Fouille de Données

Data Mining

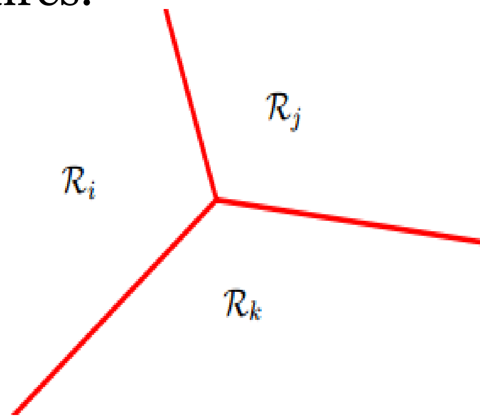
Classification - Partie 5

Plan du cours

1. Les Machines à Vecteur Support - SVM.
2. SVM linéaire.
   - Cas séparable.
   - Cas non séparable
3. Implémentation des SVM.

Machines à Vecteur Support

- Méthode de classification de données linéaires et non linéaires.
- Introduite par Vladimir Vapnik au début des années 90.
- Machine => Algorithme.
- Principe de base :



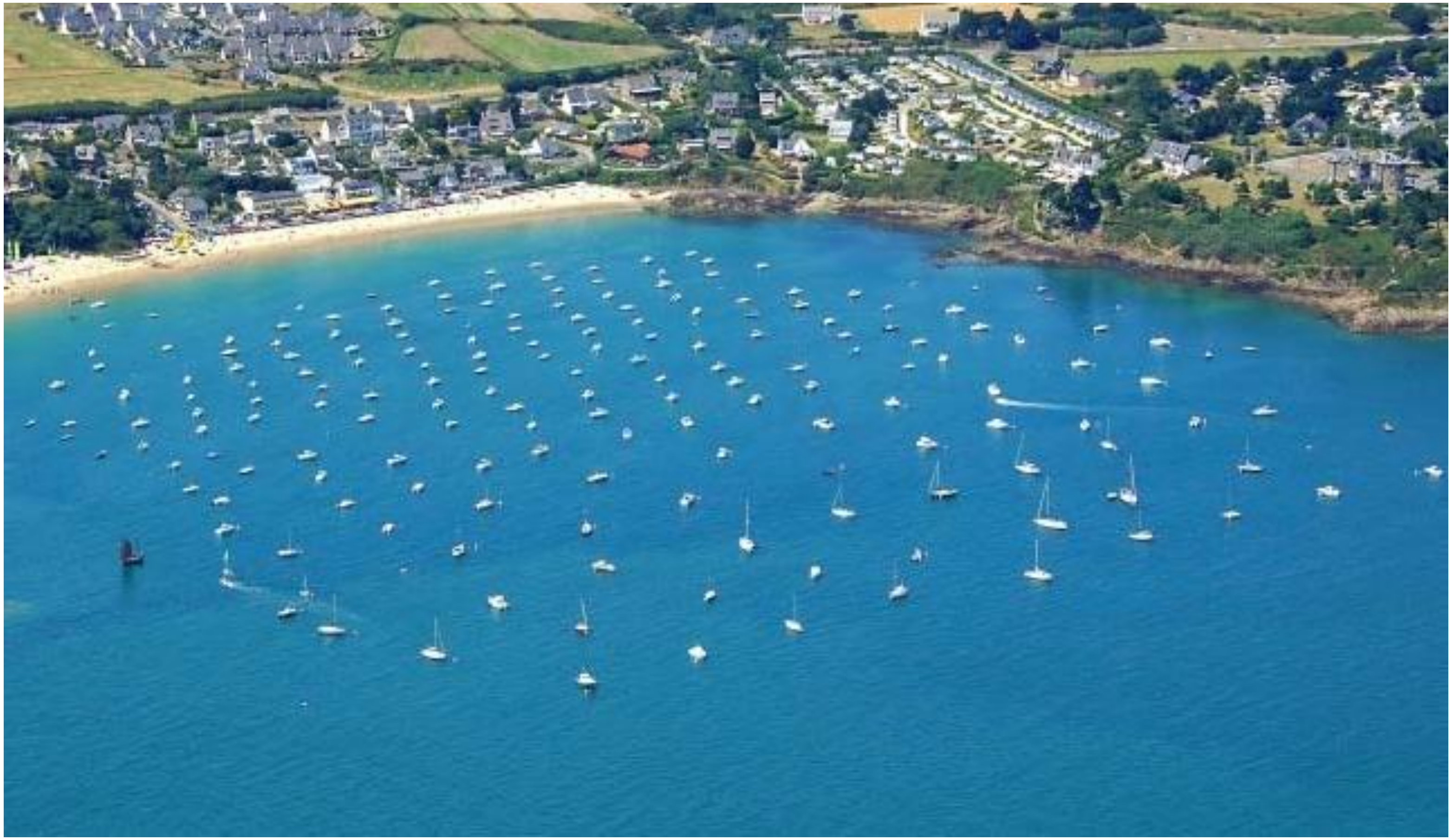
1. Rechercher un **hyperplan** qui sépare le mieux l'espace des exemples en différentes **régions de décisions**.

