Séance 07: Algorithmes de Support Vector Machines

Sébastien Gadat

Laboratoire de Statistique et Probabilités
UMR 5583 CNRS-UPS

www.lsp.ups-tlse.fr/gadat

Septième partie VII

Algorithmes de Support Vector Machines

Introduction

- Ce sont des algorithmes d'apprentissage initialement construits pour la classification binaire.
- L'idée est de rechercher une règle de décision basée sur une séparation par hyperplan de marge optimale.
- Méthode relativement récente qui découle de premiers travaux théoriques de Vapnik et Chervonenkis en 1995, démocratisés à partir de 2000.
- Le principe de l'algorithme est d'intégrer lors de la phase d'apprentissage une estimation de sa complexité pour limiter le phénomène d'*over-fitting*.
- Méthode qui ne substitue pas au problème déjà compliqué de la classification un problème encore plus complexe comme l'estimation d'une densité de probabilités (par exemple).

Principe de l'algorithme

L'algorithme se base principalement sur 3 astuces pour obtenir de très bonnes performances tant en qualité de prédiction qu'en complexité de calcul.

- On cherche l'hyperplan comme solution d'un problème d'optimisation sous-contraintes. La fonction à optimiser intègre un terme de qualité de prédiction et un terme de complexité du modèle.
- Le passage à la recherche de surfaces séparatrices non linéaires est introduit en utilisant un noyau *kernel* qui code une transformation non linéaire des données.
- Numériquement, toutes les équations s'obtiennent en fonction de certains produits scalaires utilisant le noyau et certains points de la base de données (ce sont les *Support Vectors*).

Utilisations récentes

Dans tout type de problème de classification à deux classes :

- Classification d'images : reconnaissance de visages, de chiffres manuscrits
- Interprétation textuelle : détection de Spam
- Aide au diagnostic biologique

De nouveaux centres d'intérêts se développent actuellement :

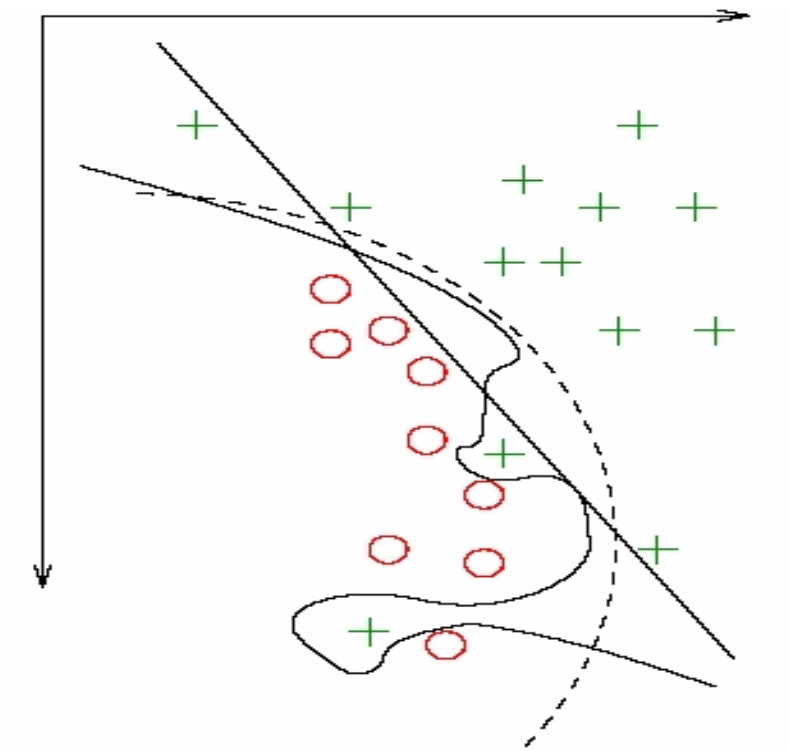
- Utilisation comme outil de sélection de variables
- *Sparse SVM* : utilisation de modèles parcimonieux pour la classification

Contraintes :

- A priori, capable d'utiliser un grand nombre de variables puisque l'astuce du noyau « envoie »le problème dans l'espace des individus.
- Plus il y a d'individus dans la base de données et meilleure est la prédiction, mais plus les calculs sont longs...

