Analyse, Classification et Indexation des données (ACID)

Apprentissage supervisé : Perceptron et SVM

Akka Zemmari

LaBRI, Université de Bordeaux

2023 - 2024

Rappel

Classification binaire

- Données : quelques éléments qui appartiennent à deux classes différentes
  - classe 1 (+1) et classe 2 (-1)
  - classe positive (+1) et classe négative (-1)

Rappel

Classification binaire

- Données : quelques éléments qui appartiennent à deux classes différentes
  - classe 1 (+1) et classe 2 (-1)
  - classe positive (+1) et classe négative (-1)
- Tâche : entrainer un classifieur sur ces données puis prédire la classe d'un nouvel élément
- Géométriquement : trouver une séparation entre les deux classes

Séparation Linéaire / Non Linéaire

Données linéairement séparables

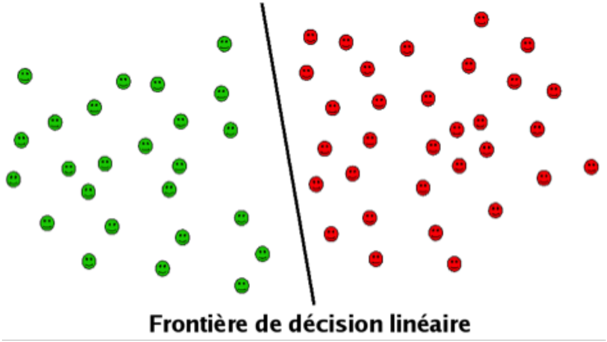
Tous les points associés aux données peuvent être séparés correctement par une frontière linéaire (hyperplan)

Séparation Linéaire / Non Linéaire

Données linéairement séparables

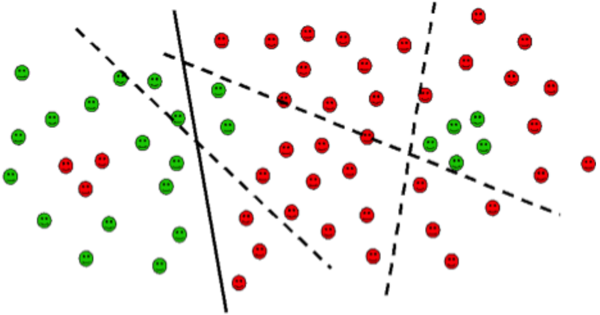
Tous les points associés aux données peuvent être séparés correctement par une frontière linéaire (hyperplan)

Exemple : données linéairement séparables



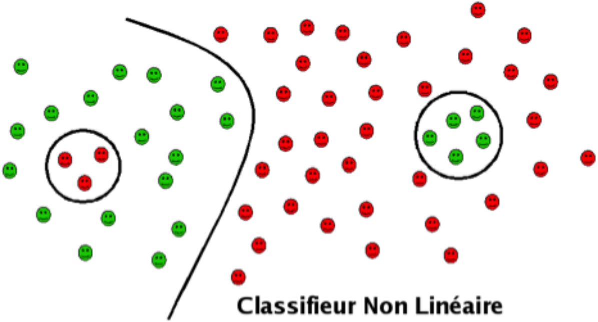
Séparation Linéaire / Non Linéaire

Exemple : pas de séparation linéaire



Séparation Linéaire / Non Linéaire

Exemple : pas de séparation linéaire, mais avec un classifieur non linéaire



Algorithmes : Séparation Linéaire / Non Linéaire

Séparation Linéaire

- Données séparables Linéairement ou Non Linéairement ?

Algorithmes : Séparation Linéaire / Non Linéaire

Séparation Linéaire

- Données séparables Linéairement ou Non Linéairement ?
  réponse empirique, se faire aider par les outils de visualisation

Algorithmes : Séparation Linéaire / Non Linéaire

Séparation Linéaire

- Données séparables Linéairement ou Non Linéairement ? réponse empirique, se faire aider par les outils de visualisation
- Algorithmes linéaires (algorithmes qui trouvent une frontière linéaire)

Algorithmes : Séparation Linéaire / Non Linéaire

Séparation Linéaire

- Données séparables Linéairement ou Non Linéairement ?
  réponse empirique, se faire aider par les outils de visualisation
- Algorithmes linéaires (algorithmes qui trouvent une frontière linéaire)
  - Quand on pense que les données sont linéairement séparables