2. Une **frontière de décision** séparant les exemples d'une classe à une autre. Chaque région est associée à une classe.

3. Une recherche qui s'appuie uniquement sur quelques exemples de la base d'entrainement, appelés **Vecteurs Support**, ainsi que la **marge** entre eux.

Machines à Vecteur Support

Principe de base:

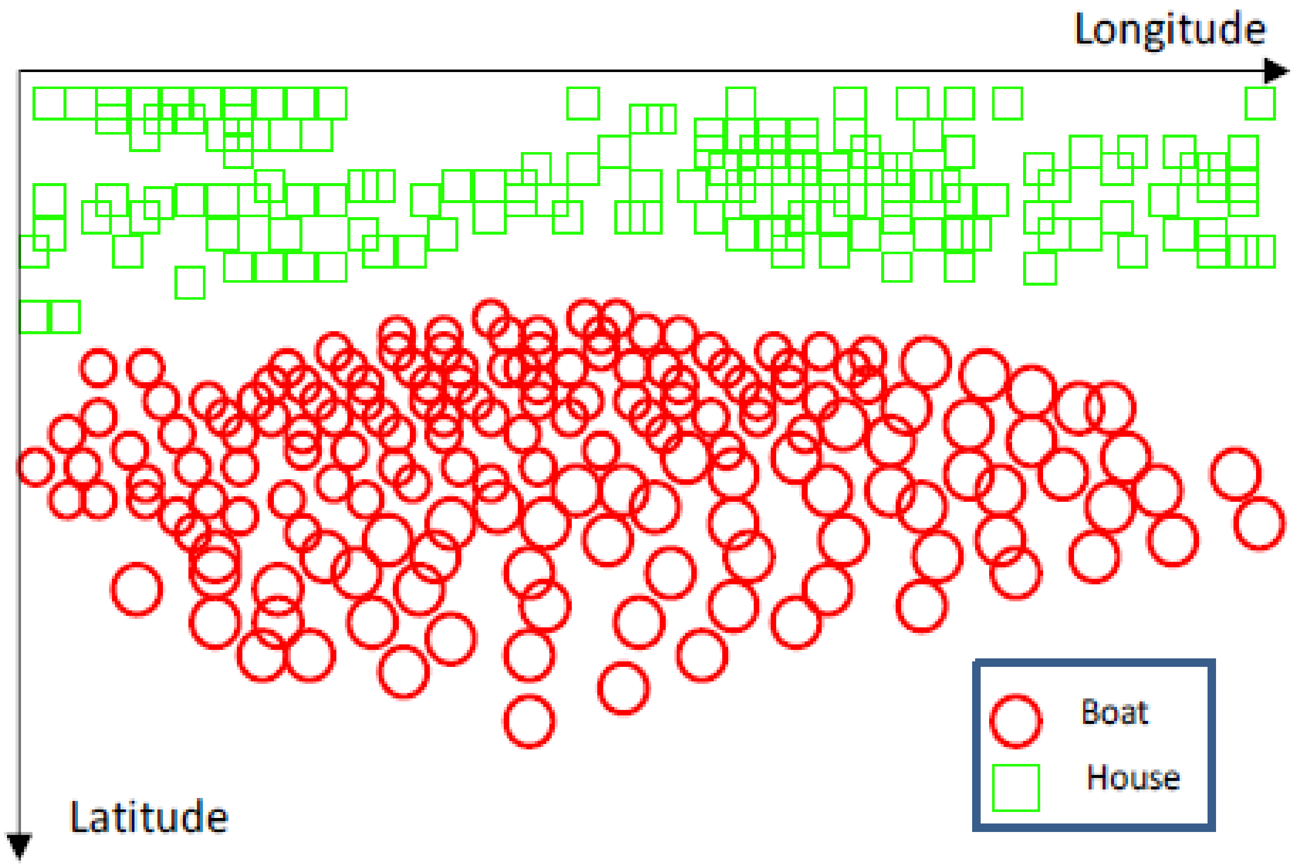


Machines à Vecteur Support

Principe de base:

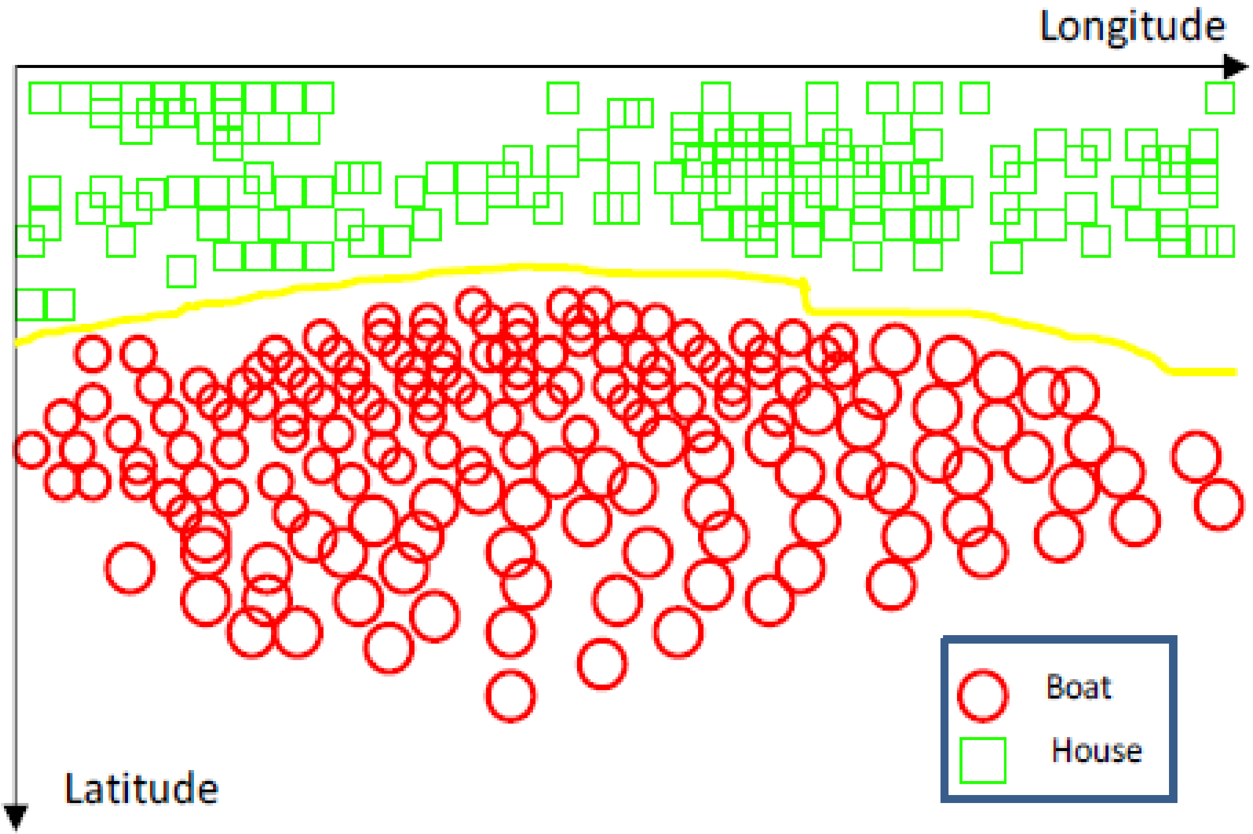
Machines à Vecteur Support

Principe de base:



