Séparateurs à Vaste Marge

SVM avec Python

MARHOUN Marouane
ECH-CHOUYYEKH Monir

Encadrer par :
Mr. CHRAYEH Mohamed

Université Abdelmalek Essaadi
École Nationale des Sciences Appliquées

26 novembre 2018

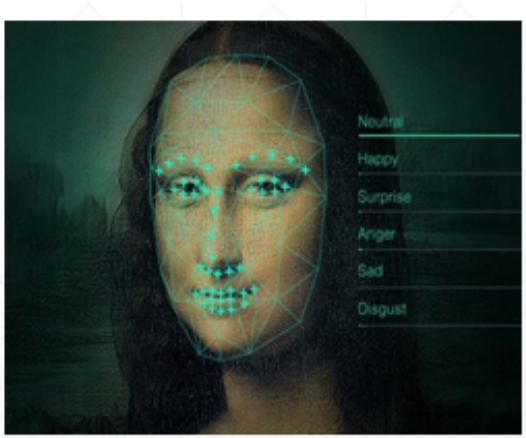
Plan

Introduction

L'apprentissage automatique (Machine Learning ML) apprend un modèle à partir des données passées afin de prédire les données futures. Le processus clé est l'apprentissage, qui est l'une des intelligences artificielles. De nombreuses techniques statistiques, probabilistes et d'optimisation différentes peuvent être mises en œuvre comme méthodes d'apprentissage telles que la régression logistique, les réseaux de neurones artificiels (RNA), K-voisin le plus proche (KNN), arbre de décision (DT), Naïve Bayes et Séparateurs à Vaste Marge (SVM).

Introduction

La technologie de reconnaissance faciale « par exemple » permet aux plateformes de médias sociaux d'aider les utilisateurs à marquer et partager des photos d'amis.



Introduction

La technologie de reconnaissance optique des caractères (OCR) convertit les images du texte en caractères mobiles.

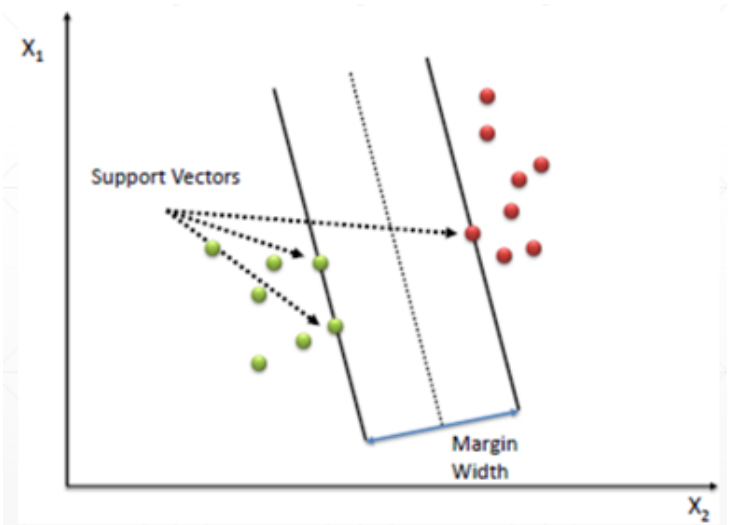
SVM

- Support Vector Machines est considéré comme une approche de classification, mais peut être utilisé dans les deux types de problèmes de classification et de régression.
- Méthode relativement récente qui découle de premiers travaux théoriques de Vapnik et Chervonenkis en 1995, démocratisés à partir de 2000.
- L'idée est de rechercher une règle de décision basée sur une séparation par hyperplan de marge optimale.
- 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.

Séparateurs à Vaste Marge SVM

Une machine à vecteurs de support (SVM) effectue la classification en recherchant l'hyperplan qui optimise la marge entre les deux classes. Les vecteurs (cas, points) qui définissent l'hyperplan sont les vecteurs de support.on définit Algorithme par les étapes :

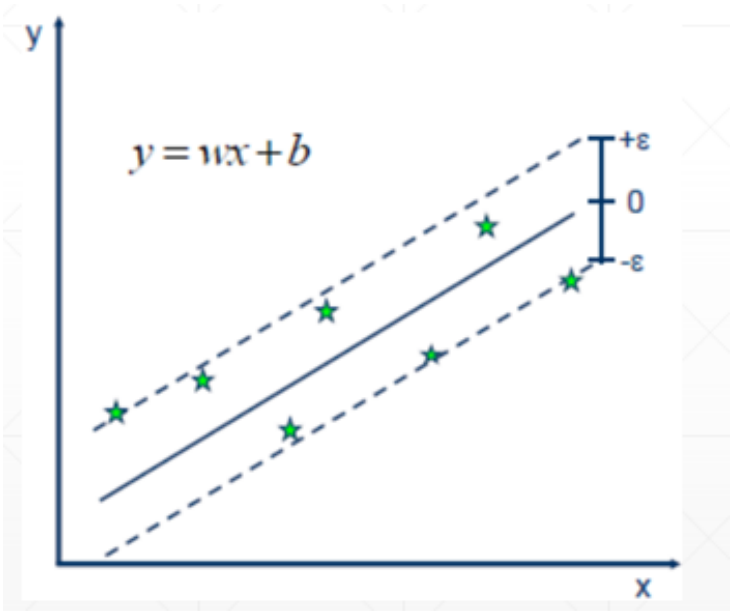
- Définir un hyperplan optimal : maximiser la marge
- Pour les problèmes séparables de manière non linéaire : prévoir un terme de pénalité pour les erreurs de classification.
- Mappez les données vers un espace de grande dimension où il est plus facile de classer avec des surfaces de décision linéaires : reformulez le problème afin que les données soient mappées implicitement dans cet espace.



SVM

La régression vectorielle de support (SVR) utilise les mêmes principes que la SVM pour la classification, avec seulement quelques différences mineures.

- En cas de régression, une marge de tolérance (epsilon) est définie comme une approximation du SVM qui l'aurait déjà demandé au problème.
- L'idée principale est toujours la même : minimiser les erreurs, individualiser l'hyperplan qui maximise la marge, en gardant à l'esprit qu'une partie de l'erreur est tolérée.



Séparation linéaire par SVM

Les termes que nous allons utiliser fréquemment dans ce travail :

- Noyau : fonction utilisée pour mapper une donnée de dimension inférieure en une donnée de dimension supérieure.
- Hyper plan : en SVM, il s'agit essentiellement de la ligne de séparation entre les classes de données. Bien que dans SVR nous allons le définir comme la ligne qui nous aidera à prédire la valeur continue ou la valeur cible.
- Ligne de limite : Dans SVM, deux lignes autres que Hyper Plan créent une marge. Les vecteurs de support peuvent se trouver sur les lignes de délimitation ou à l'extérieur. Cette ligne de démarcation sépare les deux classes. En SVR, le concept est le même.
- Vecteurs de support : Ce sont les points de données les plus proches de la limite. La distance des points est minimale ou minimale.

Séparation linéaire par SVM

Nous avons besoin de présenté une technique permettant de construire un hyperplan de séparation optimal entre deux classe parfaitement séparer (linéairement séparable).

Le problème abordé est celui de la discrimination binaire. Il s'agit de trouver un moyen permettant de construire une fonction de décision associant à chaque observation sa classe. Alors L'idée des SVMs est de rechercher un hyperplan (droite dans le cas de deux dimensions) qui sépare le mieux ces deux classes (1 et -1 ou + et -), soit x_i=(x_1,x_2,...,x_n) est l'ensembles des données et $y_i \in \{-1, 1\}$ est la classe de chacune.

