# Support Vector Machines Solver

Jules Pondard - Joseph de Vilmarest

3 janvier 2017

## Introduction

On cherche à apprendre un classifieur de points dans $\mathbb{R}^n$ en deux catégories. On dispose pour cela d'un ensemble d'entraînement : des points $x_i \in \mathbb{R}^n$ et leur classe $y_i \in \{-1, 1\}$. Un SVM essaie de trouver l'hyperplan séparateur entre les deux catégories optimal, en un certain sens à définir.

Formellement, un SVM minimise $\frac{1}{2}\|w\|_2^2 + C1^T z$ sous les contraintes $z \geq 0$ et $y_i(w^T x_i) \geq 1 - z_i$ pour $1 \leq i \leq m$, en $w \in \mathbb{R}^n$ et $z \in \mathbb{R}^m$.

Durant tout ce rapport on s'intéressera à un exemple : deux nuages de points de $\mathbb{R}^2$ obtenus comme des gaussiennes de covariance $I_2$ et d'espérances respectives $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$ et $\begin{pmatrix} 2 \\ 0 \end{pmatrix}$.

## 1 Méthode barrière

On applique tout d'abord la méthode barrière, en utilisant une méthode de Newton à chaque pas : on minimise $f(w,x) = t(\frac{w^T w}{2} + C1^T z) - \sum\limits_{i=1}^{m} log(z_i) - \sum\limits_{i=1}^{m} log(y_i(w^T x_i) - 1 + z_i)$ avec un pas $t$ qui tend vers l'infini. On a pris $t = 1$ au début et on multiplie $t$ par 10 à chaque étape.

Pour appliquer la méthode de Newton, on calcule $\nabla f$ avec

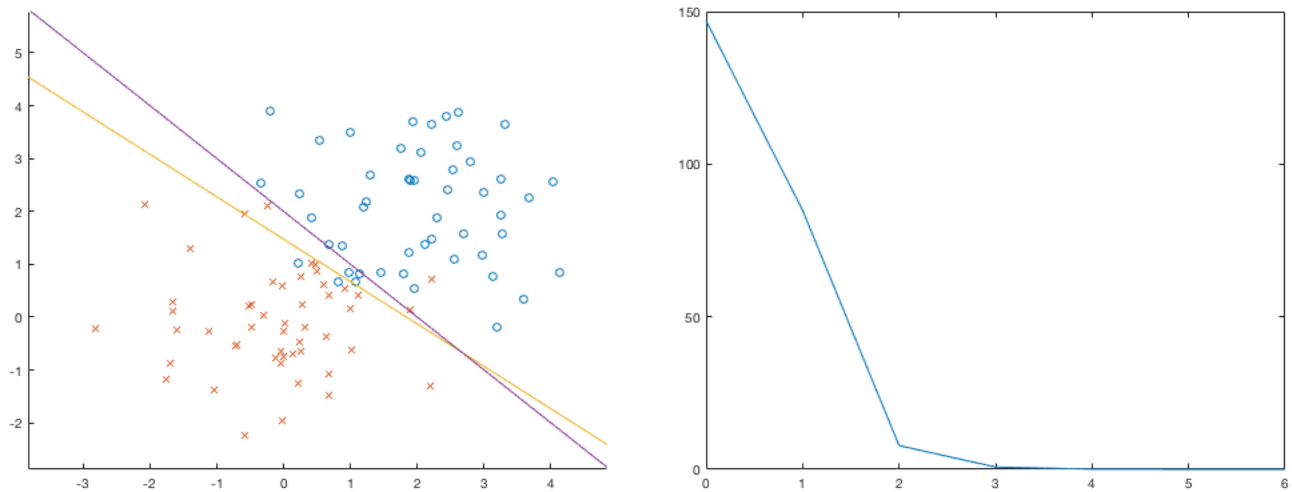
- $\frac{\partial f}{\partial w_i} = tw_i - \sum\limits_{k=1}^{m} \frac{y_k x_k^i}{y_k(w^T x_k) - 1 + z_k}$
- $\frac{\partial f}{\partial z_i} = tC - \frac{1}{z_i} - \frac{1}{y_i(w^T x_i) - 1 + z_i}$

Pour la hessienne $\nabla^2 f$, on utilise

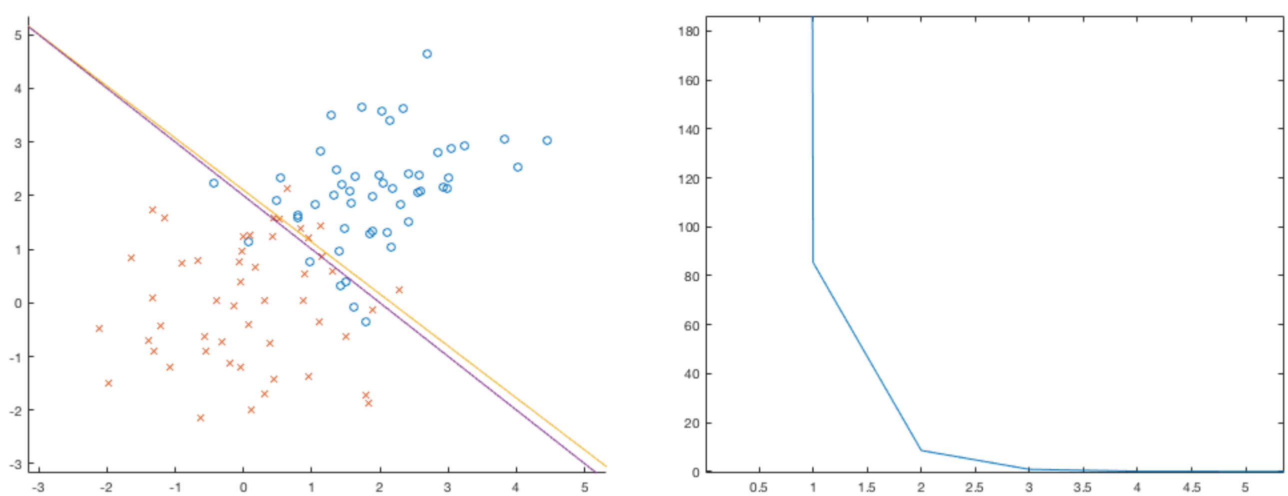
- $\frac{\partial^2 f}{\partial w_i \partial w_j} = t\delta_{i=j} + \sum\limits_{k=1}^{m} \frac{x_k^i x_k^j}{(y_i(w^T x_i) - 1 + z_i)^2}$
- $\frac{\partial^2 f}{\partial w_i \partial z_j} = \frac{y_j x_j^i}{(y_j(w^T x_j) - 1 + z_j)^2}$
- $\frac{\partial^2 f}{\partial z_i \partial z_j} = \delta_{i=j}(\frac{t}{z_i^2} + \frac{1}{(y_i(w^T x_i) - 1 + z_i)^2})$

On fait varier C et on affiche, par dessus le nuage de points, la droite séparatrice trouvée et la droite "optimale" (médiatrice des espérances des gaussiennes). C'est le graphe de gauche. On afficher de plus le gap à droite.

C=1 :



C=100 :



On se rend compte que plus C est grand, plus la solution obtenue est proche de la solution optimale : la pénalité $C1^T z$ est en effet plus importante. On donne ainsi le pourcentage d'erreur sur un ensemble de test en fonction de C.

| C | 1 | 10 | 100 |
|---|---|---|---|
| Erreur | | | |

## 2 CVX

On n'a pas réussi à utiliser Libsvm qui doit être mieux optimisée pour le problème en question.