\theta}) + B^2.
\end{aligned}
$$

Bias/Variance trade-off



Empirical risk minimization (ERM)

$$\min_{h \in \mathcal{H}} \hat{\mathcal{R}}_\ell(h) + \lambda \|h\|_p$$

Definition

The ERM principle rests on the fact that if h works well on the training set S it might also work well on new examples.

Statistical learning theory investigates under what conditions empirical risk minimization (ERM) is admissible.

Empirical Risk Minimization

PAC (Probably Approximately Correct) Condition [Valiant 84]

The ERM principle is valid if the true risk of the hypothesis $h \in \mathcal{H}$ induced from S is close to the true risk of the optimal hypothesis $h^* \in \mathcal{H}$.

$$h = \underset{h_i \in \mathcal{H}}{\arg\min}\, \hat{\mathcal{R}}(h_i)$$

$$h^* = \underset{h_i \in \mathcal{H}}{\arg\min}\, \mathcal{R}(h_i)$$

Condition of validity of the ERM principle:

$$\forall \mathcal{D}_{\mathcal{Z}}, \forall \gamma \geq 0, \forall \delta \leq 1, P(|\mathcal{R}(h) - \mathcal{R}(h^*)| \geq \gamma) \leq \delta$$

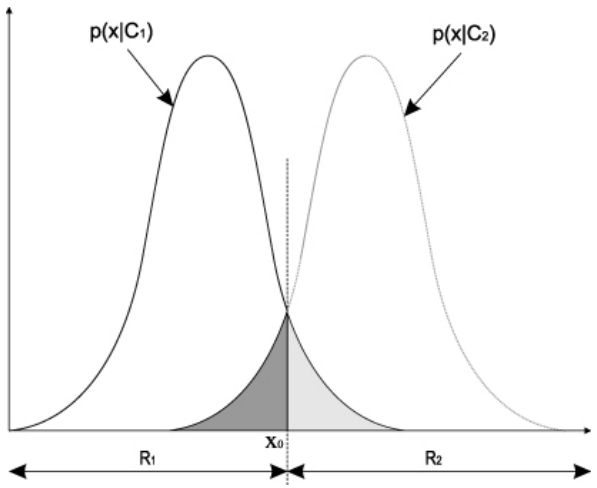
Bayesian Error

Bayesian Error

The bayesian error ϵ^* is the lowest possible error rate (or irreducible error) for any hypothesis h.

$$\epsilon^* = \int_{x \in R_i \text{ s.t. } y \neq C_i} P(C_i|x)p(x)dx$$

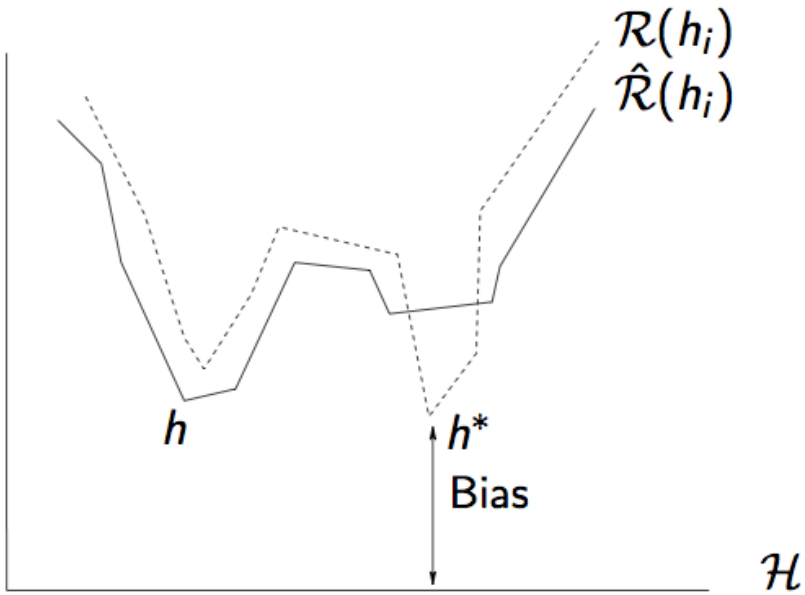
where x is an instance, y its corresponding label, R_i is the area/region that a classifier function h classifies as C_i.



Empirical Risk Minimization

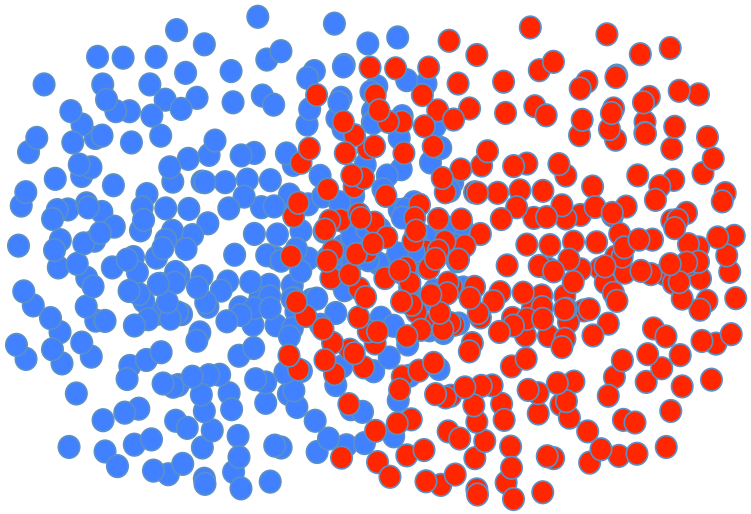
Remark

Note that **in many real world applications** $\epsilon^* > 0$. *Since* S *is finite, selecting* $h = \arg\min_{h_i \in \mathcal{H}} \hat{\mathcal{R}}(h_i)$ *does not guarantee to get the optimal hypothesis* $h^* = \arg\min_{h_i \in \mathcal{H}} \mathcal{R}(h_i)$.