Algorithmes : Séparation Linéaire / Non Linéaire

Séparation Linéaire

- Données séparables Linéairement ou Non Linéairement ? réponse empirique, se faire aider par les outils de visualisation
- Algorithmes linéaires (algorithmes qui trouvent une frontière linéaire)
  - Quand on pense que les données sont linéairement séparables
  - Avantages : Simples, peu de paramètres à régler

Algorithmes : Séparation Linéaire / Non Linéaire

Séparation Linéaire

- Données séparables Linéairement ou Non Linéairement ?
  réponse empirique, se faire aider par les outils de visualisation
- Algorithmes linéaires (algorithmes qui trouvent une frontière linéaire)
  - Quand on pense que les données sont linéairement séparables
  - Avantages : Simples, peu de paramètres à régler
  - Désavantages : les données dans un espace de grande dimension sont souvent non linéairement séparables

Algorithmes : Séparation Linéaire / Non Linéaire

Séparation Linéaire

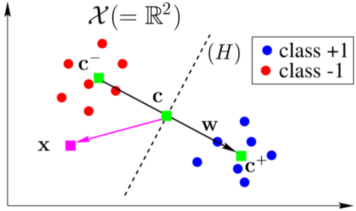
- Données séparables Linéairement ou Non Linéairement ? réponse empirique, se faire aider par les outils de visualisation
- Algorithmes linéaires (algorithmes qui trouvent une frontière linéaire)
  - Quand on pense que les données sont linéairement séparables
  - Avantages : Simples, peu de paramètres à régler
  - Désavantages : les données dans un espace de grande dimension sont souvent non linéairement séparables
  - Exemples d'algorithmes : Perceptron, SVM

Algorithmes : Séparation Linéaire / Non Linéaire

Séparation Non linéaire

- Non Linéaires
  - Quand les données sont non linéairement séparables
  - Avantages : Plus précis
  - Désavantages : Plus compliqués, plus de paramètres à régler
  - Exemple : méthodes à base de fonctions noyau
- Note : la distinction entre linéaire et non linéaire est valable pour la classification multi-classes

Un premier algorithme[1] (simple)



Centres de gravité des classes

- $\mathbf{c}^+ = \frac{1}{n^+} \sum_{i:y_i=+1} \mathbf{x}_i$
- $\mathbf{c}^- = \frac{1}{n^-} \sum_{i:y_i=-1} \mathbf{x}_i$
- $\mathbf{c} = \frac{1}{2}(\mathbf{c}^+ + \mathbf{c}^-)$
- $\mathbf{w} = \mathbf{c}^+ - \mathbf{c}^-$

Fonction de décision

Classer chaque élément x dans la classe qui a le centre le plus proche.

- Pour x, il suffit de prendre le signe du produit scalaire entre w et $x - c$
- Si $h(x) =< w, x - c >$, on a le classifieur $f(x) = signe(h(x))$
- Sur la figure, la surface de décision est représentée par l'hyper-plan (H), de vecteur normal w.

[1] voir le cours de Cécile Capponi (https://pageperso.lis-lab.fr/cecile.capponi/?page_id=32)

Classifieurs linéaires (généralités)

Définition

Un classifieur linéaire s'appuie sur une surface de décision g qui opère une séparation des données selon le côté de cette surface où se situent les points

- Fondé sur une combinaison linéaire des entrées
- Combinaison définie par un vecteur $w = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{pmatrix}$
- Surface de décision $g(x) = w^T x + w_0 = <w, x> + w_0$

Classifieurs linéaires (généralités)

Définition

Un classifieur linéaire s'appuie sur une surface de décision g qui opère une séparation des données selon le côté de cette surface où se situent les points

- Simplification de notation matricielle : ajout d'une composante constante 1 comme première composante de x
  L'expression de g devient :

$$g(x) = w^T x \text{ avec } x = \begin{pmatrix} x_0 = 1 \\ x_1 \\ x_2 \\ \vdots \\ x_d \end{pmatrix} \text{ et } w = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_d \end{pmatrix}$$