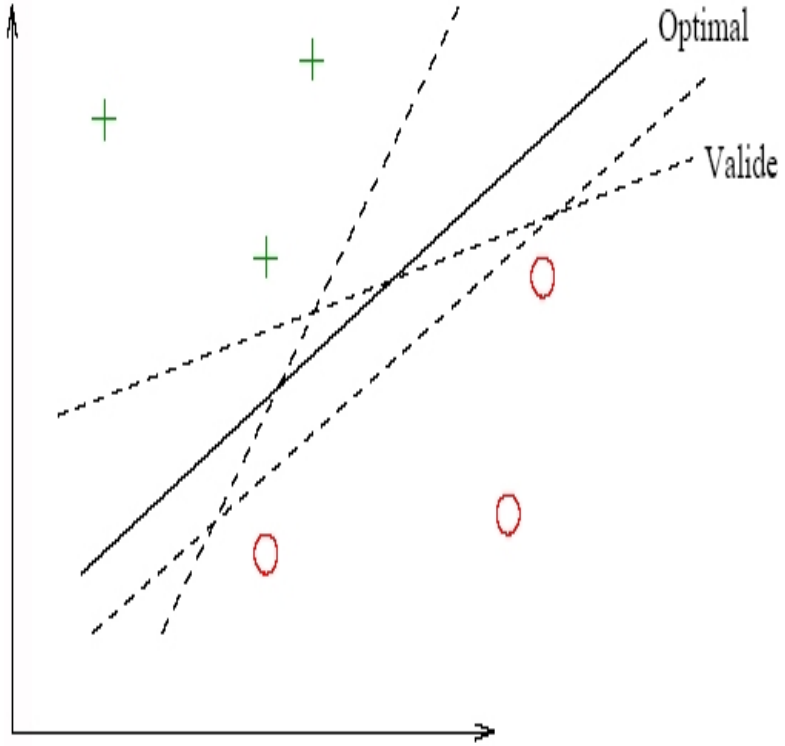
Problème

- On cherche à prédire Y (à valeurs dans $\{-1/+1\}$) en fonction de variables explicatives $X^1, \ldots X^p$.
- Chercher le modèle $\hat{\phi}$ tel que $P\left(\hat{\phi}(X) \neq Y\right)$ minimale.



Pénalisation

- On va chercher un modèle pénalisé par sa faculté d'adaptation aux données
- Comparer par exemple ceci avec la séparation précédente :



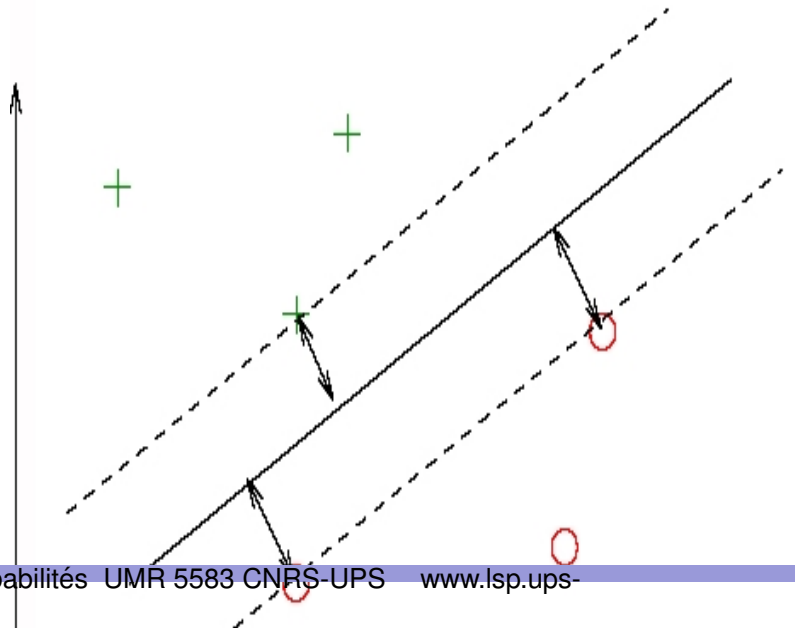
- On cherche en plus un modèle qui maximise la marge de séparation entre les classes

Notion de marge

- Plutôt que d'estimer ϕ comme fonction à valeurs dans $\{-1;1\}$, on cherche f telle que $\phi = sgn(f)$.
- L'erreur de la méthode est alors

$$P(\phi(X) \neq Y) = P(Yf(X) < 0)$$

- On définit alors la quantité $|Yf(X)|$ comme la marge de f en (X,Y). C'est un indice de confiance sur la prédiction.