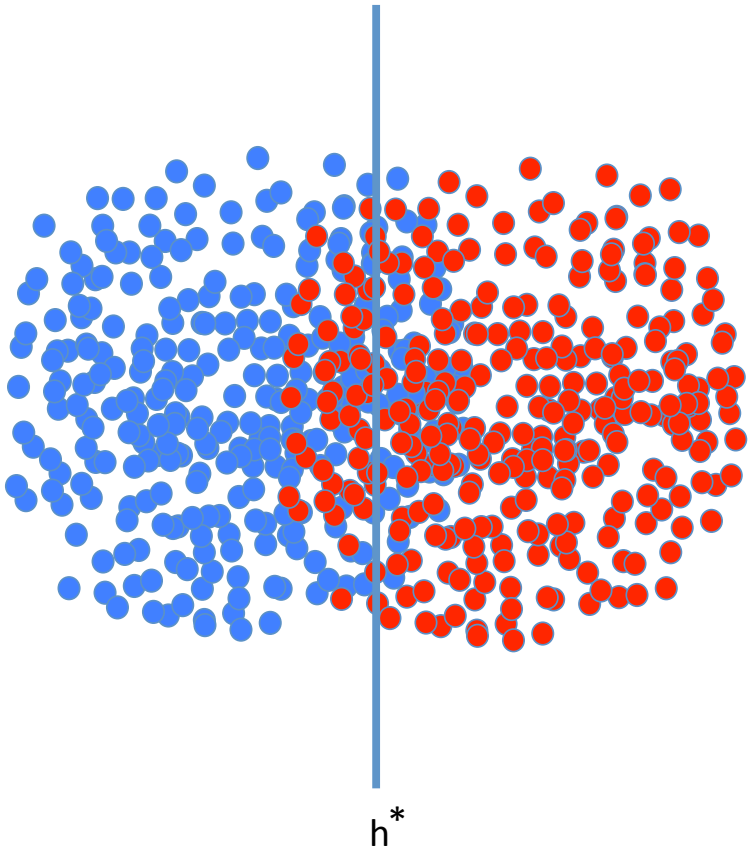
h versus h^*

Binary classification task with non null ϵ^*.



