Méthodes à noyaux

et

SVMs (Séparateurs à Vastes Marges)

Antoine Cornuéjols

Équipe *TAO* (INRIA/CNRS) - *Université de Paris-Sud, Orsay*

& *ENSIIE* (Evry)

antoine@lri.fr http://www.lri.fr/~antoine

Limsi [28-06-05]

Plan

Apprentissage inductif supervisé

À partir de l'*échantillon d'apprentissage* $S = \{(x_i, u_i)\}_{1,m}$

on cherche à identifier une <u>loi de dépendance</u> **sous-jacente**

- **Par exemple une fonction *h* aussi proche possible de *f* (fonction cible) tq :** $u_i = f(x_i)$
- **Ou bien de la distribution de probabilités** $P(x_i, u_i)$

afin de prédire l'avenir

Apprentissage inductif supervisé

$$\boxed{\begin{aligned} &P(\boldsymbol{x}, u) \\ u_i &= f(\boldsymbol{x}_i) \end{aligned}}$$

$$S_m = \{(\boldsymbol{x}_i, u_i)\}_{1 \leq i \leq m}$$

Échantillon d'apprentissage

$$\boxed{u = h(\boldsymbol{x})}$$

- ***Identification*** : ***h*** **« proche de »** ***f***
- ***Prédiction*** : ***h*** **« bonne règle de décision »**

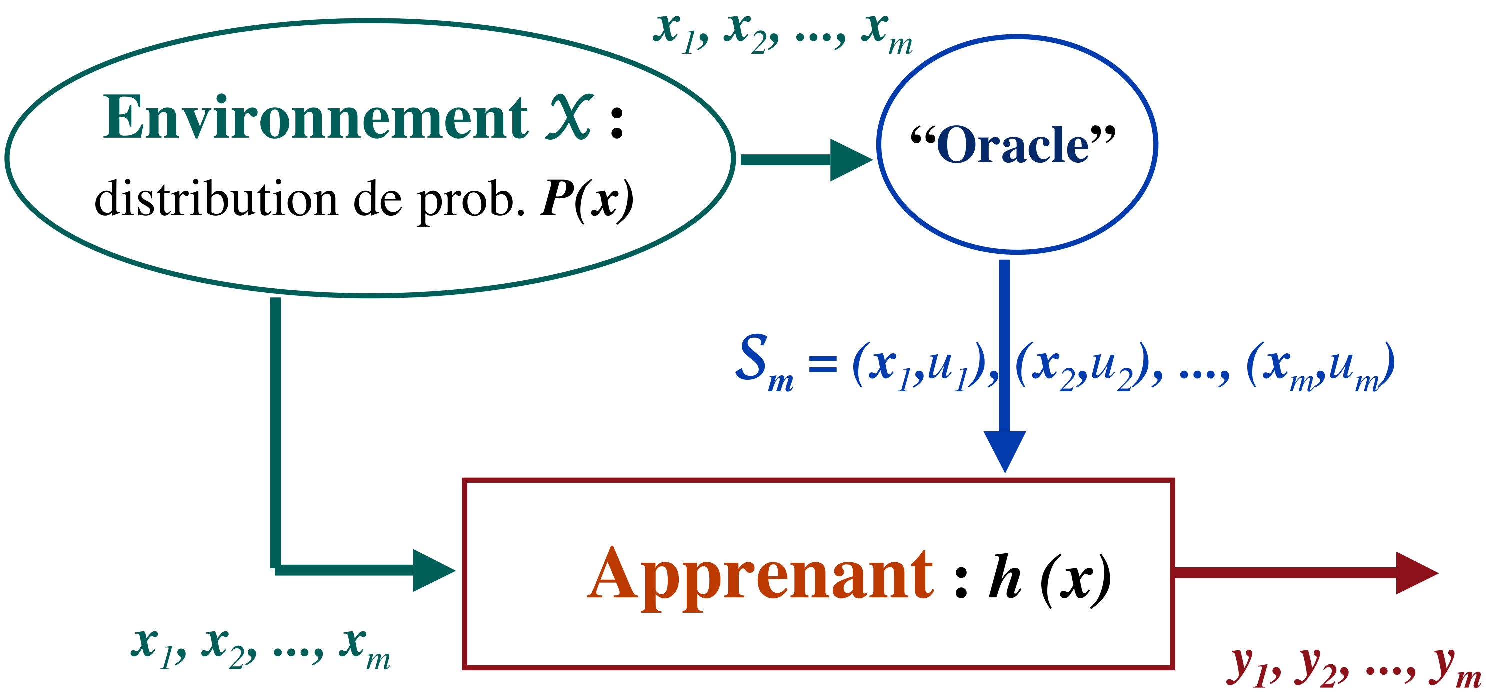
L'induction supervisée

- f : *fonction continue*
  - ➥ **Régression**
  - ➥ **Estimation de densité**

- f : *fonction discrète*
  - ➥ **Classification**

- f : *fonction binaire* (booléenne)
  - ➥ **Apprentissage de concept**

Cadre



- **Hypothèses fondamentales**
  - Données i.i.d.
  - Distribution P_{XxU} identique en apprentissage et après

Mesure de performance : le risque réel

Induction
Méthodes à noyaux
- Régression
- Fonctions noyax

Les SVMs
- Principe
- Problème associé
- Illustration

Mise en œuvre
- Validation
- Construction de noyaux

Bilan

Objectif : trouver une hypothèse $h \in \mathcal{H}$ minimisant ***le risque réel***
(*espérance de risque, erreur en généralisation*)

$$R(h) \;=\; \int_{X \times Y} l\left(\underline{h(\mathbf{x})}, \underline{u}\right)\; \underline{dP(\mathbf{x}, y)}$$

Fonction de perte

Étiquette prédite

Étiquette vraie (ou désirée)

Loi de probabilité jointe sur $\mathcal{X} \times \mathcal{Y}$

Le principe inductif ERM

$$R(h) = \int_{X \times Y} l\left(h(x), u\right) \, dP(x, y)$$

- On ne connaît pas la loi de probabilité $\boldsymbol{P}_{XxY}$.
- Le ***principe ERM*** (*minimisation du risque empirique*) prescrit de chercher l'hypothèse $h \in \mathcal{H}$ minimisant **le risque empirique**

$$\boxed{R_{\text{Emp}}(h) = \frac{1}{m} \sum_{i=1}^{m} l\left(h(\boldsymbol{x}_i), u_i\right)}$$

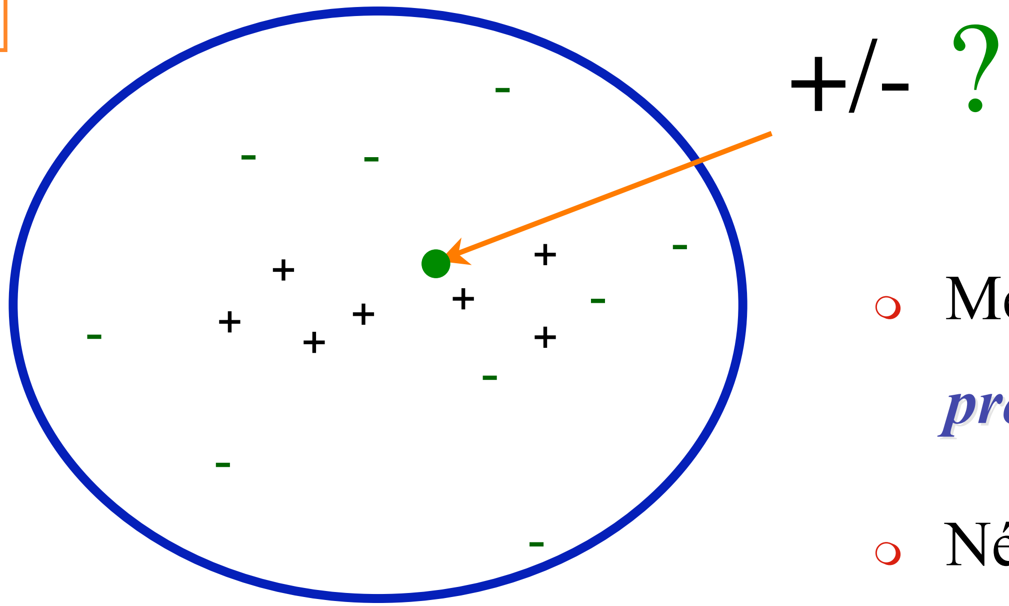
ERM régularisé

- **Pour éviter le surapprentissage (« overfitting »)**

$$\hat{h} = \underset{h \in \mathcal{H}}{\mathrm{ArgMin}} \left[R_{\mathrm{Emp}}(h) + \lambda \boxed{\Gamma(h)} \right]$$

$$\hat{h} = \underset{h \in \mathcal{H}}{\mathrm{ArgMin}} \left[R_{\mathrm{Emp}}(h) + \lambda \boxed{\Gamma(\mathcal{H})} \right]$$

Interpolation : par plus proches voisins

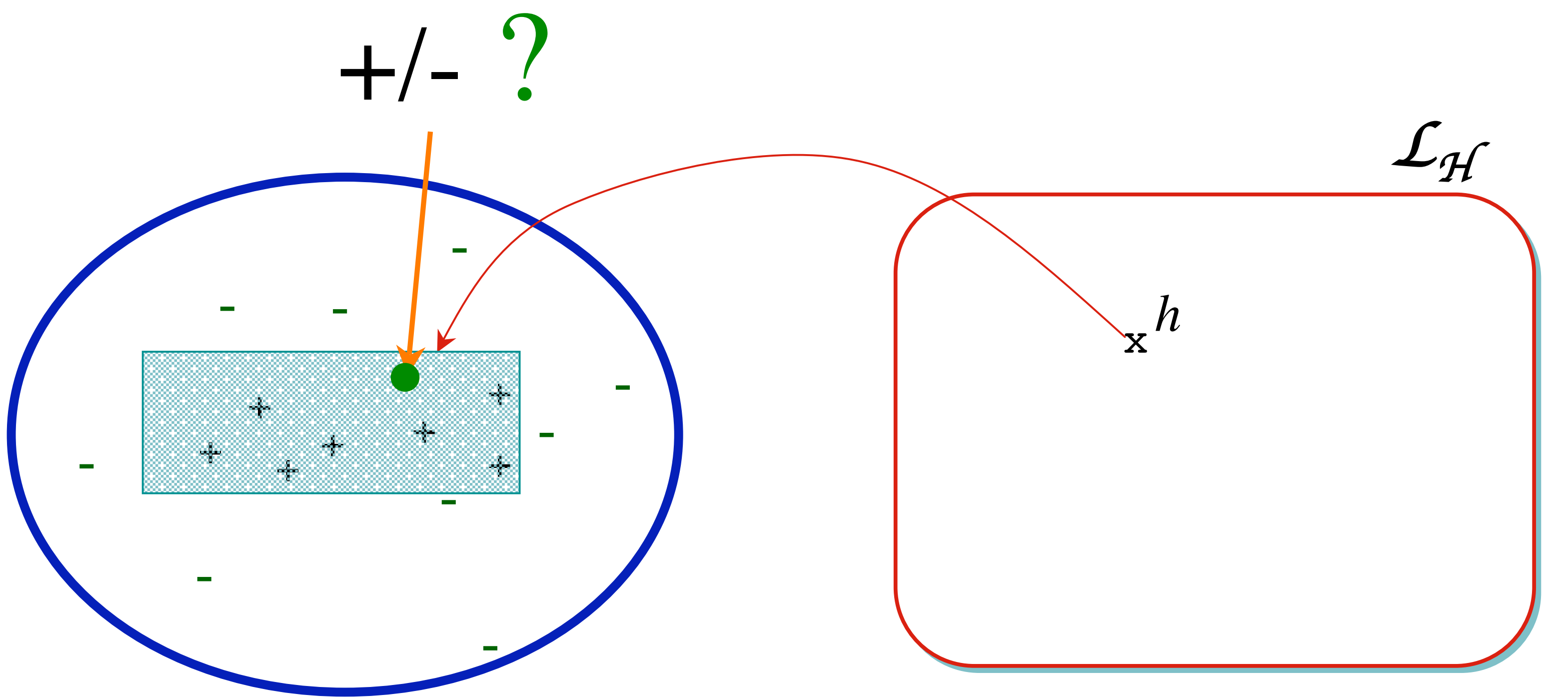


Espace des exemples : $\mathcal{X}$

- Méthodes par ***plus proches voisins***
- Nécessité d'une ***notion de distance***