Algorithme du Perceptron

Perceptron : généralités

- Linéaire
- Classification Binaire
- En ligne (apprentissage séquentiel, une donnée à la fois)
- Apprentissage sur les erreurs
- Réseau de neurones à une couche

Algorithme d'apprentissage du Perceptron (Rosenblatt, 1958)

Input

$\{(x_i, y_i)_{i=1,\cdots,n}\}$ avec $(x_i, y_i) \in \mathbb{R}^{d+1} \times \{-1, 1\}$

Principe

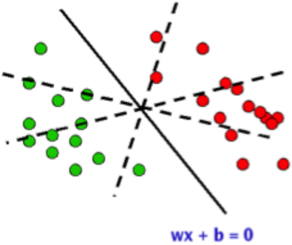
Mise à jour itérative des coefficients w de l'équation de l'hyper-plan séparateur

- Si (x, y) tel que $y = 1$ est mal classé, $<w, x> < 0$ et il faudrait augmenter $<w, x>$,
- si (x, y) tel que $y = 1$ est mal classé, $<w, x> \geq 0$ et il faudrait diminuer $<w, x>$,

Algorithme du Perceptron

Principe

- Trouver un hyperplan $(w, b) \in \mathbb{R}^{d+1}$ qui classe aussi bien que possible les données (points)
- Progressivement : un point à la fois, en modifiant les poids si nécessaire



Algorithme du Perceptron

Algorithme original

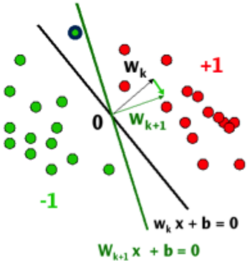
Input: $S = \{(x_1, y_1), \cdots, (x_n, y_n)\}$, un échantillon linéairement séparable de $\mathbb{R}^{d+1} \times \{-1, 1\}$

$w_0 = 0 \in \mathbb{R}^{d+1}$, $k = 0$

Répéter
    Pour $i = 1$ **à** n
        Si $y_i < w_k, x_i > \leq 0$ (mal classé) **Alors**
            $w_{k+1} = w_k + y_i x_i$
            $k = k + 1$
    Fin pour
Jusqu'à ce qu'il n'y ait plus d'erreurs
Output: w_{k+1}

Algorithme du Perceptron

Graphiquement



Algorithme du Perceptron

Avantage : très facile à implémenter

Un exemple

Séparer

$$S = \{((0,0,1);-1),((0,1,1);1),((1,0,1);1),((1,1,1);1)\}$$

Dessiner l'hyperplan obtenu.

Voir le Jupyter notebook

Algorithme du Perceptron

Analyse

- Progressif : s'adapte toujours aux nouvelles données
- Avantages
  - Simple et efficace
  - Garantie d'apprendre un problème linéairement séparable (convergence, optimum global)

Algorithme du Perceptron

Analyse

- Progressif : s'adapte toujours aux nouvelles données
- Avantages
  - Simple et efficace
  - Garantie d'apprendre un problème linéairement séparable (convergence, optimum global)
- Limitations
  - Seulement séparations linéaires
  - Converge seulement pour données séparables
  - Pas très efficace dès qu'il y a trop de features

Séparateur à vastes marges (SVM)

Séparateur à vastes marges (SVM)

- Une autre famille d'algorithmes linéaires

Séparateur à vastes marges (SVM)

- Une autre famille d'algorithmes linéaires
- Si les classes sont linéairement séparables :
  - Séparer les données
  - Mettre un hyperplan "loin" des données : marge large
  - bonne généralisation

Séparateur à vastes marges (SVM)

- Une autre famille d'algorithmes linéaires
- Si les classes sont linéairement séparables :
  - Séparer les données
  - Mettre un hyperplan "loin" des données : marge large
  - bonne généralisation

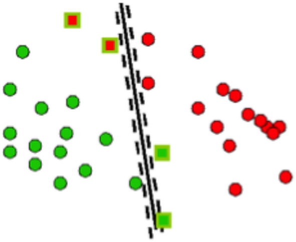


Figure: Mauvais

Séparateur à vastes marges (SVM)

- Une autre famille d'algorithmes linéaires
- intuition (Vapnik, 1965)
- Si les classes sont linéairement séparables :
  - Séparer les données
  - Mettre un hyperplan "loin" des données : marge large
  - bonne généralisation

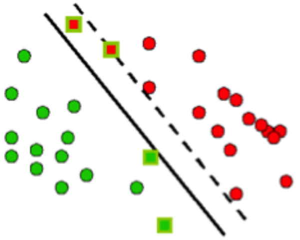
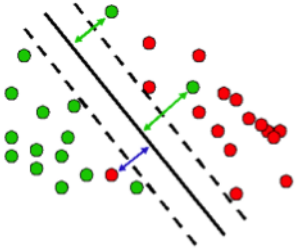


Figure: Bon ⇒ Classifieur à marge maximale.