Hyperplan séparateur - Mise en équation

- Dans le cas où la séparation est possible, on choisit celui qui obtient la marge optimale.
- L'hyperplan a une équation de la forme :

$$\underbrace{\langle w; b\rangle + b}_{=f(x)} = 0$$

où w est un vecteur orthogonal au plan

- Un point est bien classé si et seulement si $f(x)y > 0$
- Le jeu de données étant fini, on peut imposer $|yf(x)| \geq 1$
- La distance entre un point x et l'hyperplan est donnée par

$$d(x, \mathcal{H}) = \frac{|\langle w; x\rangle + b|}{\|w\|} = \frac{|f(x)|}{\|w\|}$$

Recherche de l'hyperplan optimal

- La recherche de l'hyperplan optimal se résume alors à trouver le vecteur w à la vue de la base de données
- La marge devant être maximale, on cherche w à minimiser

$$\min_w \frac{1}{2}\|w\|^2 \qquad \text{sous les contraintes} \qquad \exists b \quad \forall i \qquad \langle w; x_i \rangle + b \geq 1$$

- Ce problème d'optimisation est résolu par méthode « primal/dual »et la solution s'écrit sous la forme

$$f(x) = \sum_{i=1}^{n} \lambda_i^\star y_i \langle x; x_i \rangle + b^\star$$

- Ce qu'il faut retenir :
  - L'équation ne fait intervenir au final que des produits scalaires entre x et x_i
  - La résolution est quadratique en le nombre d'individus n
  - La plupart des $\lambda_i^\star$ sont nuls, sauf pour certaines observations, ces observations sont alors les « Support Vectors »

Recherche de l'hyperplan optimal

- Lorsque les données ne sont pas séparables par un plan, on « assouplit »les contraintes par :

$$y_i f(x_i) \geq 1 - \xi_i$$

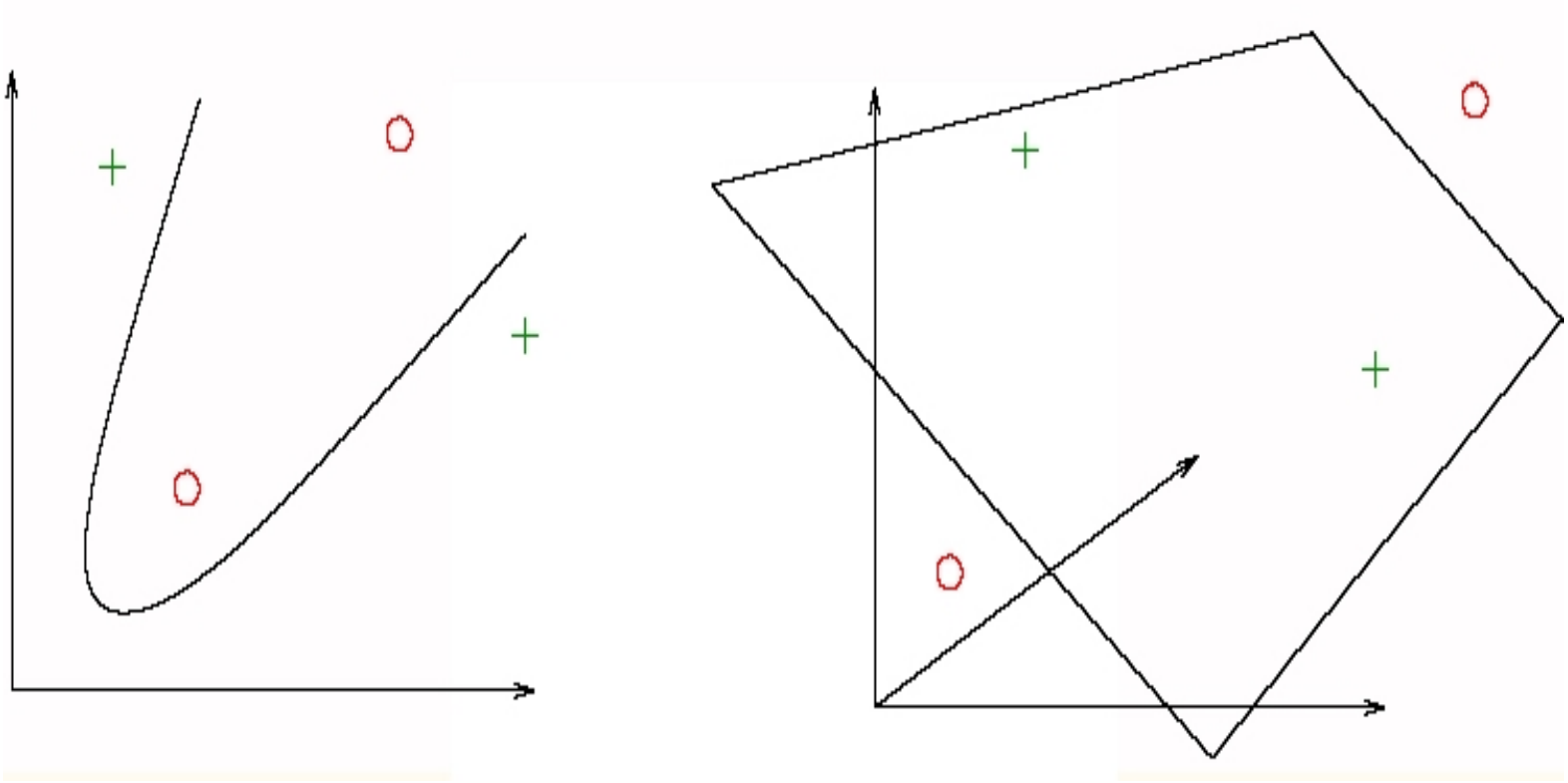
- Le modèle « se trompe »dès que $\xi_i > 1$
- La somme de tous les ξ_i doit être minimale pour occasionner le moins d'erreur de l'algorithme.
- Le problème d'optimisation devient

