Rudiments de l'apprentissage automatique (ainsi que quelques notions plus avancées !)

Pascal Germain

GRAAL

3 avril 2009

Plan de la présentation

- L'apprentissage automatique et la classification ;
- Les concepts fondamentaux et leur représentation mathématique :
  - Ensemble de données ;
  - Classificateur ;
  - Algorithme d'apprentissage.
- Le problème de l'estimation du risque :
  - Borne sur l'ensemble test ;
  - Borne (*PAC-Bayes*) sur l'ensemble d'entraînement.
- Mes travaux de maîtrise.

L'apprentissage automatique et la classification

Objectif de l'apprentissage automatique

Doter un programme informatique de la faculté d'apprendre à effectuer une tâche à partir de l'observation d'un environnement.

Application aux problèmes de classification

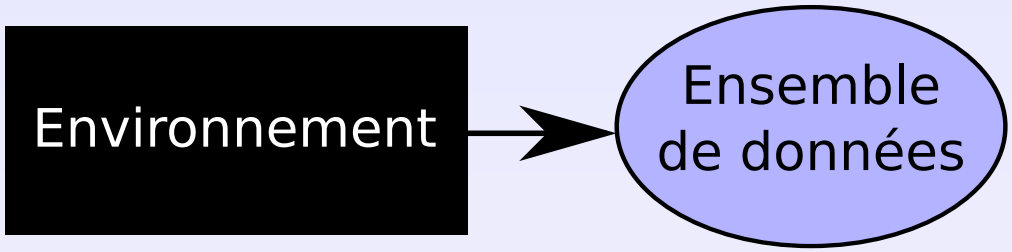
Créer un algorithme qui apprend à distinguer entre eux divers éléments (apposer une étiquette à un objet).

Exemples

- Reconnaître chacune des 26 lettres de l'alphabet ;
- Déterminer si un champignon est comestible ;
- Produire un diagnostique médical.

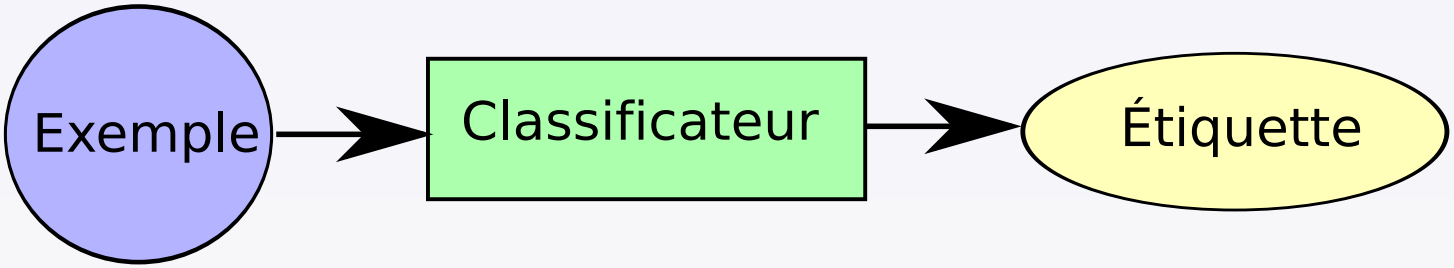
Procédures typiques

Collecte de données :

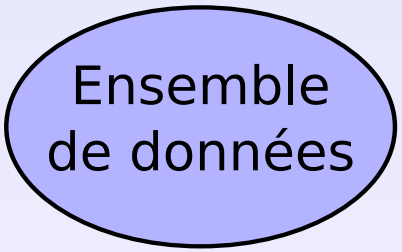


Procédure d'apprentissage :

Procédure de classification :

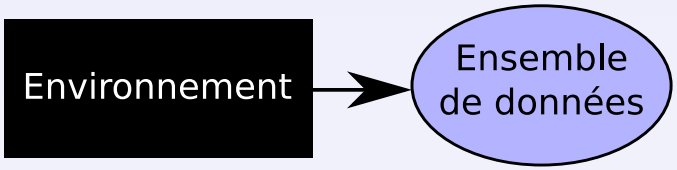


Représentation des données



Ensemble de données	$S \stackrel{\text{def}}{=} \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m\}$
Exemple d'apprentissage	$\mathbf{z} \stackrel{\text{def}}{=} (\mathbf{x}, y)$
Description de l'exemple	$\mathbf{x} \stackrel{\text{def}}{=} (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$
Classe d'appartenance	$y \in \{-1, +1\}$

L'ensemble S est issu de l'observation d'un phénomène réel :



- Chaque exemple $\mathbf{z} = (\mathbf{x}, y)$ est une paire description-étiquette ;
- La description $\mathbf{x} \in \mathbb{R}^n$ est un vecteur d'attributs ;
- L'étiquette $y \in \{-1, +1\}$ est attribuée manuellement.

NB : On se restreint ici aux problèmes de classification binaire.

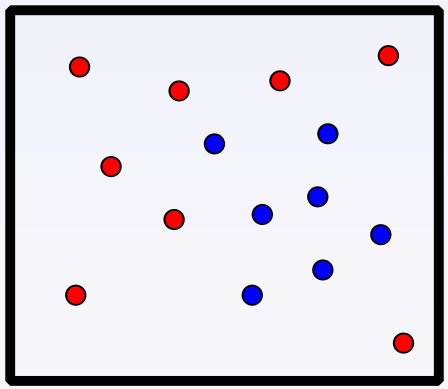
Représentation des données

Ensemble de données

Ensemble de données	$S \stackrel{\text{def}}{=} \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m\}$
Exemple d'apprentissage	$\mathbf{z} \stackrel{\text{def}}{=} (\mathbf{x}, y)$
Description de l'exemple	$\mathbf{x} \stackrel{\text{def}}{=} (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$
Classe d'appartenance	$y \in \{-1, +1\}$

Exemple (très simplifié) de l'ensemble de données « Mushrooms » :

$\Rightarrow$



Soit $(\mathbf{x}, y)$ un champignon :

Attribut x_1 : Hauteur du pied
Attribut x_2 : Diamètre du chapeau
Classe y : $\begin{cases} +1 & \text{Comestible} \\ -1 & \text{Vénéneux} \end{cases}$

Représentation d'un classificateur

Classificateur

Fonction de classification	$h(\mathbf{x}) \rightarrow y'$
Description de l'exemple	$\mathbf{x} \stackrel{\text{def}}{=} (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$
Prédiction	$y' \in \{-1, +1\}$

Un classificateur h est une fonction qui reçoit la description x d'un exemple et retourne une étiquette y'.