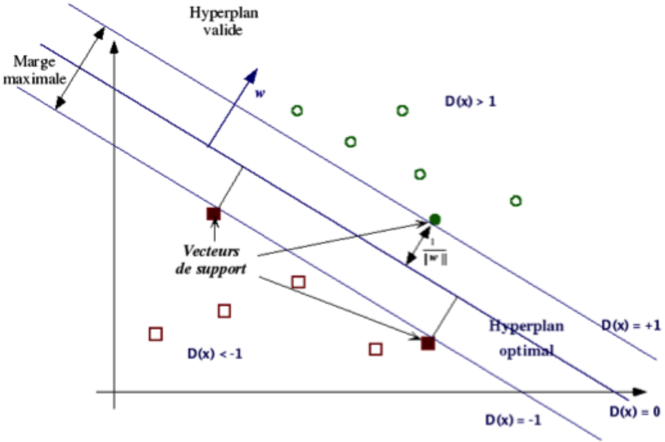
Séparateur à vastes marges (SVM)

- Si non séparable linéairement
  - Permettre quelques erreurs
  - Essayer encore de placer un hyperplan "loin" de chaque classe



Hyperplan de plus vaste marge

Optimiser la marge



Hyperplan de plus vaste marge

Optimiser la marge

- La distance d'un point à l'hyperplan est : $\frac{|wx+b|}{||w||}$
- L'hyperplan optimal est celui pour lequel la distance aux points les plus proches est maximale. La marge entre les deux classes vaut $\frac{2}{||w||}$.
- Maximiser la marge revient donc à minimiser $||w||$ sous ?les contraintes :

$$\begin{cases} min\frac{||w||^2}{2} \\ \forall i,\ y_i(wx_i + b) \geq 1 \end{cases}$$

SVM : un problème d'optimisation quadratique

Expression primaire

- Il faut donc déterminer w et b minimisant :

$$\frac{||w||^2}{2}$$

  afin de maximiser le pouvoir de généralisation.
- sous les contraintes (hyperplan séparateur):

$$y_i(wx_i + b) \geq 1, \ \forall i \in [1..n]$$

La résolution de ce problème n'est pas au programme de ce cours. L'idée est plutôt de donner l'intuition et expliquer comment la méthode marche ...

Données non linéairement séparables : SVM à noyau (kernel trick)

Intuition : un exemple (voir le notebook).
Soit les données suivantes :

Données non linéairement séparables : SVM à noyau (kernel trick)

Intuition : un exemple.
Appliquons aux données la transformation suivante :

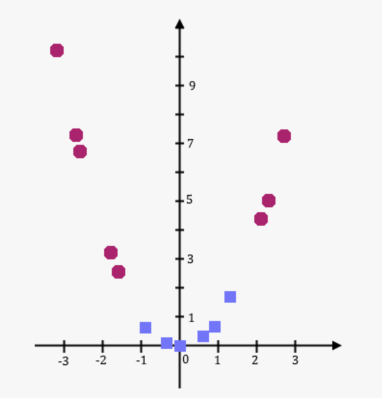
$$\phi(x) = \left(\frac{x - 150}{10}, \left(\frac{x - 150}{10} \right)^2 \right)$$

On passe donc de la dimension 1 à une dimension 2.

Données non linéairement séparables : SVM à noyau (kernel trick)

Intuition : un exemple.

Les données après application du noyau :



Données non linéairement séparables : SVM à noyau (kernel trick)

Noyaux couramment utilisés :

- Noyau polynomial.
- Noyau gaussien.
- Noyau laplacien.
- Noyau rationnel.

SVM en pratique

Voir le jupyter notebook