ENS-CACHAN, L3
Cours d'Apprentissage

Michèle Sebag

TAO: Theme Apprentissage & Optimisation

http://tao.lri.fr/tiki-index.php

21 mars 2012

Ce cours

Apprentissage statistique

Machines à Vecteurs Supports Linéaires, classification

- Cas séparable
- Cas non séparable
- L'astuce des noyaux (kernel trick)

Rappels

Vapnik, 1995, 1998

Input

$\mathcal{E} = \{(x_i, y_i)\},\ x_i \in \mathbb{R}^m,\ y_i \in \{-1, 1\}, i = 1..n\}\ \ (x_i, y_i) \sim P(x, y)$

Output : $\hat{h} : \mathbb{R}^m \mapsto \{-1, 1\}$ ou $\mathbb{R}$.

$\hat{h}$ estime y

Critère : idéalement, minimiser l'erreur en généralisation

$$Err(h) = \int \ell(y, \hat{h}(x)) dP(x, y)$$

$\ell =$ fonction de perte : $1_{y \neq \hat{h}(x)}$, $(y - \hat{h}(x))^2$

$P(x, y) =$ distribution jointe des données.

Compromis Biais-Variance

Choix de modèle : L'espace $\mathcal{H}$ dans lequel on cherche $\hat{h}$.

Biais : La distance entre y et $h^* = argmin\{Err(h), h \in \mathcal{H}\}$.

ce qu'on peut espérer au mieux

Variance : La distance entre $\hat{h}$ et h^*

entre le meilleur h^* et le $\hat{h}$ qu'on apprend

Note :
Seul le risque empirique (sur les données) est donné

$$Err_{emp,n}(\hat{h}) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \hat{h}(x_i))$$

Principe :

$$Err(\hat{h}) < Err_{emp,n}(\hat{h}) + \mathcal{B}(n, \mathcal{H})$$

Si $\mathcal{H}$ est "raisonnable", $Err_{emp,n} \to Err$ quand $n \to \infty$

Apprentissage statistique

Statistical Learning Theory

Voir l'apprentissage d'un point de vue statistique.

But de la théorie

en général

Modéliser un phénomène réel / artificiel, pour pouvoir:

* comprendre
* prédire
* exploiter

Généralités

Une théorie: hypothèses $\rightarrow$ prédictions

- Hypothèses sur le phénomène ici l'apprentissage
- Prédiction sur le comportement du phénomène erreurs

Théorie $\rightarrow$ algorithme

- Optimiser les quantités permettant la prédiction
- Nothing practical like a good theory! Vapnik

Généralités

Une théorie: hypothèses → prédictions

- Hypothèses sur le phénomène — ici l'apprentissage
- Prédiction sur le comportement du phénomène — erreurs

Théorie → algorithme

- Optimiser les quantités permettant la prédiction
- Nothing practical like a good theory! — Vapnik

Forces/faiblesses

\+ Hypothèses plus fortes → prédictions plus précises

MAIS si les hypothèses ne sont pas adaptées, rien ne va.

De quelle théorie avons-nous besoin ?

Approche moyenne

un exemple

- Un ensemble de données

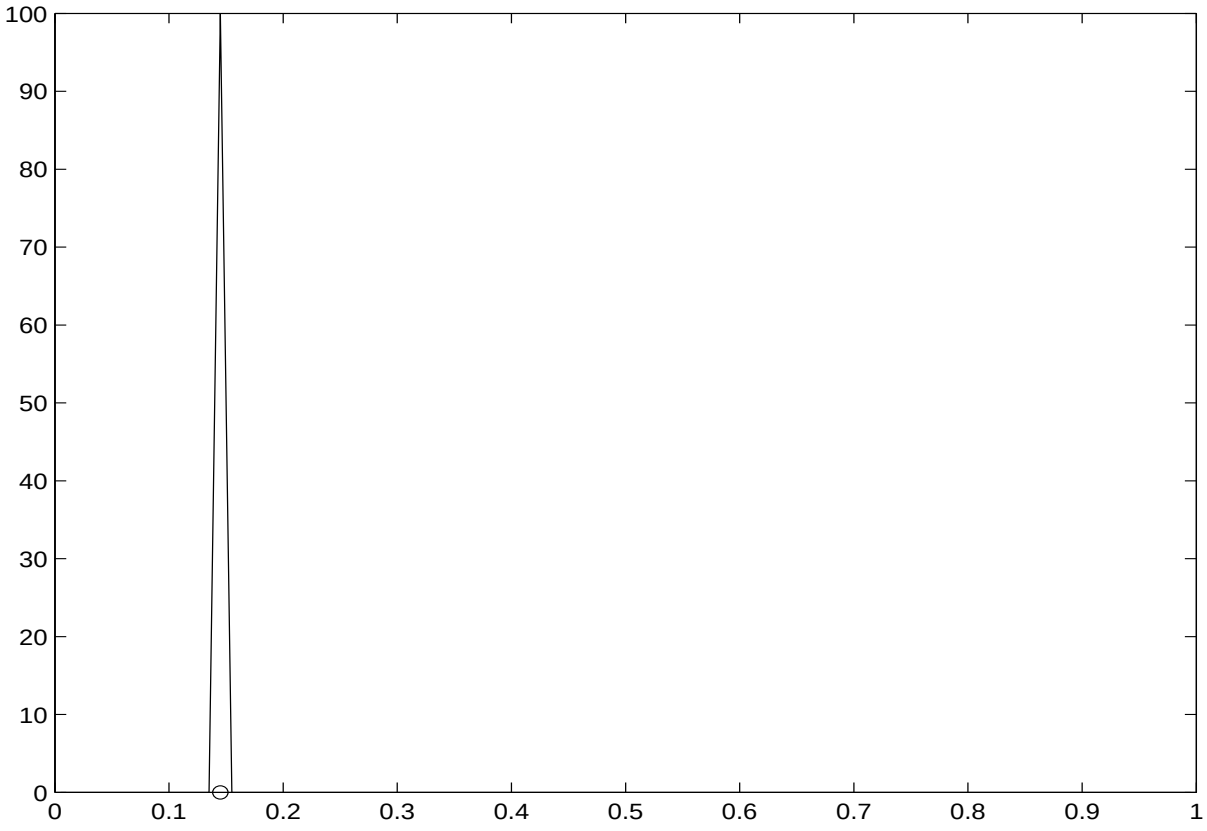
breast cancer

- $\bar{x}^+$: moyenne des exemples positifs
- $\bar{x}^-$: moyenne des exemples négatifs
- $h(x) = +1$ iff $d(x, \bar{x}^+) < d(x, \bar{x}^-)$

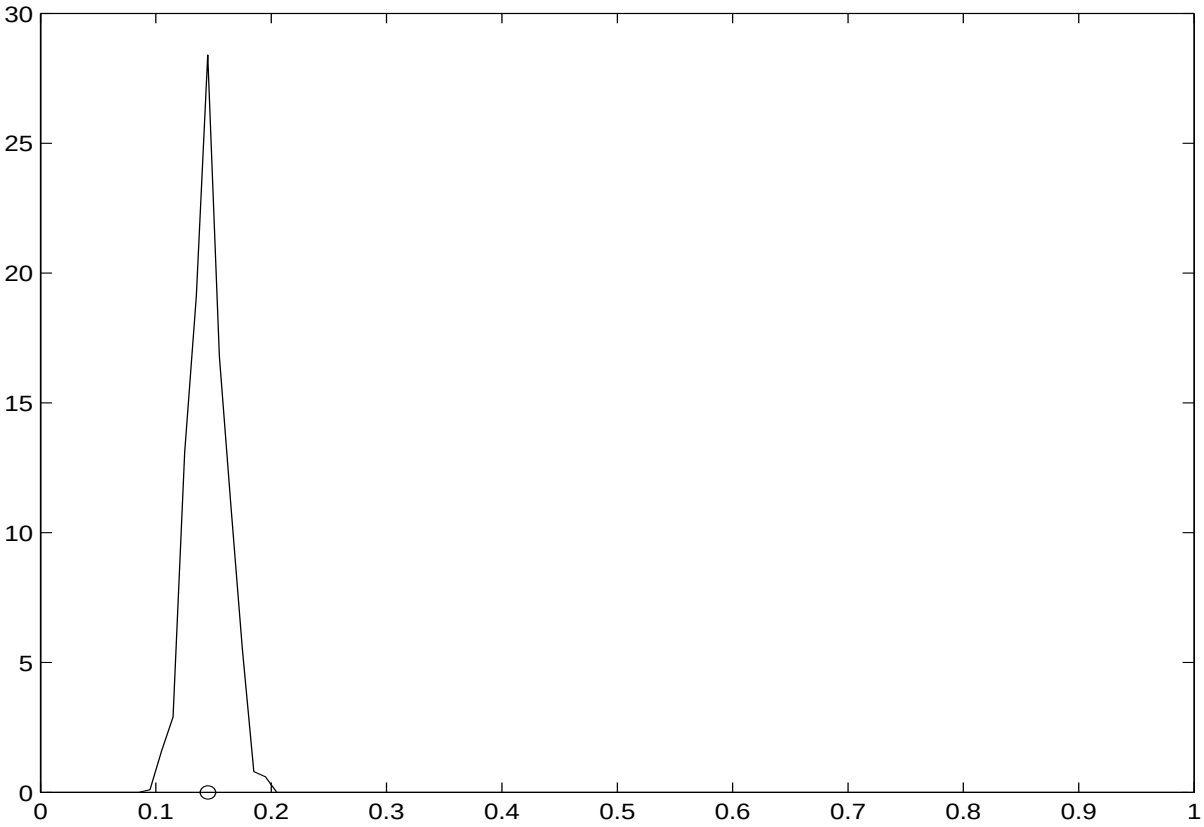
Estimer l'erreur en généralisation

- Données $\rightarrow$ Données d'apprentissage, données de test
- Apprendre $\bar{x}^+$ et $\bar{x}^-$ sur l'apprentissage, mesurer le nombre d'erreurs sur le test

