Intelligence artificielle
Support vector machines

Mastère Data Scientist

Antoine Vacavant
antoine.vacavant@ynov.com
antoine.vacavant@nxv.fr

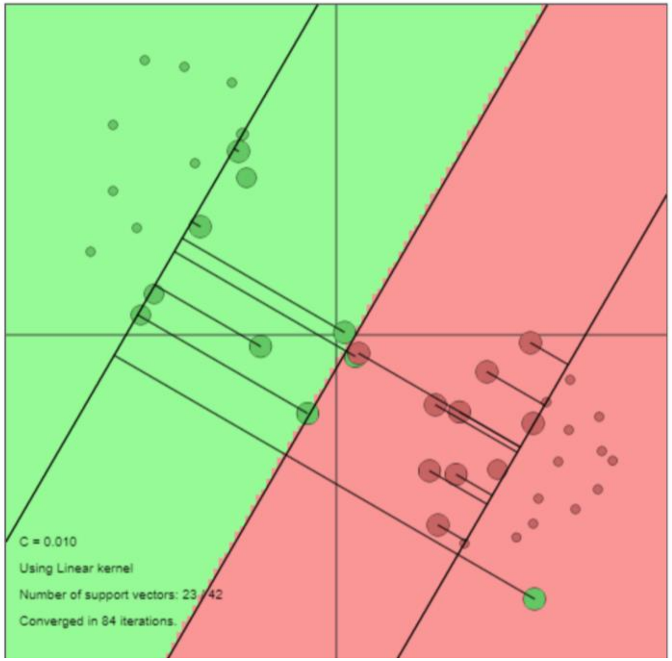
Plan

1. Principe
   – Cas linéaire
   – Première formulation
   – Cas linéairement non-séparable
2. Méthodes de résolution
   – Primal et dual
   – Cas non linéaire
3. Bilan

Principe

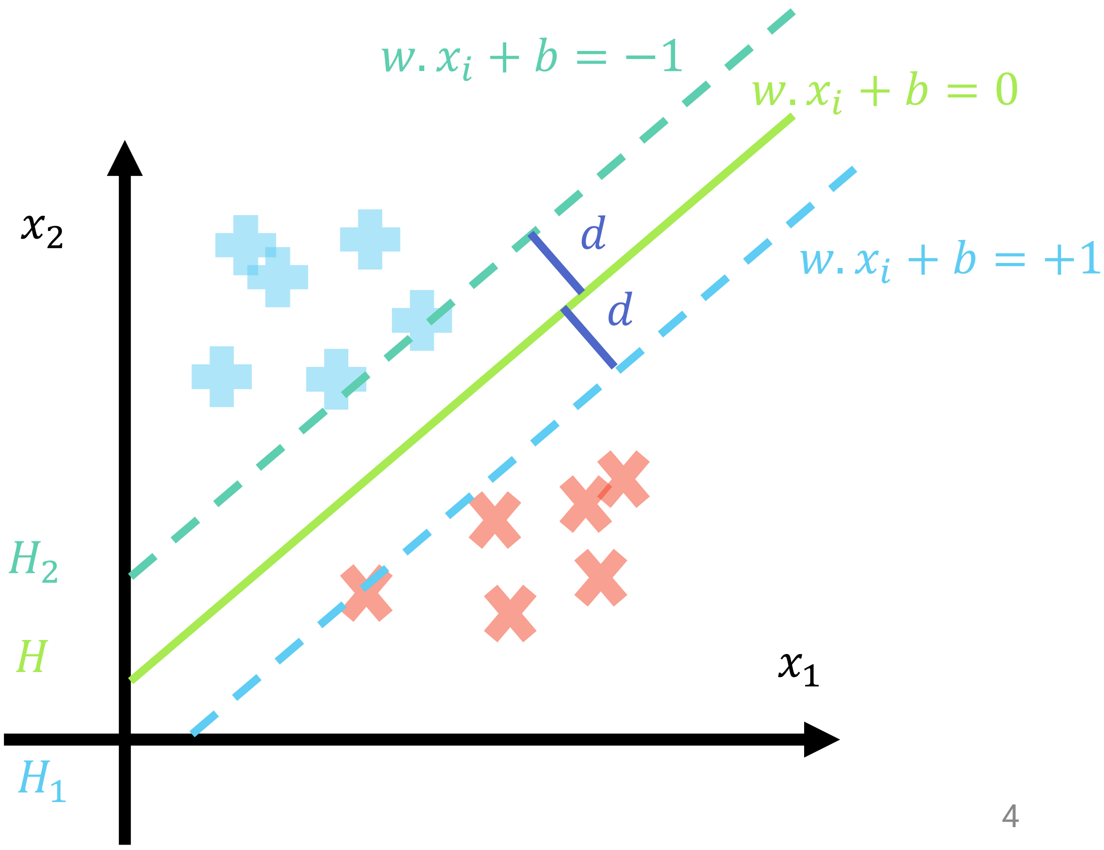
- Machines à vecteurs de support
- Calculer une fonction séparatrice des données
- En utilisant certaines d'entre elles comme support
- Inclut une marge d'erreur