➠ **Hypothèse de continuité dans $\mathcal{X}$**

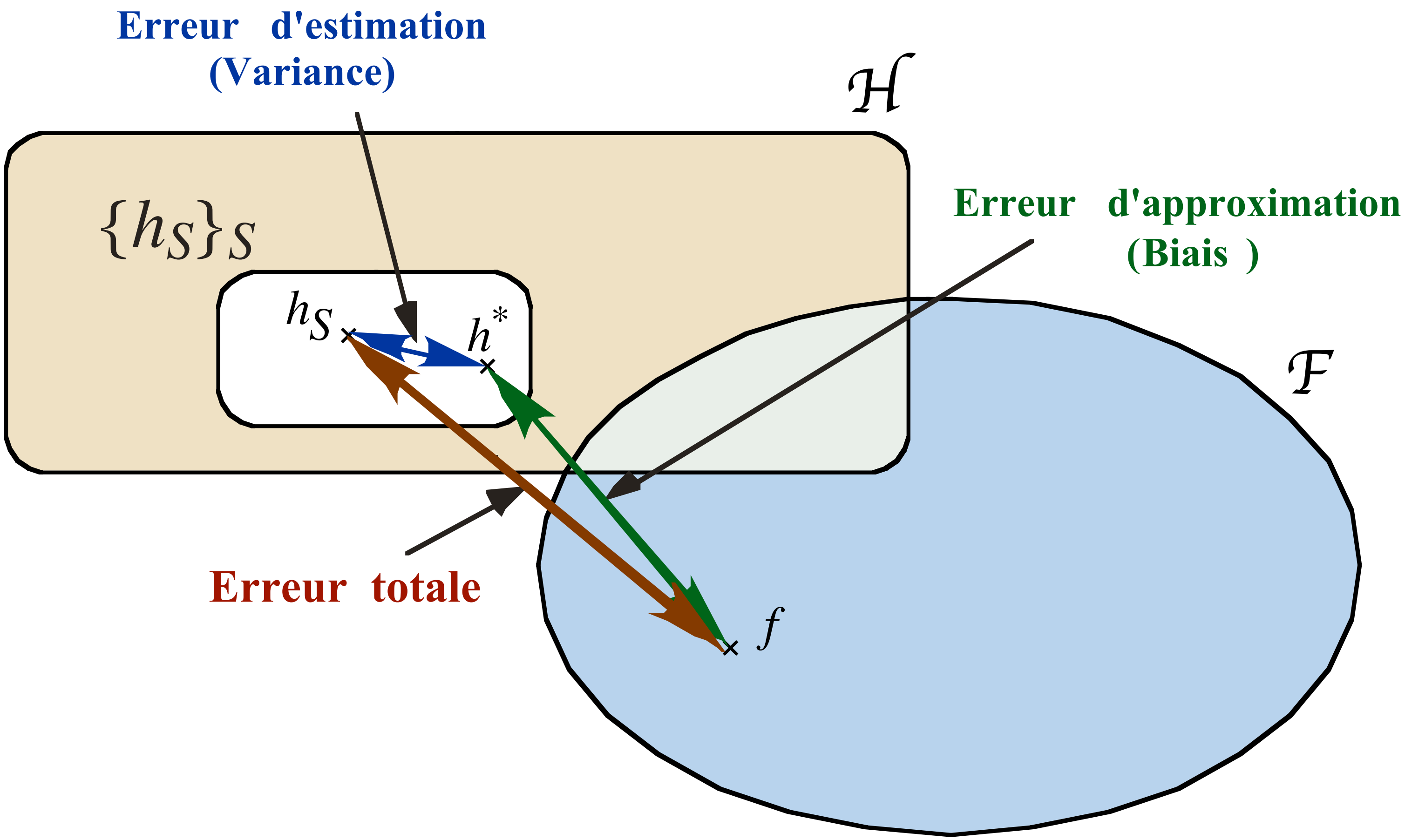
Interpolation : par fonction de décision



Espace des exemples : $\mathcal{X}$

Espace des hypothèses : $\mathcal{H}$

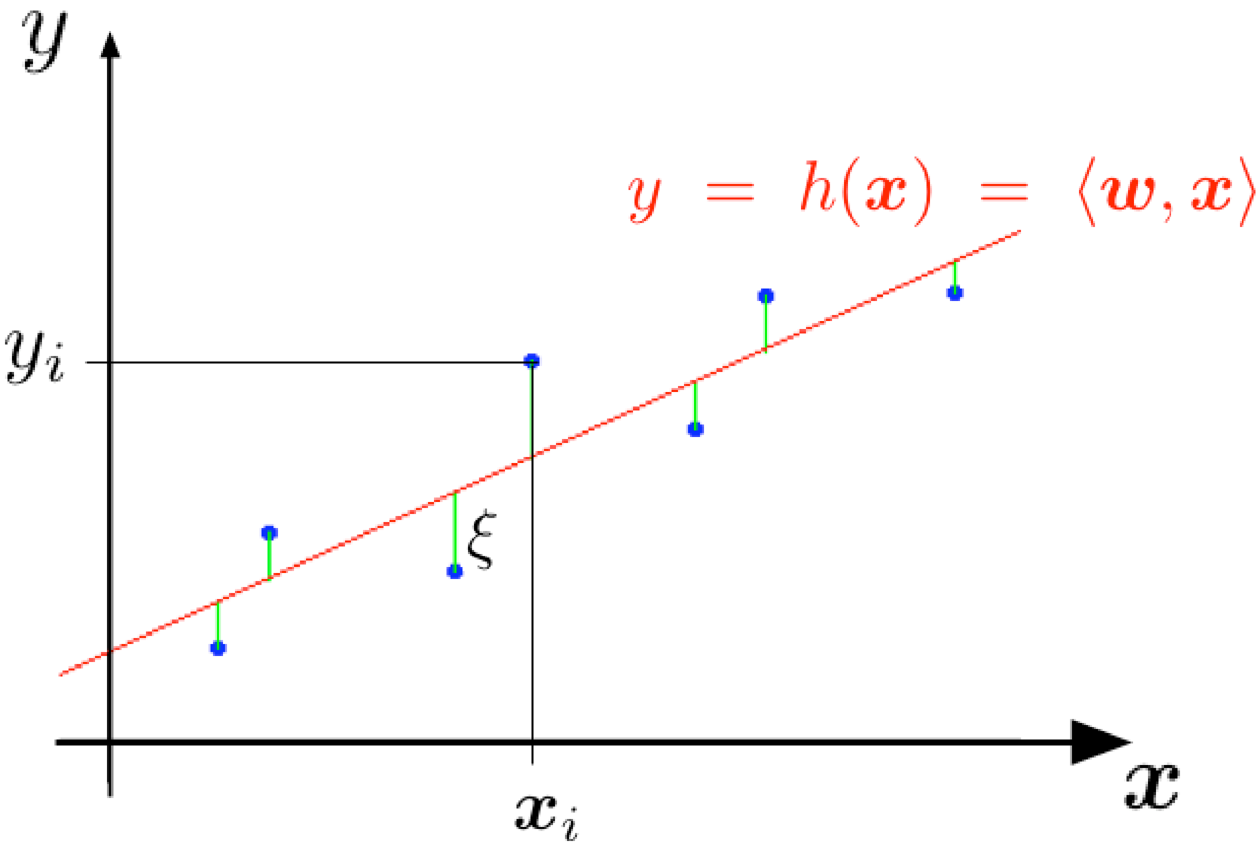
Le compromis biais-variance



Conditions pour l'induction

- **Conditions de généralisation**
  - Si espace d'hypothèses : Choix de $\mathcal{H}$
    - Contrôler la capacité de $\mathcal{H}$
  - Si directement dans $\mathcal{X}$ **(plus proches voisins)**
    - Choix de la distance
    - Choix des poids sur les voisins
- **Efficacité en gain d'information** (e.g. taille d'échantillon)
- Efficacité computationnelle

Exemple : régression linéaire



$$R_{\text{Emp}}(h) = R_{\text{Emp}}(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m} l(h(\boldsymbol{x}_i), u_i) = \frac{1}{m}\sum_{i=1}^{m} (y_i - h(\boldsymbol{x}_i))^2 = \frac{1}{m}\sum_{i=1}^{m} \xi_i^2$$

Approximation par moindres carrés

$$R_{\mathtt{Emp}}(\boldsymbol{w}) = ||\xi||_2^2 = (y - \boldsymbol{X}\boldsymbol{w})^\top (y - \boldsymbol{X}\boldsymbol{w})$$

$$\frac{\partial R_{\mathtt{Emp}}(\boldsymbol{w})}{\partial \boldsymbol{w}} = -2\boldsymbol{X}^\top y + 2\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{w}$$

$$\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{w} = \boldsymbol{X}^\top y$$

$$\boldsymbol{w} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top y$$

$$\boldsymbol{w} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top y = (\boldsymbol{X}^\top \boldsymbol{X})(\boldsymbol{X}^\top \boldsymbol{X})^{-2} \boldsymbol{X}^\top y = \sum_{i=1}^{m} \alpha_i \boldsymbol{x}_i$$

$$h(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle = \sum_{i=1}^{m} \alpha_i \langle \boldsymbol{x}_i, \boldsymbol{x} \rangle$$

Représentation duale

Ridge regression

- Si $\boldsymbol{X}^\top \boldsymbol{X}$ n'est pas inversible
  - Pas assez de données
  - Bruit
- **Problème mal-posé** → ***Régularisation***

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w} \in \mathcal{W}}{\mathrm{ArgMin}} \left[\frac{1}{m} \sum_{i=1}^{m} (y_i - \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle)^2 + \lambda \, ||\boldsymbol{w}||^2 \right]$$

$\lambda \geq 0$ (Coefficient de régularisation)

Solution pour la « ridge regression »

$$\hat{\boldsymbol{w}} = \underset{\boldsymbol{w}\in\mathcal{W}}{\text{ArgMin}}\left[\frac{1}{m}\sum_{i=1}^{m}(y_i - \langle \boldsymbol{w}, \boldsymbol{x}_i\rangle)^2 + \lambda\,||\boldsymbol{w}||^2\right]$$

$$\frac{\partial R_{\text{Emp}}(\boldsymbol{w})}{\partial \boldsymbol{w}} = \frac{1}{m}[-2\boldsymbol{X}^\top y + 2\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{w}] + 2\lambda\boldsymbol{w}$$

$$\lambda \leftarrow m\cdot\lambda$$

$$\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{w} + \lambda\boldsymbol{w} = (\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I}_d)\boldsymbol{w} = \boldsymbol{X}^\top y$$

$$\boxed{\boldsymbol{w} = (\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I}_d)^{-1}\boldsymbol{X}^\top y}$$

(Primale)

Système de d équations linéaires à d inconnues : $O(d^3)$

(Duale)

$$\boldsymbol{w} = \lambda^{-1}\boldsymbol{X}^\top(y - \boldsymbol{X}\boldsymbol{w}) = \boldsymbol{X}^\top\boldsymbol{\alpha}$$

$$\boxed{\boldsymbol{\alpha} = \lambda^{-1}(y - \boldsymbol{X}\boldsymbol{w}) = \lambda^{-1}(y - \boldsymbol{X}\boldsymbol{X}^\top\boldsymbol{\alpha}) = (\boldsymbol{X}\boldsymbol{X}^\top + \lambda\boldsymbol{I}_m)^{-1}y}$$

$$\boxed{h(\boldsymbol{x})} = \langle \boldsymbol{w}, \boldsymbol{x}\rangle = \left\langle \sum_{i=1}^{m}\alpha_i\boldsymbol{x}_i, \boldsymbol{x}\right\rangle = \sum_{i=1}^{m}\alpha_i\langle\boldsymbol{x}_i, \boldsymbol{x}\rangle = \boxed{\boldsymbol{y}^\top(\boldsymbol{G} + \lambda\boldsymbol{I}_m)^{-1}\langle\boldsymbol{x}_i, \boldsymbol{x}\rangle}$$

Formule duale

$$h(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i \langle \boldsymbol{x}_i, \boldsymbol{x} \rangle = \boldsymbol{y}^\top (\boldsymbol{G} + \lambda \boldsymbol{I}_m)^{-1} \langle \boldsymbol{x}_i, \boldsymbol{x} \rangle$$