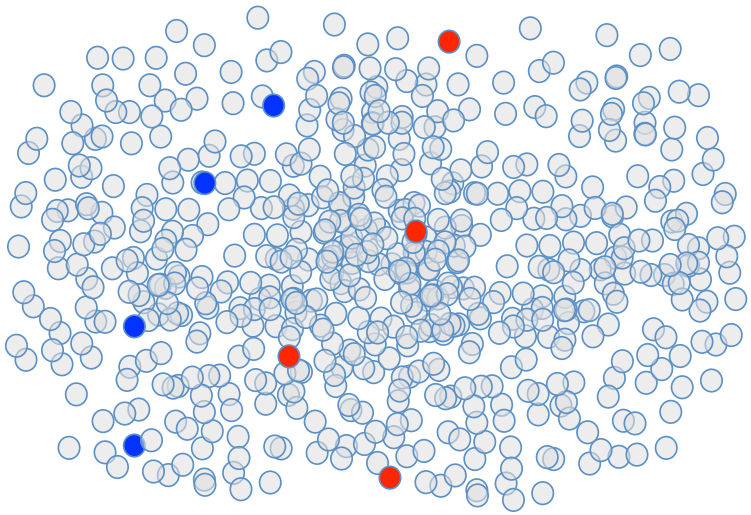
h versus h^*

$h^* = \arg\min_{h_i \in \mathcal{H}} \mathcal{R}(h_i)$



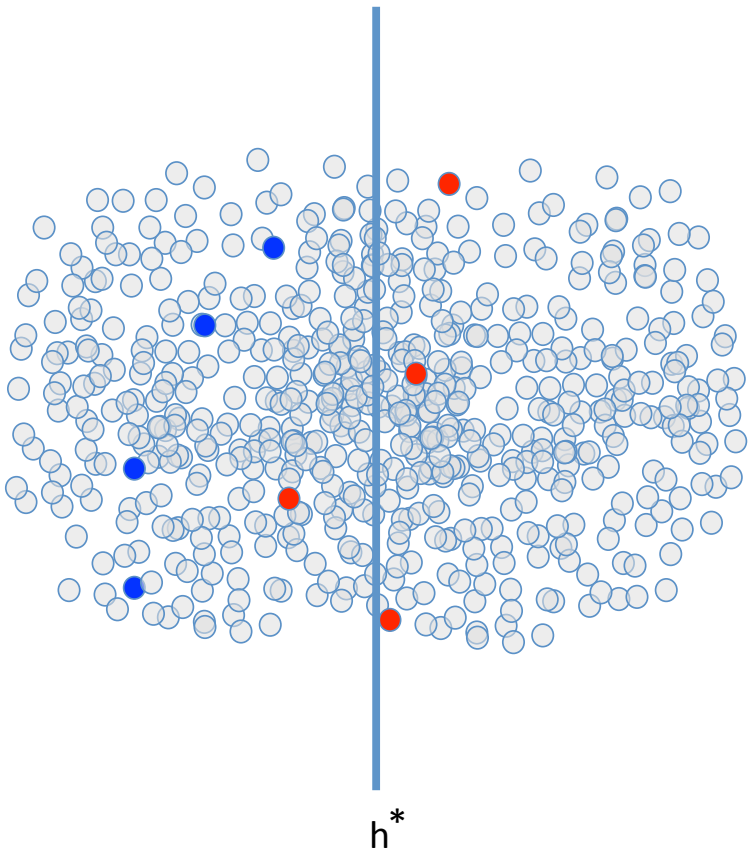
h versus h^*

Sample S drawn from $\mathcal{D}_{\mathcal{Z}}$.



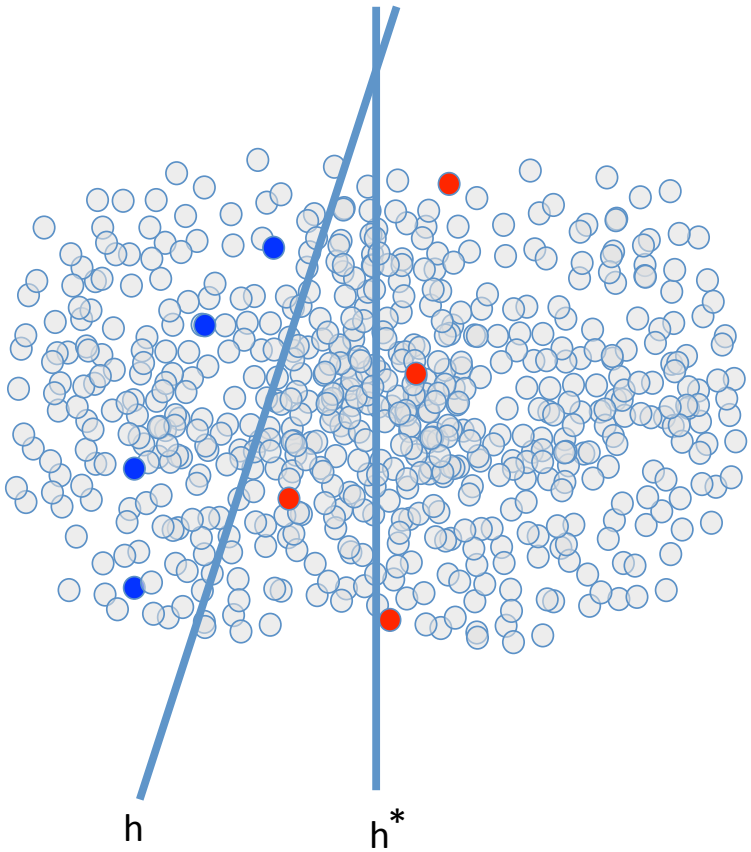
h versus h^*

h^* makes errors on S...



h versus h^*

...while h does not...

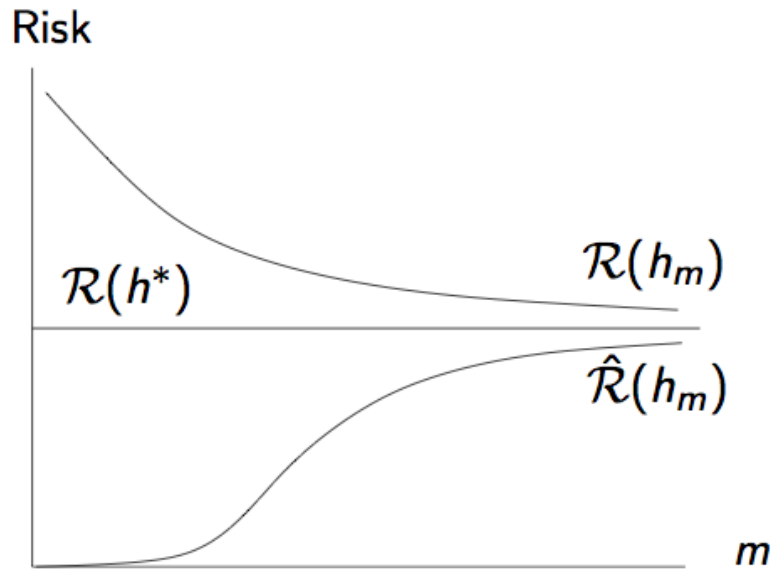


Empirical Risk Minimization

The law of large numbers prompts us to increase the size of the learning set and to search for the minimal size m that allows us to fulfill the PAC condition.

$$\forall \mathcal{D}_{\mathcal{Z}}, \forall \gamma \geq 0, \forall \delta \leq 1, \exists \text{ m s.t.} P(|\mathcal{R}(h_m) - \mathcal{R}(h^*)| \geq \gamma) \leq \delta$$

where h_m is the hypothesis learned from a training set of size m.



Empirical Risk Minimization

Question

Under what conditions (on the **minimum number of required examples***) do the empirical and true risks of the induced hypothesis h converge towards the true risk of h^*?*

One has to differentiate two different situations:

- When $|\mathcal{H}| = k$ is finite. $\mathcal{H} = \{h_1, h_2, \ldots, h_k\}$.
- When $|\mathcal{H}|$ is infinite.