Machines à Vecteur Support

Principe de base:

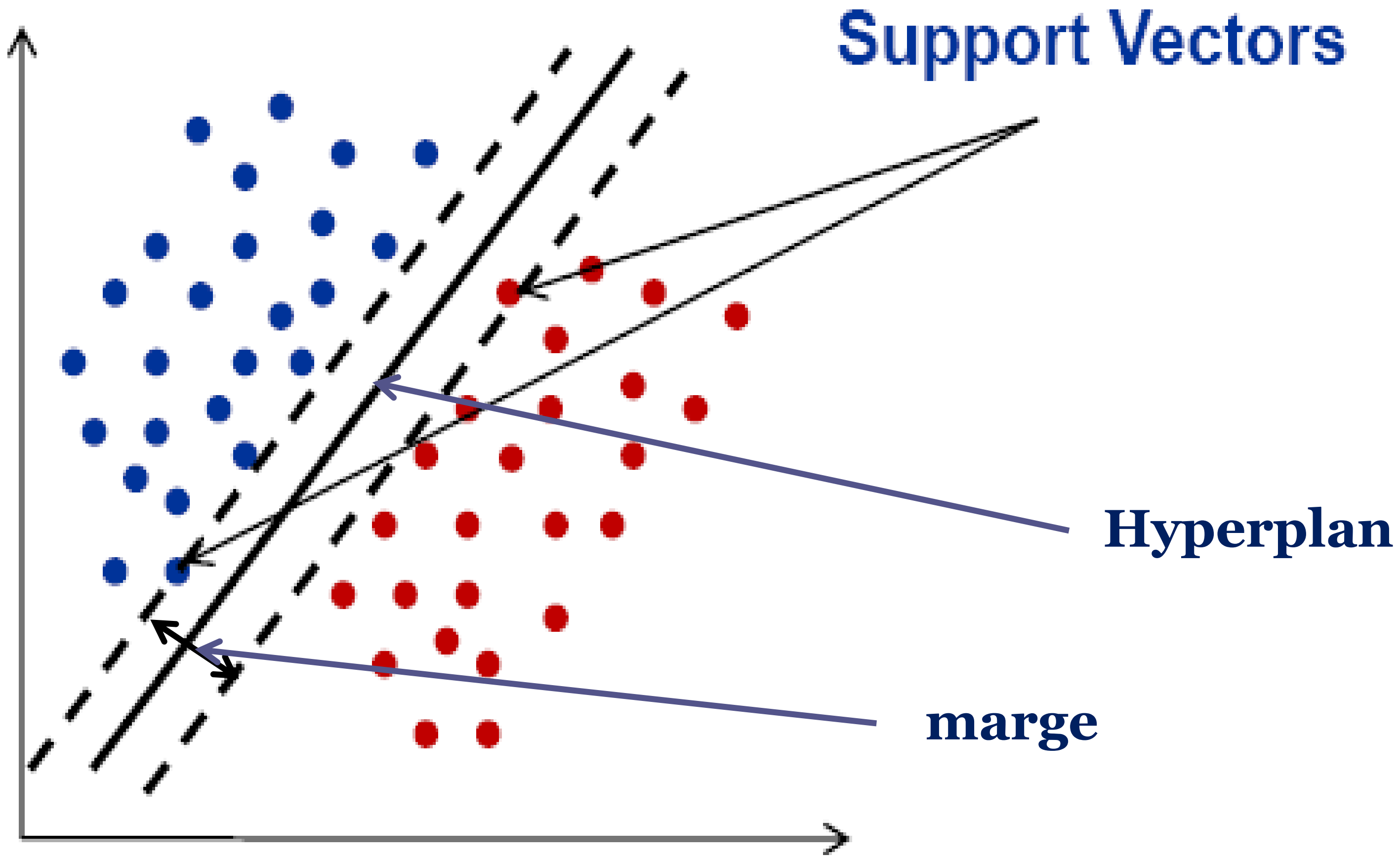


frontière de décision

Machines à Vecteur Support

C’est-à-dire ?