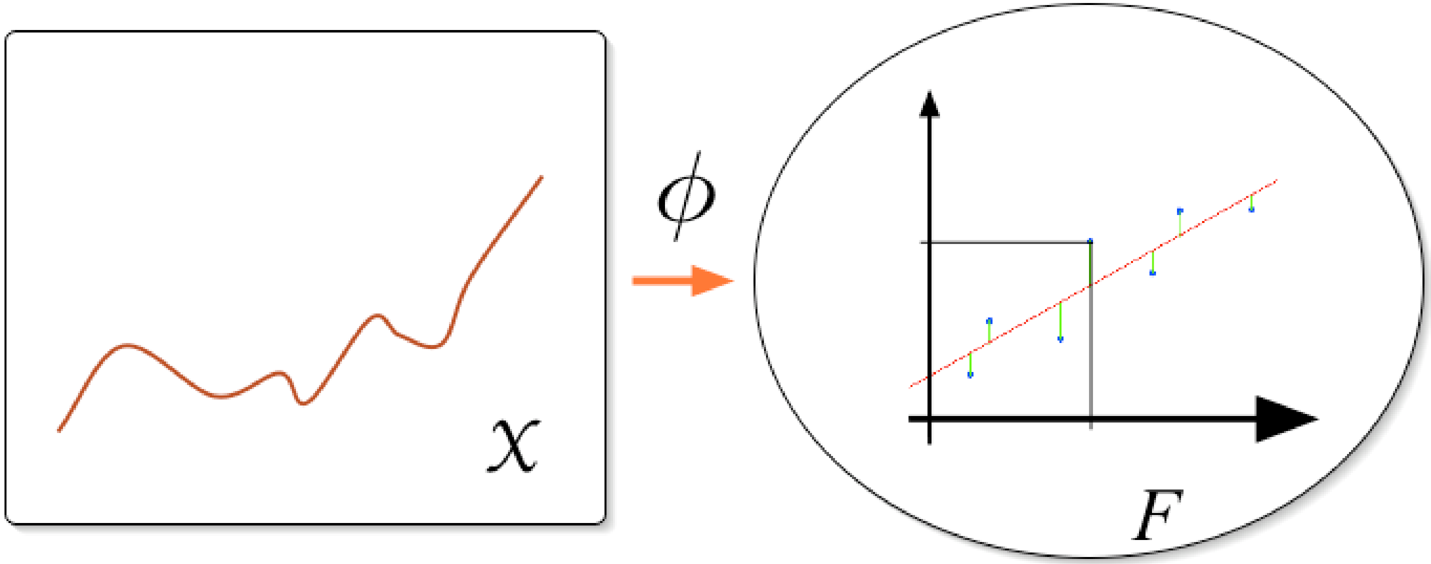
Provient directement des données

- L'information sur les exemples est **entièrement contenue dans les produits scalaires** (matrice de Gram $\boldsymbol{G}$ et les $<\boldsymbol{x}_i, \boldsymbol{x}>$)
- L'équation en $\boldsymbol{\alpha}$ requiert $O(m)$ opérations
- Le calcul de $h(\boldsymbol{x})$ requiert $O(m\ l)$ opérations

Régression non linéaire

- Idée : re-décrire les données dans un espace dans lequel la relation cherchée puisse avoir la forme d'une droite

$$\phi : \boldsymbol{x} \in \Re^d \mapsto \phi(\boldsymbol{x}) \in F \subseteq \Re^N$$



Régression non linéaire

- Idée : re-décrire les données dans un espace dans lequel la relation cherchée puisse avoir la forme d'une droite

$$\phi : \boldsymbol{x} \in \Re^d \mapsto \phi(\boldsymbol{x}) \in F \subseteq \Re^N$$

- Expression primale :

$$\boldsymbol{w} = (\phi(\boldsymbol{X})^\top \phi(\boldsymbol{X}) + \lambda \boldsymbol{I}_N)^{-1} \phi(\boldsymbol{X})^\top y \qquad O(N^3)$$

$$y = h(\boldsymbol{x}) = \langle \boldsymbol{w}, \phi(\boldsymbol{x}) \rangle \qquad O(N)$$

- Expression duale :

$$\boxed{h(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}) \rangle = \overbrace{\boldsymbol{y}^\top (\boldsymbol{G} + \lambda \boldsymbol{I}_m)^{-1}}^{O(m^3 + m^2 N)} \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}) \rangle} \qquad O(mN)$$

$$G_{ij} = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$$

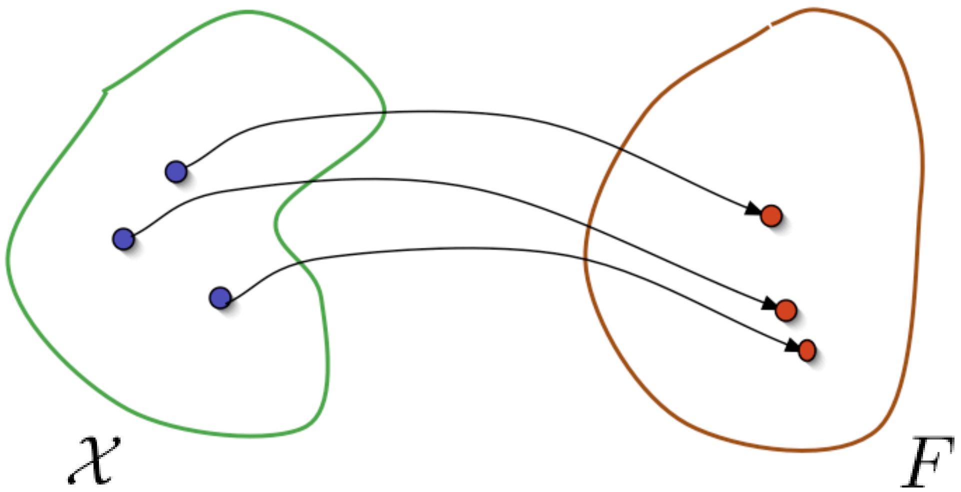
Les fonctions noyau (kernel functions)

- Fonction ***k*** telle que :

$$k : \mathcal{X} \times \mathcal{X} \rightarrow \Re$$
$$\forall \boldsymbol{x}, \boldsymbol{z} \in \mathcal{X} : \quad k(\boldsymbol{x}, \boldsymbol{z}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z}) \rangle$$

où : $\phi : \boldsymbol{x} \mapsto \phi(\boldsymbol{x}) \in F$

Espace de redescription muni d'un produit interne



Les fonctions noyau : exemple

Induction

Méthodes à noyaux

- Régression
- Fonctions noyax

Les SVMs

- Principe
- Problème associé
- Illustration

Mise en œuvre

- Validation
- Construction de noyaux

Bilan

Soit : $\mathcal{X} \subseteq \Re^2$

$$\phi : \quad \boldsymbol{x} = (x_1, x_2) \mapsto \phi(\boldsymbol{x}) = (x_1^2, x_2^2, \sqrt{2}\, x_1 x_2) \in F = \Re^3$$

$$h(\boldsymbol{x}) = w_{11}\, x_1^2 + w_{22}\, x_2^2 + w_{12}\, \sqrt{2}\, x_1 x_2$$

$$\begin{aligned} \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z}) \rangle &= \langle (x_1^2, x_2^2, \sqrt{2}\, x_1 x_2), (z_1^2, z_2^2, \sqrt{2}\, z_1 z_2) \rangle \\ &= x_1^2 z_1^2 + x_2^2 z_2^2 + 2 x_1 x_2 z_1 z_2 \\ &= (x_1 z_1 + x_2 z_2)^2 = \langle x, z \rangle^2 \end{aligned}$$

$$\boxed{k(\boldsymbol{x}, \boldsymbol{z}) = \langle \boldsymbol{x}, \boldsymbol{z} \rangle^2}$$

- Rq (non unicité de l'espace *F* défini par *Φ*) :

$$\phi : \quad \boldsymbol{x} = (x_1, x_2) \mapsto \phi(\boldsymbol{x}) = (x_1^2, x_2^2, x_1 x_2, x_2 x_1) \in F = \Re^4$$

Les fonctions noyau

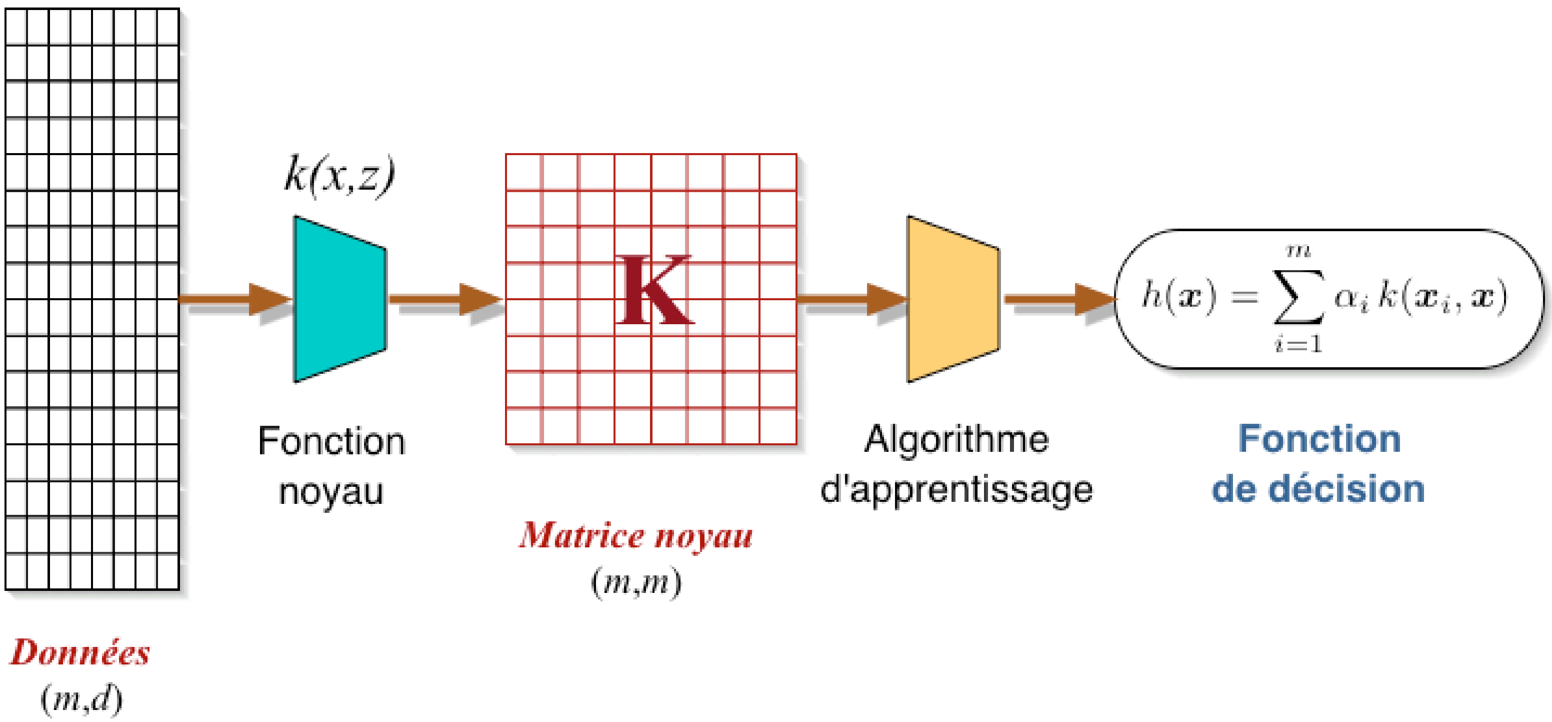
- **Efficacité computationnelle :**

$$\boxed{h(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}) \rangle = \boldsymbol{y}^\top \overbrace{(\boldsymbol{G} + \lambda \boldsymbol{I}_m)^{-1}}^{O(m^3 + m^2 d)} k(\boldsymbol{x}_i, \boldsymbol{x})} \quad O(m\,d)$$

$$G_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

Les méthodes à noyau

- **Modularité**
  - Découplage entre
    - Les **algorithmes** (linéaires)
    - La **description des données**



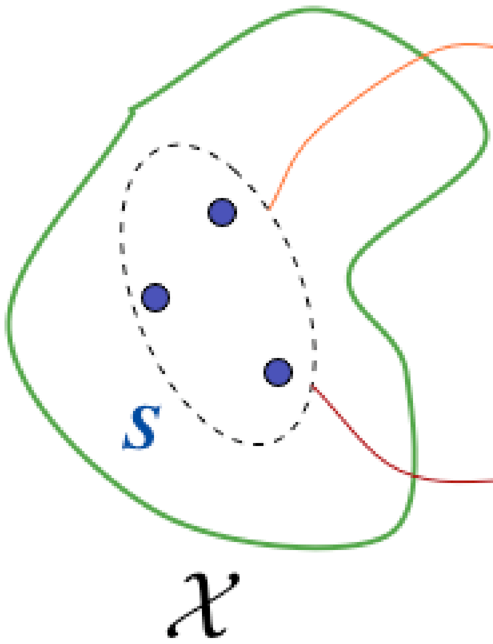
Leçons (provisoires)

L'emploi de fonctions noyau permet :

- D'utiliser les algorithmes de recherche de régularités linéaires pour la ***recherche de régularités non linéaires***
- D'employer ces algorithmes même sur des ***données non vectorielles*** (du moment que l'on sait trouver une fonction noyau adéquate)
- De redécrire implicitement les données dans des ***espaces de grande dimension*** sans en avoir le coût computationnel

Les méthodes à noyaux

Tout passe par les produits internes dans F !!!