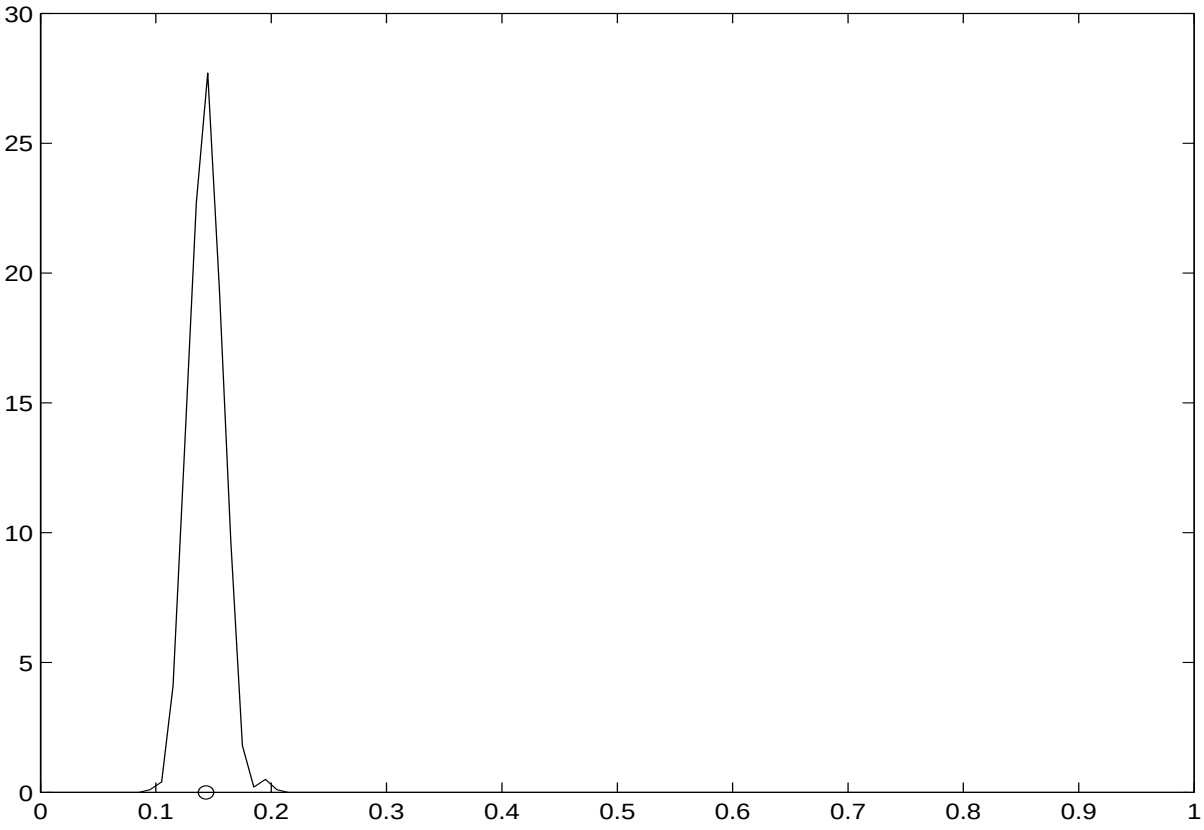
Error distribution: full dataset



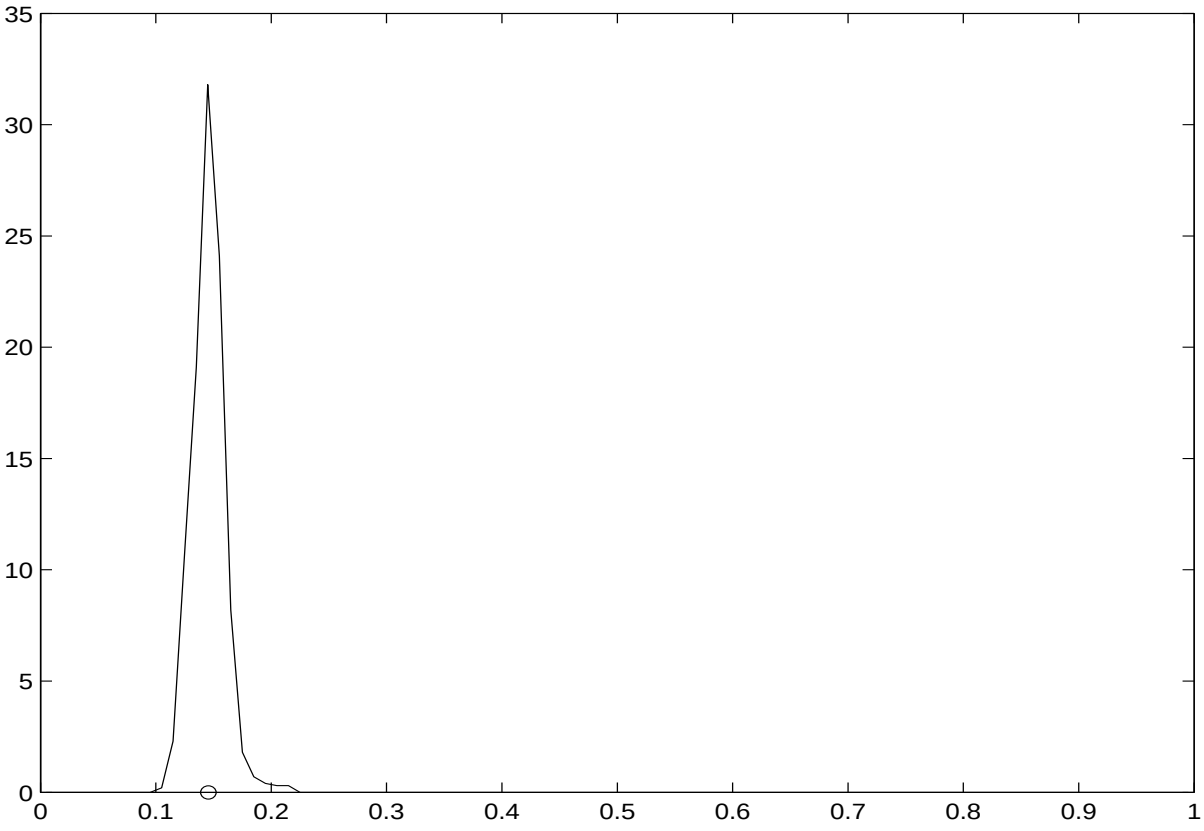
Error distribution: dataset size: 342



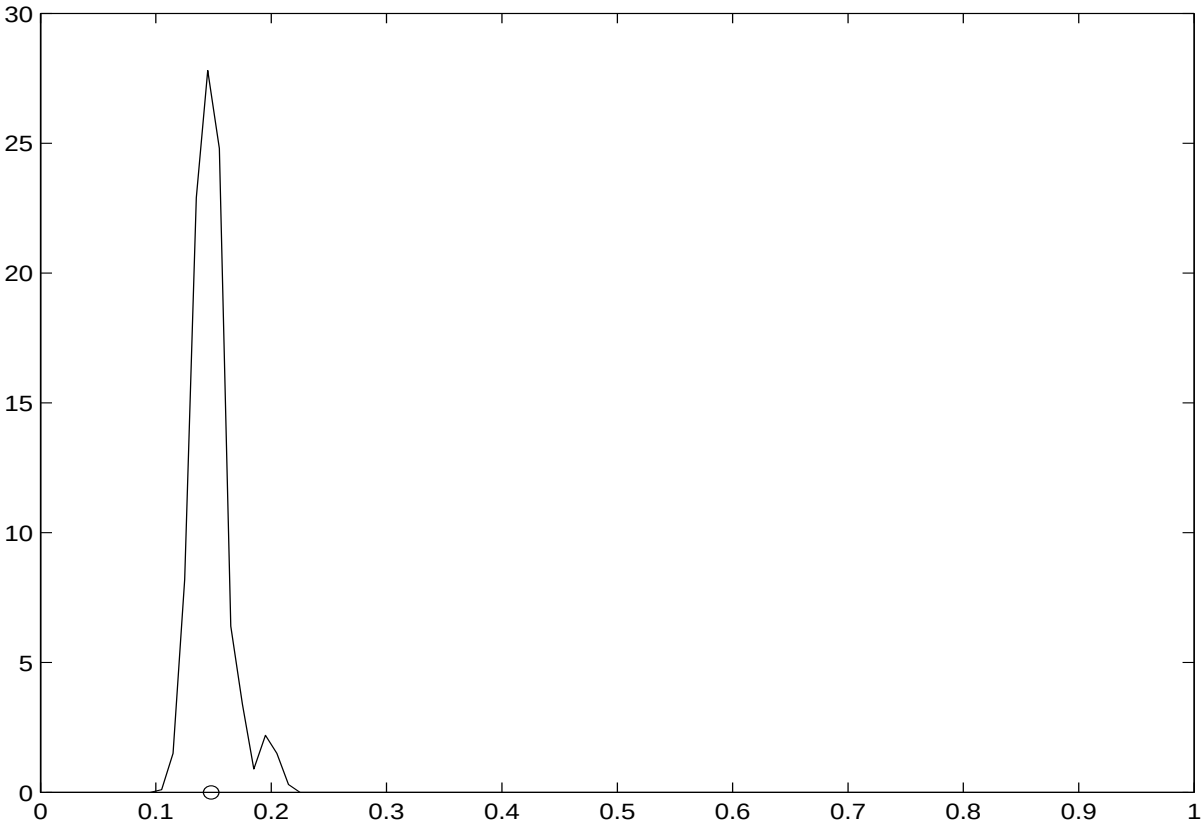
Error distribution: dataset size: 273



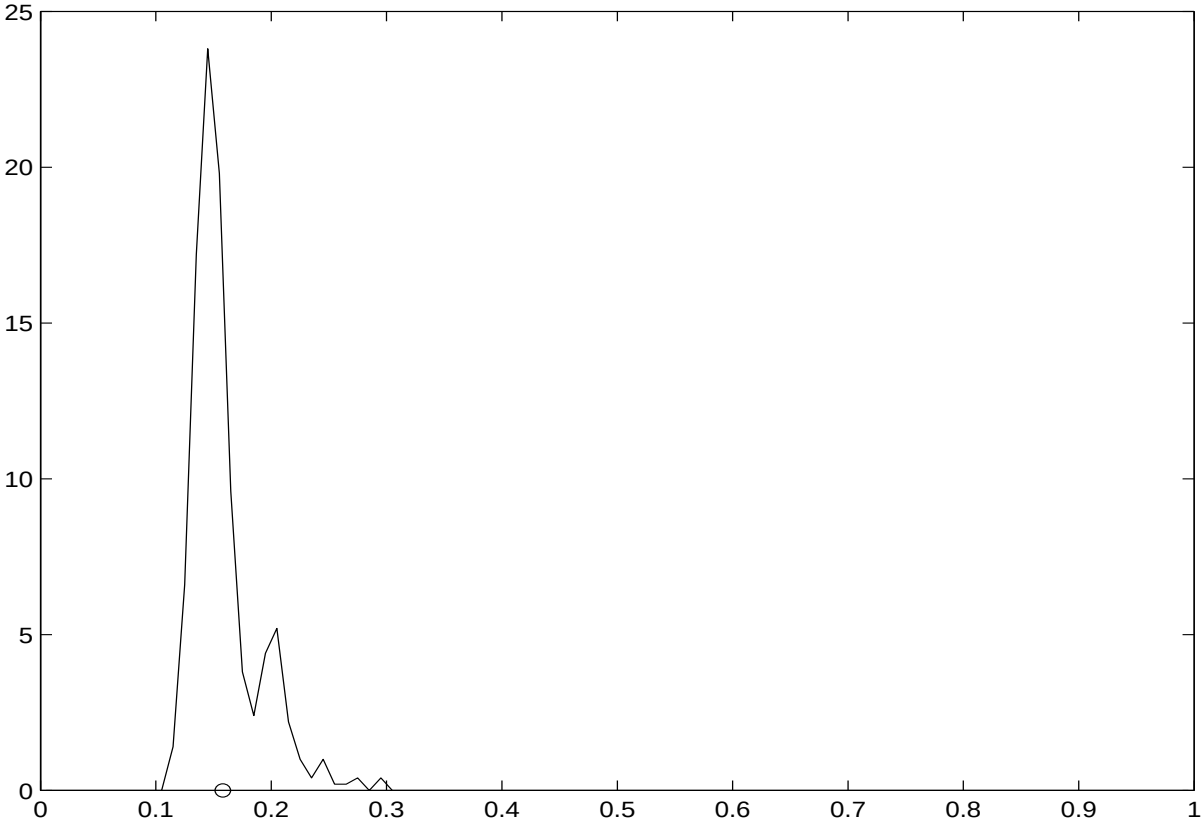
Error distribution: dataset size: 205



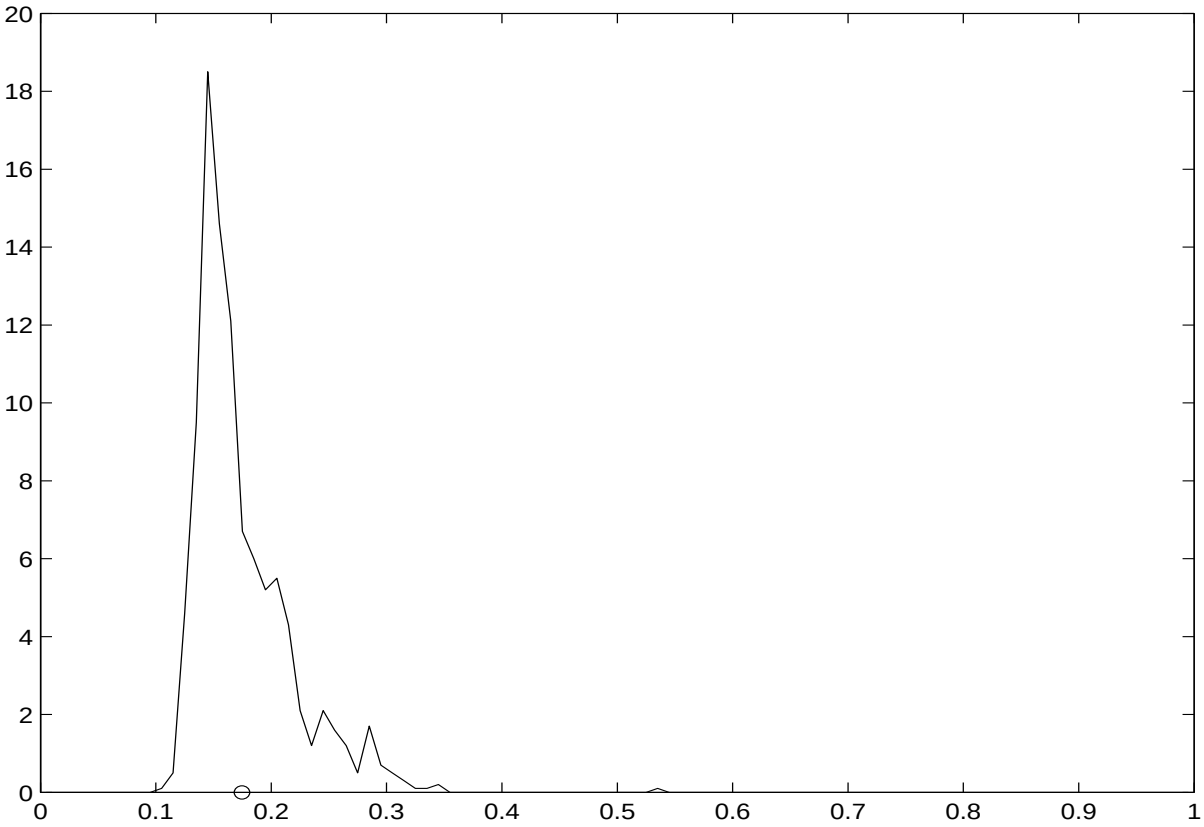
Error distribution: dataset size: 137



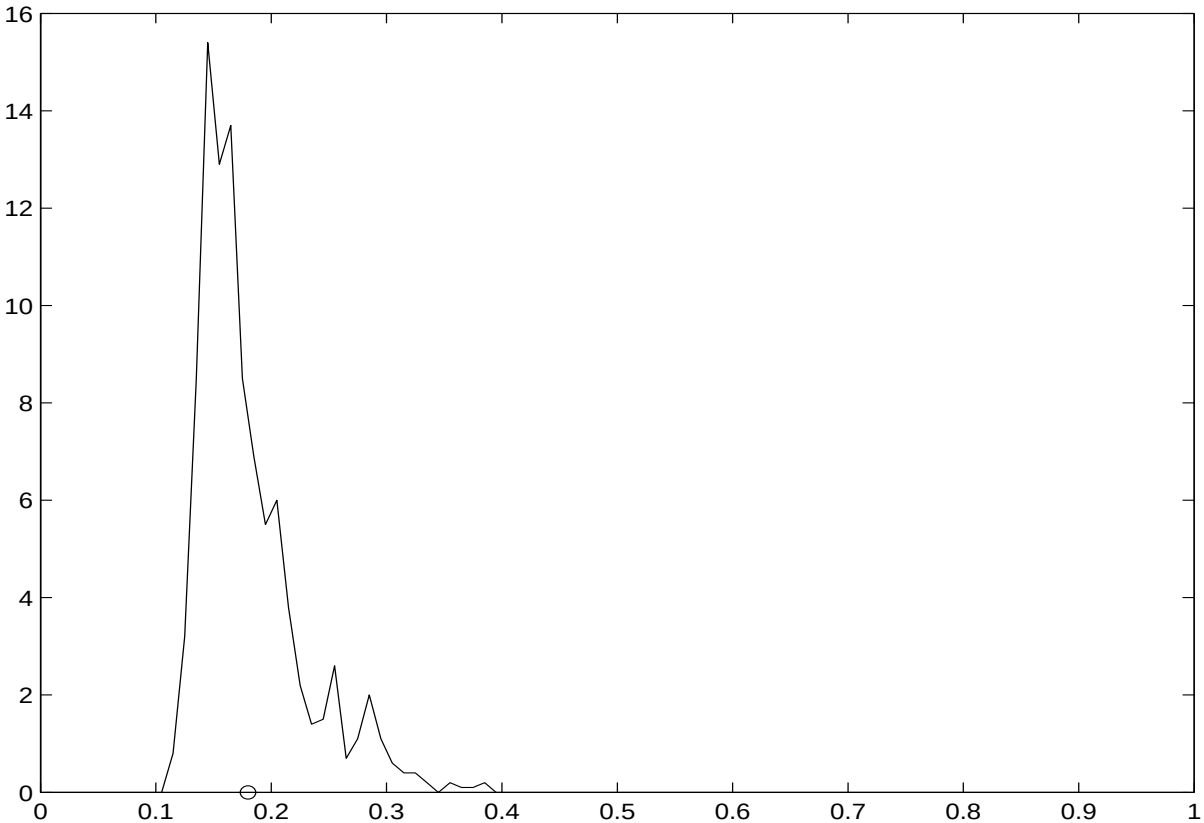
Error distribution: dataset size: 68



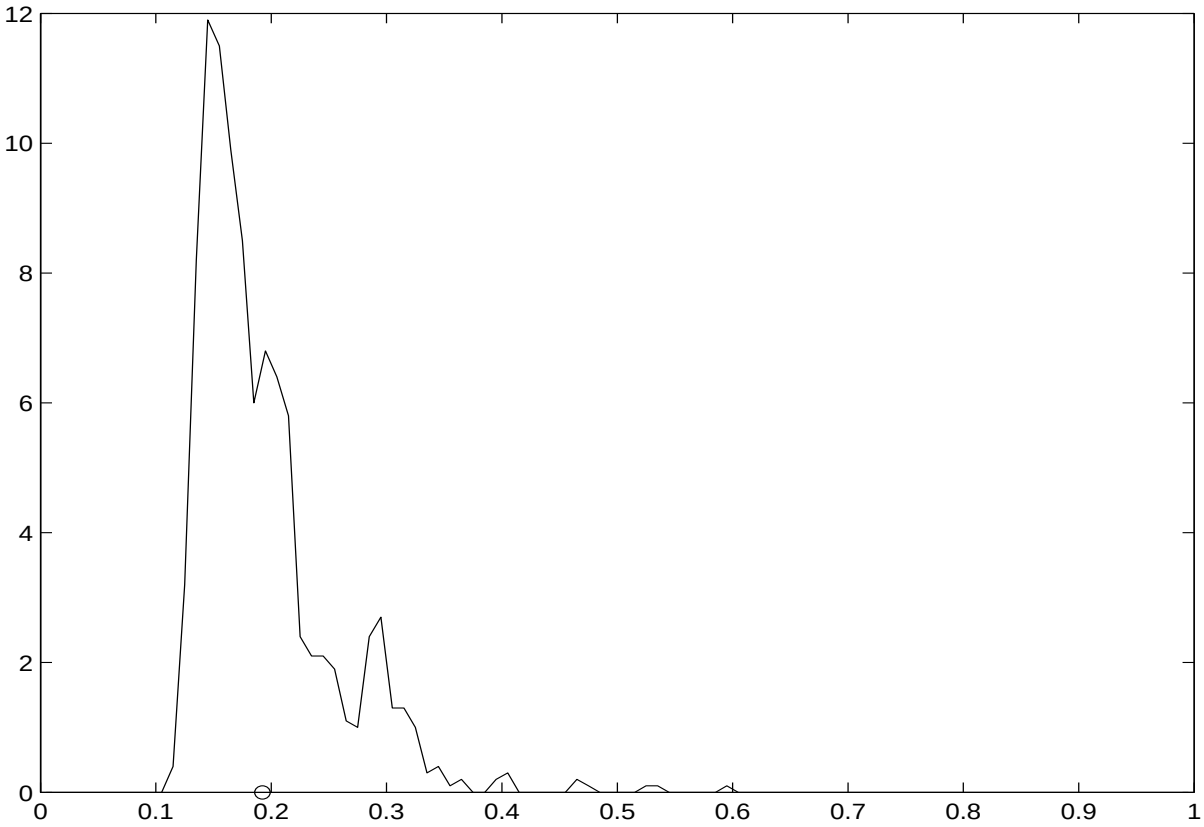
Error distribution: dataset size: 34



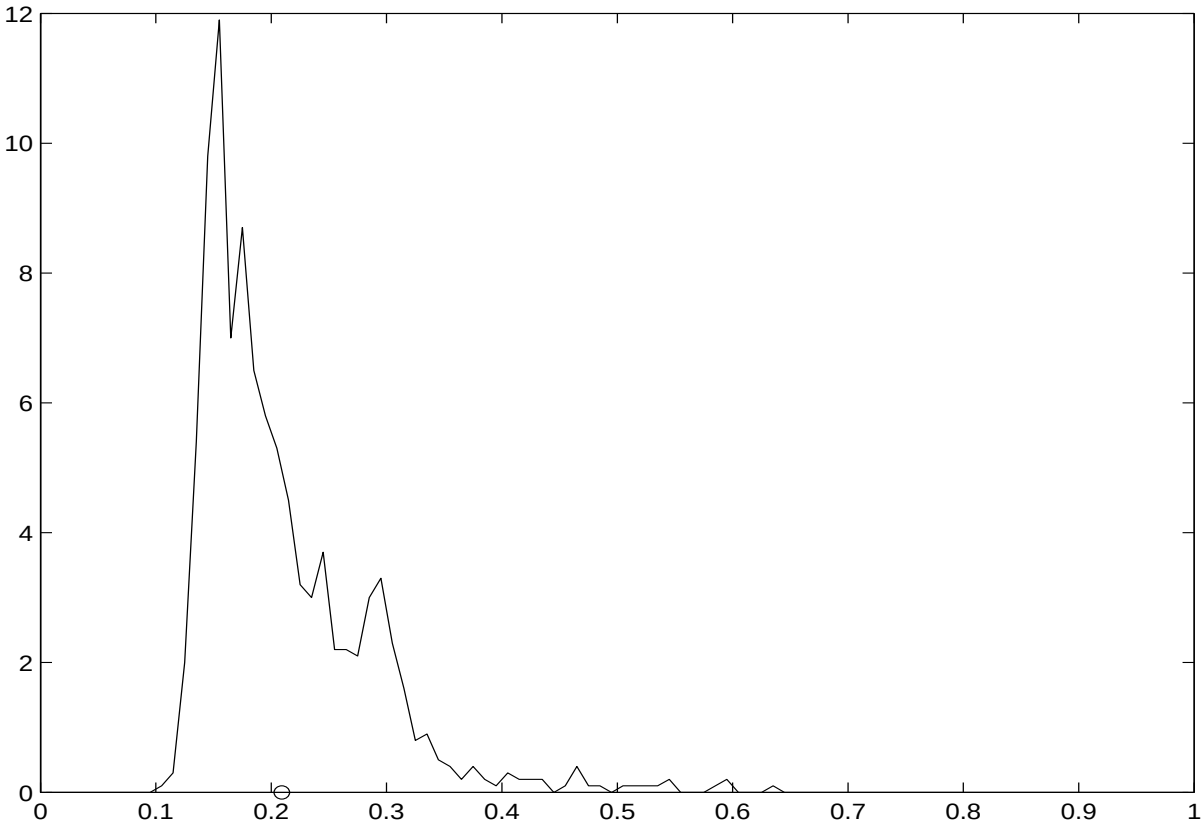
Error distribution: dataset size: 27



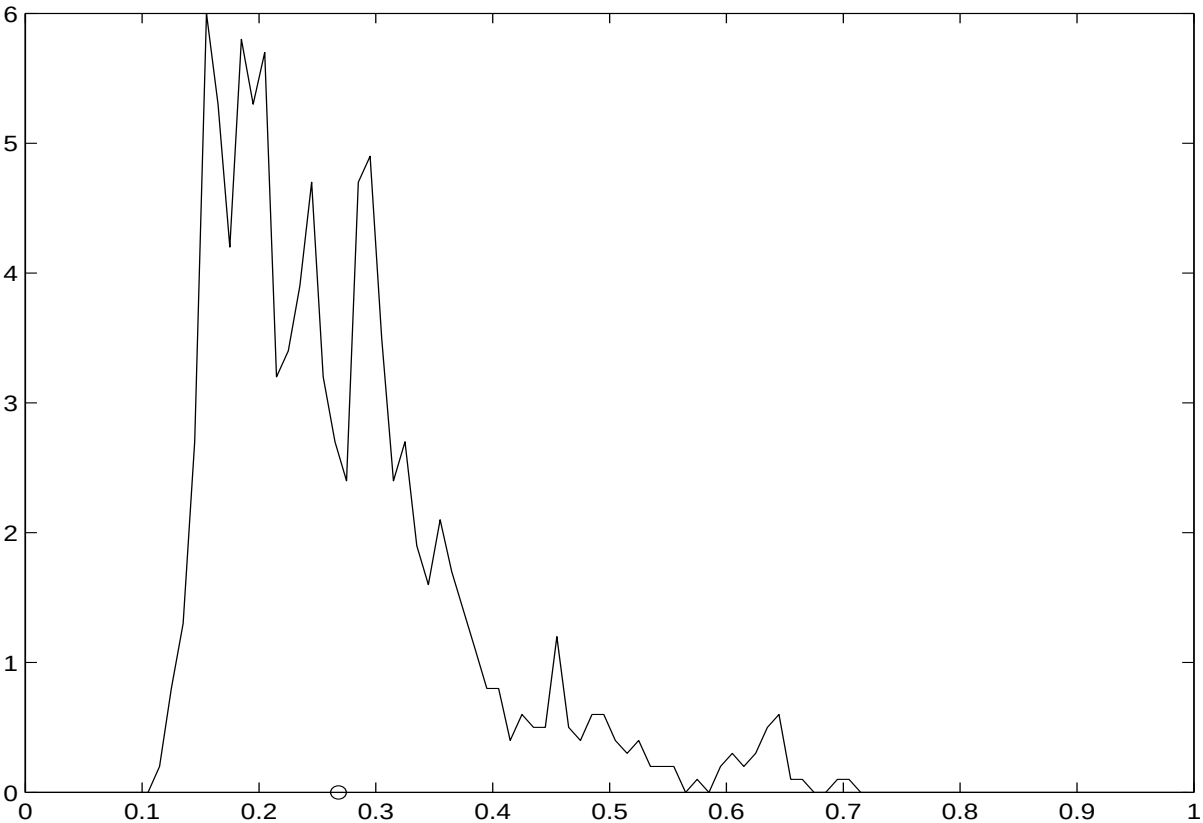
Error distribution: dataset size: 20



Error distribution: dataset size: 14