Cas d’une classification binaire à deux attributs => Plan 2D / Exemple = vecteur

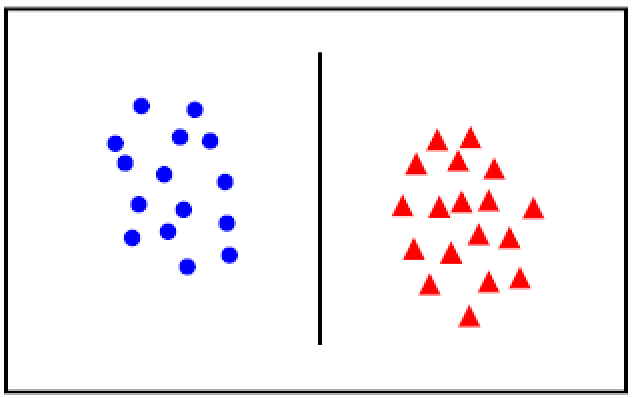


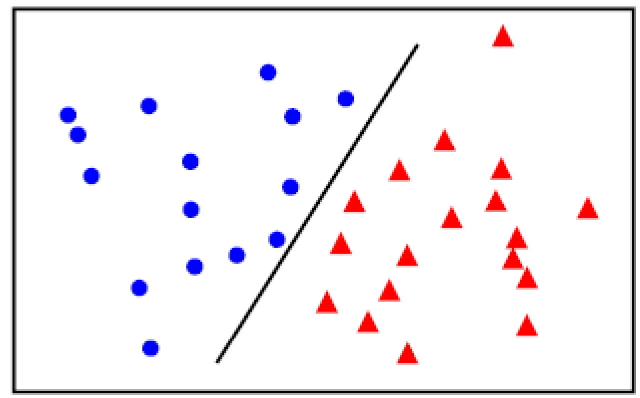
SVM Linéaire

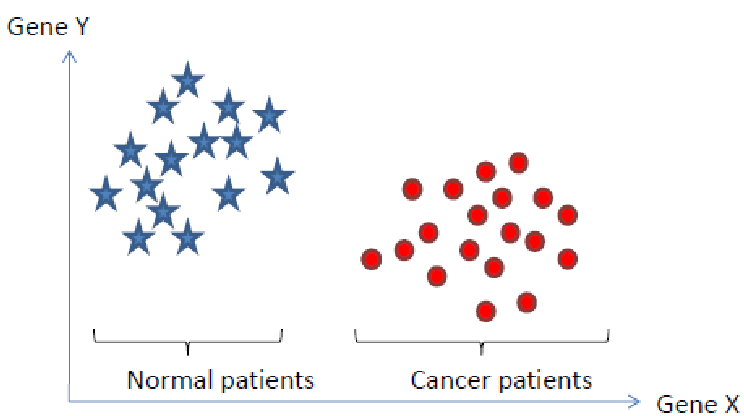
Deux cas du problème posé :

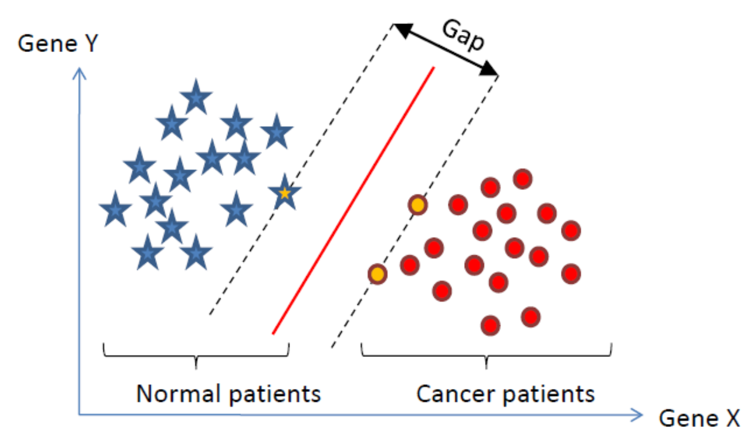
- ✓ Séparable linéairement.
- ✓ Non séparable linéairement.