When $|\mathcal{H}| = k$ is finite

Lemma 1: Union Bound

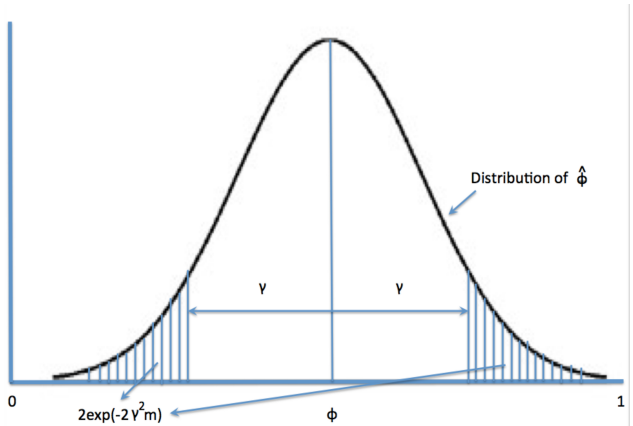
Let $A_1, A_2, ..., A_k$ be k events (not necessary independent). Then

$$P(A_1 \cup A_2, ..., A_k) \leq P(A_1) + P(A_2) + ... + P(A_k)$$

Lemma 2: Hoeffding Inequality

Let $Z_1, Z_2, ..., Z_m$ be m Bernouilli random variables with mean ϕ (i.e. $P(Z_i = 1) = \phi$). Let $\hat{\phi} = \frac{1}{m} \sum_{i=1}^{m} Z_i$ and let any $\gamma > 0$ be fixed. Then

$$P(|\hat{\phi} - \phi| > \gamma) \leq 2exp(-2\gamma^2 m)$$



When $|\mathcal{H}| = k$ is finite

Consider that $\mathcal{H} = \{h_1, h_2, \ldots, h_k\}$. Hoeffding inequality (Lemma 2) can be applied on $\mathcal{R}(h)$ and $\hat{\mathcal{R}}(h)$ with $\ell(h, z_i)$ a Bernouilli random variable with mean $\mathcal{R}(h)$.

Theorem

For a given $h \in \mathcal{H}, \forall \gamma \geq 0, \forall m > 0, \forall \mathcal{D}_{\mathcal{Z}}$

$$P(|\mathcal{R}(h) - \hat{\mathcal{R}}(h)| \geq \gamma) \leq 2e^{-2\gamma^2 m}$$

However, we need a bound that holds uniformly over the whole space of hypotheses. Therefore, we are interested in:

$$P(sup_{h \in \mathcal{H}} |\mathcal{R}(h) - \hat{\mathcal{R}}(h)| \geq \gamma)$$

When $|\mathcal{H}| = k$ is finite

Uniform convergence

Let A_j be the event $|\mathcal{R}(h_j) - \hat{\mathcal{R}}(h_j)| \geq \gamma$. By Lemma 2, we get

$$P(A_j) = 2e^{-2\gamma^2 m}$$

$$\begin{aligned} P(sup_{h\in\mathcal{H}}|\mathcal{R}(h) - \hat{\mathcal{R}}(h)| \geq \gamma) &= P(A_1 \cup ... \cup A_k) \\ &\leq \sum_i P(A_i) \text{ (Lemma 1)} \\ &= \sum_i 2e^{-2\gamma^2 m} \\ &= 2ke^{-2\gamma^2 m} \end{aligned}$$

Bound on m

From $P(|\mathcal{R}(h) - \hat{\mathcal{R}}(h)| \geq \gamma) \leq 2ke^{-2\gamma^2 m}$, given a fixed probability δ:

$$\begin{aligned} \text{If} \quad 2ke^{-2\gamma^2 m} &= \delta \\ \Leftrightarrow \quad e^{2\gamma^2 m} &= \frac{2k}{\delta} \\ \Leftrightarrow \quad 2\gamma^2 m &= ln\frac{2k}{\delta} \\ \Leftrightarrow \quad m &= \frac{1}{2\gamma^2} ln\frac{2k}{\delta} \end{aligned}$$

So, if $m \geq \frac{1}{2\gamma^2} ln\frac{2k}{\delta}$ then with probability $1 - \delta$, $\forall \gamma$, we have

$$\mathcal{R}(h) \leq \hat{\mathcal{R}}(h) + \gamma$$

Bound on γ

From the Uniform Convergence, given m and a fixed probability δ, we get:

$$\begin{aligned} \text{If} \quad & 2ke^{-2\gamma^2 m} && = \delta \\ \Leftrightarrow \quad & e^{2\gamma^2 m} && = \frac{2k}{\delta} \\ \Leftrightarrow \quad & 2\gamma^2 m && = ln\frac{2k}{\delta} \\ \Leftrightarrow \quad & \gamma && = \sqrt{\frac{1}{2m} ln\frac{2k}{\delta}} \end{aligned}$$

So, with probability $1-\delta$, we have $\forall h \in \mathcal{H}$:

$$\mathcal{R}(h) \leq \hat{\mathcal{R}}(h) + \sqrt{\frac{1}{2m} ln\frac{2k}{\delta}}$$

Condition of validity of the ERM principle

$$\forall \mathcal{D}_{\mathcal{Z}}, \forall \gamma \geq 0, \forall \delta \leq 1, P(|\mathcal{R}(h) - \mathcal{R}(h^*)| \geq \gamma) \leq \delta$$