- Deux approches
  - Linéaire
  - Non-linéaire



Cas linéaire

- Dans un cadre 2-classes
- On cherche à déterminer l'hyperplan tel que
  - $w.x_i + b \geq +1$ lorsque $y_i = +1$
  - $w.x_i + b \leq -1$ lorsque $y_i = -1$
- La marge : d



Cas linéaire

- La distance d s'exprime de la manière suivante :

$$d_i = \frac{y_i(w.x_i + b)}{\|w\|}$$

- Notre objectif : maximiser la marge $2d_i$
- Plus simplement, on cherche à maximiser $\frac{1}{2}\|w\|^2$
- Forme quadratique
- Résolution plus simple par une approche lagrangienne

Cas linéaire

- On ne souhaite pas de points entre H_1 et H_2
  - $w.x_i + b \geq +1$ lorsque $y_i = +1$
  - $w.x_i + b \leq -1$ lorsque $y_i = -1$

- Combinés ensemble :

$$y_i(w.x_i + b) \geq 1$$

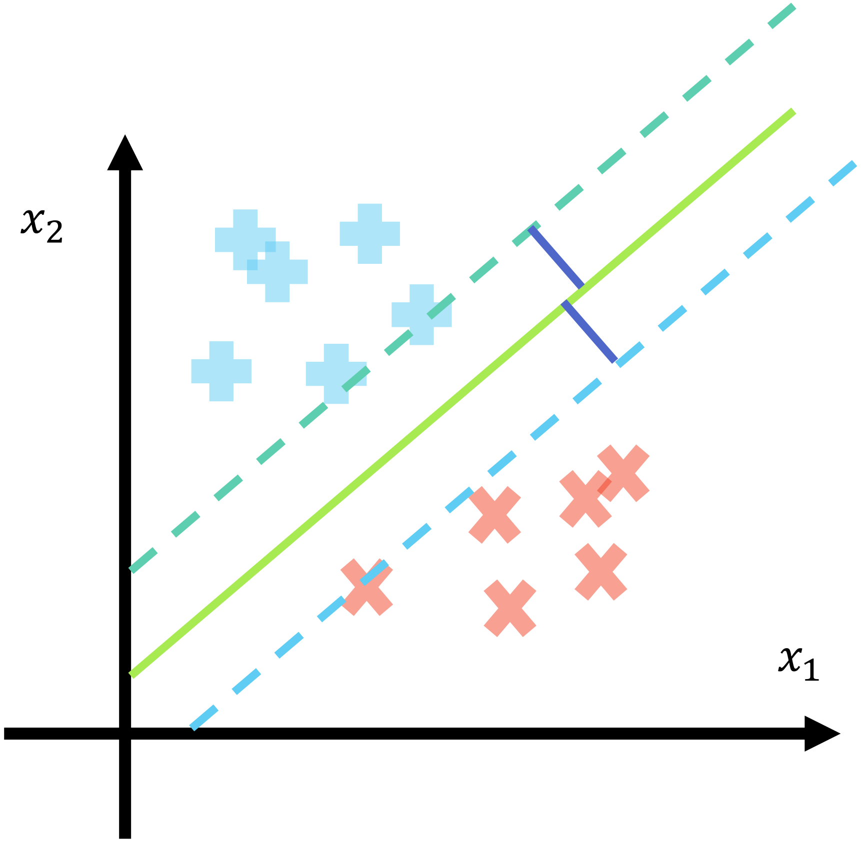
Première formulation

- On cherche à résoudre le problème suivant :

$$\min_{w,b} \frac{1}{2} ||w||^2$$
$$\text{tel que } y_i(w.x_i + b) > 1 \quad \forall i = 1, \dots, n$$