$$\min_w \frac{1}{2}\|w\|^2 + \delta \sum_{i=1}^{n} \xi_i \qquad \text{sous les contraintes} \qquad \exists b \quad \forall i \qquad \langle w; x_i \rangle + b \geq 1$$

- δ est un paramètre à régler entre le bon ajustement et la bonne généralisation
- Le problème se résout sous la même forme que précédemment, les $\lambda_i^\star$ sont bornés par δ
- Le modèle est d'autant plus fiable que le nombre de *Support Vectors* est faible

Sortir de l'espace de séparation linéaire

- On souhaite pouvoir discriminer les classes de points par des fonctions plus complexes que des hyperplans
- Voici un exemple de situation significatif :



- On souhaite soit discriminer par des courbes polynomiales, soit

Utilisation de Noyaux

- On utilise une application Φ de $\mathbb{R}^p$ dans $\mathcal{H}$ muni d'un produit scalaire, et plus « gros »que $\mathbb{R}^p$
- On définit dans le même temps l'application k, (*kernel*-noyau) qui vérifie :

$$\forall (x, x') \in \mathbb{R}^p \times \mathbb{R}^p \qquad k(x, x') = \langle \Phi(x); \Phi(x') \rangle_{\mathcal{H}}$$

- On sépare les données dans $\mathcal{H}$ en utilisant la même formulation par hyperplan linéaire (mais dans $\mathcal{H}$!).
- Ceci ne se traduit pas forcément par une séparation linéaire dans $\mathbb{R}^p$!
- L'équation de séparation, c'est-à-dire la construction du modèle, s'écrit en utilisant uniquement les termes $k(., x_i)$

Exemple de noyaux

- La fonction $k(x, x') = \langle x; x' \rangle_{\mathcal{R}^p}$ correspond à un hyperplan linéaire, dans ce cas : $\mathcal{H} = \mathbb{R}^p$.
- $k(x, x') = (c + \langle x; x' \rangle)^d$ cherche une séparation par courbe polynômiale de degré au plus d
- Le noyau Gaussien $k(x, x') = e^{\frac{-\|x-x'\|^2}{2\sigma^2}}$ cherche une séparation par frontière radiale (*Radial Basis Function*)
- On peut choisir le « bon »noyau en utilisant une stratégie de validation croisée. En général, ce choix permet de gagner quelques pourcentages d'erreur.

SVM pour la régression

- Cas où Y est quantitative.
- Les fonctions 'interpolantes' sont de la forme

$$\phi(x, w) = \sum_{i=1}^{\infty} w_i v_i(x)$$

où v_i est une base de fonctions type Fourier par exemple.

- On cherche à minimiser une fonction de coût en norme L1 et non L2 :

$$E(w, \lambda) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \phi(x_i, w)|_\epsilon + \lambda \|w\|_2^2$$

- λ est un paramètre de régularisation, $|.|_\epsilon$ est une application type valeur absolue, mais nulle sur l'intervalle $[-\epsilon; \epsilon]$.
- Là encore, les w_i sont souvent nuls sauf pour certains points x_i qui sont ici encore les supports vectors.