When $|\mathcal{H}| = k$ is finite

We know that with a probability $\geq 1 - \delta, \forall h \in \mathcal{H}$

$$\gamma = \sqrt{\frac{1}{2m} ln \frac{2k}{\delta}}$$

and

$$\mathcal{R}(h) - \gamma < \hat{\mathcal{R}}(h) < \mathcal{R}(h) + \gamma$$

Therefore,

$$\begin{aligned}
\mathcal{R}(h) &\leq \hat{\mathcal{R}}(h) + \gamma \\
&\leq \hat{\mathcal{R}}(h^*) + \gamma \text{ (because } h = \arg\min_{h_i \in \mathcal{H}} \hat{\mathcal{R}}(h_i)) \\
&< (\mathcal{R}(h^*) + \gamma) + \gamma \\
&= \mathcal{R}(h^*) + 2\gamma \\
&= \mathcal{R}(h^*) + 2\sqrt{\frac{1}{2m} ln \frac{2k}{\delta}}
\end{aligned}$$

When $|\mathcal{H}| = k$ is finite

Condition of validity of the ERM principle

$$\forall \mathcal{D}_{\mathcal{Z}}, \forall \gamma \geq 0, \forall \delta \leq 1, P(|\mathcal{R}(h) - \mathcal{R}(h^*)| \geq \gamma) \leq \delta$$

Theorem

Let $|\mathcal{H}| = k$ and let m, δ be fixed. Then, with probability $1 - \delta$,

$$\mathcal{R}(h) \leq \mathcal{R}(h^*) + 2\sqrt{\frac{1}{2m} ln \frac{2k}{\delta}}$$

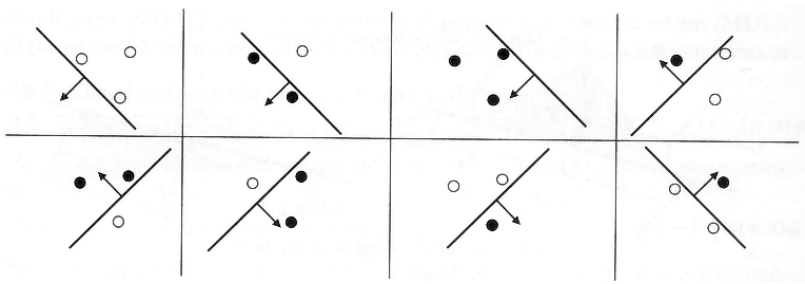
- The first term on the right hand side corresponds (roughly speaking) to the **bias**.
- The second term plays the role of the **variance**.

When $|\mathcal{H}|$ is infinite

The VC dimension $d_{\mathcal{H}}$ (for Vapnik-Chervonenkis dimension) is a measure of the **capacity** (or **complexity**) of the class of hypotheses $\mathcal{H}$.

Definition

- The VC dimension $d_{\mathcal{H}}$ of a class of hypotheses $\mathcal{H}$ is defined as the cardinality of the largest set of points that a hypothesis $h \in \mathcal{H}$ can **shatter**.
- A set of points is **shattered** if for all assignments of labels to those points, there exists a hypothesis $h \in \mathcal{H}$ that makes no error. Said differently, S is shattered by $\mathcal{H}$ if $\mathcal{H}$ realizes **all possible dichotomies of** S.



When $|\mathcal{H}|$ is infinite - Uniform convergence analysis

From the VC dimension $d_{\mathcal{H}}$, we can define an upper bound in $\mathcal{O}(1/\sqrt{n})$ on the true error:

Theorem

Let $\mathcal{H}$ be a class of hypotheses, $\forall h \in \mathcal{H}, \forall \delta \geq 0, \forall m > 0$, the following bound holds:

$$\mathcal{R}(h) \leq \hat{\mathcal{R}}(h) + \sqrt{\frac{d_{\mathcal{H}}(ln\frac{2m}{d_{\mathcal{H}}} + 1) + ln\frac{4}{\delta}}{m}}$$

Using the same math as before, we also get:

Theorem

Let $\mathcal{H}$ be a class of hypotheses, $\forall h \in \mathcal{H}, \forall \delta \geq 0, \forall m > 0$, the following bound also holds:

$$\mathcal{R}(h) \leq \mathcal{R}(h^*) + 2\sqrt{\frac{d_{\mathcal{H}}(ln\frac{2m}{d_{\mathcal{H}}} + 1) + ln\frac{4}{\delta}}{m}}$$

When $|\mathcal{H}|$ is infinite - Uniform convergence analysis

Corollary

To guarantee that $\mathcal{R}(h) \leq \mathcal{R}(h^*) + 2\gamma$ it suffices that m is on the order of the VC-dim, i.e.:

$$m = \mathcal{O}_{\delta,\gamma}(d_{\mathcal{H}}),$$

where we treat γ and δ as constant.

Intuition behind the Corollary

The number of training examples you need is roughly linear in the VC-dimension of the hypothesis class. For most reasonable hypothesis classes, it turns out that the VC-dimension is very similar to the number of parameters of your model. (e.g. Linear classifier in d dimensions $\rightarrow$ $d_{\mathcal{H}} = d + 1$).