Première formulation

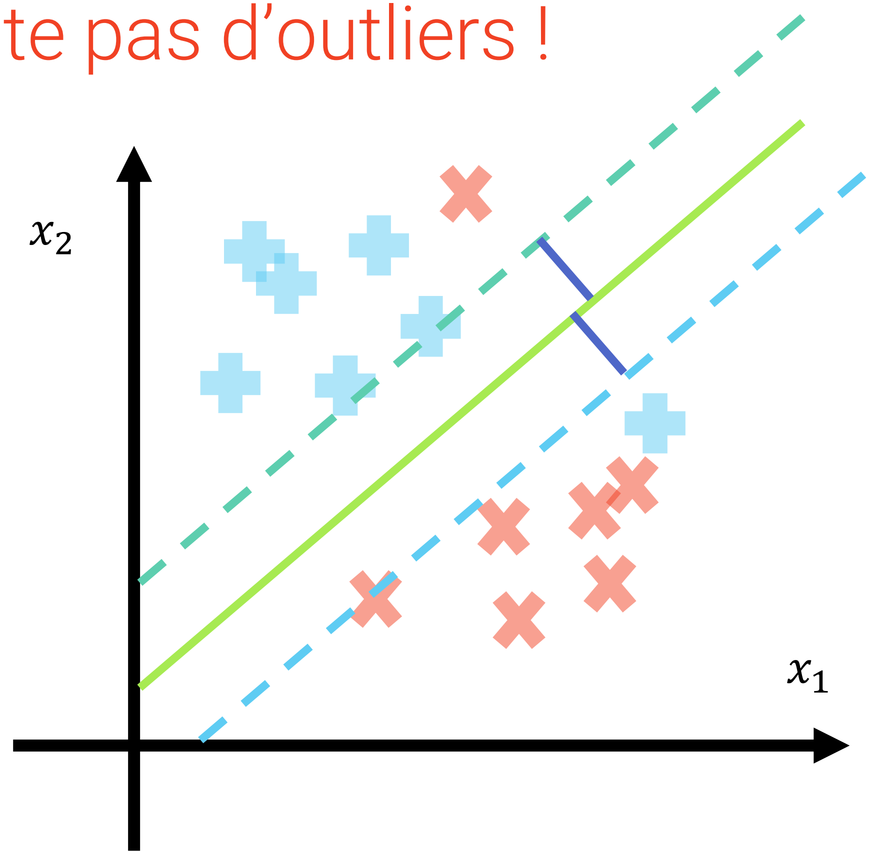
- Problème de cette formulation ?



Première formulation

- Problème de cette formulation ?

Elle n'accepte pas d'outliers !



Cas linéairement non-séparable

- On cherche à résoudre le problème suivant :

$$\min_{w,b} \frac{1}{2}\|w\|^2 + \frac{C}{n}\sum_{i=1}^{n} \xi_i$$

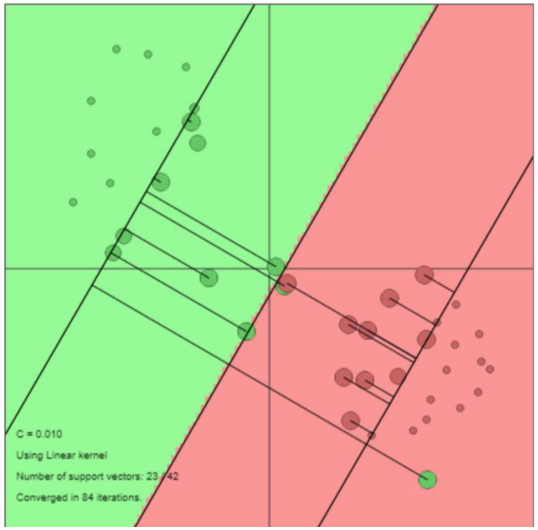
$$\text{tel que } y_i(w.x_i + b) > 1 - \xi_i \quad \forall i = 1, \dots, n$$