Error distribution: dataset size: 7



Statistiques classiques vs apprentissage statistique

Statistique classique

- Erreur moyenne

Nous voulons des garanties

- Modèle PAC Probably Approximately Correct
- Quelle est la probabilité que l'erreur soit supérieure à un seuil donné ?

Exemple

Supposons

$$Err(h) > \varepsilon$$

Quelle est la probabilité pour que $Err_{emp,n}(h) = 0$?

$$\begin{aligned} Pr(Err_{emp,n}(h) = 0, Err(h) > \varepsilon) &= (1 - Err(h))^n \\ &< (1 - \varepsilon)^n \\ &< exp(-\varepsilon n) \end{aligned}$$

Exemple

Supposons

$$Err(h) > \varepsilon$$

Quelle est la probabilité pour que $Err_{emp,n}(h) = 0$?

$$\begin{aligned} Pr(Err_{emp,n}(h) = 0, Err(h) > \varepsilon) &= (1 - Err(h))^n \\ &< (1 - \varepsilon)^n \\ &< exp(-\varepsilon n) \end{aligned}$$

Donc si nous voulons garantir un risque δ

$$Pr(Err_{emp,n}(h) = 0, Err(h) > \varepsilon) < \delta$$

Exemple

Supposons

$$Err(h) > \varepsilon$$

Quelle est la probabilité pour que $Err_{emp,n}(h) = 0$?

$$\begin{aligned} Pr(Err_{emp,n}(h) = 0, Err(h) > \varepsilon) &= (1 - Err(h))^n \\ &< (1 - \varepsilon)^n \\ &< exp(-\varepsilon n) \end{aligned}$$

Donc si nous voulons garantir un risque δ

$$Pr(Err_{emp,n}(h) = 0, Err(h) > \varepsilon) < \delta$$

L'erreur est au plus

$$\varepsilon < \frac{1}{n} \ln \frac{1}{\delta}$$

Apprentissage statistique

Principe

- Trouver une borne sur l'erreur en généralisation;
- Minimiser la borne.

Note