Some remarks about the VC theory

1. The only property that matters is the **size of the hypothesis space** and not on **how the algorithm searches the space**.
2. Therefore, the VC theory is meaningful when the learning algorithm performs minimization of $\hat{\mathcal{R}}(h)$ in the **full hypothesis space**.
3. It is **useless for local algorithms**, like the k-NN which has an infinite $d_{\mathcal{H}}$.
4. Two analytical frameworks to take into account the algorithm L to derive generalization bounds: **Uniform stability** and **Algorithmic robustness**. The goal is to bound:

$$P(|\mathcal{R}(L, h_S) - \hat{\mathcal{R}}(L, h_S)| \geq \gamma)$$

which differs from what we studied before:

$$P(sup_{h \in \mathcal{H}} |\mathcal{R}(h) - \hat{\mathcal{R}}(h)| \geq \gamma)$$

Uniform stability

We only focus here on Uniform Stability.

Variance versus Stability

- Statistical learning theory prompts us to **reduce the variance without altering the bias**.
- Having a **low variance** is equivalent to having **high stability**.
- How to **relate the generalization error $\mathcal{R}_h$ to the stability** of an algorithm L which induces h?

Intuitively, an algorithm L is said **stable** if it is robust to small changes in the training sample, i.e., the variation in its output h is small.

Uniform stability

Given a training set S of size m, we build $\forall i = 1, ..., m$:

- $S^{\setminus i} = \{z_1, ..., z_{i-1}, z_{i+1}, ..., z_m\}$ by removing the i-th element of S.
- $S^i = \{z_1, ..., z_{i-1}, z'_i, z_{i+1}, ..., z_m\}$ by replacing the i-th element by z'_i drawn i.i.d. from $\mathcal{D}_{\mathcal{Z}}$.

Definition (Uniform stability [Bousquet and Elisseef 2002])

An algorithm L has uniform stability $\frac{\beta}{m}$ with respect to a loss function ℓ if the following holds:

$$\forall S, \forall i \in \{1, ..., m\}, sup_z |\ell(h_S, z) - \ell(h_{S_i}, z)| \leq \frac{\beta}{m},$$

where β is a positive constant, h_S and h_{S_i} are the hypothesis learned by L from S and S_i respectively.

Uniform stability

Generalization bound using uniform stability

Let S be a training sample of size m and $\delta > 0$. For any algorithm L with uniform stability $\frac{\beta}{m}$ with respect to a loss function ℓ bounded by M, with probability $1 - \delta$, we have:

$$\mathcal{R}_{h_S} \leq \hat{\mathcal{R}}_{h_S} + \frac{2\beta}{m} + (4\beta + M)\sqrt{\frac{ln\frac{1}{\delta}}{2m}}$$

Theorem [Kearns and Ron 1999]

An algorithm L having an hypothesis space of finite VC-dimension is stable in the sense that its stability is bounded by its VC-dimension.

Corollary

Using the stability as a complexity measure does not give worse bounds than using the VC-dimension.

Proof of the Generalization Bound

The proof is based on **McDiarmid's Theorem (1989)**.

Theorem [McDiarmid 1989]

Let S and S^i defined as above, let $F : \mathcal{Z}^m \to \mathbb{R}$ be a function for which there exists constants c_i $(i = 1, ..., m)$ such that

$$sup_{S \in \mathcal{Z}^m, z_i' \in \mathcal{Z}} |F(S) - F(S^i)| \leq c_i,$$

then

$$P_S[F(S) - \mathbb{E}_S[F(S)] \geq \gamma] \leq e^{-2\gamma^2 / \sum_{i=1}^m c_i^2}$$

Let's show that $F(S) = \mathcal{R}_{hs} - \hat{\mathcal{R}}_{hs}$ (for the sake of simplicity $F(S) = \mathcal{R} - \hat{\mathcal{R}}$) and $F(S^i) = \mathcal{R}^i - \hat{\mathcal{R}}^i$ satisfy the previous condition.

Proof of the Generalization Bound

By definition, L is β stable. Therefore, we have

$$\begin{aligned} |\mathcal{R} - \mathcal{R}^{\setminus i}| &= |\mathbb{E}_z[\ell(h_S, z)] - \mathbb{E}_z[\ell(h_{S^{\setminus i}}, z)| \\ &\leq \frac{\beta}{m} \end{aligned}$$

We deduce that

$$\begin{aligned} |\mathcal{R} - \mathcal{R}^{i}| &\leq |\mathcal{R} - \mathcal{R}^{\setminus i}| + |\mathcal{R}^{\setminus i} - \mathcal{R}^{i}| \text{ (triangle inequality)} \\ &\leq 2\frac{\beta}{m} \end{aligned} \tag{1}$$

Moreover,

$$\begin{aligned} |\hat{\mathcal{R}} - \hat{\mathcal{R}}^{\setminus i}| &= |\frac{1}{m}\sum_{j\neq i}(\ell(h_S, z_j) - \ell(h_{S^{\setminus i}}, z_j)) + \frac{1}{m}\ell(h_S, z_i)| \\ &\leq \frac{1}{m}\sum_{j\neq i}|\ell(h_S, z_j) - \ell(h_{S^{\setminus i}}, z_j)| + \frac{1}{m}|\ell(h_S, z_i)| \\ &\leq \frac{\beta}{m} + \frac{M}{m}. \end{aligned} \tag{2}$$