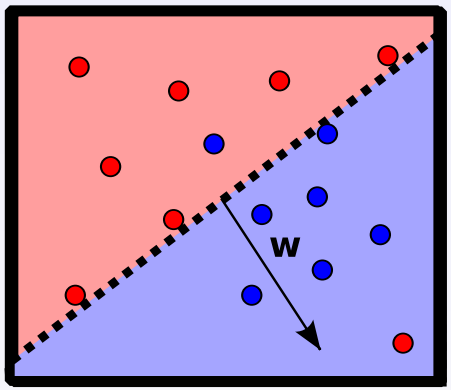
Représentation d'un classificateur

Classificateur

Fonction de classification	$h(\mathbf{x}) \to y'$
Description de l'exemple	$\mathbf{x} \stackrel{\text{def}}{=} (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$
Prédiction	$y' \in \{-1, +1\}$

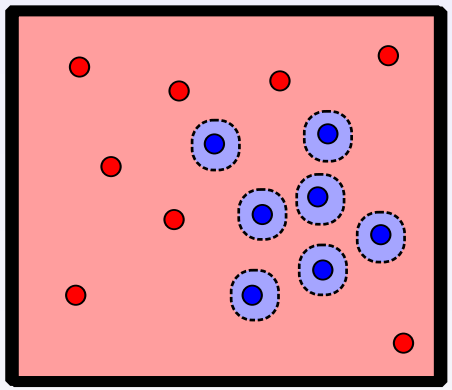
On peut représenter un classificateur comme une frontière de décision dans l'espace des données.

Séparateurs linéaires



$$h_{\mathbf{w},b}(\mathbf{x}) = \begin{cases} +1 & \text{si } \mathbf{w} \cdot \mathbf{x} + b > 0 \\ -1 & \text{sinon.} \end{cases}$$

Boules centrées sur les exemples



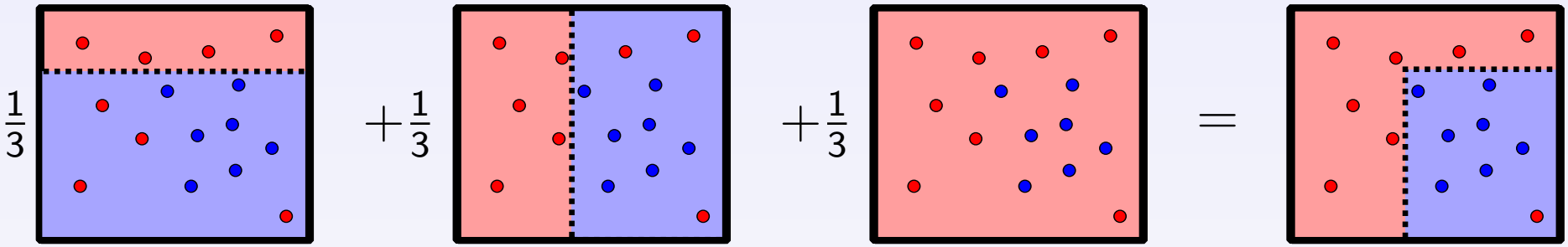
$$h_d(\mathbf{x}) = \begin{cases} +1 & \text{si } (\bar{\mathbf{x}}, +1) \in S \\ & \text{et } \|\bar{\mathbf{x}} - \mathbf{x}\| < d \\ -1 & \text{sinon.} \end{cases}$$

Représentation d'un classificateur

Classificateur

Fonction de classification	$h(\mathbf{x}) \to y'$
Description de l'exemple	$\mathbf{x} \stackrel{\text{def}}{=} (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$
Prédiction	$y' \in \{-1, +1\}$

Votes de majorité (B_Q)



$$B_Q(\mathbf{x}) = \operatorname{sgn}\left[\sum_{h \in \mathcal{H}} Q(h) h(\mathbf{x})\right]$$

Classificateurs de base	$\mathcal{H} = \{h_1, \ldots, h_n\}$
Distribution de poids	Q *t.q.* $\sum_{h \in \mathcal{H}} Q(h) = 1$

Algorithme d'apprentissage

Algorithme d'apprentissage

Fonction d'apprentissage	$A(S) \rightarrow h$
Ensemble de données	$S \stackrel{\text{def}}{=} \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m\}$
Fonction de classification	$h(\mathbf{x}) \rightarrow y'$

Un algorithme d'apprentissage A prend en entrée un ensemble S et fournit en sortie un classificateur h.

Algorithme d'apprentissage

Algorithme d'apprentissage

Fonction d'apprentissage	$A(S) \rightarrow h$
Ensemble de données	$S \stackrel{\text{def}}{=} \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m\}$
Fonction de classification	$h(\mathbf{x}) \rightarrow y'$

Le défi de la généralisation

Un bon algorithme d'apprentissage doit construire un classificateur qui généralise bien l'information contenue dans les données observées, afin de bien représenter le phénomène étudié.

Chaque algorithme propose une stratégie pour généraliser les données :

- Support Vectors Machines (SVM) ;
- AdaBoost ;
- Réseaux de neurones ;
- ...

Algorithme d'apprentissage

Stratégie des SVM (cas linéairement séparable)

Trouver le séparateur linéaire dont la marge est maximale.