Voir $\hat{h}$ comme une variable aléatoire, dépendant de l'ensemble d'apprentissage $\mathcal{E}$ et du nombre n d'exemples. $\hat{h}_n$

Résultats

- deviation of the empirical error

$$Err(\hat{h}_n) \leq Err_{emp,n}(\hat{h}_n) + \mathcal{B}_1(n, \mathcal{H})$$

- biais-variance

$$Err(\hat{h}_n) \leq Err(h^*) + \mathcal{B}_2(n, \mathcal{H})$$

Types d'approches

Minimisation du risque empirique

- Model selection: Choose hypothesis space $\mathcal{H}$
- Choose $\hat{h}_n = argmin\{Err_n(h),\ h \in \mathcal{H}\}$

risque de surapprentissage

Minimisation du risque structurel

Given $\mathcal{H}_1 \subset \mathcal{H}_2 \subset ... \subset \mathcal{H}_k$,

$$\text{Find } \hat{h}_n = argmin\{Err_n(h) + pen(n,k),\ h \in \mathcal{H}_k\}$$

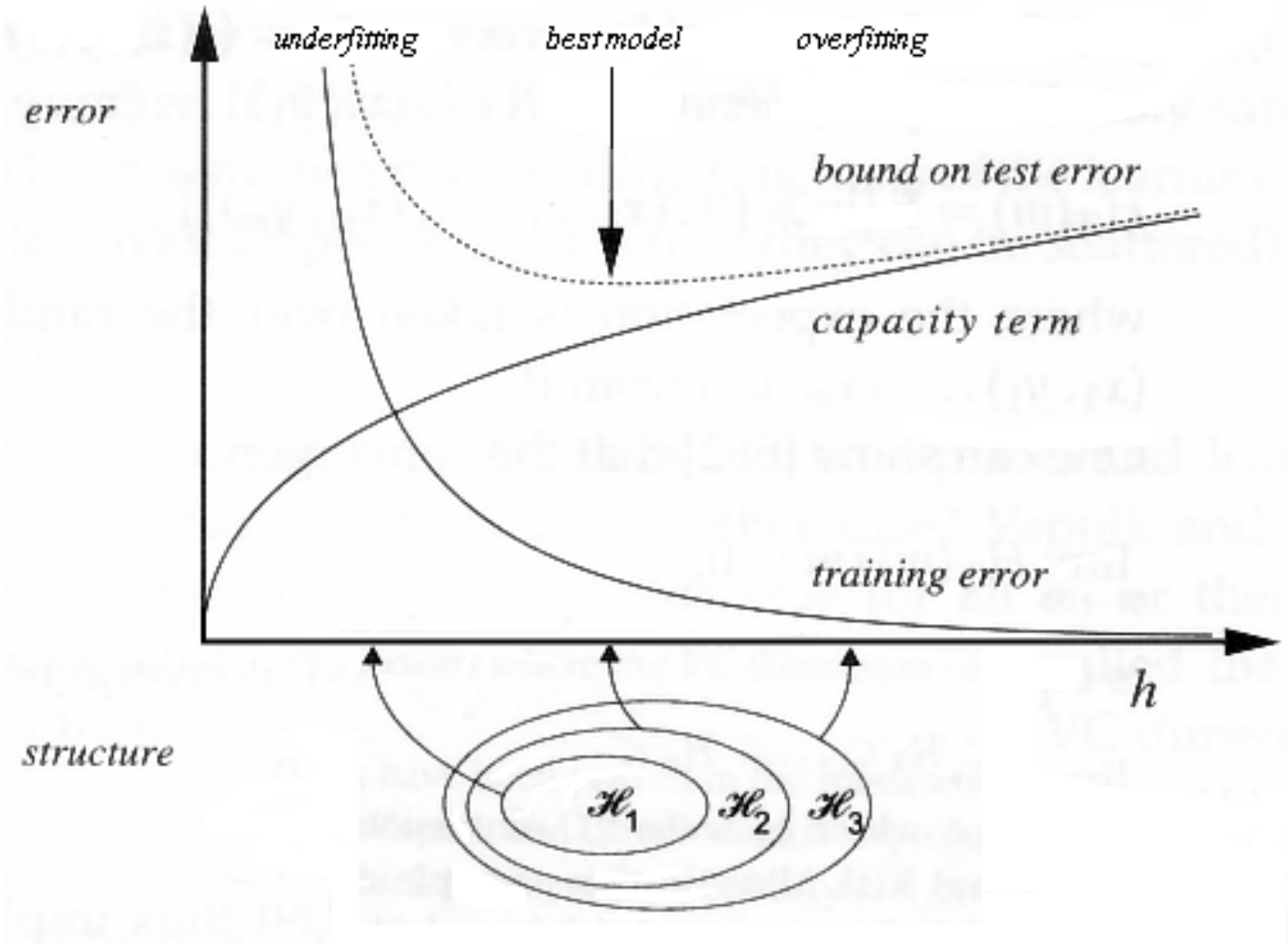
Quelle pénalisation ?