Proof of the Generalization Bound

From $|\hat{\mathcal{R}} - \hat{\mathcal{R}}^{\setminus i}| \leq \frac{\beta}{m} + \frac{M}{m}$ we deduce that

$$\begin{aligned} |\hat{\mathcal{R}} - \hat{\mathcal{R}}^{i}| &\leq |\hat{\mathcal{R}} - \hat{\mathcal{R}}^{\setminus i}| + |\hat{\mathcal{R}}^{\setminus i} - \hat{\mathcal{R}}^{i}| \text{ (triangle inequality)} \\ &\leq 2\frac{\beta}{m} + 2\frac{M}{m} \end{aligned} \tag{3}$$

However, a closer look reveals that the second factor of 2 is not needed. Indeed,

$$\begin{aligned} |\hat{\mathcal{R}} - \hat{\mathcal{R}}^{i}| &\leq \frac{1}{m}\sum_{j\neq i} |\ell(h_S, z_j) - \ell(h_{S^i}, z_j)| + \frac{1}{m}|\ell(h_S, z_i) - \ell(h_S, z_i')| \\ &\leq \frac{1}{m}\sum_{j\neq i} |\ell(h_S, z_j) - \ell(h_{S^{\setminus i}}, z_j)| + \frac{1}{m}\sum_{j\neq i} |\ell(h_{S^{\setminus i}}, z_j) - \ell(h_{S^i}, z_j)| \\ &\quad + \frac{1}{m}|\ell(h_S, z_i) - \ell(h_S, z_i')| \\ &\leq 2\frac{\beta}{m} + \frac{M}{m}. \end{aligned} \tag{4}$$

Proof of the Generalization Bound

We now know that:

$$|\mathcal{R} - \mathcal{R}^i| \leq 2\frac{\beta}{m} \text{ and } |\hat{\mathcal{R}} - \hat{\mathcal{R}}^i| \leq 2\frac{\beta}{m} + \frac{M}{m}$$

Considering the random variable $F(S) = \mathcal{R} - \hat{\mathcal{R}}$, we deduce that:

$$\begin{aligned} |F(S) - F(S^i)| &= |(\mathcal{R} - \hat{\mathcal{R}}) - (\mathcal{R}^i - \hat{\mathcal{R}}^i)| \\ &\leq |\mathcal{R} - \mathcal{R}^i| + |\hat{\mathcal{R}} - \hat{\mathcal{R}}^i| \\ &= 4\frac{\beta}{m} + \frac{M}{m} \\ &= c_i \text{ (of McDiarmid's inequality)} \end{aligned} \quad (5)$$

We conclude that the random variable $\mathcal{R} - \hat{\mathcal{R}}$ satisfies McDiarmid's conditions. It remains to bound the expectation of this random variable to apply the following bound:

$$P_S[F(S) - \mathbb{E}_S[F(S)] \geq \gamma] \leq e^{-2\gamma^2 / \sum_{i=1}^m c_i^2}$$

Proof of the Generalization Bound

$$
\begin{aligned}
\mathbb{E}_S[\mathcal{R} - \hat{\mathcal{R}}] &= \mathbb{E}_{S,z_i'}[\ell(h_S, z_i')] - \mathbb{E}_S[\hat{\mathcal{R}}] \\
&= \mathbb{E}_{S,z_i'}[\ell(h_S, z_i')] - \frac{1}{m}\sum_{j=1}^{m}\mathbb{E}_S[\ell(h_S, z_j)] \\
&= \mathbb{E}_{S,z_i'}[\ell(h_S, z_i')] - \frac{1}{m}\sum_{j=1}^{m}\mathbb{E}_{S,z_i'}[\ell(h_{S^i}, z_i')] \text{ (it holds } \forall S \in \mathcal{Z}^m) \\
&= \mathbb{E}_{S,z_i'}[\ell(h_S, z_i')] - \ell(h_{S^i}, z_i')] \\
&\leq \mathbb{E}_{S,z_i'}[|\ell(h_S, z_i')] - \ell(h_{S^i}, z_i')|] \\
&\leq \mathbb{E}_{S,z_i'}[|\ell(h_{S^i}, z_i')] - \ell(h_{S^{\setminus i}}, z_i')|] + \mathbb{E}_{S,z_i'}[|\ell(h_{S^{\setminus i}}, z_i')] - \ell(h_S, z_i')|] \\
&\quad \text{(since } sup_{S,z}|\ell(h_S, z) - \ell(h_{S_i}, z)| \leq \frac{\beta}{m} \\
&\quad \implies \mathbb{E}_{S,z}[|\ell(h_S, z)] - \ell(h_{S^{\setminus i}}, z)|] \leq \frac{\beta}{m}) \\
&\leq 2\frac{\beta}{m} \qquad (6)
\end{aligned}
$$

Proof of the Generalization Bound

McDiarmid's Theorem tells us that

$$P_S[F(S) - \mathbb{E}_S[F(S)] \geq \gamma] \leq e^{-2\gamma^2 / \sum_{i=1}^{m} c_i^2}$$

If we set $F(S) = \mathcal{R} - \hat{\mathcal{R}}$ we get:

$$P_S[\mathcal{R} - \hat{\mathcal{R}} > \gamma] \leq exp(-\frac{2m\gamma^2}{(4m\frac{\beta}{m} + M)^2}).$$

Thus, setting the right hand side to δ, we obtain with probability at least $1 - \delta$,

$$\mathcal{R} \leq \hat{\mathcal{R}} + 2\frac{\beta}{m} + (4\beta + M)\sqrt{\frac{ln1/\delta}{2m}}$$

which gives the bound.