On dit qu'un problème est linéairement séparable si une surface linéaire permet de classifier parfaitement.







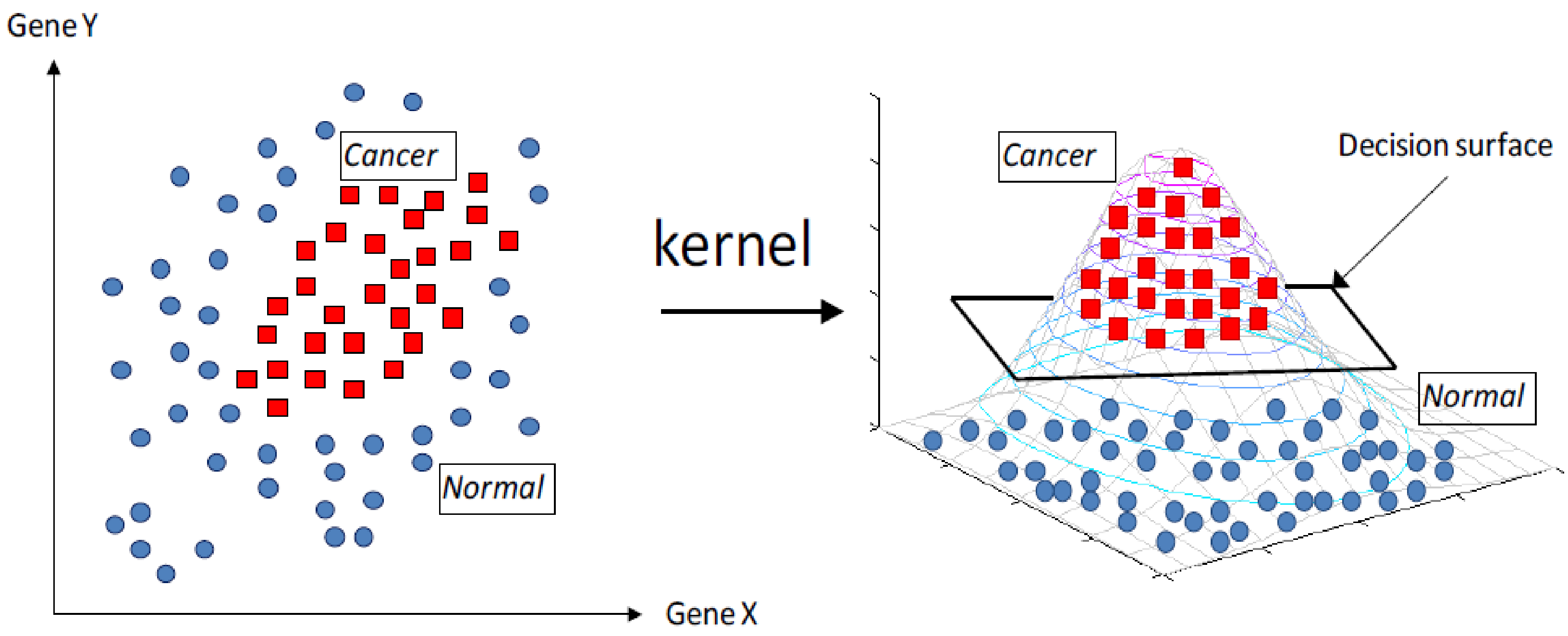


SVM Linéaire

Deux cas du problème posé :

- ✓ Séparable linéairement.
- ✓ Non séparable linéairement.
    - **Kernel Trick**

On dit qu'un problème est linéairement séparable si une surface linéaire permet de classifier parfaitement.