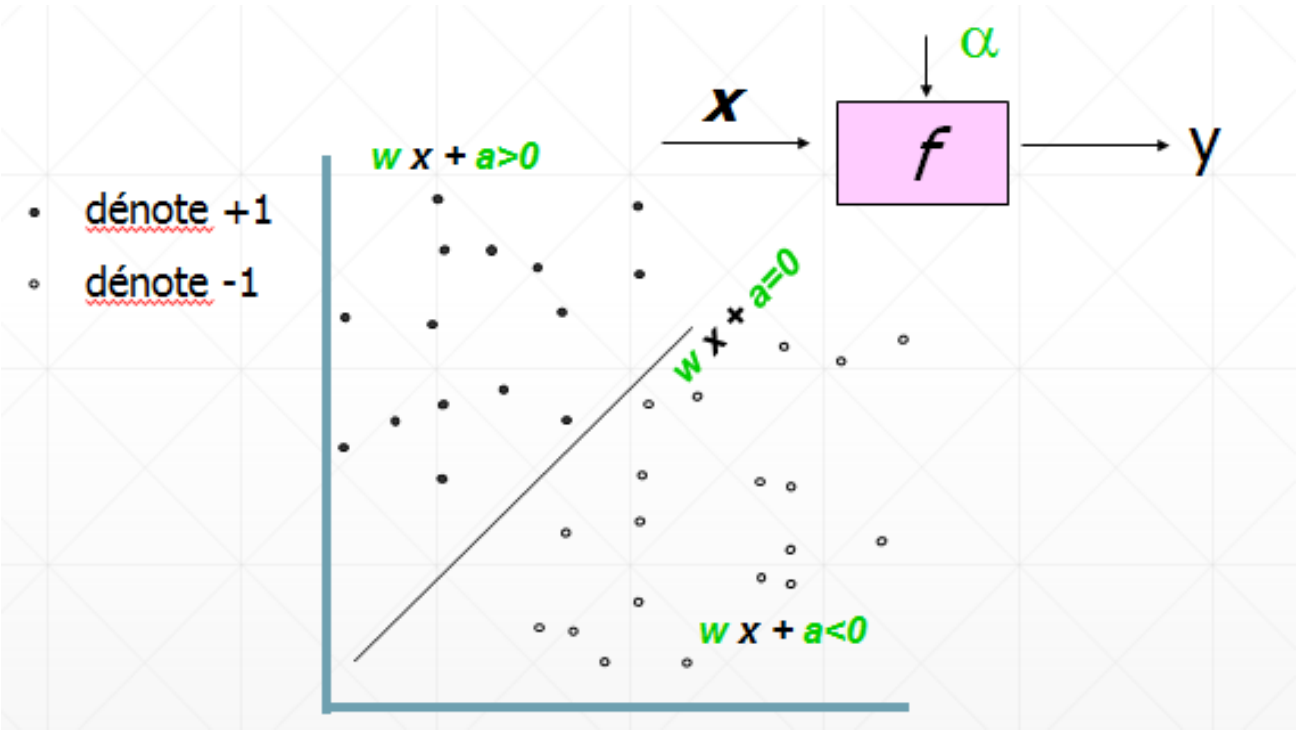
L'objectif est de trouver une séparation linéaire permettant de distinguer les '+' des '-'. Le classifieur (fonction de discrimination) se présente sous la forme d'une combinaison linéaire des variables.

Séparation linéaire par SVM

Classificateurs Linéaires

Comment classeriez-vous ces données ?

f(x,w,a) = sign(w^T x + a)

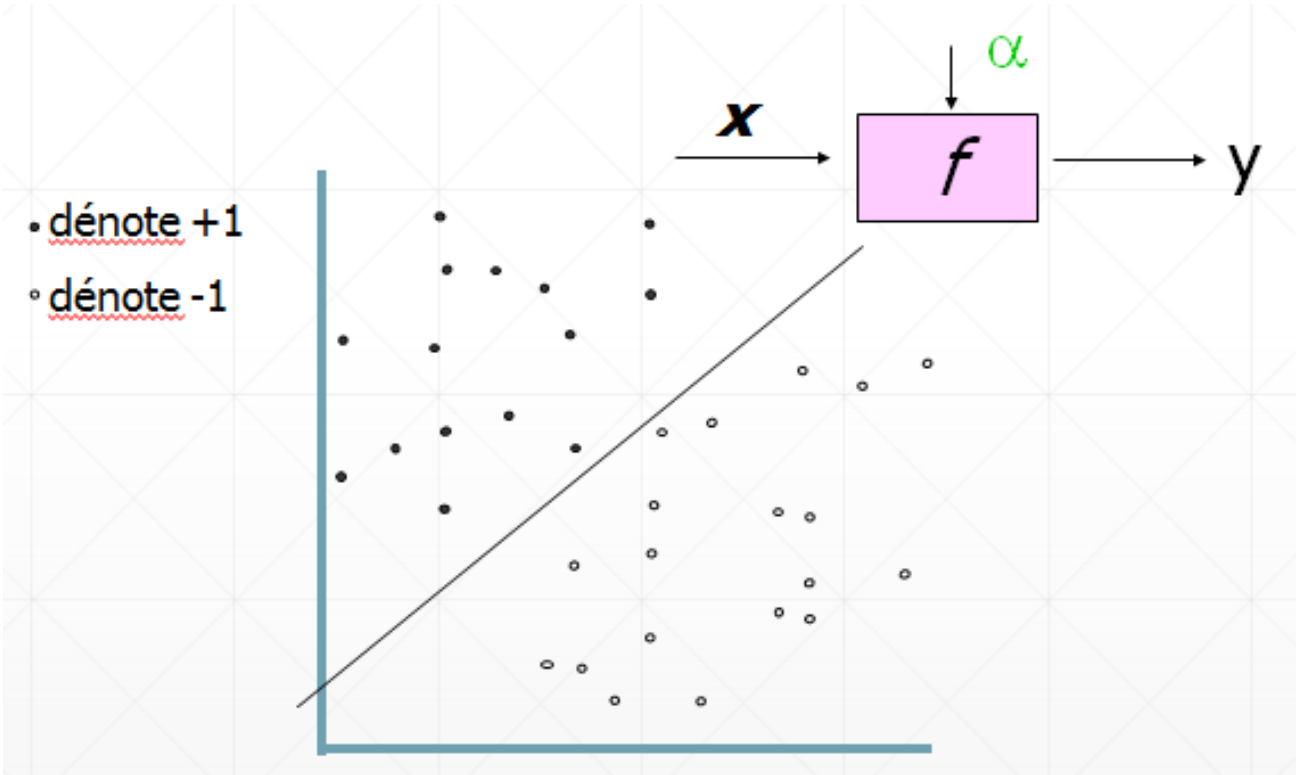


Séparation linéaire par SVM

Classificateurs Linéaires

Comment classeriez-vous ces données ?