$$\xi_i \geq 0$$

  - $\xi_i \geq 0$: échantillon mal classifié
  - $\xi_i = 0$: échantillon bien classifié
  - C grand : pas d'erreur dans le calcul de la marge

Retour sur SVMjs

- Sur ce site
  →https://cs.stanford.edu/~karpathy/svmjs/demo/
- Passer en mode linéaire
- Etudier l'influence du paramètre C



Résolution

- Méthodes d’optimisation
- Descente de gradient stochastique

- On peut également reformuler le calcul de w :

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

Où $\alpha_i \in \mathbb{R}$

Résolution

- Méthodes d'optimisation
- Descente de gradient stochastique

- On peut également reformuler le calcul de w :

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

Où $\alpha_i \in \mathbb{R}$

La solution est combinaison linéaire des données

Résolution

- Changement de variable de w, b à α
- Calcul original : primal
- Nouvelle version : dual

Résolution

- Changement de variable de w, b à α
- Calcul original : primal
- Nouvelle version : dual
- Le problème devient alors :

$$\max_{\alpha_i \geq 0} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i x_j)$$

tel que $0 \leq \alpha_i \leq C$

$$\sum_i \alpha_i y_i = 0$$

Résolution

- Changement de variable de w, b à α
- Calcul original : primal
- Nouvelle version : dual
- Le problème devient alors :

$$\max_{\alpha_i \geq 0} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i x_j)$$

tel que $0 \leq \alpha_i \leq C$

$$\sum_i \alpha_i y_i = 0$$

Interlude

- Le calcul $x_i . x_j$ représente l'« éloignement » des données x_i et x_j

Interlude

- Le calcul $x_i . x_j$ représente l'« éloignement » des données x_i et x_j
- Si $x_i . x_j = \mathbf{1}$ les vecteurs sont colinéaires
  - Donc complètement similaires
- Si $x_i . x_j = \mathbf{0}$ les vecteurs sont perpendiculaires
  - Donc complètement différents

Interlude

- Le calcul $x_i . x_j$ représente l'« éloignement » des données x_i et x_j
- Si $x_i . x_j = 1$ les vecteurs sont colinéaires
  - Donc complètement similaires
- Si $x_i . x_j = 0$ les vecteurs sont perpendiculaires
  - Donc complètement différents

Comment améliorer cette mesure de dissimilarité ?

Augmentation de dimension

- Supposons une feature map suivante :

$$\phi(x): x \rightarrow \phi(x)$$
$$\mathbb{R}^d \rightarrow \mathbb{R}^D$$

- Permet d'augmenter la représentation des données
- On pourrait être tenté de calculer $\phi(x_i).\phi\left(x_j\right)$
- Mais calcul complexe… voire impossible numériquement

Augmentation de dimension

- A la place, on utilise des noyaux de la forme
$$K(x_i, x_j) = \phi(x_i).\phi(x_j)$$
- Inutile de calculer ϕ directement
- Le noyau devient une mesure de dissimilarité

- Exemples de noyaux pour un SVM non-linéaire
  - Polynomial : $K(x_i, x_j) = (x_i.x_j + 1)^p$
  - RBF (radial basis function) : $K(x_i, x_j) = \exp\left({\|x_i - x_j\|^2}/_{2\sigma^2}\right)$
  - Sigmoide : $K(x_i, x_j) = \tanh(\kappa\, x_i.x_j - \delta)$

Bilan

- Le SVM permet une classification efficace
- Surtout dans le cadre de 2 classes
- Possibilité d'une approche multi-classes
- Avec moins de paramètres qu'un réseau de neurones artificiels standard

- Mais nécessite généralement l'extraction de caractéristiques au préalable (ex. ondelettes)