S = {aatcgagtcac, atggactct, tgcactact}

$$K = \begin{bmatrix} 1 & 0.5 & 0.3 \\ 0.5 & 1 & 0.6 \\ 0.3 & 0.6 & 1 \end{bmatrix}$$

Philosophie de représentation des données radicalement différente

Conséquences d'une représentation par noyau

- **Des informations sont perdues**
  - ***Orientation*** (invariance de la matrice K par rotation)
  - ***Alignement*** des données avec les axes (idem)

Les fonctions noyau : définition

- **Fonction noyau ⟺ positive définie**
  - Symétrique : $k(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{z}, \boldsymbol{x})$
  - Positive définie :

$$\forall n > 0, \forall \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}, \forall c_1, \ldots, c_n \in \Re : \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$$

- **Théorème de Mercer**
  - *Toute fonction positive définie peut être exprimée comme un produit interne dans un espace de description*

Fonctions noyau et similarité

- $k(x, z)$ grand $\Leftrightarrow$ x ***similaire*** à z

- Évident pour le noyau gaussien : $k(\boldsymbol{x}, \boldsymbol{z}) = \exp(-\frac{d(\boldsymbol{x}, \boldsymbol{z})^2}{2\,\sigma^2})$

- Plus généralement :

$$k(\boldsymbol{x}, \boldsymbol{z}) = \frac{||\phi(\boldsymbol{x})||^2 + ||\phi(\boldsymbol{z})||^2 - d(\phi(\boldsymbol{x}), \phi(\boldsymbol{z}))^2}{2}$$

Si tous les points sont de même « longueur » dans F,
($||\phi(\boldsymbol{x})||^2 = k(\boldsymbol{x}, \boldsymbol{x}) = \texttt{Const.} \quad , \forall \boldsymbol{x} \in \mathcal{X}$),
alors le noyau est une fonction décroissante de d.

Inversement : $d(\boldsymbol{x}, \boldsymbol{z}) = \sqrt{k(\boldsymbol{x}, \boldsymbol{x}) - 2\,k(\boldsymbol{x}, \boldsymbol{z}) + k(\boldsymbol{z}, \boldsymbol{z})}$

Fonctions noyau pour des vecteurs

- **Noyaux polynomiaux**

$$k_{\texttt{poly1}}(\boldsymbol{x}, \boldsymbol{z}) = (\boldsymbol{x}^\top \boldsymbol{z})^d$$

Tous les produits d'exactement ***d*** *variables*

$$k_{\texttt{poly2}}(\boldsymbol{x}, \boldsymbol{z}) = (\boldsymbol{x}^\top \boldsymbol{z} + c)^d$$

Tous les produits d'au plus ***d*** *variables*

- **Noyaux gaussiens**

$$k_G(\boldsymbol{x}, \boldsymbol{z}) \;=\; \exp\left(-\frac{d(\boldsymbol{x}, \boldsymbol{z})^2}{2\,\sigma^2}\right)$$

Sorte de décomposition en série de Fourrier

- **Noyaux sigmoïdes**

$$k(\boldsymbol{x}, \boldsymbol{z}) \;=\; \tanh(\kappa\, \boldsymbol{x}^\top \boldsymbol{z} + \theta)$$

Pas définie positive.
Mais fonction de décision proche des réseaux connexionnistes

Les SVMs (Séparateurs à Vastes Marges)

- **Tâche de discrimination** (entre deux classes)
  - **Cas de la** séparation linéaire

- On cherche h sous forme d'une fonction linéaire : $h(\boldsymbol{x}) = w.\,\boldsymbol{x} + b$

- La ***surface de séparation*** est donc l'hyperplan :

$$\boxed{w.\boldsymbol{x} + b = 0}$$

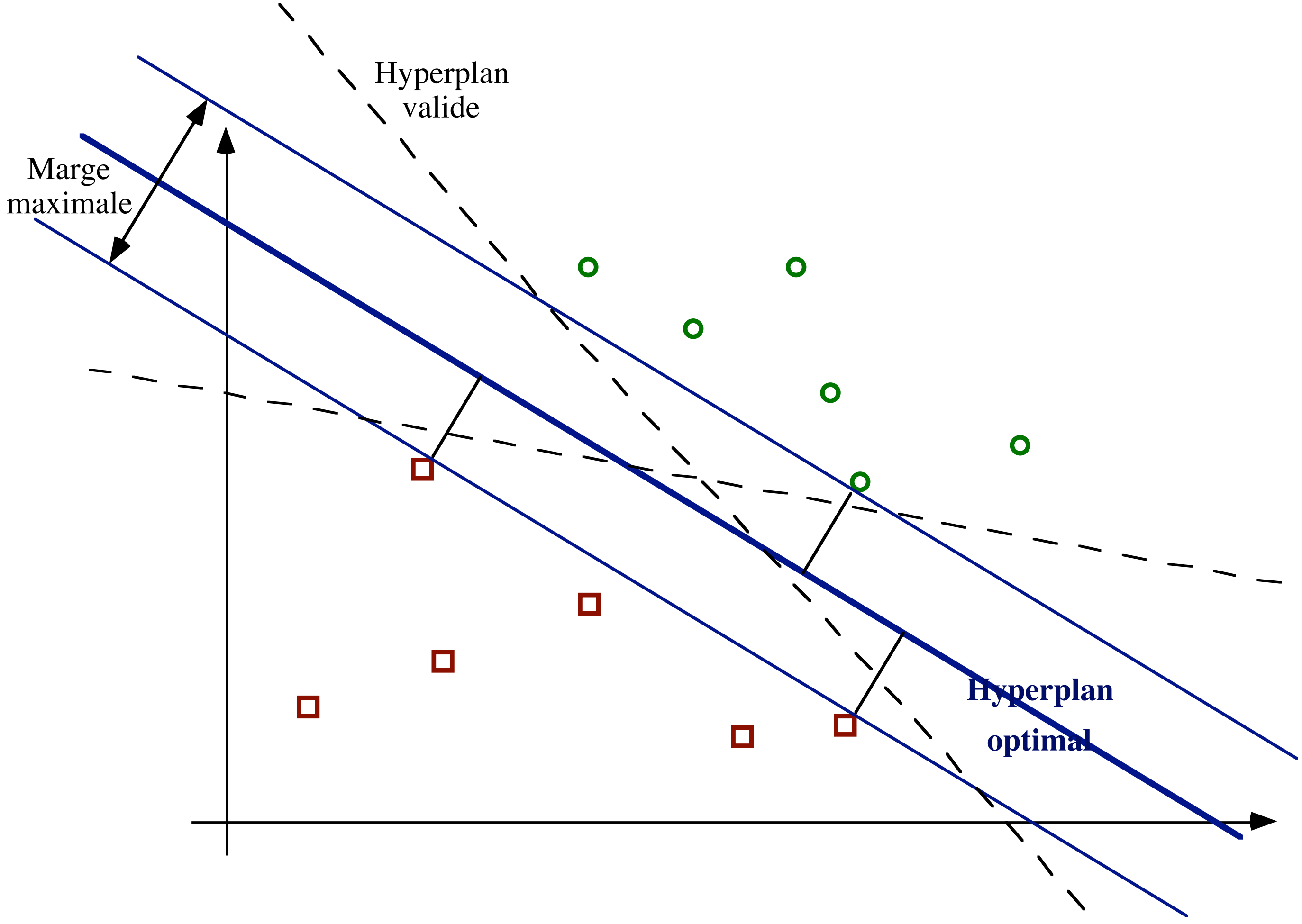
- Elle est valide si $\quad \forall i \quad u_i\, h(\boldsymbol{x}_i) \geq 0$

- L'hyperplan est dit sous forme canonique lorsque $\quad \min_i |w.\boldsymbol{x} + b| = 1$

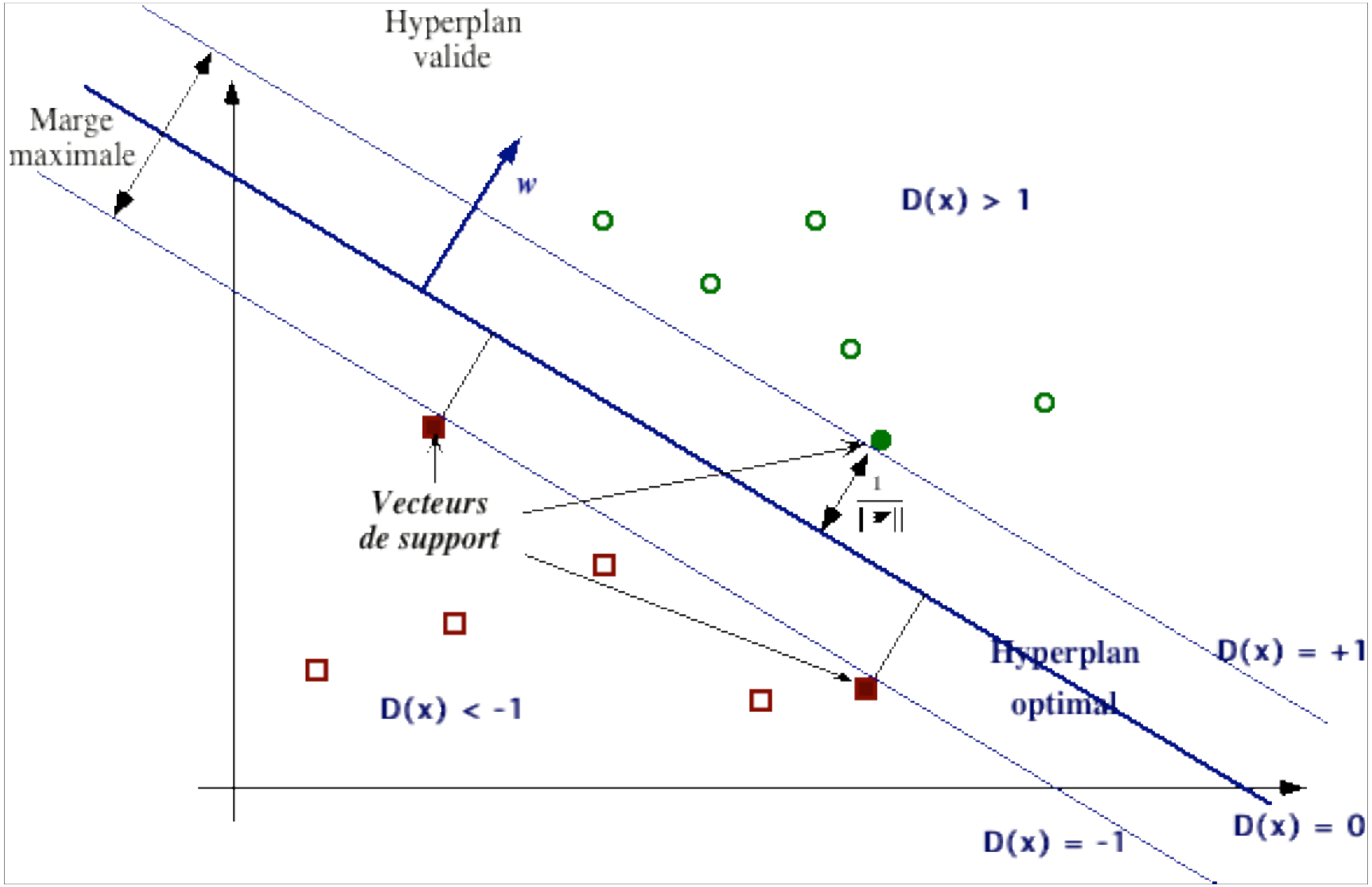
ou encore

$$\forall i \quad u_i\,(w.\boldsymbol{x}_i + b) \geq 1$$

Hyperplan de plus vaste marge



Optimisation de la marge



Optimisation de la marge

- La distance d'un point à l'hyperplan est : $d(x) = \dfrac{|w.x + w_0|}{\|w\|}$

- L'hyperplan optimal est celui pour lequel la distance aux points les plus proches (***marge***) est maximale. Cette distance vaut $\dfrac{2}{\|w\|}$