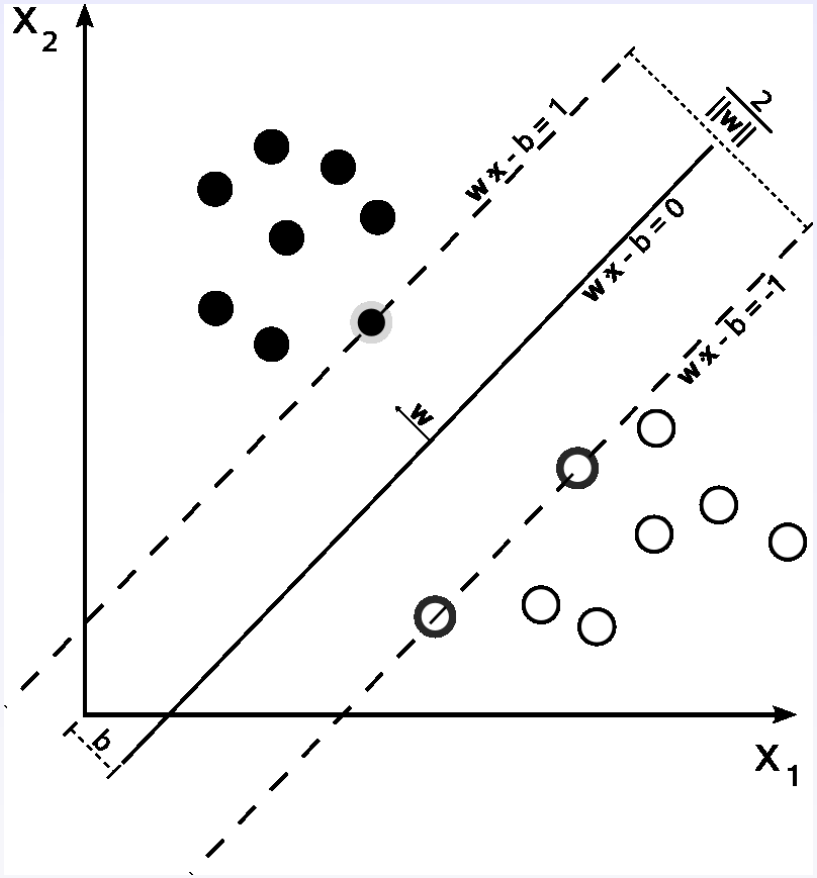


Image : Wikipédia

Problème de l'estimation du risque

Suite à l'apprentissage, nous désirons évaluer la qualité d'un classificateur.

Risque d'un classificateur

Probabilité d'effectuer une erreur sur un exemple **qui n'a pas servi à l'apprentissage**.

Pour estimer le risque d'un classificateur, il est nécessaire de faire quelques suppositions sur l'environnement qui génère les exemples.

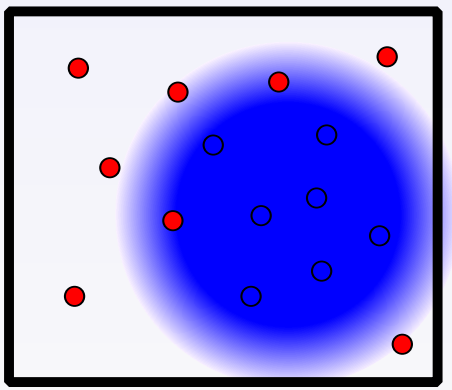
Problème de l'estimation du risque

Environnement

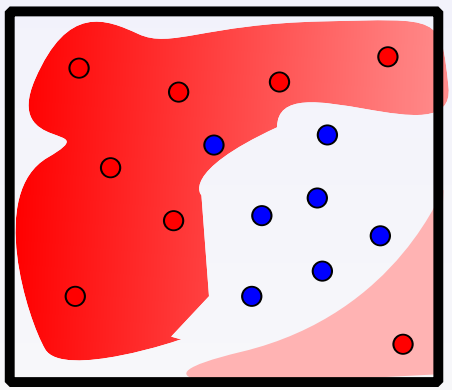
On représente l'environnement générateur par une distribution de probabilité D sur l'espace des exemples

- Chaque exemple $\mathbf{z} = (\mathbf{x}, y)$ est généré selon D ;
- L'ensemble de données S est une collection de réalisations d'une variable aléatoire ;
- Ces réalisations sont considérées *indépendantes et identiquement distribuées (iid)* .

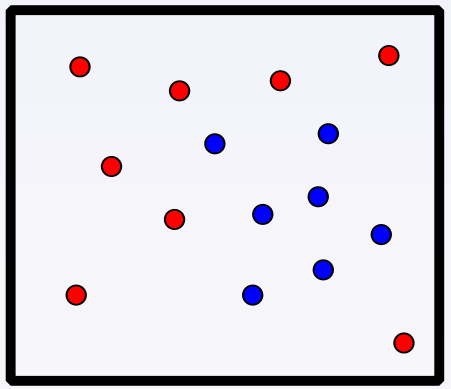
$\Pr_{(\mathbf{x},y)\sim D}(\mathbf{x}=(x_1,x_2)|y=+1)$



$\Pr_{(\mathbf{x},y)\sim D}(\mathbf{x}=(x_1,x_2)|y=-1)$



$S = (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m) \sim D^m$



Problème de l'estimation du risque

Environnement

On représente l'environnement générateur par une distribution de probabilité D sur l'espace des exemples

Risque d'un classificateur (ou vrai risque)

Probabilité qu'un classificateur classe incorrectement un exemple généré par la distribution D :

$$R(h) \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{(\mathbf{x},y)\sim D} I(h(\mathbf{x}) \neq y) \qquad \text{où } I(a) = \begin{cases} 1 & \text{si } a \text{ est vrai} \\ 0 & \text{sinon} \end{cases}$$

Risque empirique d'un classificateur

Taux d'erreur sur l'ensemble d'entraînement $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$:

$$R_S(h) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^{m} I(h(\mathbf{x}_i) \neq y_i)$$

Problème de l'estimation du risque

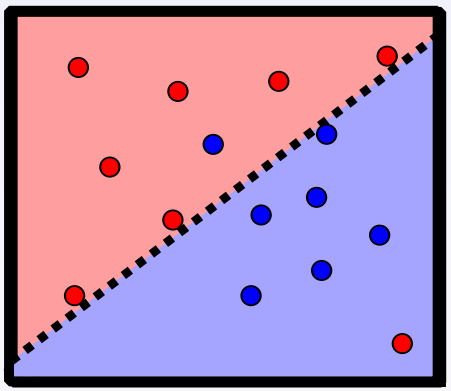
Risque empirique

$$R_S(h) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^{m} I(h(\mathbf{x}_i) \neq y_i)$$