Stability Constant and Generalization Bound

Stability Constant and Generalization Bound

$$\mathcal{R} \leq \hat{\mathcal{R}} + 2\frac{\beta}{m} + (4\beta + M)\sqrt{\frac{ln1/\delta}{2m}}$$

This upper bound tells us that if we can find the stability constant β of an algorithm A, then β comes with a generalization guarantee.

How to find the stability constant?

Stability with regularization of the form $||h||_{\mathcal{F}}^2$

When the learning problem takes the following form:

$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_i \ell(h, z_i) + \lambda ||h||_{\mathcal{F}}^2,$$

we can show [Bousquet and Elisseef 2002] that the stability constant is defined as follows:

$$\beta \leq \frac{\sigma^2}{2\lambda}$$

where σ comes from the σ-admissibility of $\ell(h, z) = c(h(x), y)$ where c is the associated cost function.

σ-admissibility

A loss function ℓ is σ-admissible if the associated cost function $c(h(x), y)$ is convex w.r.t. its first argument and the following condition holds:

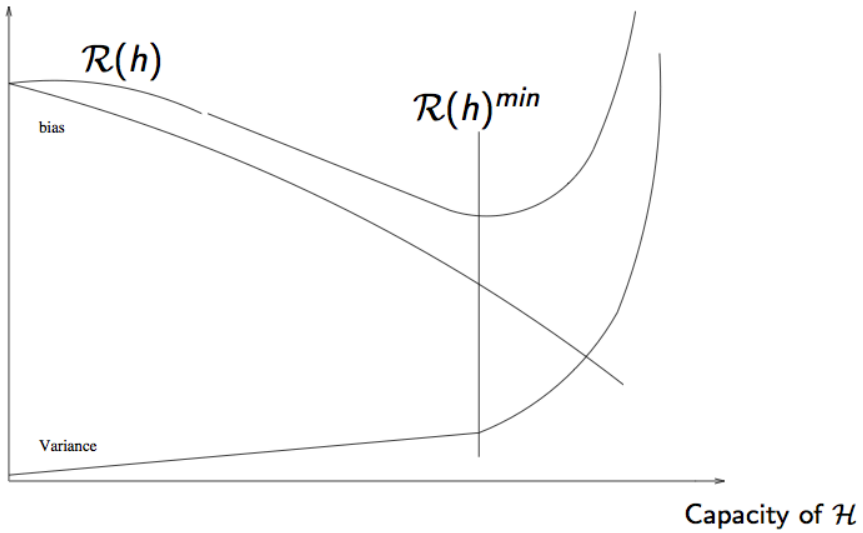
$$\forall y_1, y_2 \in \mathcal{Z}, \forall y' \in \mathcal{Y}, |c(y_1, y') - c(y_2, y')| \leq \sigma |y_1 - y_2|$$

See examples during the tutorials...

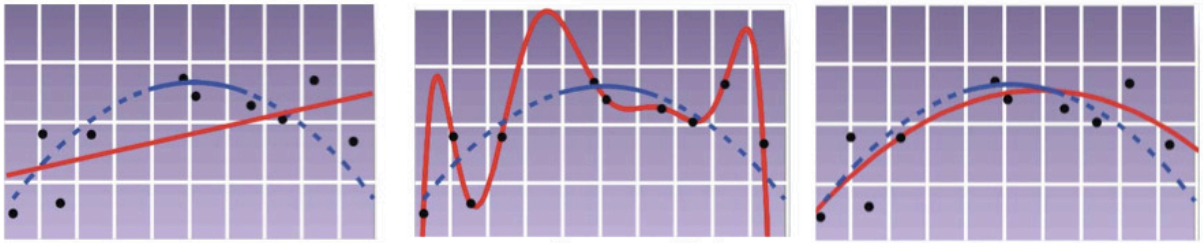
Model Selection

Model Selection

We saw there is often a trade-off between the bias and variance: it is important not to choose a hypothesis that is either too simple (underfitting) or too complex (overfitting). Model selection algorithms provide a method that automatically makes the trade-off between bias and variance.



Model Selection



capacity too low
⇨under-fitting

capacity too high
⇨over-fitting

optimal capacity
⇨good generalisation

Example 1: Linear Regression

$$h_1 = \theta_0 + \theta_1 x$$

$$h_2 = \theta_0 + \theta_1 x + \theta_2 x^2$$

$$\ldots$$

$$h_n = \theta_0 + \theta_1 x + \ldots + \theta_n x^n$$

What degree of the polynomial do you need to select?

Example 2: k nearest-neighbors

What is the right number k of neighbors?

Model Selection

$$M = \{h_1, h_2, ...\}$$

How to choose the best hypothesis in M?

1. Bad idea: choose the one with the lowest training error $\hat{\mathcal{R}}(h)$ (risk of overfitting).
2. Good idea: Hold-out k cross-validation.

Hold-out k cross-validation algorithm

Input: A learning algorithm L and a learning set S of m examples
Output: An estimate $\hat{\mathcal{R}}'(h)$
Split S randomly in k subsets $S_1, ..., S_k$ (if $k = m$, leave-one-out CV);
for *i=1 to k* **do**
  Run L on $S - S_i$ and induce classifier h_i;
end
Deduce the estimate $\hat{\mathcal{R}}'(h)$ of the true risk s.t. $\hat{\mathcal{R}}'(h) = \frac{1}{k}\sum_{i=1}^{k} \hat{\mathcal{R}}'(h_i)$ where $\hat{\mathcal{R}}'(h_i)$ is the error of h_i on S_i;