Régularisation

$$\text{Find } \hat{h}_n = argmin\{Err_n(h) + \lambda||h||,\ h \in \mathcal{H}\}$$

λ determiné par validation croisée

Minimisation du risque structurel



Outil 1. Borne de Hoeffding

Hoeffing 1963

Soit $X_1 \ldots, X_n$ des variables aléatoires indépendantes, X_i à valeur dans $[a_i, b_i]$
Soit leur moyenne empirique $\overline{X} = (X_1 + \cdots + X_n)/n$

Théorème

$$\Pr(|\overline{X} - \mathrm{E}[\overline{X}]| \geq \varepsilon) \leq 2 \exp\left(-\frac{2\varepsilon^2 n^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right)$$

où $\mathrm{E}[\overline{X}]$ est l'espérance de $\overline{X}$.

Borne de Hoeffding, suite

Application: si

$$Pr(|Err(g) - Err_n(g)| > \varepsilon) < 2e^{-2n\varepsilon^2}$$

alors avec probabilité au moins $1 - \delta$

$$Err(g) \leq Err_n(g) + \sqrt{\frac{\log 2/\delta}{2n}}$$

mais ceci ne dit rien sur $\hat{h}_n$...

Uniform deviations

$$|Err(\hat{h}_n) - Err_n(\hat{h}_n)| \leq sup_{h \in H} |Err(h) - Err_n(h)|$$

- si $\mathcal{H}$ fini, considérer la somme des $|Err(h) - Err_n(h)|$
- si $\mathcal{H}$ est infini, considérer sa trace sur les données

Apprentissage statistique. Définitions

Vapnik 92, 95, 98

Trace de $\mathcal{H}$ sur $\{x_1, \ldots x_n\}$

$$Tr_{x_1,..x_n}(\mathcal{H}) = \{(h(x_1), ..h(x_n)),\ h \in \mathcal{H}\}$$

Fonction de croissance

$$S(\mathcal{H}, n) = sup_{(x_1,..x_n)} |Tr_{x_1,..x_n}(\mathcal{H})|$$

Définitions, suite

Capacité d'un espace d'hypothèses $\mathcal{H}$

Si une fonction de $\mathcal{H}$ peut faire "n'importe quoi" sur n exemples, et que la base d'apprentissage comprend n exemples, on ne peut être sur de rien.

$\mathcal{H}$ pulvérise $(x_1, \ldots x_n)$ ssi shatters

$$\forall (y_1, \ldots y_n) \in \{1, -1\}^n, \exists h \in \mathcal{H} \text{ s.t. } \forall i = 1 \ldots n, h(x_i) = y_i$$

Dimension de Vapnik Cervonenkis

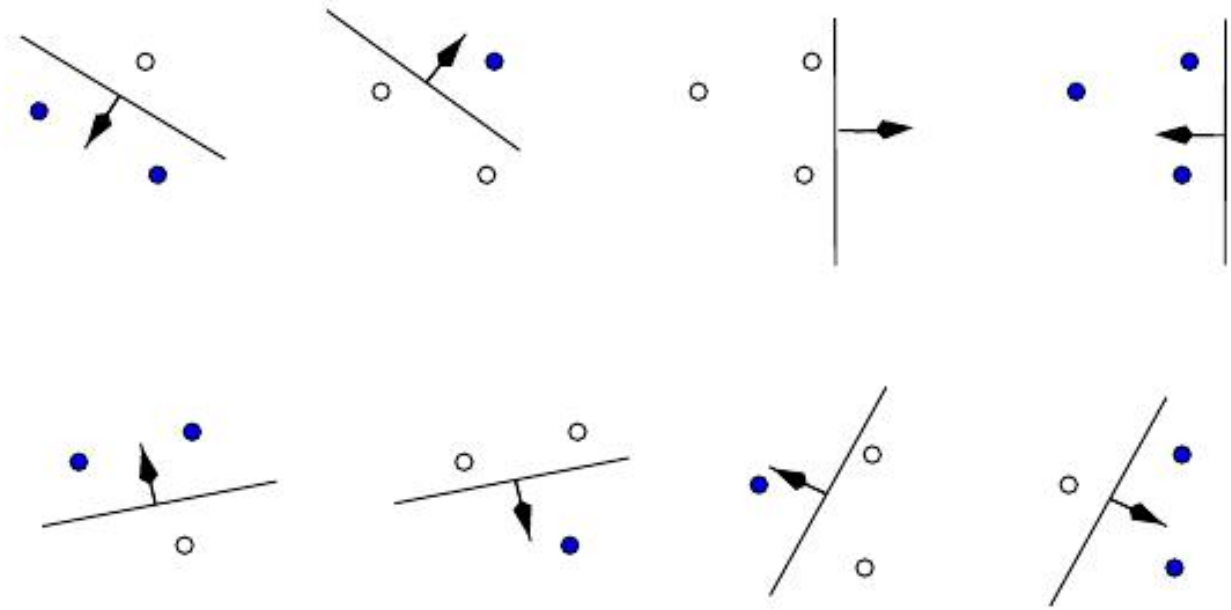
$$\text{VC}(\mathcal{H}) = \max \{n, (x_1, \ldots x_n) \text{ pulvérisé par } \mathcal{H}\}$$

$$VC(\mathcal{H}) = max\{n \ / \ S(\mathcal{H}, n) = 2^n\}$$

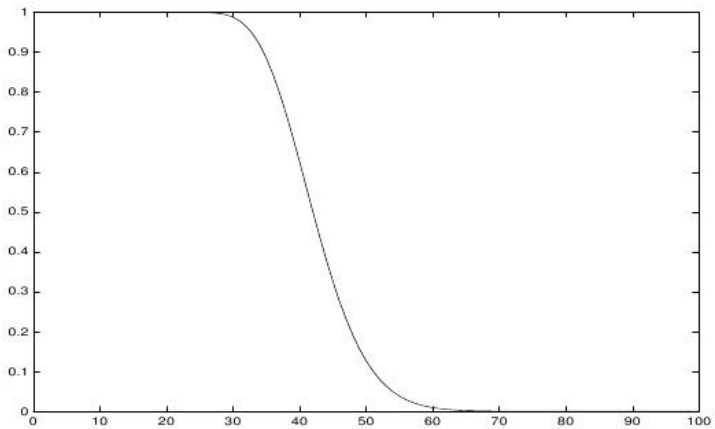
Un ensemble pulvérisé

3 points dans $\mathbb{R}^2$

$\mathcal{H}=$ droites du plan



Fonction de croissance des fonctions linéaires sur $\mathbb{R}^{20}$



abscisse: n
ordonnée: $S(\mathcal{H}, n) \times \frac{1}{2^n}$

La fonction de croissance est exponentielle en n pour $n < d = VC(\mathcal{H})$, puis polynomiale (en n^d).

Théorème, cas séparable

$\forall \delta > 0$, avec probabilité au moins $1 - \delta$

$$Err(h) \leq Err_n(h) + \sqrt{2 \frac{log(S(H, 2n)) + log(2/\delta)}{n}}$$

Idée 1: Double sample trick

Considerer un second échantillon $\mathcal{E}'$

$$Pr(sup_h(Err(h) - Err_n(h)) \geq \varepsilon) \leq$$

$$2Pr(sup_h(Err'_n(h) - Err_n(h)) \geq \varepsilon/2)$$

où $Err'_n(h)$ est l'erreur empirique sur $\mathcal{E}'$.

Double sample trick

- Il existe h t.q.
- A: $Err_{\mathcal{E}}(h) = 0$
- B: $Err(h) \geq \varepsilon$
- C: $Err_{\mathcal{E}'} \geq \frac{\varepsilon}{2}$

$$\begin{aligned} P(A(h)\&C(h)) &\geq P(A(h)\&B(h)\&C(h)) \\ &= P(A(h)\&B(h)).P(C(h)|A(h)\&B(h)) \\ &\geq \tfrac{1}{2}P(A(h)\&B(h)) \end{aligned}$$

Outil 2. Lemme de Sauer

Lemme de Sauer

Si $d = VC(\mathcal{H})$

$$S(\mathcal{H}, n) = \sum_{i=1}^{d} \binom{n}{i}$$

For $n > d$,

$$S(H, n) \leq \left(\frac{en}{d}\right)^d$$

Idée 2: Symmétrisation

Compter les permutations qui échangent $\mathcal{E}$ et $\mathcal{E}'$.

Résumé

$$\boxed{Err(h) \leq Err_n(h) + \mathcal{O}(\sqrt{\frac{d \log n}{n}})}$$

Ce cours

Apprentissage statistique

Machines à Vecteurs Supports Linéaires, classification

- Cas séparable
- Cas non séparable
- L'astuce des noyaux (kernel trick)

Souvenir du perceptron...

On pourrait demander plus...