$f(x,w,a) = \text{sign}(w^T x + a)$

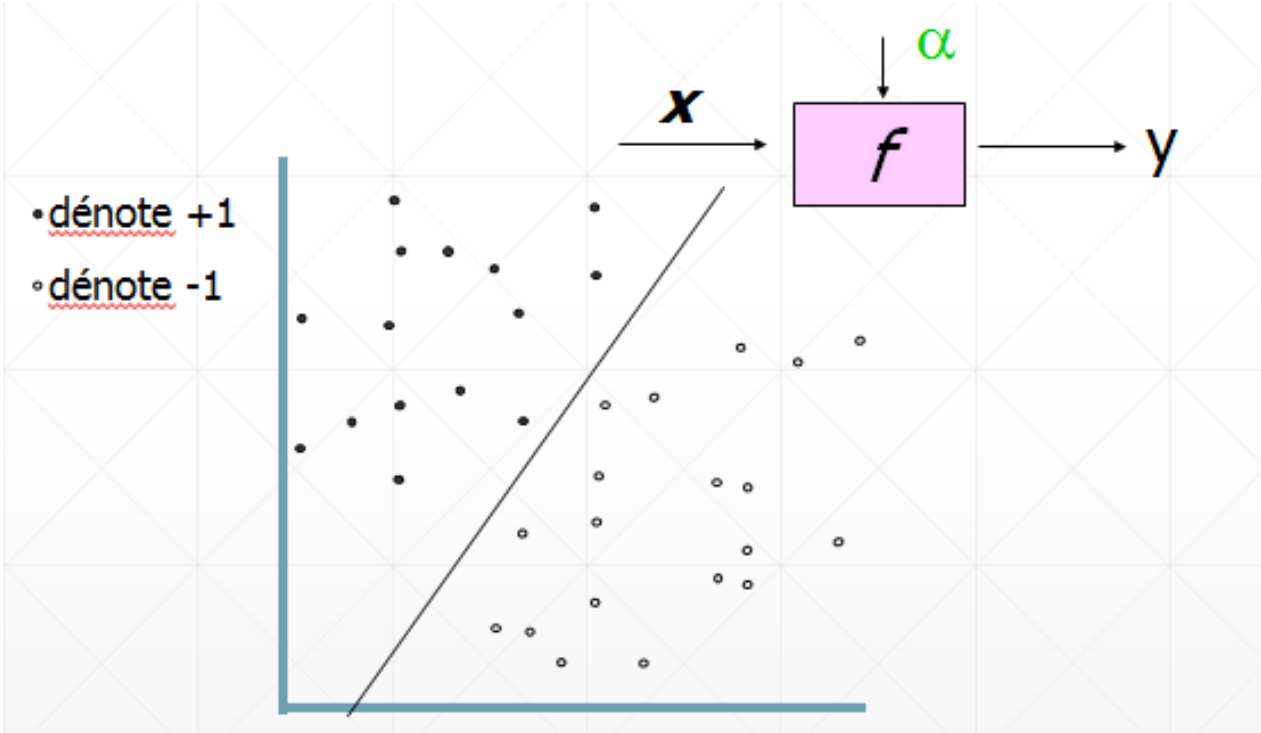


Séparation linéaire par SVM

Classificateurs Linéaires

Comment classeriez-vous ces données ?

$f(x,w,a) = \text{sign}(w^T x + a)$

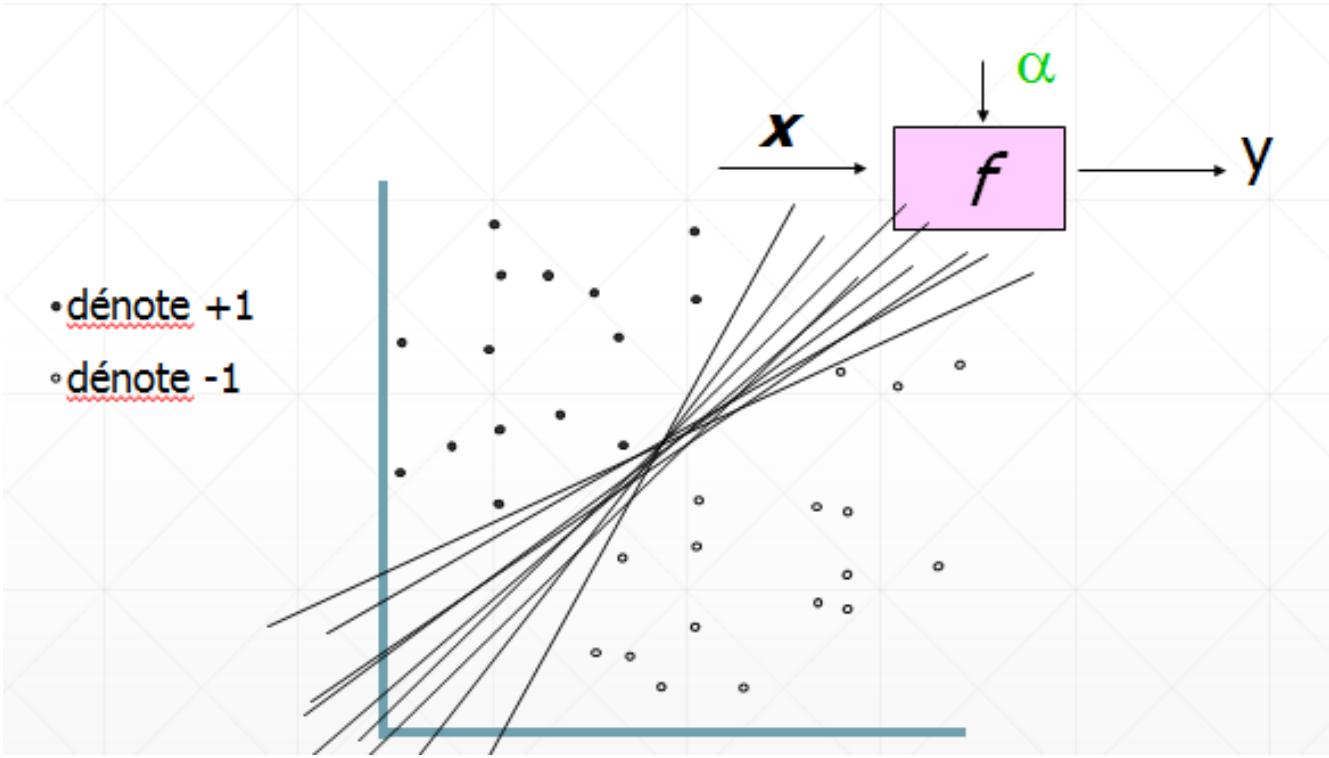


Séparation linéaire par SVM

Classificateurs Linéaires

N'importe lequel de ces serait bien .. ..mais qui est le meilleur ?