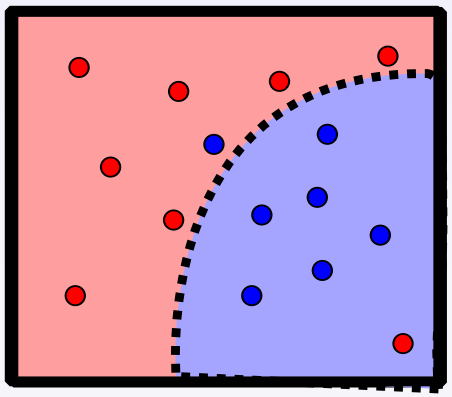
Risque (ou vrai risque)

$$R(h) \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{(\mathbf{x},y)\sim D} I(h(\mathbf{x}) \neq y)$$

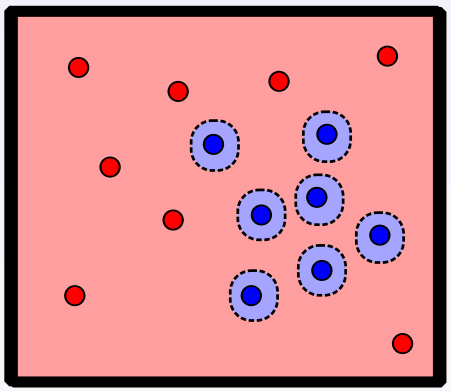
Le risque empirique $R_S(h)$ d'un classificateur n'est pas nécessairement un bon indicateur de son vrai risque $R(h)$.



$R_S(h) = \frac{3}{15} = 20\%$



$R_S(h) = \frac{2}{15} \simeq 13\%$



$R_S(h) = \frac{0}{15} = 0\%$

Problème de l'estimation du risque

Afin d'évaluer la qualité d'un classificateur h, nous désirons connaître son risque $R(h)$.

Risque (ou vrai risque)

$$R(h) \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{(\mathbf{x},y)\sim D} I(h(\mathbf{x}) \neq y)$$

La valeur précise de $R(h)$ peut être calculée si :

- Nous disposons de tous les exemples possibles du phénomène étudié ;
- Nous connaissons la distribution de probabilité D qui génère les exemples.

Ces deux cas ne correspondent pas aux problèmes réels.

$\Longrightarrow$ Nous devons donc trouver un moyen d'estimer $R(h)$.

Problème de l'estimation du risque

Afin d'évaluer la qualité d'un classificateur h, nous désirons connaître son risque $R(h)$.

Risque (ou vrai risque)

$$R(h) \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{(\mathbf{x},y)\sim D} I(h(\mathbf{x}) \neq y)$$

La théorie statistique permet d'obtenir une garantie sur le risque d'un classificateur à l'aide d'outils mathématiques.

Borne supérieure sur le risque d'un classificateur

Avec probabilité $1-\delta$, *le risque du classificateur* h *est d'au plus* $B(\cdots)$

$$\Pr\left(R(h) \leq B(\cdots)\right) \geq 1-\delta$$

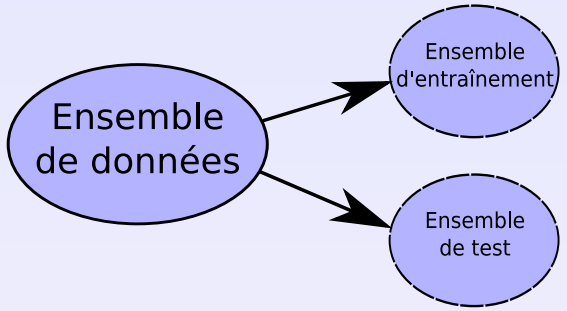
Il existe deux catégories de borne sur le risque :

- Bornes sur l'ensemble test ;
- Bornes sur l'ensemble d'entraînement.

Borne sur l'ensemble test

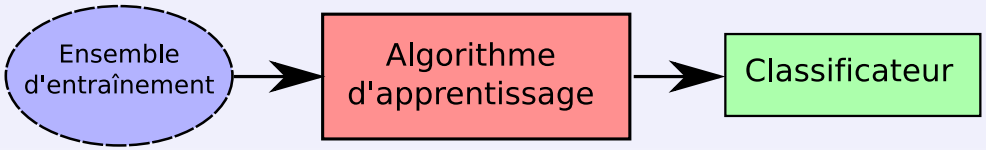
(A) Diviser les exemples étiquetés en :

- Un ensemble d'entraînement $S = \{\mathbf{z}_1, \ldots, \mathbf{z}_m\}$;
- Un ensemble test $T = \{\mathbf{z}'_1, \ldots, \mathbf{z}'_n\}$.



(B) Entraîner le classificateur sur l'ensemble d'entraînement :

$$h \leftarrow A(S)$$



(C) Calculer le risque sur l'ensemble test :

$$R_T(h) \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} I(h(\mathbf{x}'_i) \neq y'_i)$$

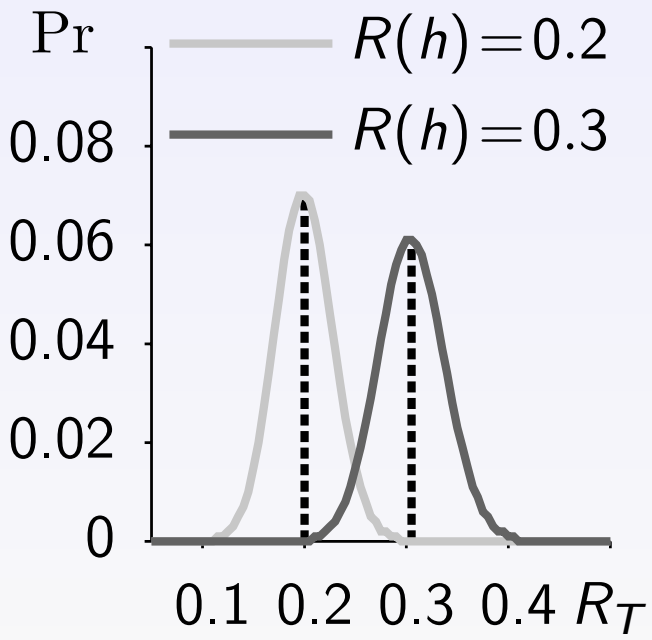
(D) Calculer la borne sur l'ensemble test (...)