Programmation linéaire

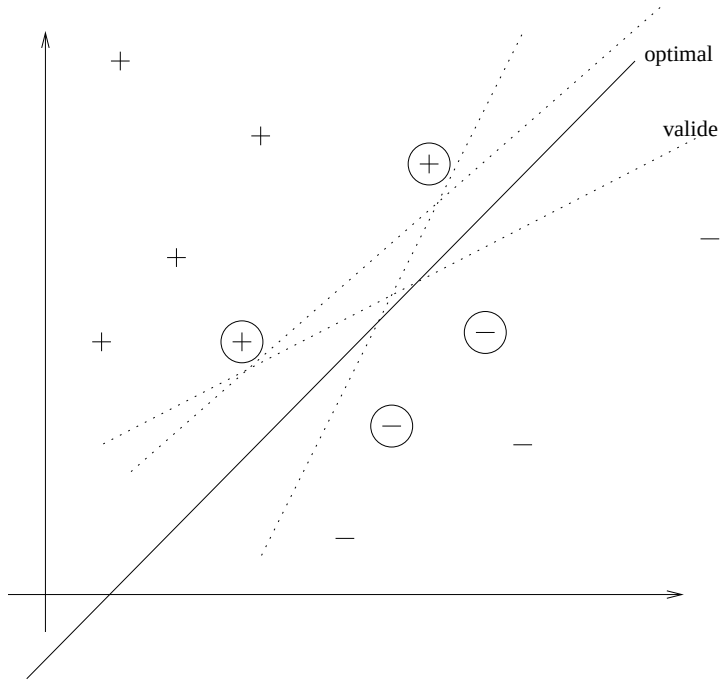
Trouver $\mathbf{w}, \delta$ tel que:

$$Max\ \delta, \quad subject\ to$$
$$\forall\, i = 1 \ldots n,\ \ y_i \left\langle \mathbf{w}, \mathbf{x}_i \right\rangle > \delta$$

Ici, se brancheront les machines à vecteurs supports...

Plusieurs hyperplans séparateurs

M. S.



Machines à vecteurs supports linéaires

Séparateurs linéaires

$$f(x) = \langle w, x \rangle + b$$

Région $\hat{y} = 1 : \ f(x) > 0$

Région $\hat{y} = -1 : \ f(x) < 0$

Critère

$$\forall i, \ y_i(\langle w, x_i \rangle + b) > 0$$

Remarque

Invariant par multiplication de w et b par une constante > 0

Forme canonique

Fixer l'échelle :

$$min_i \ \{y_i(\langle w, x_i \rangle + b)\} = 1$$

$\Leftrightarrow$

$$\forall i, \ y_i(\langle w, x_i \rangle + b) \geq 1$$

Maximisation de la marge

Critère

maximiser la distance min (points, hyperplan).

La bande la plus large possible

Marge

$$\langle w, x_+ \rangle + b = 1 \quad \langle w, x_- \rangle + b = -1$$

$$\langle w, x_+ - x_- \rangle = 2$$

Marge = projection de $x_+ - x_-$ sur le vecteur normal à l'hyperplan, $\frac{w}{||w||_2}$

$\Rightarrow$ On cherche à maximiser $\frac{1}{||w||}$

$\Leftrightarrow$ minimiser $||w||^2$

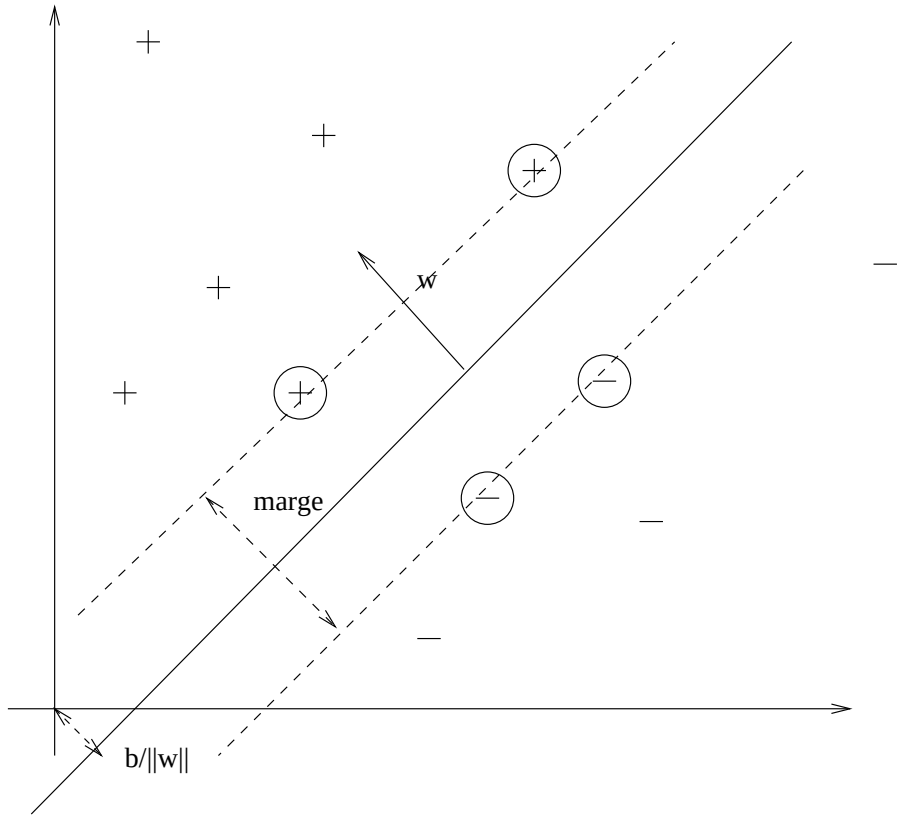
Maximisation de la marge

Problème

$$\begin{cases} \text{Minimiser} & \frac{1}{2}\ ||w||^2 \\ \text{sous la contrainte} & \forall\ i,\ y_i(\langle w, x_i\rangle + b) \geq 1 \end{cases}$$

Hyperplan de marge maximale

M. S.



Rappels optimisation

Optimisation de f sous contraintes $f_i \geq 0$ Si f et f_i convexes

introduire multiplicateurs de Lagrange α_i ($\alpha_i \geq 0$),
considérer

(termes de pénalisation des contraintes violées)

$$F(x, \alpha) = f(x) - \sum_i \alpha_i \, f_i(x)$$

Principe de Kuhn-Tucker (1951) A l'optimum (x_0, α^*)

$$F(x_0, \alpha^*) = min_{\alpha \geq 0} F(x_0, \alpha) = max_x F(x, \alpha^*)$$

Problème primal

$$L(w, b, \alpha) = \frac{1}{2}||w||^2 - \sum_i \alpha_i (y_i(\langle x_i, w \rangle + b) - 1), \ \alpha_i \geq 0$$

- Dérivons par rapport à b : à l'optimum,

$$\frac{\partial L}{\partial b} = 0 = \sum \alpha_i y_i$$

- Dérivons par rapport à w :

$$\frac{\partial L}{\partial w} = 0 = w - \sum \alpha_i y_i x_i$$

- Remplaçons dans $L(w, b, \alpha)$:

Problème dual (Wolfe)

$$\begin{cases} \text{Maximiser} & W(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \ < x_i, x_j > \\ \text{sous la contrainte} & \forall\ i,\ \alpha_i \geq 0 \\ & \sum_i \alpha_i y_i = 0 \end{cases}$$

Une forme quadratique des α optimisation quadratique

Solution: α_i^*

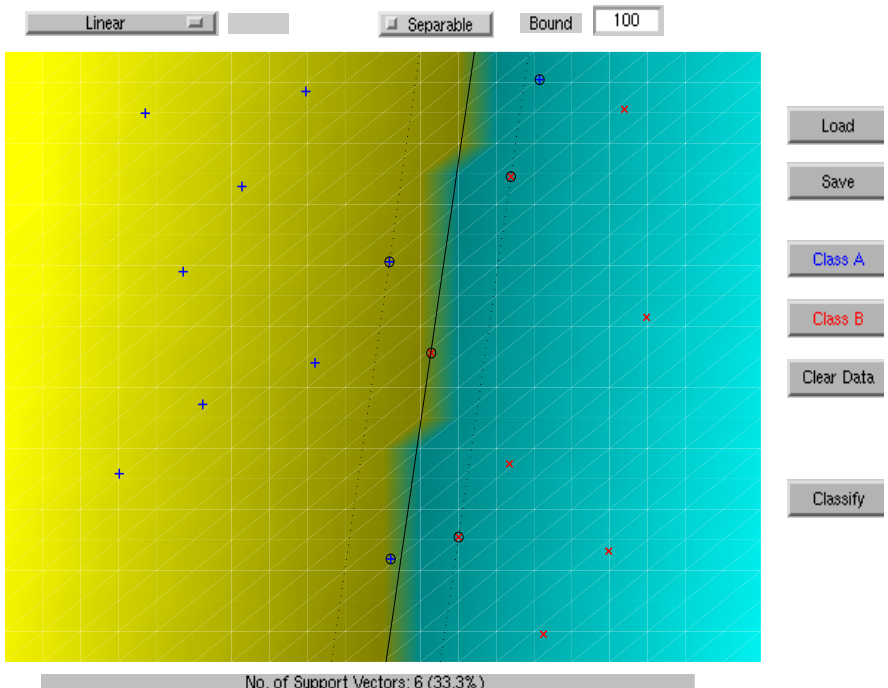
- On en tire w^* :

$$w^* = \sum_i \alpha_i^* y_i x_i$$

- Si $(\langle x_i, w^* \rangle + b) y_i > 1$, $\alpha_i^* = 0$.
- Si $(\langle x_i, w^* \rangle + b) y_i = 1$, $\alpha_i^* > 0$, x_i vecteur support
- On en tire b^* :

$$b^* = -\frac{1}{2}(\langle w^*, \bar{x}^+ \rangle + \langle w^*, \bar{x}^- \rangle)$$

S.