- Maximiser la marge revient donc à minimiser $\|w\|$ sous contraintes:

$$\begin{cases} \min \frac{1}{2}\|w\|^2 \\ \forall i \quad u_i(w.x_i + w_0) \geq 1 \end{cases}$$

EXPRESSION PRIMALE

Remarques sur la justification de ce critère inductif

- **Intuitivement satisfaisant**
  - Si il y a du bruit dans les données, le séparateur à marge maximale sera plus robuste

- **Risque empirique régularisé**
  - Satisfaire les données : $u_i \left[(\boldsymbol{w} \cdot \boldsymbol{x_i}) + w_0 \right] \geq 1 , \qquad i = 1, \ldots, n$
  - Régulariser : $\mathrm{Min} \left(\eta(\boldsymbol{w}) = \frac{1}{2} \|\boldsymbol{w}\|^2 \right)$

Transformation du problème d'optimisation

- **Méthode des multiplicateurs de Lagrange**

$$\begin{cases} L(\boldsymbol{w}, w_0, \boldsymbol{\alpha}) = \frac{1}{2}||\boldsymbol{w}||^2 - \sum_{i=1}^{m} \alpha_i [(\boldsymbol{x}_i \cdot \boldsymbol{w} + w_0)\, u_i - 1] \\ \alpha_i \geq 0 \;\; (\forall i) \end{cases}$$

EXPRESSION DUALE

$$\begin{cases} \max\limits_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j u_i u_j \, k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ \sum_{i,j=1}^{m} \alpha_i u_i = 0 \\ \alpha_i \geq 0 \;\; (\forall i) \end{cases}$$

Justification impliquant la fonction noyau

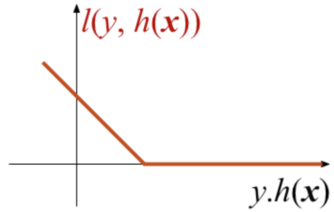
- Norme du vecteur de poids

$$
\begin{aligned}
\|\boldsymbol{w}\|^2 &= \langle \boldsymbol{w}, \boldsymbol{w} \rangle = \langle \sum_{i=1}^{m} \alpha_i \phi(\boldsymbol{x}_i), \sum_{i=1}^{m} \alpha_j \phi(\boldsymbol{x}_j) \rangle \\
&= \sum_{i,j=1}^{m} \alpha_i \alpha_j \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle = \sum_{i,j=1}^{m} \alpha_i \alpha_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\
&= \boldsymbol{\alpha}^\top \boldsymbol{K} \boldsymbol{\alpha}
\end{aligned}
$$

- Espace d'hypothèses de norme bornée $\mathcal{F}_B$

$$\|\boldsymbol{w}\|^2 = \boldsymbol{\alpha}^\top \boldsymbol{K} \boldsymbol{\alpha} \leq B^2$$

- Fonction de perte (hinge loss) $\big(1 - y\, h(\boldsymbol{x})\big)_+$



- Alors : $\boxed{\hat{R}_m(\mathcal{F}_B) \leq \frac{2\,B}{m} \sqrt{\mathrm{tr}(\boldsymbol{K})}}$

Complexité de Rademacher de $\mathcal{F}_B$

Avec *prob* ≥ 1-δ

$$\boxed{R_{\mathtt{R\acute{e}el}}(h) \leq R_{\mathtt{Emp}}(h) + \hat{R}_m(\mathcal{F}_B) + \sqrt{\frac{\ln(2/\delta)}{2\,m}}}$$

Morale

- **Les données s'expriment à travers la matrice noyau**
- **La matrice noyau contrôle la régularisation du risque**

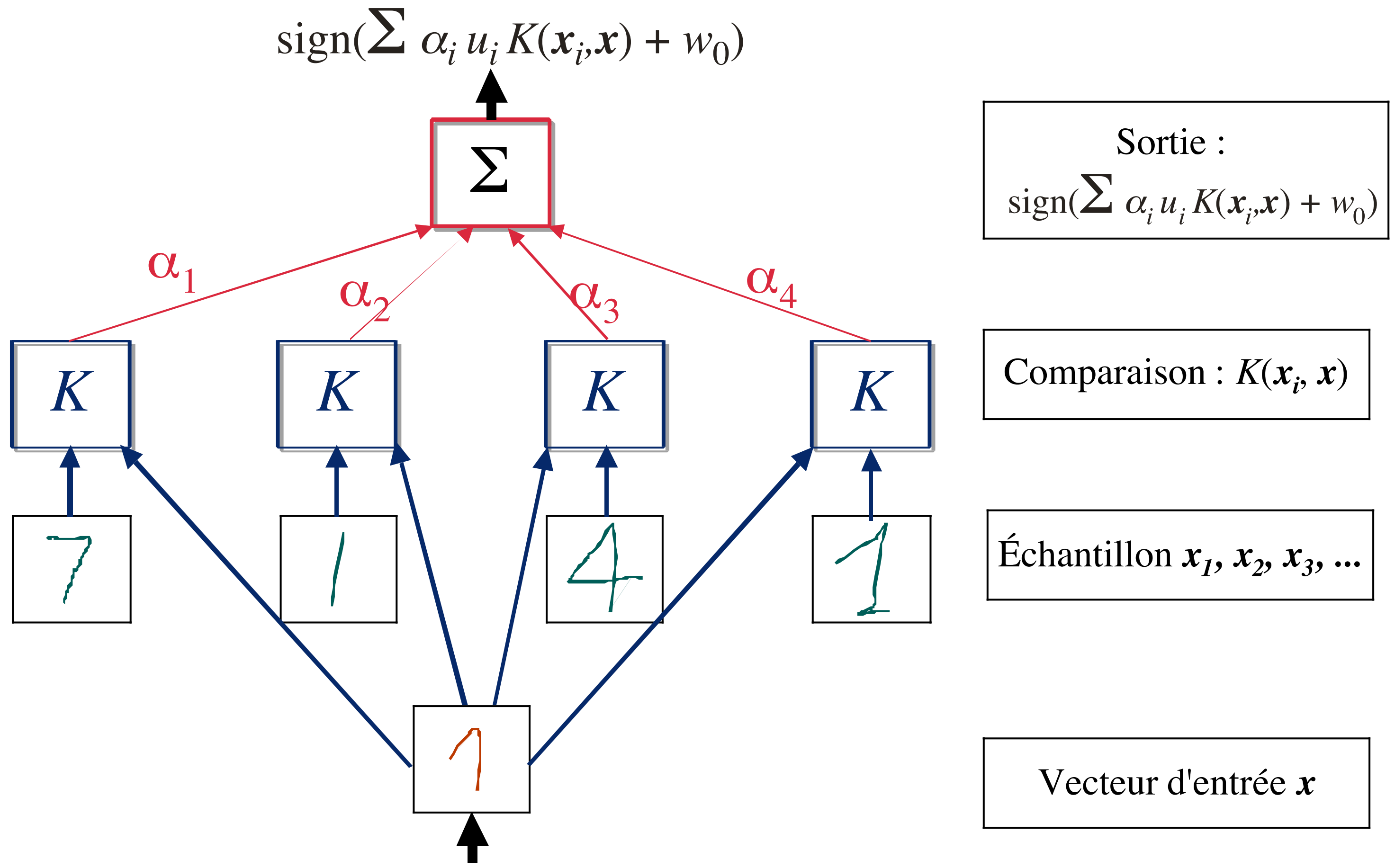
Solution du problème d'optimisation dual

- Dans la forme duale :

$$h(\boldsymbol{x}) = \mathtt{sign}\left(\sum_{i=1}^{m_S} \alpha_i\, u_i\, k(\boldsymbol{x}_i, \boldsymbol{x}) + w_0\right)$$

m_S : nb de points de support

Schéma de fonctionnement des SVMs



Cas du problème non séparable : marges douces

- On introduit des variables “ressort” qui pénalisent l’erreur commise :

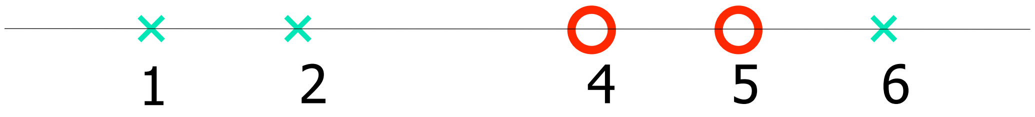
$$\begin{cases} \min \frac{1}{2} \| \mathbf{w} \|^2 + \boxed{C \sum_{i=1}^{l} \xi_i} \\ \forall i \quad \mathbf{u}_i (\mathbf{w} . \mathbf{x}_i + w_0) \geq 1 \boxed{- \xi_i} \end{cases}$$

- **Le problème dual a la même forme à l’exception d’une constante *C***

$$\begin{cases} \max\limits_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j u_i u_j \, k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ \sum_{i,j=1}^{m} \alpha_i u_i = 0 \\ 0 \leq \alpha_i \leq \boxed{C} \quad (\forall i) \end{cases}$$

Illustration

- Soient 5 points sur la droite :
  $\{(x_1=1, u_1=1), (x_2=2, u_2=2), (x_3=4, u_3=-1), (x_4=5, u_4=-1), (x_5=6, u_5=1)\}$



- Utilisation d'un noyau polynomial de degré 2
  - $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i\, \boldsymbol{x}_j + 1)^2$
  - $C = 100$

- **Recherche de α_i par :**

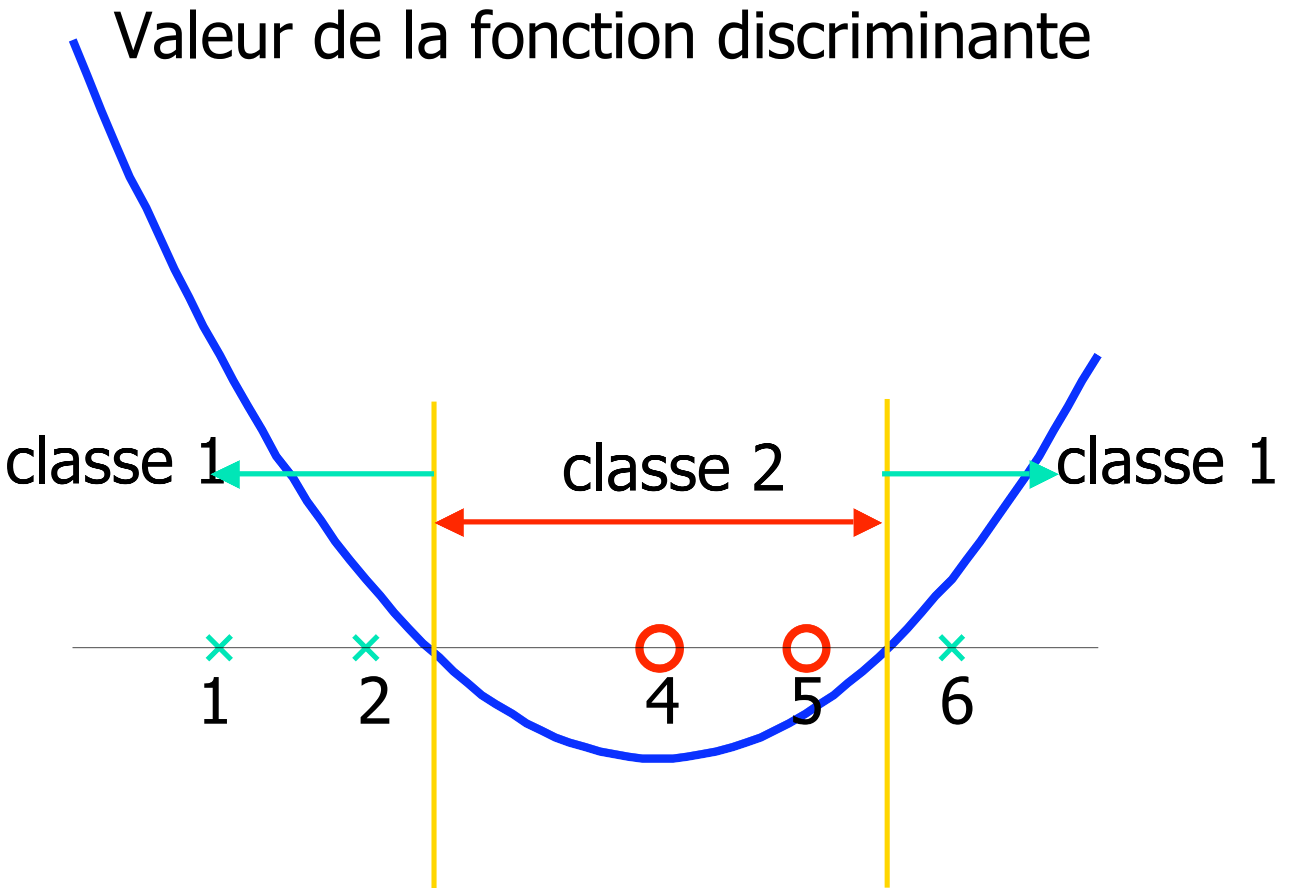
$$\begin{cases} \max\limits_{\alpha} \sum_{i=1}^{5} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{5} \alpha_i \alpha_j u_i u_j (\boldsymbol{x}_i \cdot \boldsymbol{x}_j + 1)^2 \\ \sum_{i,j=1}^{5} \alpha_i u_i = 0 \\ 0 \leq \alpha_i \leq 100 \quad (\forall i) \end{cases}$$