$f(x,w,a) = \text{sign}(w^T x + a)$

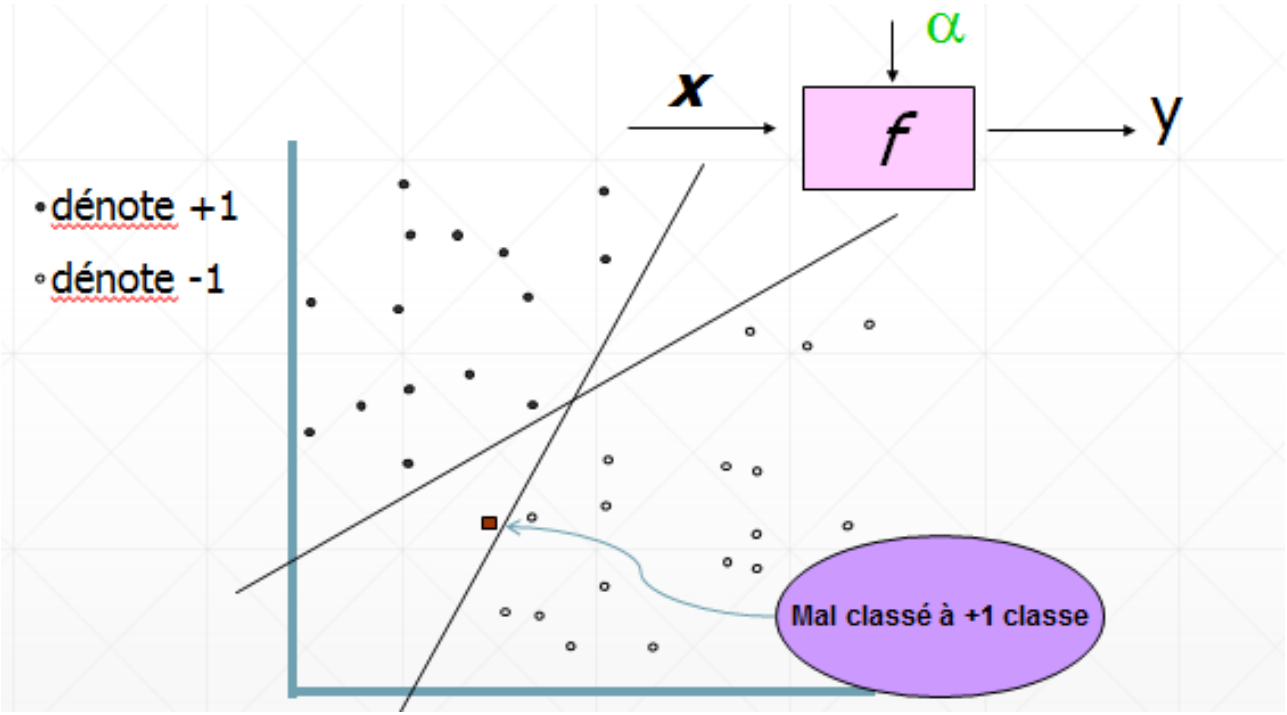


Séparation linéaire par SVM

Classificateurs Linéaires

Comment classeriez-vous ces données ?

$f(x,w,a) = \text{sign}(w^T x + a)$



Séparation linéaire par SVM

Un problème de discrimination est dit linéairement séparable lorsqu'il existe une fonction de décision linéaire (appelé aussi séparateur linéaire), de la forme :

$$D(x) = \text{signe}(f(x)) \text{ avec } f(x) = w^T x + a$$

$w \in \mathbb{R}^p$, classant correctement toutes les observations de l'ensemble d'apprentissage $D(x) = y_i, i \in [1, n]$, La fonction f est appelée fonction caractéristique.

A toute fonction de décision et donc aux fonction de décision linéaire ont peut associer une frontière de décision :

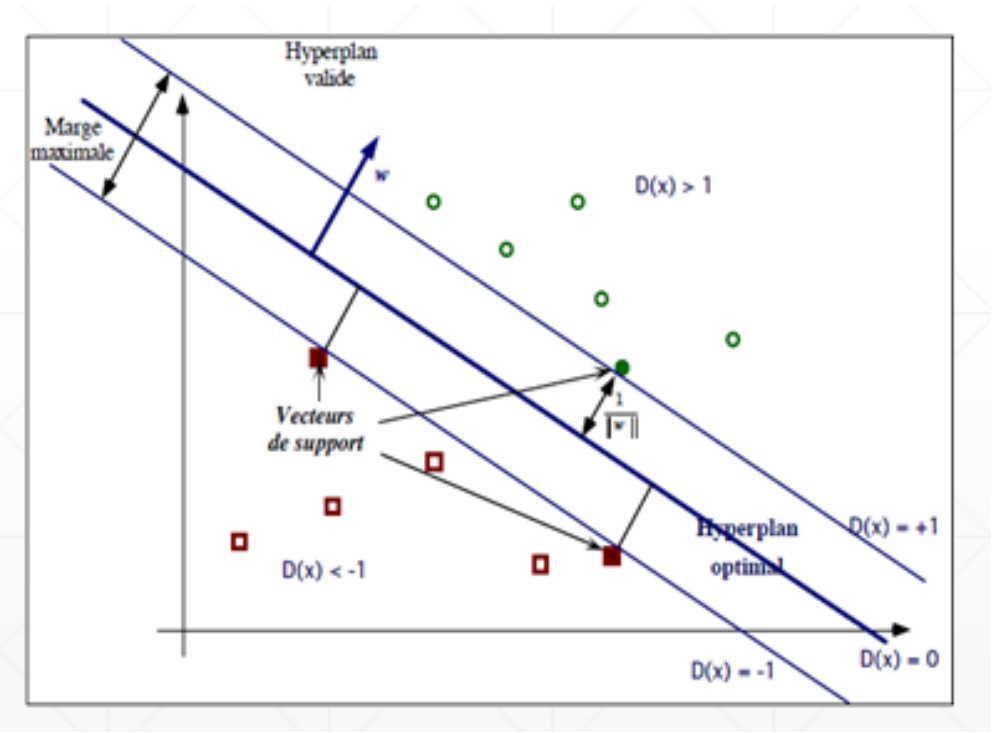
$$\Delta(w, a) = \{ x \in \mathbb{R}^p \mid w^T x + a = 0 \}$$

Séparation linéaire par SVM

Classificateurs Linéaires

Hyperplan optimal doit maximiser la distance entre la frontière de séparation et les points de chaque classe qui lui sont le plus proche .

Recherche d'un hyperplan de séparation optimal au sens de la marge maximale.



Séparation linéaire par SVM

Principe de la maximisation de la marge

- La distance d'un point à l'hyperplan est :

$$d(x, \Delta) = \frac{|w^T x + a|}{||w||} = \textit{Marge}.$$

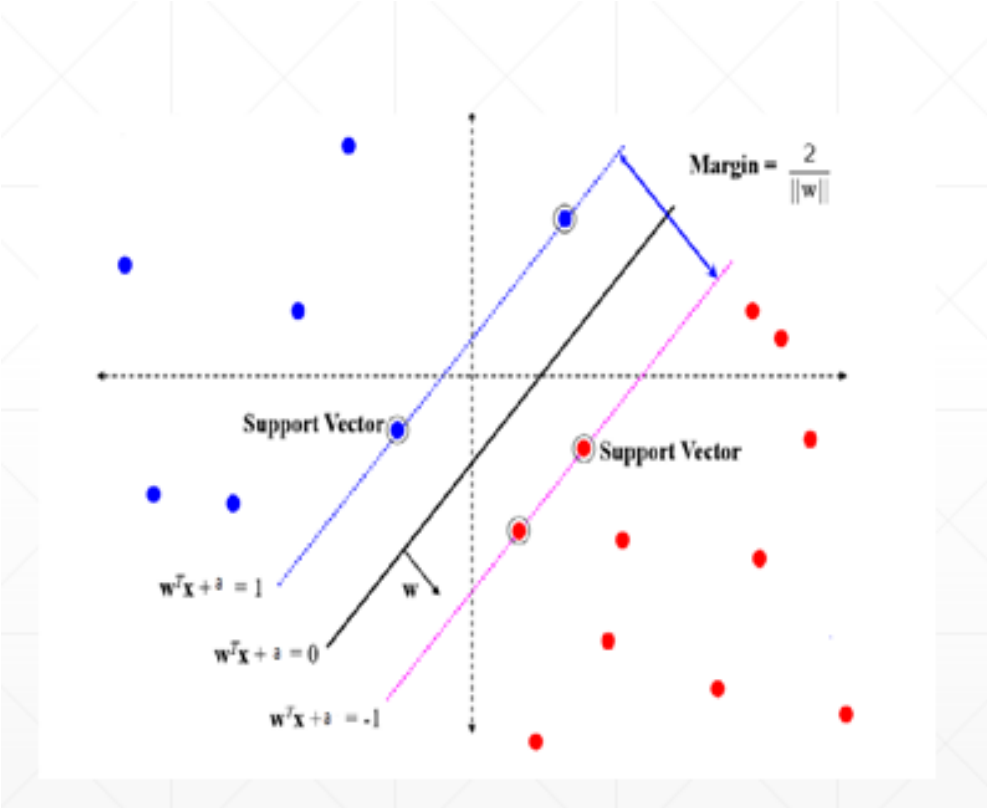
- L'hyperplan optimal est celui pour lequel la distance aux points les plus proches est maximale.
- La marge entre les deux classes vaut $\frac{2}{||w||}$
- Maximiser la marge revient donc à minimiser ||w|| sous contraintes :

$$\begin{cases} \min\limits_{w,a} \frac{1}{2}||w||^2 \\ \text{avec} \qquad y_i(w^T x_i + a) \geq 1, \forall i \in \{1, \cdots, n\} \end{cases}$$

Séparation linéaire par SVM

Principe de la maximisation de la marge

Les données qui vérifient l'égalité de la contrainte s'appellent les vecteurs supports, et ce sont ces données seules qui contribuent à la détermination de l'hyperplan.