Borne sur l'ensemble test

Le risque sur l'ensemble test $R_T(h)$ est un estimateur du risque $R(h)$.

Comme chaque exemple $\mathbf{z}'$ de l'ensemble T est généré de manière *iid* selon la distribution D, la possibilité que h classifie incorrectement $\mathbf{z}'$ est exactement $R(h)$:

$$R(h) = \Pr_{(\mathbf{x}',y')\sim D}\left(h(\mathbf{x}'_i) \neq y'_i \right)$$



Ainsi, la probabilité qu'un classificateur de risque $R(h)$ fasse k erreurs parmi n exemples test est donnée par la loi binomiale :

$$\Pr_{T\sim D^n}\left(R_T(h) = \frac{k}{n}\right) = \binom{n}{k} R(h)^k (1-R(h))^{n-k}$$

Borne sur l'ensemble test

Le calcul de « l'inverse de la queue de la distribution binomiale » permet de calculer la plus grande valeur de risque $R(h)$ telle que la probabilité est au moins δ d'obtenir un risque sur l'ensemble test d'au plus $R_T(h)$ sur n exemples.

Théorème de l'inverse de la queue de la binomiale (*Langford, 2005*)

Pour tout classificateur h de risque $R(h)$ et pour tout $\delta \in]0,1]$, on a :

$$\Pr_{T \sim D^n} \left(R(h) \leq \max \left\{ r : \sum_{i=0}^{n \cdot R_T} \binom{n}{i} r^i (1-r)^{n-i} \geq \delta \right\} \right) \geq 1 - \delta$$

Borne sur l'ensemble test

$$B(R_T, n, \delta) = \max \left\{ r : \sum_{i=0}^{n \cdot R_T} \binom{n}{i} r^i (1-r)^{n-i} \geq \delta \right\}$$

Borne sur l'ensemble test

Borne sur l'ensemble test

$$B(R_T, n, \delta) = \max \left\{ r : \sum_{i=0}^{n \cdot R_T} \binom{n}{i} r^i (1-r)^{n-i} \geq \delta \right\}$$

La borne sur l'ensemble test dépend donc de trois paramètres :

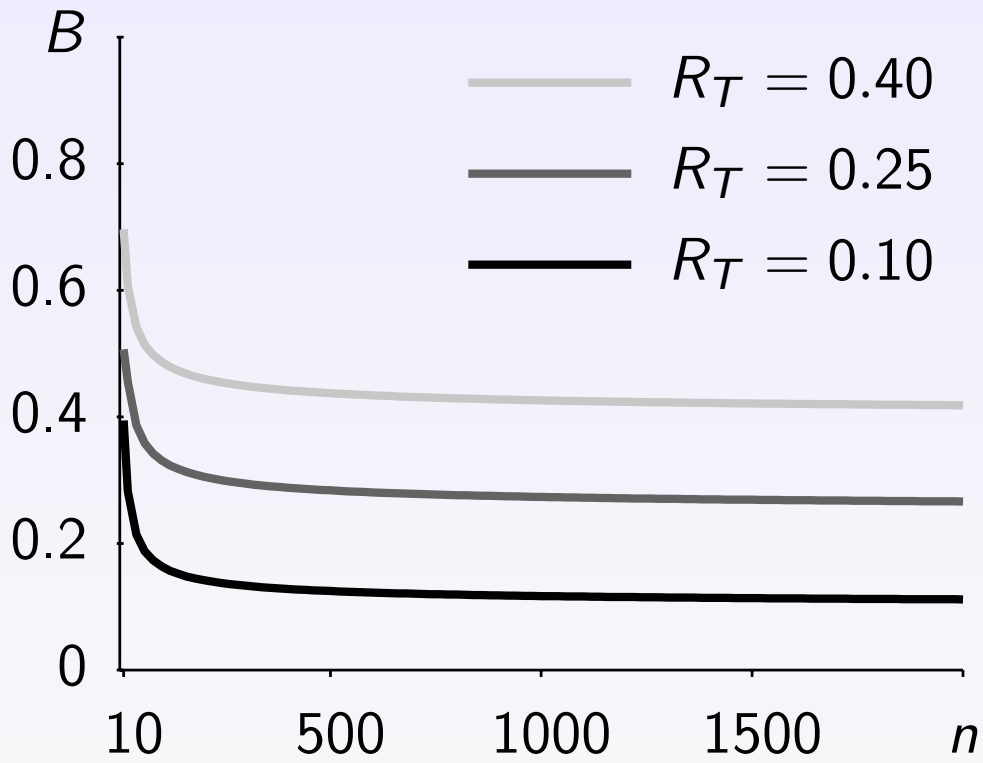
- R_T : Le risque sur l'ensemble test.
- n : Le nombre d'exemples de test.
- δ : Le paramètre de confiance.

Étudions l'influence des paramètres R_T et n sur la valeur de la borne...

Influence du nombre n d'exemples sur la borne

Borne sur l'ensemble test

$$B(R_T, n, \delta) = \max \left\{ r : \sum_{i=0}^{n \cdot R_T} \binom{n}{i} r^i (1-r)^{n-i} \geq \delta \right\}$$



Paramètre fixe : $\delta = 0.05$

Borne sur l'ensemble test

Avantage

- La borne obtenue est très serrée.

Inconvénient

- Nécessite de déterminer la taille de l'ensemble d'entraînement et de l'ensemble test :
  - Augmenter la taille de l'ensemble d'entraînement améliore (potentiellement) la qualité du classificateur ;
  - Diminuer la taille de l'ensemble test détériore la qualité de la borne.