Illustration

- **Utilisation d'un programme de résolution de problème quadratique**
  - α_1=0, α_2=2.5, α_3=0, α_4=7.333, α_5=4.833
  - Les points de supports sont : { x_2=2, x_4= 5, x_5= 6}
- La ***fonction de décision*** est :
  - $h(x) = (2.5)(1)(2x+1)^2 + 7.333(1)(5x+1)^2 + 4.833(1)(6x+1)^2+b$
    $= 0.6667\ x^2 - 5.333\ x + b$
  - Avec b obtenue par $h(2)$=1 ou par $h(5)$=-1 ou par $h(6)$=1, puisque x_2, x_4 et x_5 sont sur la droite $u_i(w^T\Phi(x)+b)=1$
    ce qui donne b=9
- **D'où :**

$$\boxed{h(x) = 0.6667\ x^2 - 5.333\ x + 9}$$

Illustration



$\{x=2, x=5, x=6\}$ sont points supports

La mise en pratique

- **Il faut choisir :**
  - Le **type de fonction noyau *k***
    - **Sa forme**
    - **Ses paramètres**
  - La **valeur de la constante *C***

- La sélection de ces paramètres requiert l'utilisation de **méthodes empiriques pour faire le meilleur choix (validation croisée)**

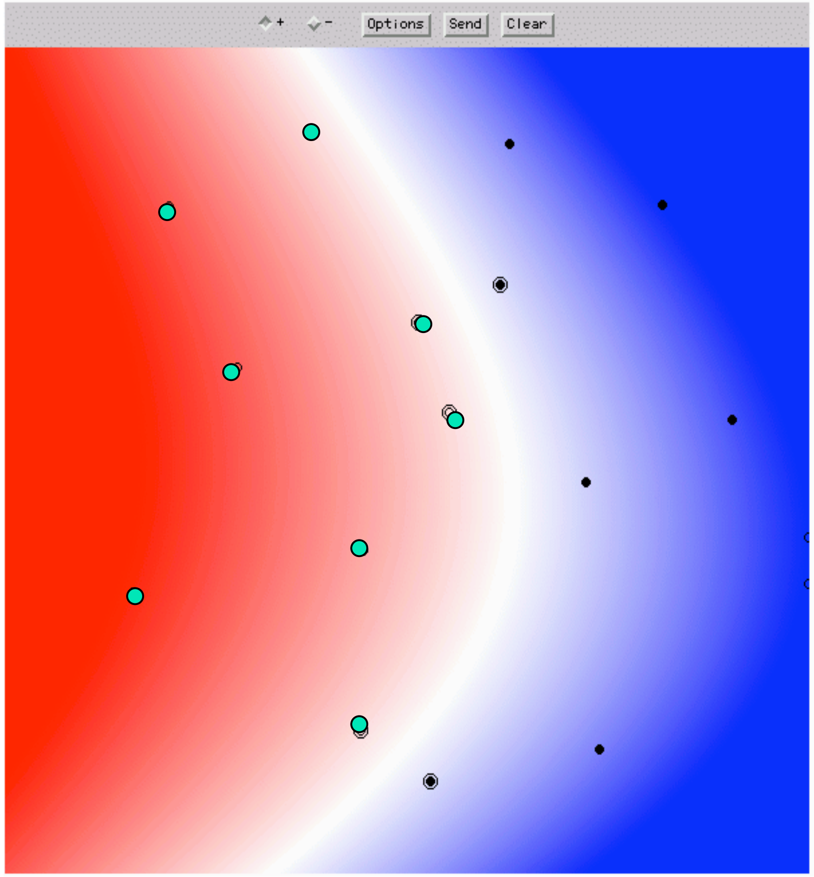
Exemple

● : exemple +

● : exemple -

Dans cercle : points de support

Fct noyau polynomiale de degré 3



Démo :

http://svm.research.bell-labs.com/

http://svm.dcs.rhbnc.ac.uk/pagesnew/GPat.shtml

Les données d'apprentissage

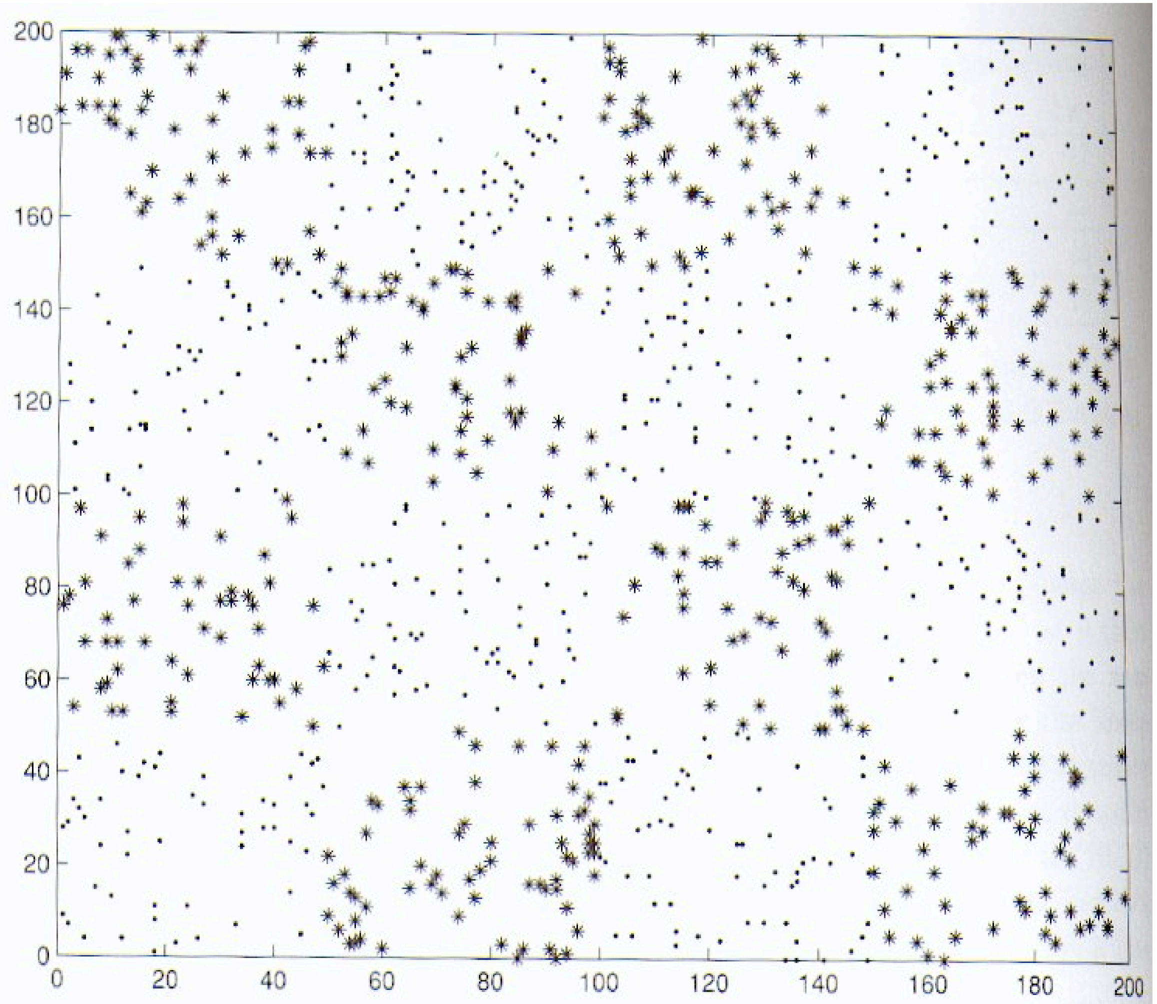


Figure 10.2 Training points for checkerboard pattern

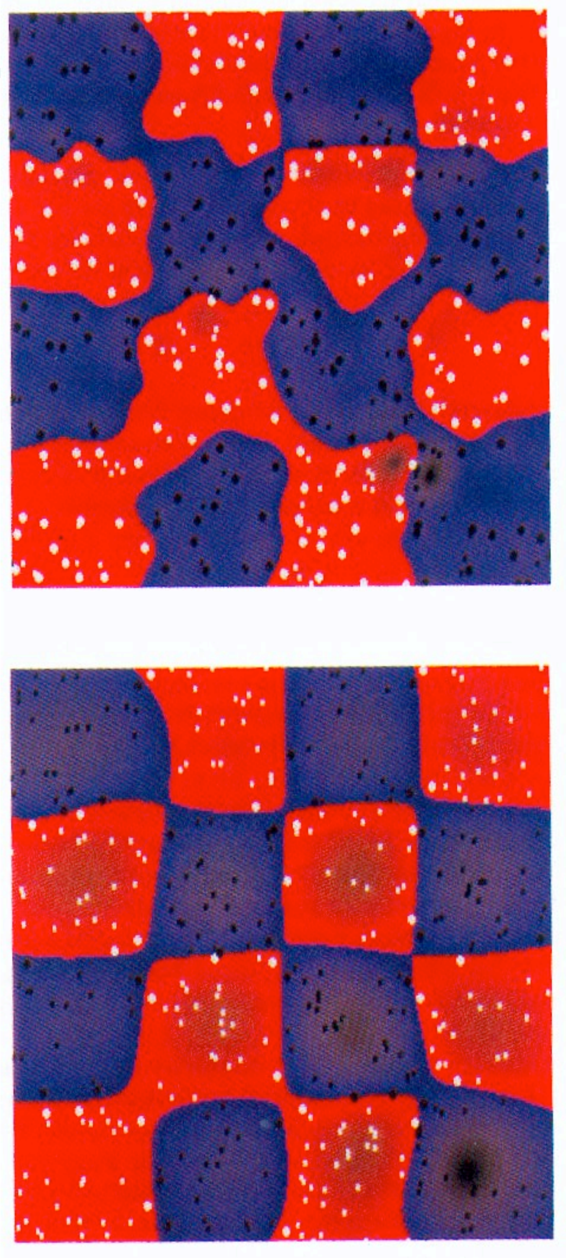
Effet des paramètres de contrôle

Induction
Méthodes à noyaux
- Régression
- Fonctions noyax
Les SVMs
- Principe
- Problème associé
- Illustration
Mise en œuvre
- Validation
- Construction de noyaux
Bilan