Séparation linéaire par SVM

Résolution de la forme primaire du problème

Le problème de l'équation est un problème de programmation quadratique avec contraintes linéaires. Dans ce problème, les variables sont w et a, c-à-d que le nombre de variables est égal à p + 1.

- Il faut régler p + 1 paramètres.
- Possible quand p est assez petit avec des méthodes d'optimisation quadratique.
- Impossible quand p est grand.

Transformation du problème d'optimisation

- Méthode des multiplicateurs de Lagrange :

$$\begin{cases} L_p(w, a, \alpha) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} \alpha_i [y_i * (w^T x_i + a) - 1] \\ \forall i, \alpha_i \geq 0 \end{cases}$$

Où α_i sont les multiplicateurs de Lagrange

Séparation linéaire par SVM

Expression duale Optimisation

En introduisant les informations issues de l'annulation des dérivées partielles du Lagrangien, on obtient une optimisation ne dépendant que des multiplicateurs En remplaçant dans la fonction objective, on obtient le problème dual à maximiser suivant :

$$\max_{\alpha} L_p(\alpha) = \sum_{i=1}^{n} \sum_{j}^{n} \alpha_i \alpha_j y_i y_j < x_i, x_j >$$

s.c

$$\begin{cases} \alpha_i \geq 0, \forall i \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases}$$

d'ou

- $\alpha_i \geq 0$ vont définir les points importants c.à-d. les points supports
- Forcément, il y aura des points supports d'étiquettes différentes sinon cette condition ne peut pas être respectée.
- $< x_i, x_j >= x_i^T x_j$: est le produit scalaire entre les observations i et j

Séparation linéaire par SVM

Solution du problème d'optimisation

- $f(x) = \widehat{w}^T x + \widehat{a} \quad = \sum_{i=1}^{n} \alpha_i y_i < x_i, x > + a \quad = \sum_{j \in S} \alpha_j y_j < x_j, x > + a$
- $\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \Rightarrow \widehat{w} \sum_{i=1}^{n} \alpha_i' y_i x_i$
- A partir des conditions de KKT,on peut obtenir : A partir des conditions de KKT, on peut obtenir :

$$\widehat{a} = \frac{1 - y_i * w^T x_i}{y_i}$$

- D'un point de vue précision, on prend la moyenne de a pour tous les vecteurs supports :

$$\widehat{a} = \frac{1}{||S||} \sum_{i \in S} y_i - w^T x_i$$

- La fonction de décision f peut être calculée : $f(x) = \sum_{i=1}^{n} \alpha_i' y_i < x_i, x > + \widehat{a}$

Séparation linéaire par SVM

Classement d'un individu supplémentaire

La fonction de décision f peut être calculée, donc, pour chaque nouvel point x par la fonction f(x) et la décision peut être prise comme suit :

$$\begin{cases} x \in \textit{classe} + 1 & \textit{si} \quad f(x) > 0 \\ x \in \textit{classe} - 1 & \textit{si} \quad f(x) < 0 \\ x \ \textit{inclassifiable} & \textit{si} \quad f(x) = 0 \end{cases}$$

- seuls les α_i des points les plus proches sont non-nuls : points de support
- seuls interviennent les produits scalaires entre les observations x dans le problème d'optimisation.

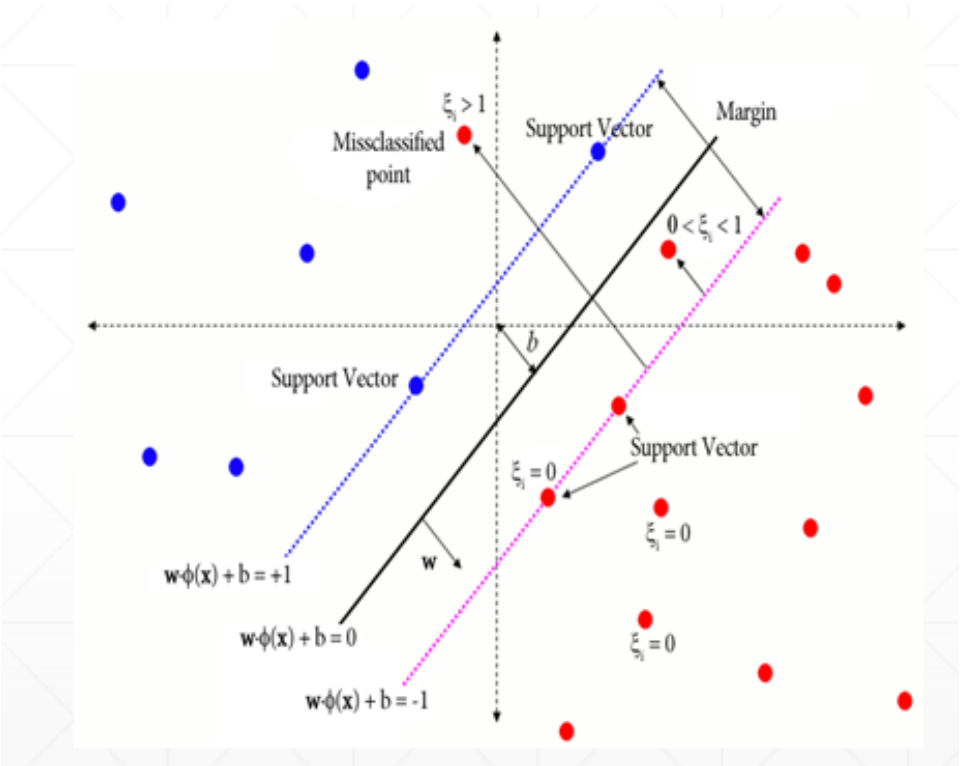
L'hyperplan solution ne dépend que du produit scalaire entre le vecteur d'entrée et les vecteurs de support. Cette particularité permet l'utilisation de fonctions noyau pour aborder des problèmes non linéaires

Séparation non linéaire par SVM

Le cas non séparable

Souvent il arrive que même si le problème est linéaire, les données sont affectées par un bruit (par ex. de capteur) et les deux classes se retrouvent mélangées autour de l'hyperplan de séparation. Pour gérer ce type de problème on utilise une technique dite de marge souple, qui tolère les mauvais classements :

- Rajouter des variables de relâchement des contraintes ξ_i.
- Pénaliser ces relâchements dans la fonction objectif.