Borne sur l'ensemble d'entraînement

Objectif

Énoncer une borne supérieure sur le risque d'un classificateur en analysant ses réalisations sur l'ensemble d'entraînement.

Défi

Comme le risque empirique $R_S(h)$ d'un classificateur n'est pas un bon indicateur du (vrai) risque $R(h)$, une borne sur l'ensemble d'entraînement doit dépendre de quantités autres.

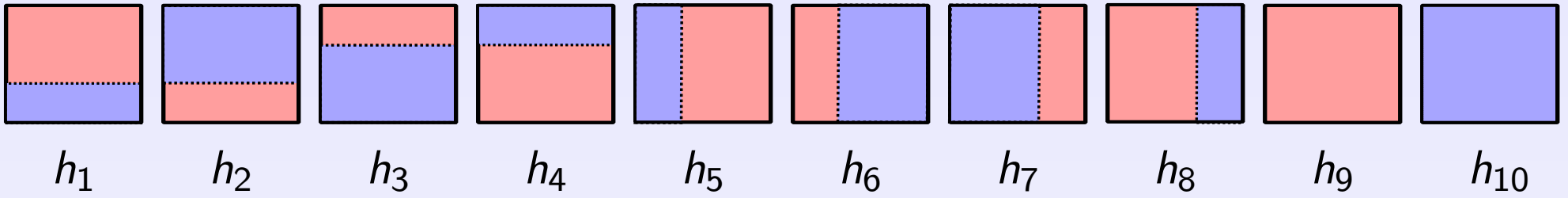
Exemple de la théorie PAC-Bayes

Théorie statistique de l'apprentissage qui permet de borner le risque d'un **vote de majorité** à partir de deux quantités :

- La « divergence » entre la connaissance du problème de classification « *a priori* » et « *a posteriori* » ;
- Le « risque de Gibbs » du vote de majorité.

Connaissance du problème *a priori*

Nous possédons un ensemble $\mathcal{H}$ de classificateurs de base :



La tâche de l'algorithme d'apprentissage est d'affecter un poids $Q(h)$ à chacun des classificateurs de base $h \in \mathcal{H}$ afin de créer le vote de majorité B_Q :

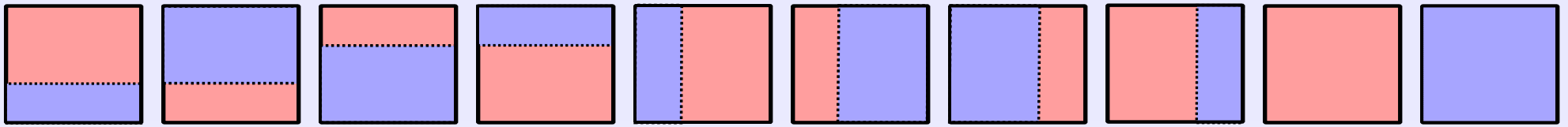
$$B_Q(\mathbf{x}) = \operatorname{sgn}\left[\sum_{h \in \mathcal{H}} Q(h) h(\mathbf{x})\right]$$

Classificateurs de base	$\mathcal{H} = \{h_1, \ldots, h_n\}$
Distribution de poids	Q *t.q.* $\sum_{h \in \mathcal{H}} Q(h) = 1$

Nous représentons par une distribution *a priori* P sur $\mathcal{H}$ les connaissances que nous avons du problème de classification avant d'étudier les données :

Distribution de poids *a priori*	P *t.q.* $\sum_{h \in \mathcal{H}} P(h) = 1$

Connaissance du problème *a priori*



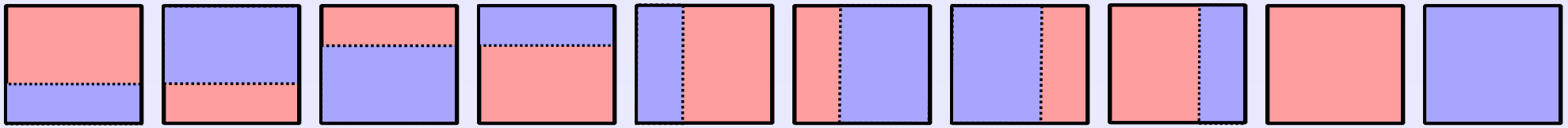
Nous quantifions la différence entre Q et P par la divergence de Kullback-Leibler :

Divergence de Kullback-Leibler entre les distributions Q et P

$$\mathrm{KL}(Q\|P) \stackrel{\text{def}}{=} \sum_{h\in\mathcal{H}} Q(h) \ln \frac{Q(h)}{P(h)} \qquad \textit{(cas discret)}$$

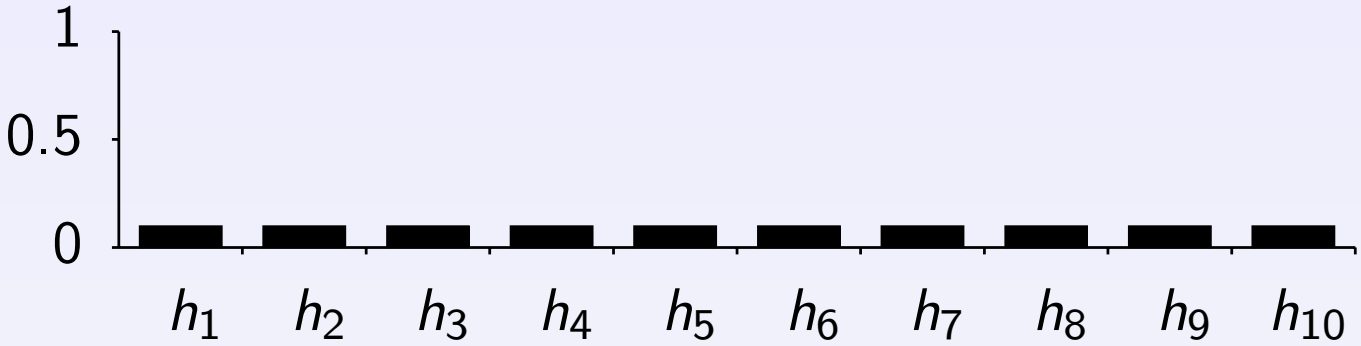
Idée générale : Si le classificateur obtenu par l'algorithme d'apprentissage est semblable au classificateur que nous croyons bon *a priori*, son risque empirique sera un bon indicateur de son vrai risque.