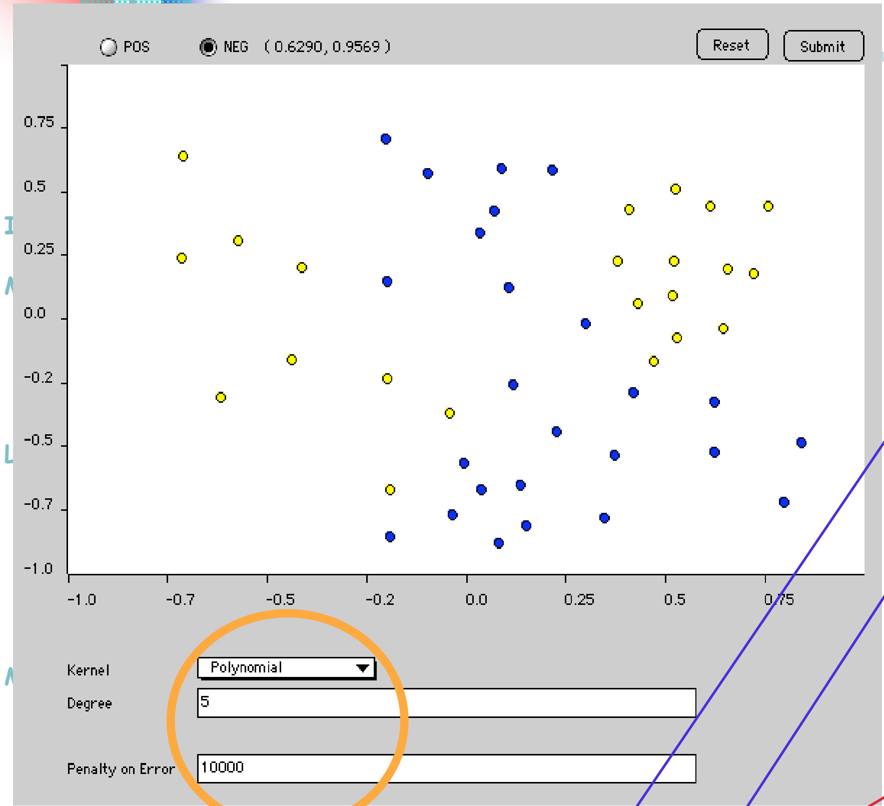
- Apprentissage de **deux classes**
  - exemples tirés uniformément sur l'échiquier
- SVM à **fonctions noyau gaussienne**

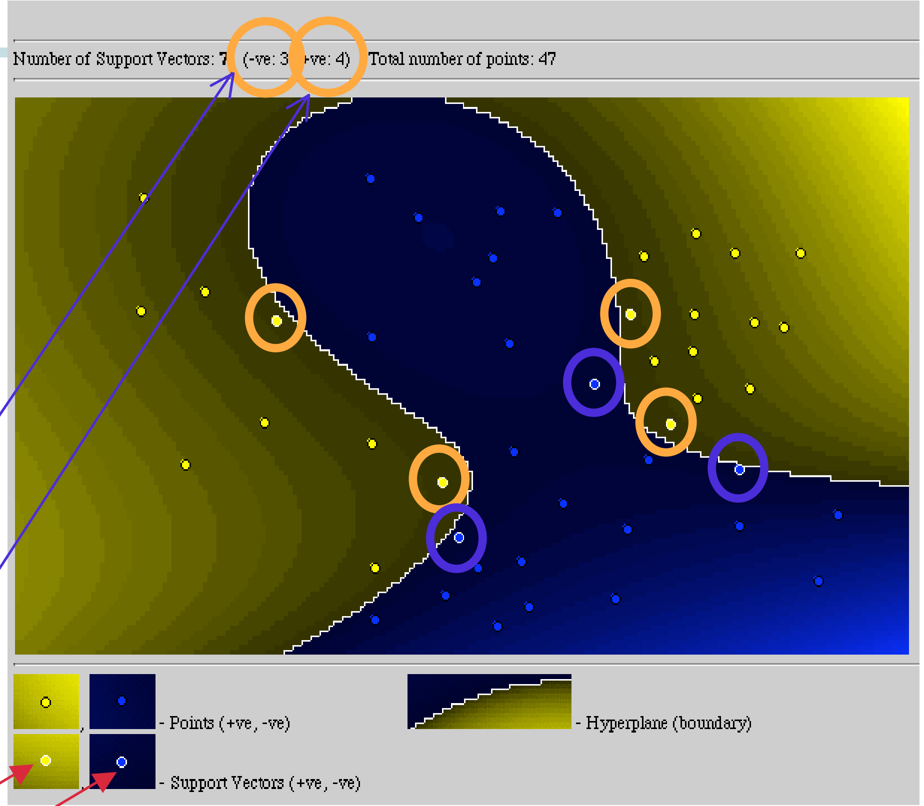
$$K(x, x') = e^{-\frac{\|x - x'\|^2}{2\sigma^2}}$$

- Ici deux valeurs de σ
  - **En haut : petite valeur**
  - **En bas : grande valeur**
- Les gros points sont des **exemples critiques**
  - **Plus en haut qu'en bas**
- Dans les deux cas : $R_{emp} = 0$



Paramètres de contrôle : les fonctions noyau

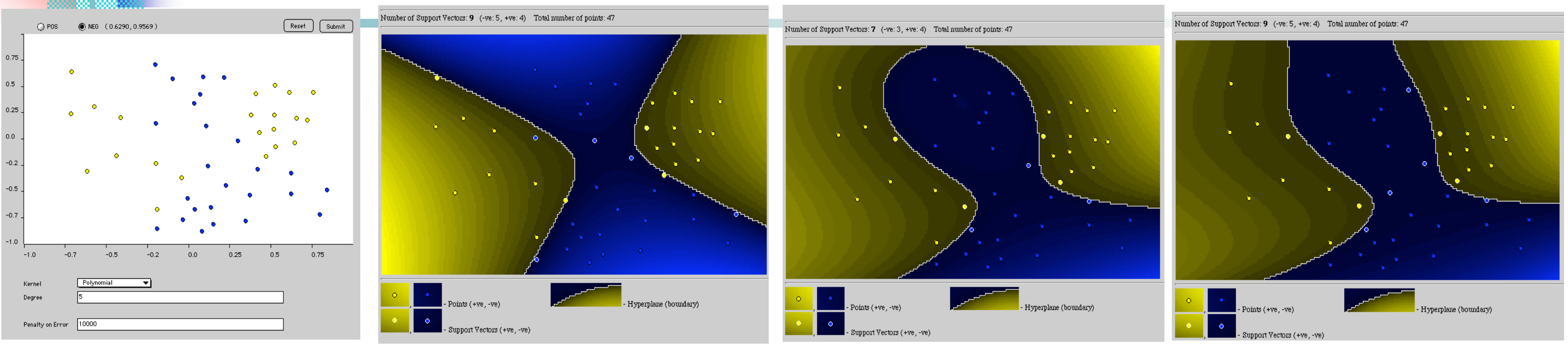




noyaux

- **http://svm.dcs.rhbnc.ac.uk/pagesnew/GPat.shtml**
- **47 exemples (22 +, 25 -)**
- ***Exemples critiques*** **: 4 + et 3 -**
- Ici ***fonction polynomiale*** de ***degré 5*** et ***C = 10000***

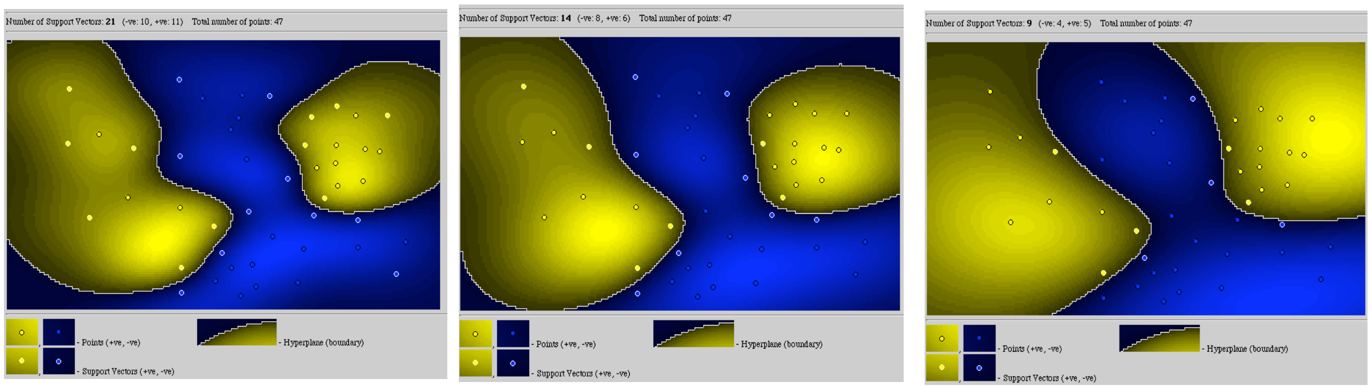
Paramètres de contrôle : les fonctions noyau



(5-, 4+) (3-, 4+) (5-, 4+)

- **47 exemples (22 +, 25 -)**
- ***Exemples critiques* : 4 + et 3 -**

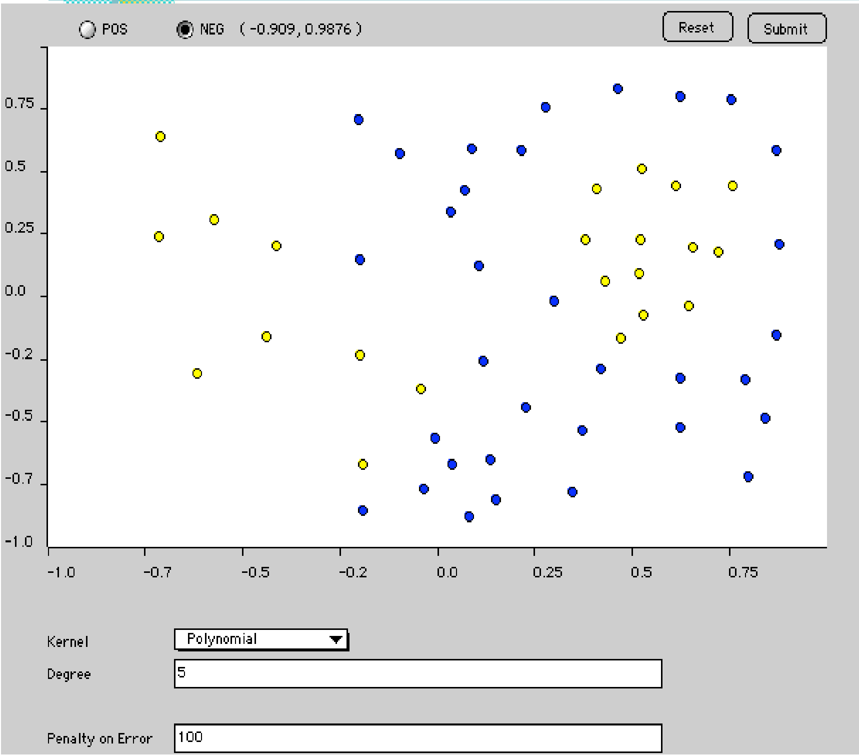
Ici *fonction polynomiale* **de** *degré 2, 5*, 8 **et** *C = 10000*

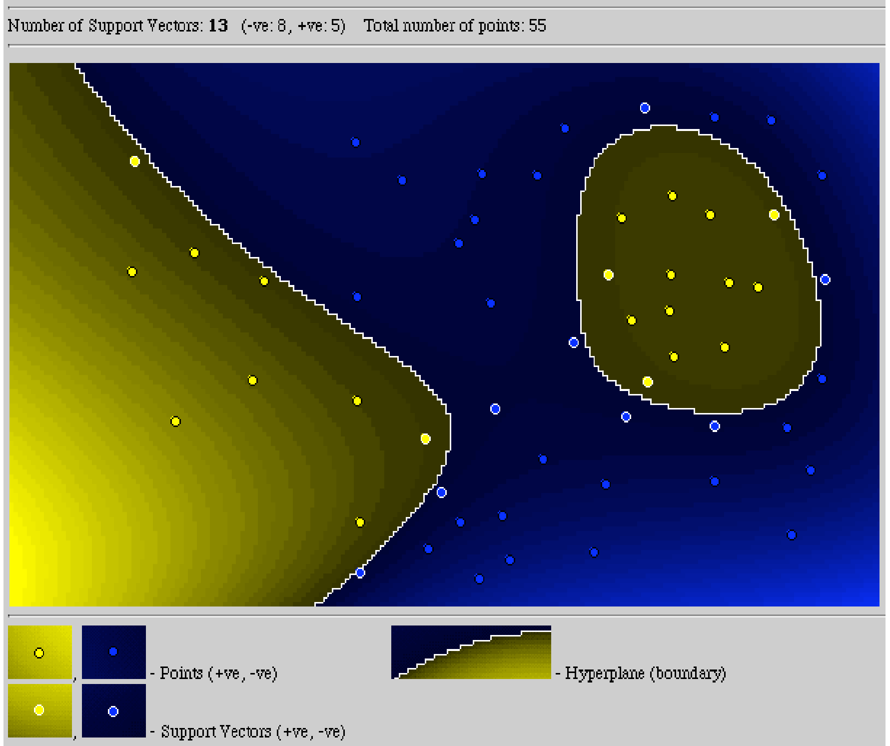


(10-, 11+) (8-, 6+) (4-, 5+)

Ici *fonction Gaussienne* **de** σ = *2, 5*, 10 **et** *C = 10000*

Ajout de quelques points ...