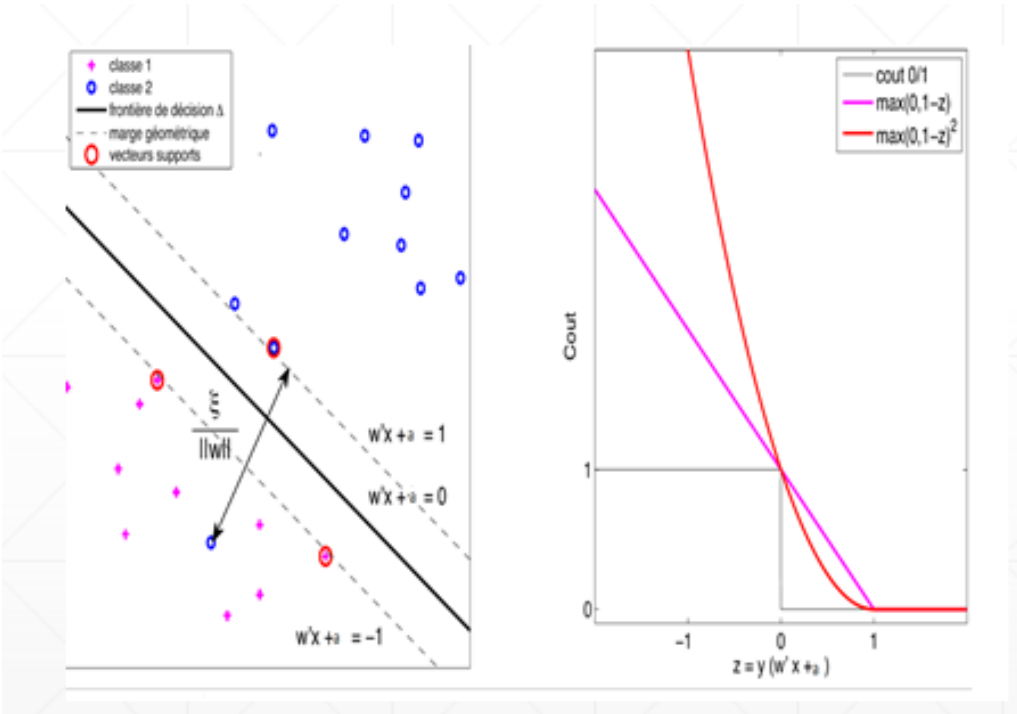


Séparation non linéaire par SVM

Le cas non séparable

Si un point (x_i, y_i) vérifie la contrainte de marge $y_i * (w^T x_i + a) \geq 1$ alors la variable d'écart (qui est une mesure du coût de l'erreur) est nulle.Nous avons donc deux situations :

- Pas d'erreur :
  $y_i * (w^T x_i + a) \geq 1 \Rightarrow \xi_i = 0$
  .
- Erreur :
  $y_i * (w^T x_i + a) 1 - \xi_i \Rightarrow \xi_i =$
  $1 - y_i * (w^T x_i + a)$.
- $\xi_i = max(0, 1 - y_i * (w^T x_i + a)$.



Séparation non linéaire par SVM

Le cas non séparable

Le problème d'optimisation dans le cas des données non-séparables est donc :

$$\begin{cases} min\frac{1}{2}||w||^2 \\ \sum_{i=1}^{n} \xi_i \end{cases}$$

Tel que :

$$\begin{cases} y_i * (w^T x_i + a) \geq 1 - \xi_i \\ \xi_i \geq 0 \ \forall i \in \{1, \cdots, n\} \end{cases}$$

Puisqu'il faut minimiser les deux termes simultanément, on introduit une variable d'équilibrage C>0 qui permet d'avoir une seule fonction objectif dans le problème d'optimisation :

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

Tel que :

$$\begin{cases} y_i * (w^T x_i + a) \geq 1 - \xi_i \\ \xi_i \geq 0 \quad \forall i \in \{1, \cdots, n\} \end{cases}$$

Séparation non linéaire par SVM

Le cas non séparable

Par la même procédure qu'avant, on obtient le problème dual :

$$\max_{\alpha} L\alpha_p(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j < x_i, x_j >$$

s.c :

$$\begin{cases} \sum_{i=1}^{n} \alpha_i y_i = 0 & \text{(stationarité)} \\ 0 \leq \alpha_i \leq C, \forall i \in \{1, \cdots, n\} & \text{(admissibilitéduale)} \end{cases}$$

La fonction de décision permettant de classer une nouvelle observation x est toujours :

$$f^*(x) = \sum_{i=1}^{n} \alpha_i^* y_i < x_i, x > + a^*$$

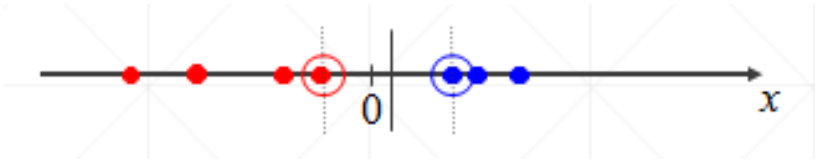
Observations :

- La différence pour le problème dual entre le cas séparable et non séparable est que les valeurs des α_i sont majorées par C.
- Les points mal classés ou placés dans la marge ont un α_i=C.
- a est calculé de sorte que $y_i * f(x_i)$=1 pour les points tels que C>α_i>0

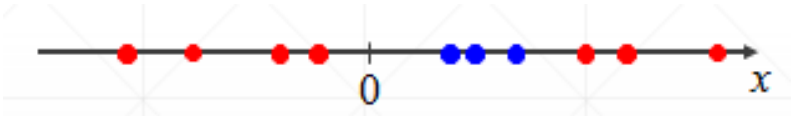
Séparation non linéaire par SVM

Le cas non linéaire : les noyaux

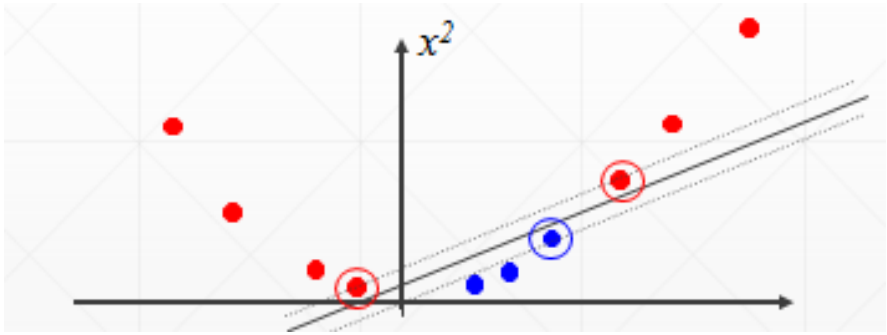
- Les jeux de données qui peuvent être séparés linéairement avec du bruit sont parfaits :



- Mais qu'allons-nous faire si l'ensemble de données est trop difficile ?



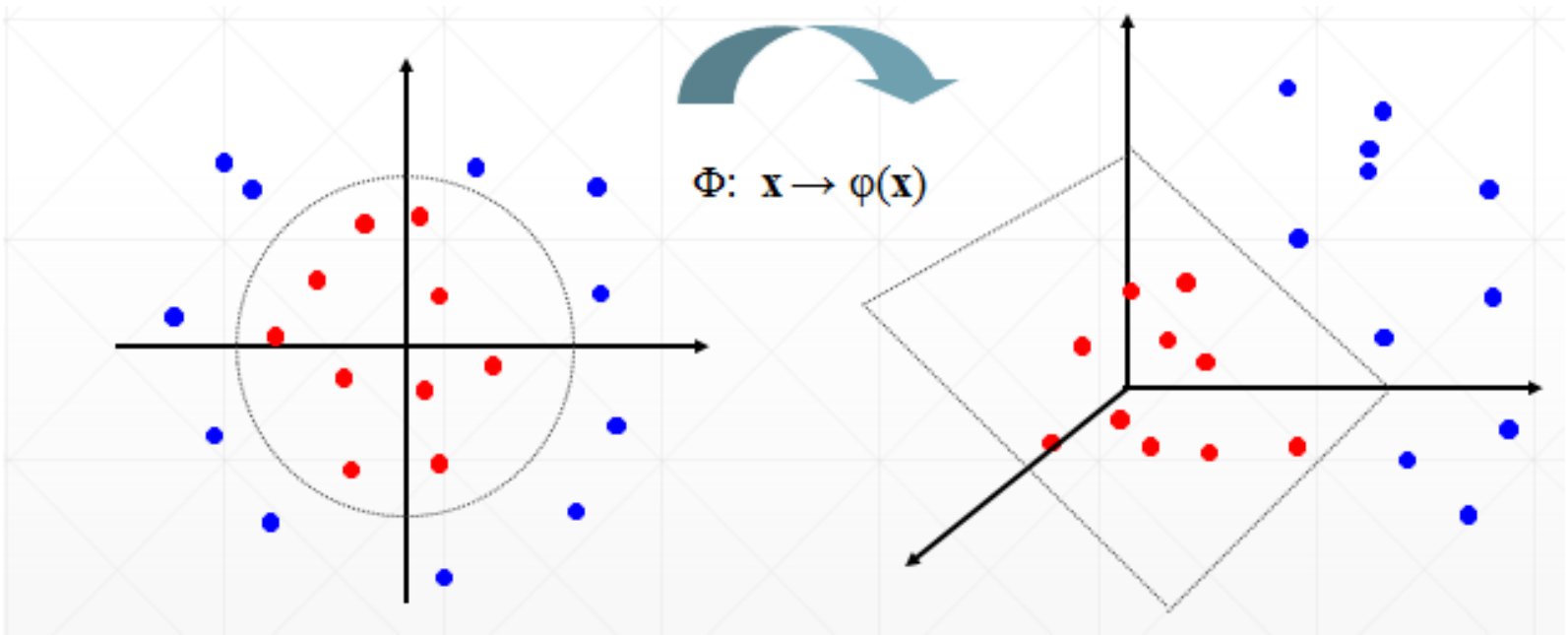
- Que diriez-vous de. . . mapper les données dans un espace de plus grande dimension :