Linear
Separable
Bound
100
Load
Save
Class A
Class B
Clear Data
Classify
No. of Support Vectors: 6 (33.3%)

Ce cours

Apprentissage statistique

Machines à Vecteurs Supports Linéaires, classification

- Cas séparable
- Cas non séparable
- L'astuce des noyaux (kernel trick)

Cas $\mathcal{H}$ linéaire, données NON séparables

Les contraintes peuvent être violées:

- Introduire des termes de violation (variables ressort ξ_i)
- Les pénaliser:

Problème

$$\begin{cases} \text{Minimiser} & \frac{1}{2}\ ||w||^2 + \ C \sum_i \xi_i \\ \text{Sous la contrainte} & \forall i,\ y_i(\langle w, x_i \rangle + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \end{cases}$$

Idem : multiplicateurs de Lagrange α_i et β_i, avec $\alpha_i \geq 0,\ \beta_i \geq 0$

$$\begin{aligned} L(w, b, \xi, \alpha, \beta) = \quad & Min\ \tfrac{1}{2}||w||^2 + C \sum_i \xi_i \\ & - \sum_i \alpha_i (y_i(\langle w, x_i \rangle + b) - 1 + \xi_i) \\ & - \sum_i \beta_i \xi_i \end{aligned}$$

- On dérive par rapport à w:

$$w = \sum_i \alpha_i y_i x_i$$

- On dérive par rapport à b :

$$\sum_i \alpha_i y_i = 0$$

- On dérive par rapport à ξ_i :

$$C - \alpha_i - \beta_i = 0$$

Problème dual

$$Min \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j, \ 0 \leq \alpha_i \leq C$$

Délicat : fixer C = le coût d'une erreur.

Ce cours

Apprentissage statistique

Machines à Vecteurs Supports Linéaires, classification

- Cas séparable
- Cas non séparable
- L'astuce des noyaux (kernel trick)

Données non séparables

Remarques

- Les bornes de généralisation ne dépendent pas de X mais de $\mathcal{H}$.
- La résolution ne fait intervenir que des produits scalaires $< x_i, x_j >$.

Intuition: changer de représentation

$$\Phi : X \mapsto \Phi(X)$$

... et considérer le produit scalaire dans $\Phi(X)$ Espace des features.

L'astuce des noyaux

Principe

séparation complexe = séparation linéaire

dans un espace complexe.

Ex : Polynôme(x) = forme linéaire $(1, x, x^2, ...)$

$\Rightarrow$ Se ramener au problème linéaire avec

$$\mathcal{E} = \{(x_i, y_i)\} \rightarrow \mathcal{E} = \{(\Phi(x_i), y_i)\}$$

Ex :

$$\Phi : \mathbb{R} \mapsto \mathbb{R}^n$$
$$\Phi : x \mapsto \Phi(x) = (1, x, x^2, ..x^n)$$

Séparation :

$$\hat{y}(x) = \sum_i \alpha_i y_i \ < \Phi(x_i), \Phi(x) > \ + b$$

L'astuce des noyaux, suite

Inconvénient

Le calcul de Φ est cher...

Mais seul $< \Phi(x_i), \Phi(x_j) >$ intervient.

Définition du noyau

$$K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle$$

On ne passe plus dans l'espace des features

On ne borne à calculer le produit scalaire sur $\Phi(X)$

Alors :

$$h(x) = Sign(\sum_i \alpha_i y_i K(x_i, x) + b)$$

Résolution

Cas linéaire

$$\begin{cases} \text{Maximiser} & W(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \ < x_i, x_j > \\ \text{sous la contrainte} & \forall\ i,\ \alpha_i \geq 0 \\ & \sum_i \alpha_i y_i = 0 \end{cases}$$

Cas avec noyau

$$\begin{cases} \text{Maximiser} & W(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \ K(x_i, x_j) \\ \text{sous la contrainte} & \forall\ i,\ \alpha_i \geq 0 \\ & \sum_i \alpha_i y_i = 0 \end{cases}$$

Condition de Mercer (optimisation d'une forme définie positive)

$$\forall g \in L_2, \int K(x, x') g(x) g(x') dx \geq 0$$

Noyaux usuels

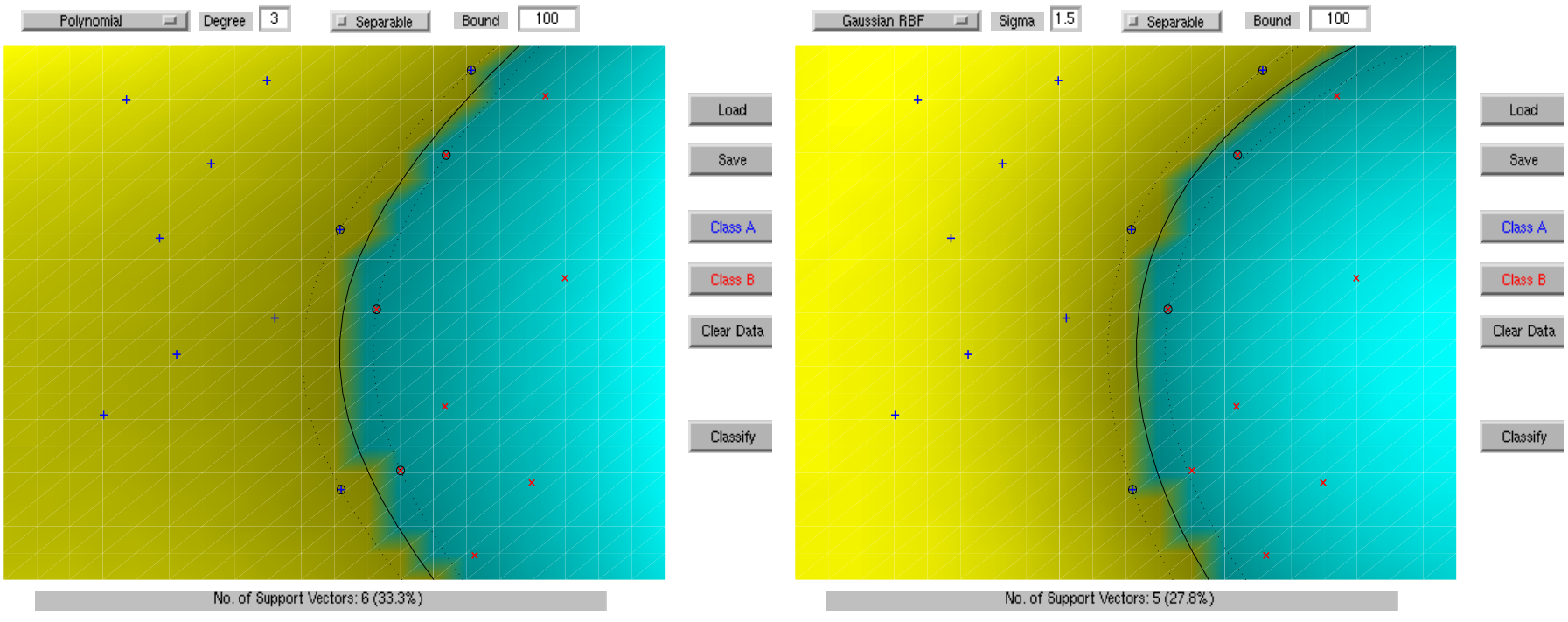
- Scalaire : $K(x, x') = xx'$
- Polynomial : $K(x, x') = (1 + xx')^p$
- RBF : $K(x, x') = e^{-\frac{||x-x'||}{2\sigma^2}}$

Radius Basis Function

- Sigmoide : $K(x, x') = tanh(cxx' - d)$

tangente hyperbolique

S.


Polynomial
Degree
3
Separable
Bound
100
Load
Save
Class A
Class B
Clear Data
Classify
No. of Support Vectors: 6 (33.3%)
Gaussian RBF
Sigma
1.5
Separable
Bound
100
Load
Save
Class A
Class B
Clear Data
Classify
No. of Support Vectors: 5 (27.8%)

Expérience des SVMs

- Très efficaces sur $\mathbb{R}^n$
- Paramètres parfois difficiles à régler (validation croisée)
- Difficile passage à l'échelle
- Données manquantes ?

Sources

- Vapnik, The nature of statistical learning, Springer Verlag 1995; Statistical Learning Theory, Wiley 1998
- Cristianini & Shawe Taylor, An introduction to Support Vector Machines, Cambridge University Press, 2000.
- http://www.kernel-machines.org/tutorials

Priorités

Représentation

Apprendre $\Leftrightarrow$ Apprendre à représenter

Sélection des exemples

Active learning / Plan d'expériences / Expert.

Optimisation

Apprendre $\Leftrightarrow$ Optimiser