- **http://svm.dcs.rhbnc.ac.uk/pagesnew/GPat.shtml**
- **47 + 8 exemples (30 +, 25 -)**
- ***Exemples critiques* : 5 + et 8 -**
- Ici ***fonction polynomiale*** de ***degré 5*** et ***C = 10000***

Estimation de la performance

- ***Empiriquement*** : par **validation croisée**
- ***Heuristiquement*** (mais théoriquement fondé)
  - **Nombre de points de supports**
    - Moins il y en a, mieux c'est
  - **Caractéristiques de la matrice noyau**
    - Si pas de structure dans K, aucune régularité ne peut-être trouvée
    - E.g.
      - Si les termes hors diagonale sont très petits : sur-adaptation
      - Si matrice uniforme : sous-apprentissage : tous les points sont attribués à la même classe

Construction de fonctions noyau

- Construction à partir de fonctions noyau de base (Propriétés de clôture)
  - $K(\boldsymbol{x},z) = K_1(\boldsymbol{x},z) + K_2(\boldsymbol{x},z)$
  - $K(\boldsymbol{x},z) = a\, K_1(\boldsymbol{x},z)$
  - $K(\boldsymbol{x},z) = K_1(\boldsymbol{x},z)\, .\, K_2(\boldsymbol{x},z)$
  - …
- Construction de fonctions noyau dédiées
  - **Splines B_m**
  - **Expansion de Fourrier**
  - **Ondelettes**
  - …

Construction de noyaux

- **Noyau invariant par translation**

$$k(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{x} - \boldsymbol{z})$$

- **Noyau défini sur des ensembles**

$$k(A_1, A_2) = 2^{|A_1 \cap A_2|} \Longleftrightarrow \phi(A)_U = \begin{cases} 1 & \texttt{si} U \subseteq A, \\ 0 & \texttt{sinon.} \end{cases}$$

Stratégies de construction

- **Noyau vu comme un moyen de coder de l'information a priori**
  - Invariance: synonymie, longueur de document, …
  - Traitements linguistiques: normalisation des mots, semantique, stopwords, weighting scheme, …
- **Noyaux de convolution** :

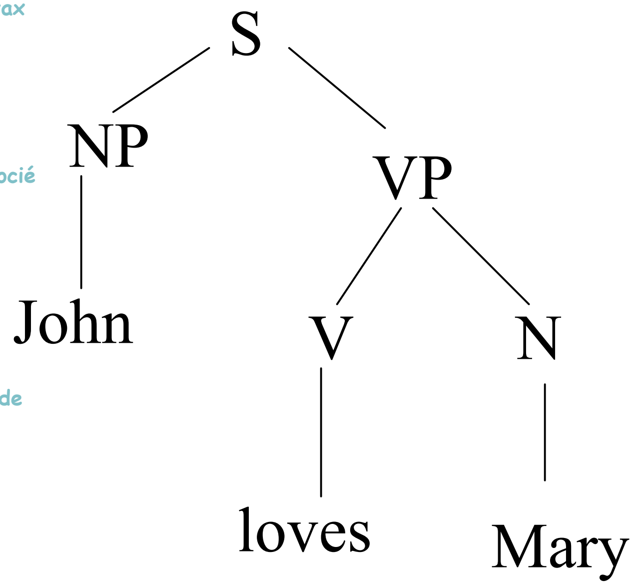
  le texte est une structure de données récursivement définie.

  Pb : ***construire un noyau global à partir de noyaux locaux ?***
- **Noyaux à partir de modèles génératifs** :

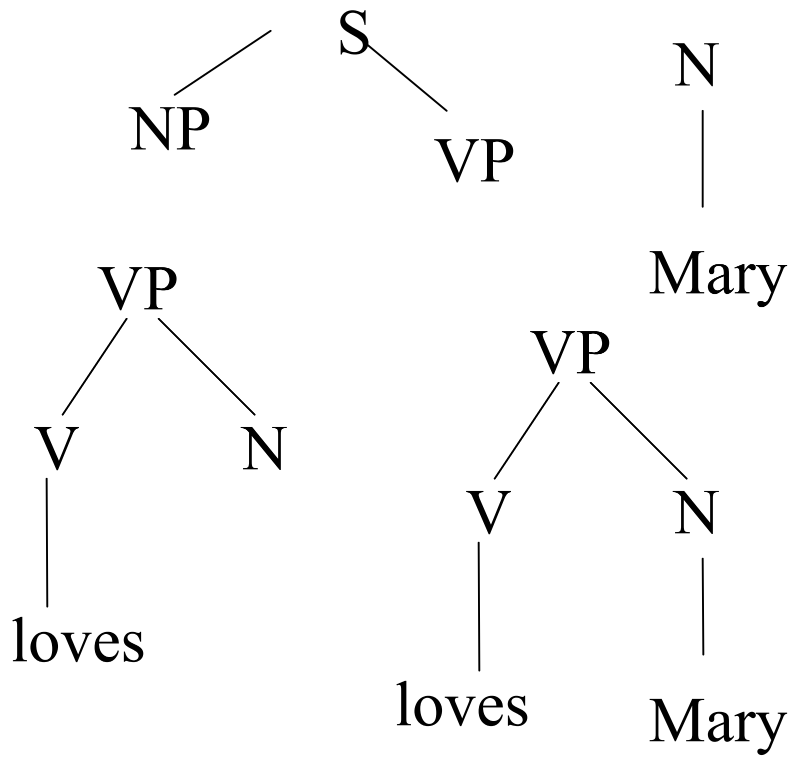
  la "topologie" du problème est traduite en une fonction noyau

Noyaux pour arbres : exemple

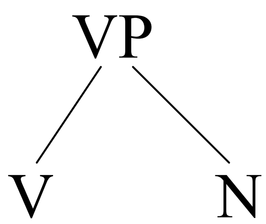
- **Exemple :**



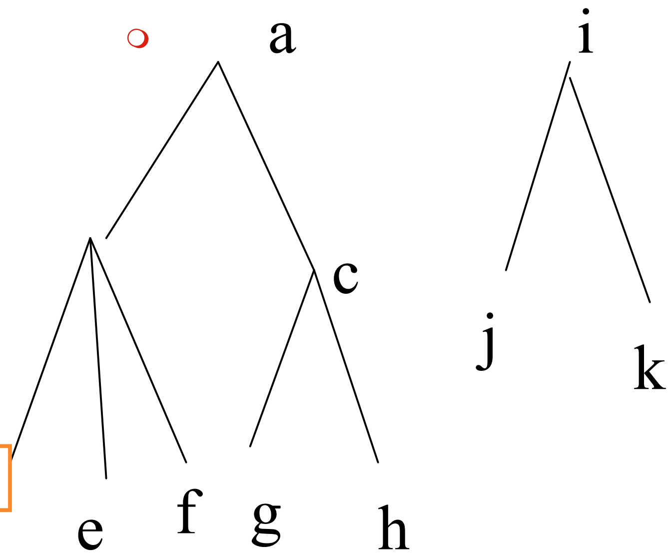
A Parse Tree



… quelques sous-arbres parmi ceux de l'arbre !



Illustration

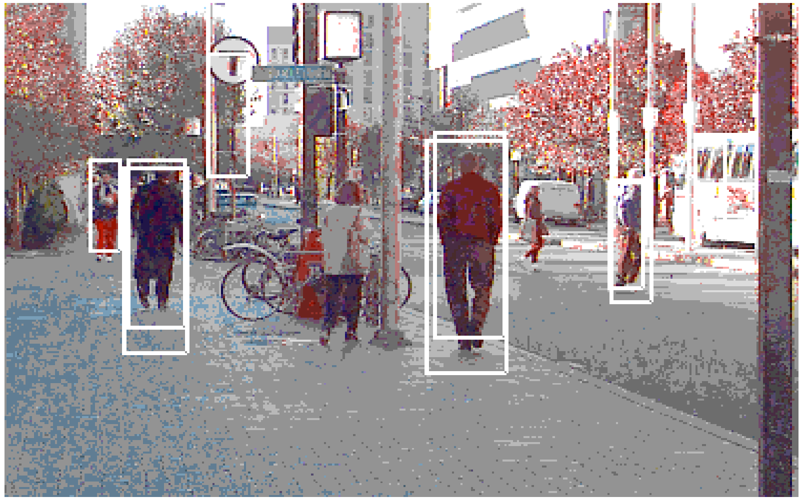


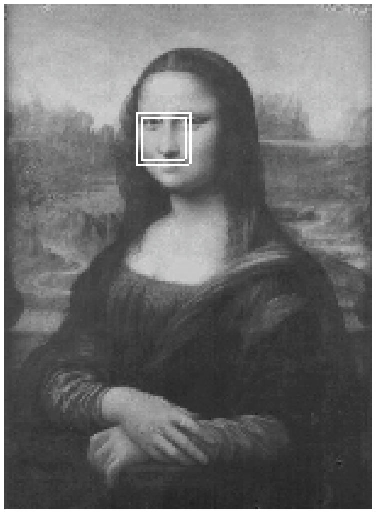
Kco-root	i	j	k
a	1	0	0
b	0	0	0
c	1	0	0
d	0	0	0
e	0	0	0
f	0	0	0
g	0	0	0
h	0	0	0

→K=2

Applications

- Catégorisation de textes
- Reconnaissance de caractères manuscrits
- Détection de visages
- Diagnostic de cancer du sein
- Classification de protéines
- Prévision de consommation électrique
- Recherche de vidéos par du texte





Trained SVM classifiers for pedestrian and face object detection (Papageorgiou, Oren, Osuna and Poggio, 1998)

Implémentation des SVMs

- **Minimisation de fonctions différentiables convexes à plusieurs variables**
  - **Pas d'optima locaux**
  - **Mais :**
    - **Problèmes de stockage de la matrice noyau (si milliers d'exemples)**
    - **Long dans ce cas**
  - D'où mise au point de méthodes spécifiques
    - Gradient sophistiqué
    - Méthodes itératives, optimisation par morceaux
  - Plusieurs **packages publics** disponibles
    - SVMTorch
    - SVM^{Light}
    - SMO
    - …

Bilan : état des recherches

- **Deux tâches évidentes**
  - ***Conception de noyaux***
    - Commence à être bien étudié
    - Encore des recherches pour certains types de données
  - ***Noyautiser*** les algorithmes classiques (« kernelization »)
    - SVM
    - Kernel Régression
    - Kernel PCA
    - Clustering (K-means, …)
    - Estimation de densité, détection de nouveauté
    - Tri (ranking)
    - …
- **Recherche sur la sélection automatique des modèles (choix des paramètres)**

Bilan

- **Les méthodes à noyau sont :**
  - Une bonne idée
  - Destinées à durer
- Offrent une **boîte à outils**
  - Très versatile
  - Avec de bons fondements théoriques
    - E.g. garanties de performance

Bilan

Nouvelle philosophie de représentation

- **Toute l'information** sur les données passe par le filtre de **la matrice noyau**
  - De l'information est perdue
  - Permet des manipulations particulières
    - E.g. ajout d'une constante sur la diagonale → marge souple ou terme de régularisation
    - Incorporation de connaissances a priori
  - Matrice noyau : interface entre les modules de traitement
- **La qualité de l'apprentissage** peut être estimée à partir des **caractéristiques de la matrice noyau**

Sources documentaires

Ouvrages / articles

- Cornuéjols & Miclet (02) : *Apprentisage artificiel. Concepts et algorithmes*. Eyrolles, 2002.
- Herbrich (02) : *Learning kernel classifiers*. MIT Press, 2002.
- Schölkopf, Burges & Smola (eds) (98) : *Advances in Kernel Methods : Support Vector Learning*. MIT Press, 1998.
- Schölkopf & Smola (02) : *Learning with kernels*. MIT Press, 2002.
- Shawe-Taylor & Cristianini(04) : *Kernel methods for pattern analysis*. Cambridge University Press, 2004.
- Smola, Bartlett, Schölkopf & Schuurmans (00) : *Advances in large margin classifiers*. MIT Press, 2000.
- Vapnik (95) : *The nature of statistical learning*. Springer-Verlag, 1995.

Sites web

- `http://www.kernel-machines.org/` **(point d'entrée)**
- `http://www.support-vector.net` **(point d'entrée)**