Séparation non linéaire par SVM

Changement de représentation

En réalisant les transformations de variables adéquates, on peut rendre linéairement séparable un problème qui ne l'est pas dans l'espace initial.



Séparation non linéaire par SVM

Dans le cas général, la frontière optimale est non linéaire. Dans le cadre des SVM, la prise en compte de non linéarités dans le modèle s'effectue par l'introduction de noyaux non linéaires.Des exemples typiques de noyaux sont le noyaux polynômial $k(x, x') = (C + (x, x'))^p$ et le noyau gaussien $k(x, x') = e^{\frac{||x-x'||^2}{2\sigma^2}}$.
Etant donné le noyau k, la fonction de décision s'écrit D(x) = signe(f(x) + a)et :

- Cas linéaire : $f(x) = \sum_{i \in S} \alpha_i y_i x_i^T x$
- Cas non linéaire : $f(x) = \sum_{i \in S} \alpha_i y_i k(x_i, x)$

La fonction de discrimination est une combinaison linéaire des noyaux dont le signe de l'influence dépend de la classe. L'ensemble actif S, les coefficients α_i associés et le biais a sont donné par la résolution du même problème dual que dans le cas des SVM linéaires . Seule la définition de la matrice G change (mais pas sa taille) :

- Cas linéaire : $G_{ij} = y_i y_j x_i^T x_j$
- Cas non linéaire : $G_{ij} = y_i y_j k(x_i, x_j)$

Séparation non linéaire par SVM

L'idée ici est de considérer dans le dual l'influence de chacune des observations dans la construction de la solution. Le calcul de cette influence passe par la résolution d'un programme quadratique de taille n. L'utilisation de noyaux permet d'introduire de la non linéarité sans modifier la complexité algorithmique du problème à résoudre

- les noyaux : un noyau k est défini de manière générale comme une fonction de deux variables sur $\mathbb{R}$:
$$k : X \times X \to \mathbb{R} \qquad (x, x') \mapsto k(x, x')$$
- Matrice de Gram : La matrice de Gram du noyau k(., .) pour les observations (x_1,x_2,. . . ,x_n)est la matrice carrée K de taille n et de terme général $k_{ij} = k(x_i, x_j)$
- Noyau positif : Un noyau k est dit positif si, pour tout entier n fini et pour toutes les suite de n observations possibles x_i ,i = 1,n, la matrice de Gram associée est une matrice symétrique définie positive. L'intéret des noyaux positifs c'est qu'il est possible de leur associer un produit scalaire.
- Construction des noyaux : Il existe deux façons de construire un noyau positif :

Séparation non linéaire par SVM

1. soit on s'appuie sur une transformation $\Phi(x)$ de X sur un espace H muni d'un produit scalaire et l'on défini le noyau à travers ce produit scalaire :

$$k(x, x') =< \Phi(x), \Phi(x') >_H$$

2. soit on utilise les propriétés algébriques des noyaux positifs :
   - un noyau séparable est un noyau positif
   - la somme de deux noyaux positifs est un noyau positif,
   - le produit de deux noyaux positifs est un noyau positif
   - le produit tensoriel ainsi la somme directe de deux noyaux positifs est un noyau positif
   - le passage à la limite conserve la positivité : si la limite d'une suite de noyaux positif existe, c'est aussi un noyau positif.

Séparation non linéaire par SVM

- Noyaux polynomiaux :

  $k_{poly1}(x, z) = (x^T z)^d$ Tous les produits d'exactement d variables.

  $k_{poly2}(x, z) = (x^T z + c)^d$ Tous les produits d'au plus d variables.

- Noyaux gaussiens :

  $k_G(x, z) = exp(-\frac{d(x,z)^1}{2\sigma^2})$ Sorte de décomposition en série de Fourrier.

- Noyaux sigmoïdes :

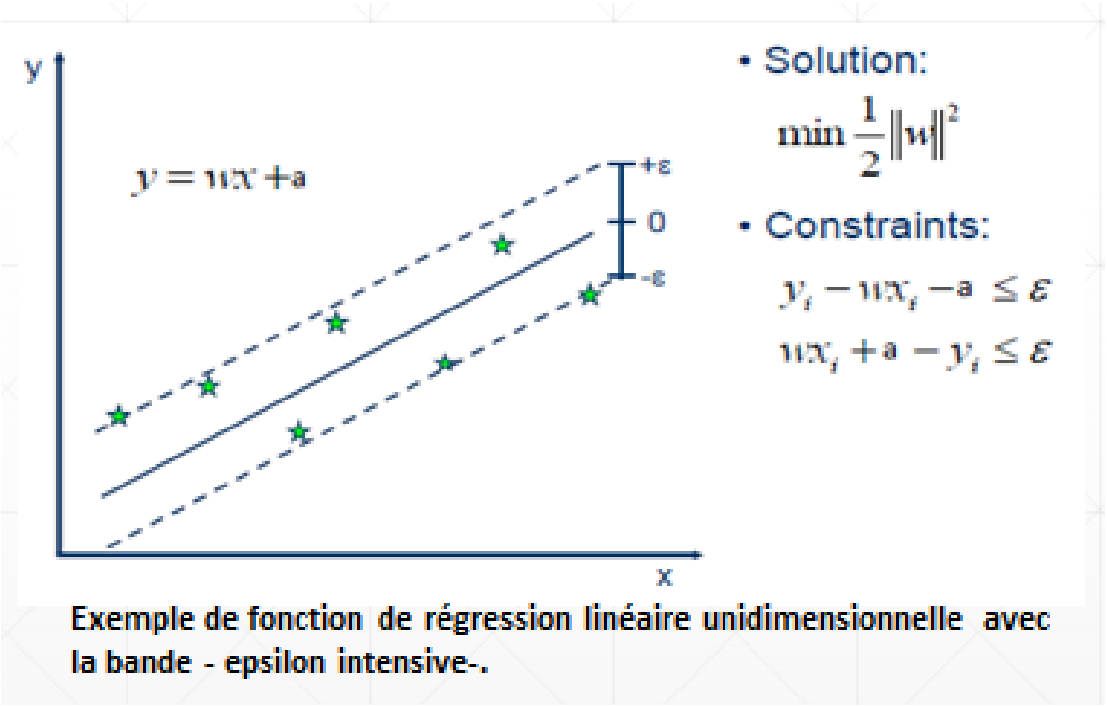
  $k(x, z) = tanh(kx^T z + \theta$ Pas définie positive.Mais fonction de décision proche des réseaux connexionnistes.

support vector regression

De la même manière que l'approche de classification, il existe une motivation pour rechercher et optimiser les limites de généralisation données pour la régression. Ils se sont appuyés sur la définition de la fonction de perte qui ignore les erreurs, situées à une certaine distance de la valeur réelle. Ce type de fonction est souvent appelé - fonction de perte intensive - epsilon.