Connaissance du problème *a priori*



h_1 h_2 h_3 h_4 h_5 h_6 h_7 h_8 h_9 h_{10}

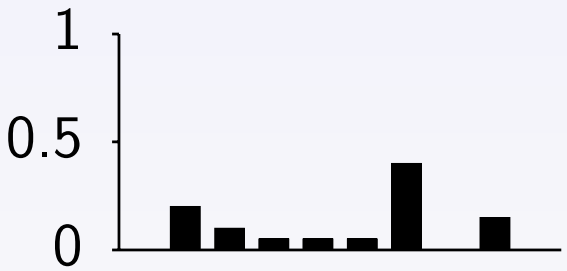
Distribution *a priori* P (non informative) :



$P(h) = \frac{1}{10}$
$\forall h \in \mathcal{H}$

Distributions *a posteriori* Q :

$\mathrm{KL}(Q\|P) = 0.0$



$\mathrm{KL}(Q\|P) = 0.65$

$\mathrm{KL}(Q\|P) = 2.3$

Classificateur de Gibbs

La théorie PAC-Bayes ne permet pas d'obtenir directement une borne sur le risque du vote de majorité mais plutôt une borne sur le classificateur de Gibbs relié au vote de majorité.

Classificateur de Gibbs relié au vote de majorité

Classificateur stochastique $G_Q(\mathbf{x})$: Pige aléatoirement un classificateur de base $h \in \mathcal{H}$ selon la distribution Q et retourne $h(\mathbf{x})$.

Risque empirique de Gibbs

$$R_S(G_Q) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^{m} \mathop{\mathbf{E}}_{h \sim Q} I(h(\mathbf{x}_i) \neq y_i)$$

Risque de Gibbs

$$R(G_Q) \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{(\mathbf{x},y) \sim D} \mathop{\mathbf{E}}_{h \sim Q} I(h(\mathbf{x}) \neq y)$$

Une borne sur le risque de Gibbs d'un classificateur se convertit simplement en une borne sur le vote de majorité correspondant, puisque :

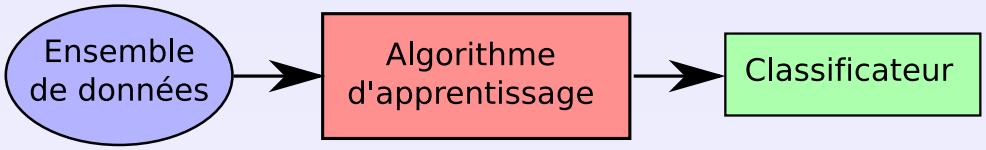
$$R(B_Q) \leq 2R(G_Q)$$

Borne PAC-Bayes sur l'ensemble d'entraînement

(A) Déterminer la distribution de poids P *a priori*.

(B) Entraîner le classificateur sur l'ensemble d'entraînement :

$$G_Q \leftarrow A(S)$$



(C) Calculer le risque de Gibbs sur l'ensemble d'entraînement :

$$R_S(G_Q) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^{m} \mathop{\mathbf{E}}_{h \sim Q} I(h(\mathbf{x}_i) \neq y_i)$$

(D) Calculer la divergence Kullback-Leibler :

$$\mathrm{KL}(Q \| P) \stackrel{\text{def}}{=} \sum_{h \in \mathcal{H}} Q(h) \ln \frac{Q(h)}{P(h)}$$

(E) Calculer la borne PAC-Bayes sur l'ensemble d'entraînement (...)

Théorème PAC-Bayes

Théorème PAC-Bayes (*McAllester, 2003*)

Pour toute distribution D, pour tout ensemble $\mathcal{H}$ de classificateurs, pour toute distribution P sur $\mathcal{H}$ et pour tout $\delta \in]0,1]$, on a :

$$\Pr_{S\sim D^m}\left(\forall Q \text{ sur } \mathcal{H}\colon R(G_Q) \leq R_S(G_Q) + \sqrt{\frac{\mathrm{KL}(Q\|P) + \ln\frac{m+1}{\delta}}{2m}}\right) \geq 1-\delta$$

Borne PAC-Bayes

$$B(R_S, m, \mathrm{KL}, \delta) = R_S + \sqrt{\frac{\mathrm{KL} + \ln\frac{m+1}{\delta}}{2m}}$$

Théorème PAC-Bayes

Borne PAC-Bayes

$$B(R_S, m, \mathrm{KL}, \delta) = R_S + \sqrt{\frac{\mathrm{KL} + \ln \frac{m+1}{\delta}}{2m}}$$

La borne PAC-Bayes dépend donc de quatre paramètres :