Utilisation de la fonction de perte intensive epsilon, nous assurons l'existence du minimum global et en même temps l'optimisation avec support de généralisation fiable.

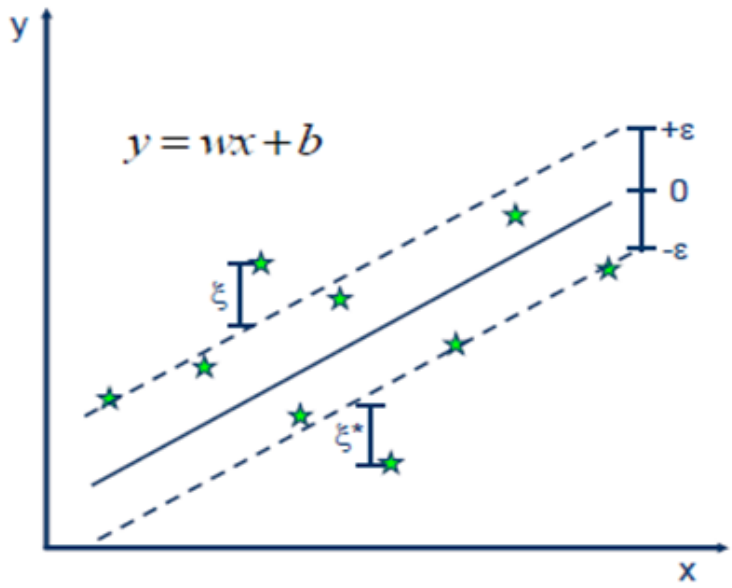
Le paramètre ϵ contrôle la largeur de la zone-insensitive, utilisé pour ajuster les données de formation.



Exemple de fonction de régression linéaire unidimensionnelle avec la bande - epsilon intensive-.

support vector regression

SVR linéaire :



Minimiser:

$$\frac{1}{2}||w||^2 + C\sum_{i=1}^{n}(\xi_i + \check{\xi}_i)$$

Le paramètre pour contrôler le problème d'over-fitting.

Contraintes:

$$y_i - w^T x_i + a \leq \xi_i + \varepsilon$$

$$w^T x_i + a - y_i \leq \check{\xi}_i + \varepsilon$$

$$\check{\xi}_i, \xi_i \geq 0$$

Variables d'écart

❖ **SVR linéaire**

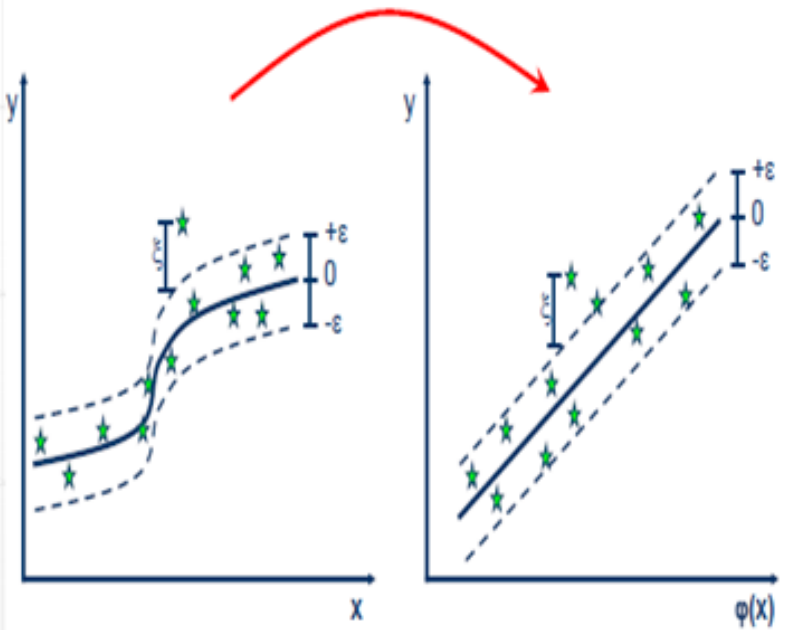
$$Y = \sum_{i=1}^{n}(\alpha_i - \grave{\alpha}_i) < x_i, x > + a$$

Biais

support vector regression

SVR non linéaire :

Les fonctions du noyau transforment les données en un espace de fonctions de dimension supérieure pour permettre la séparation linéaire.



$$Y=\sum_{i=1}^{n}(\alpha_i-\grave{\alpha}_i)<\Phi(x_i),\Phi(x_j)>+a$$

$$=\sum_{i=1}^{n}(\alpha_i-\grave{\alpha}_i)k(x_i,x)+a$$

- **Fonctions du noyau**

  polynomial:

  $$k(x_i,x_j)=(x_i.x_j)^p$$

  Fonction de base radiale gaussienne:

  $$k(x_i,x_j)=\exp\left(-\frac{||x_i-x_j||^2}{2\sigma^2}\right)$$

Simulation

- pandas : Pour importer les données ‘csv’ et les manipuler
- numpy : Pour les calculs matricielle
- matplotlib.pyplot : Pour l’affichage des figures
- Scikit-learn (sklearn) : est une bibliothèque libre dans Python destinée à l’apprentissage automatique.

Simulation

Les données ont été extraites des images qui ont été prises pour l'évaluation d'une procédure d'authentification des billets de banque. Wavelet Transform tool.

5 variables et 1372 individus.

- Variance de l'image transformée en ondelettes
- Asymétrie de l'image transformée en ondelettes
- Curtose de l'image transformée en ondelettes
- Entropie d'image
- Classe

Simulation

Conclusion

Dans cette présentation, nous avons couvert de nombreux sujets sur l'algorithme de machine à vecteurs de support, son fonctionnement, les noyaux, le réglage de l'hyperparamètre, la construction de modèles et l'évaluation de jeux de données de billets de banque à l'aide du package Scikit-learn.

Les avantages :

- SVM fonctionne bien avec une marge de séparation nette et un espace dimensionnel élevé.
- Toujours efficace dans les cas où le nombre de dimensions est supérieur au nombre d'échantillons.
- Utilise un sous-ensemble de points d'entraînement dans la fonction de décision (appelés vecteurs de support), ce qui en fait une mémoire efficace.

Les inconvénients :

- SVM ne convient pas pour les grands ensembles de données en raison de son temps de formation et il prend plus de temps dans la formation.
- SVM fonctionne mal avec des classes qui se chevauchent et est également sensible au type de noyau utilisé.

Réferences

- http : //www-lium.univ-lemans.fr/ barrault/cours/m1-aan/m1-aan-2016-svm.pdf.
- http ://eric.univ-lyon2.fr/ ricco/cours/slides/svm.pdf.
- http ://www.math.univ-toulouse.fr/ besse/Wikistat/pdf/st-m-app-svm.pdf.
- http ://pageperso.univ-lr.fr/arnaud.revel/MesPolys/SVM.pdf.
- https ://www.quora.com/What-is-the-difference-between-Support-Vector-Machine-and-Support-Vector-Regression
- https ://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python
- http ://www-igm.univ-mlv.fr/ dr/XPOSE2014/Machin-Learning/D-Machine-Learning.html
- https ://dataanalyticspost.com/Lexique/svm/