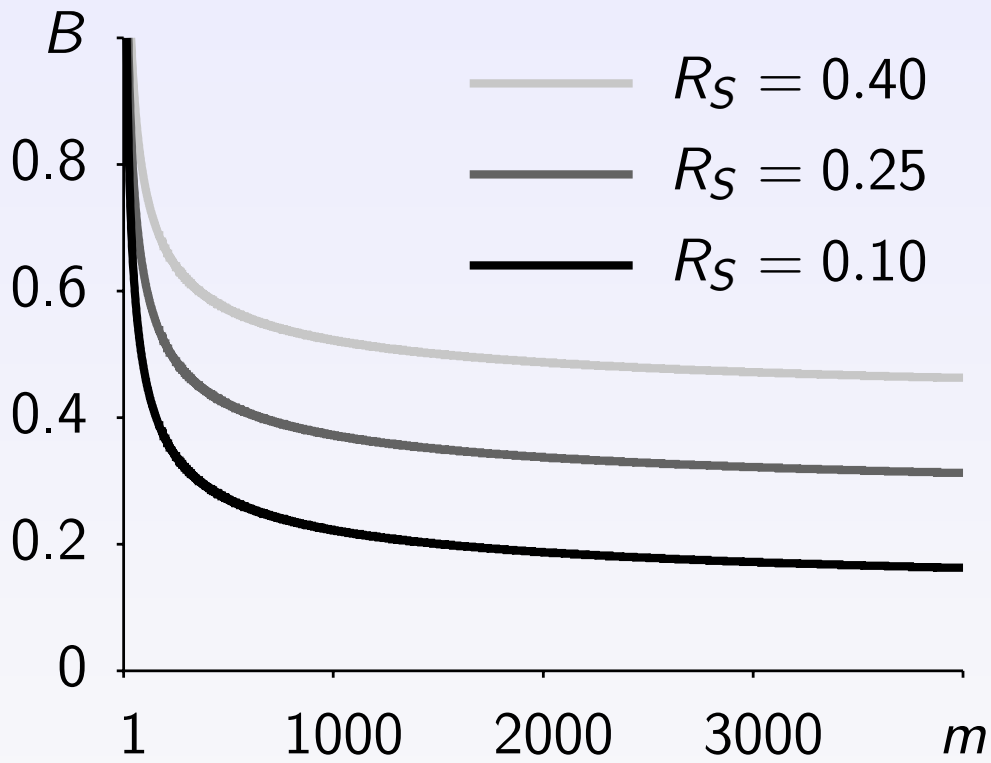
- R_S : Le risque empirique.
- m : Le nombre d'exemples d'entraînement.
- KL : La divergence entre les connaissances *a priori* et *a posteriori*.
- δ : Le paramètre de confiance.

Étudions l'influence des paramètres R_S, m et KL sur la valeur de la borne...

Influence du nombre m d'exemples sur la borne

Borne PAC-Bayes

$$B(R_S, m, \mathrm{KL}, \delta) = R_S + \sqrt{\frac{\mathrm{KL} + \ln \frac{m+1}{\delta}}{2m}}$$

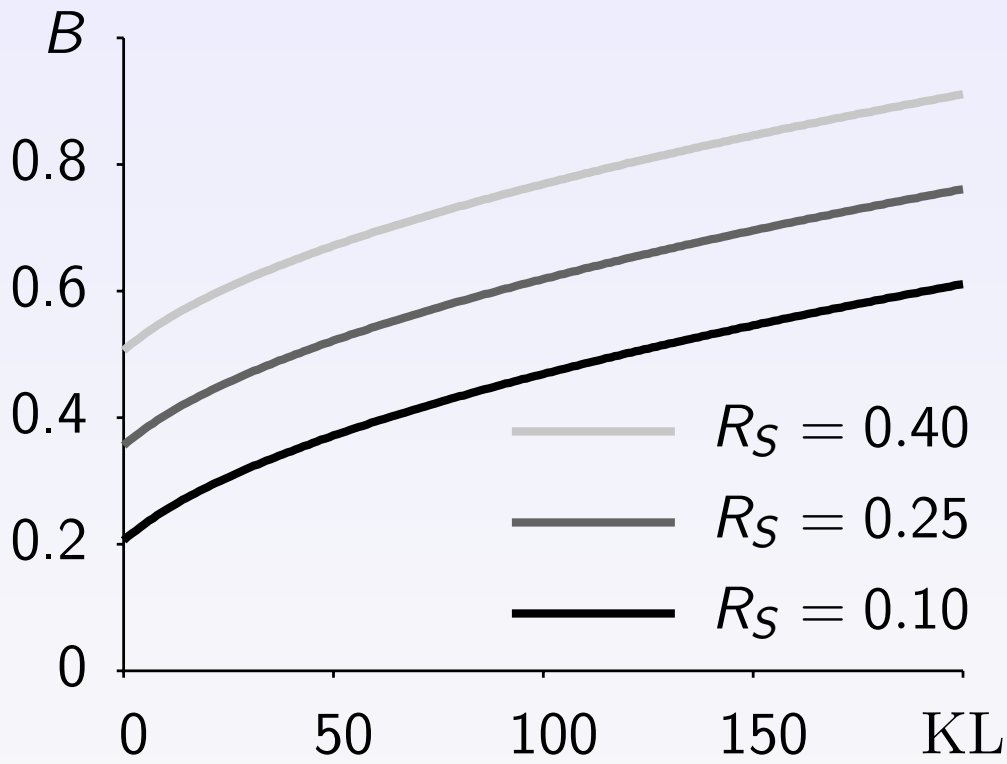


Paramètres fixes : $\mathrm{KL} = 20$ et $\delta = 0.05$

Influence de la divergence $\mathrm{KL}(Q\|P)$ sur la borne

Borne PAC-Bayes

$$B(R_S, m, \mathrm{KL}, \delta) = R_S + \sqrt{\frac{\mathrm{KL} + \ln \frac{m+1}{\delta}}{2m}}$$



Paramètres fixes : $m = 400$ et $\delta = 0.05$

Borne sur l'ensemble d'entraînement

Avantages

- Permet de dédier toutes les données étiquetées disponibles à l'apprentissage ET d'obtenir une borne sur le risque ;
- Contribue à mieux comprendre ce qui caractérise un bon classificateur.

Inconvénient

- Les bornes sur l'ensemble d'entraînement sont généralement moins serrées que les bornes sur l'ensemble test.

Et mes travaux de maîtrise dans tout cela ?

Création d'algorithmes d'apprentissage inspirés de la théorie PAC-Bayes.

Idée de départ

- Les bornes PAC-Bayes prédisent bien le risque d'un classificateur ;
- Elles sont exprimées par de simples expressions mathématiques.

$\Longrightarrow$ Trouver le classificateur qui minimise l'expression d'une borne.

Méthodologie

- Spécialiser les bornes PAC-Bayes à la famille des séparateurs linéaires ;
- Minimisation par la technique de descente de gradient conjugué.

Résultats

- Certaines versions de mes algorithmes sont compétitives avec les algorithmes populaires (SVM et AdaBoost) ;
- Mais ils sont un peu lents d'exécution ...