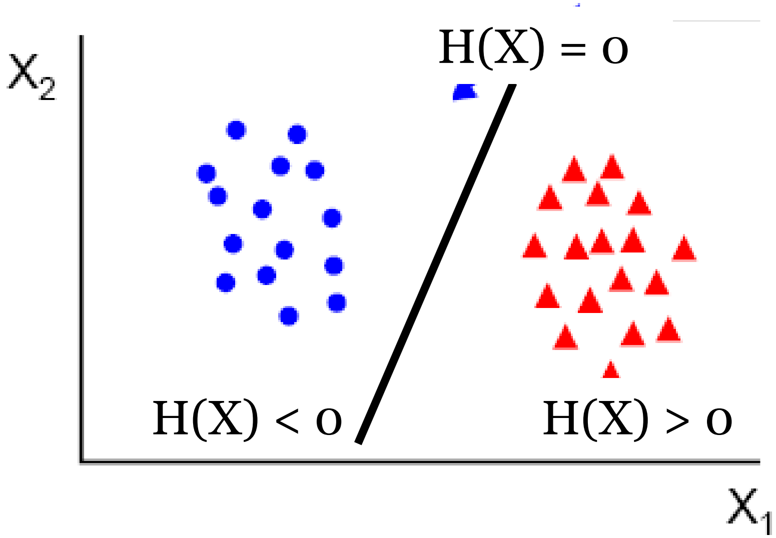
SVM Linéaire

Cas de séparabilité linéaire : Linear SVM

- Supposons deux classes :
  - Yi = {+1, -1}
- Chaque exemple **X** est décrit par des attributs.
- Chaque exemple est un vecteur.
- L'hyperplan séparateur (la droite) a cette forme :

$$H(X) = W^T X + b$$

W: Vecteur de poids
b : biais



SVM Linéaire

Problème : rechercher une **fonction discriminante H(X)** qui prend X en entrée et donne sa classe Ck en sortie.

<u>Cas de séparabilité linéaire</u>

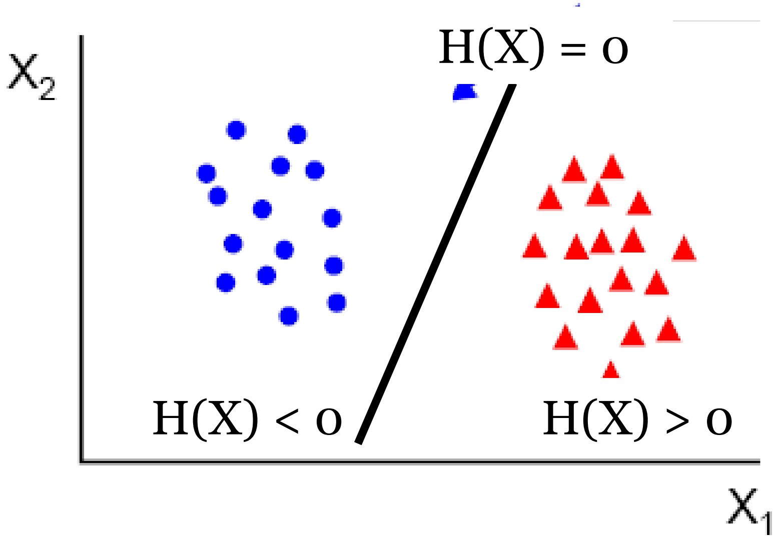
- Supposons deux classes :
  - Yi = {+1, -1}

=> Rechercher les meilleures valeurs de **W** et de **b**.

- Chaque exemple **X** est décrit par des attributs.
- Chaque exemple est un vecteur.
- L'hyperplan séparateur (la droite) a cette forme :

$$H(X) = W^T X + b$$

W: Vecteur de poids
b : biais



SVM Linéaire

Problème : rechercher une **fonction discriminante** qui prend X en entrée et donne sa classe Ck en sortie.

Cas de séparabilité linéaire

$$H(X) = W^T X + b$$

$$\begin{cases} Ck = +1 \quad \text{Si } H(X) > 0 \quad : \quad H+ \\ Ck = -1 \quad \text{Si } H(X) < 0 \quad : \quad H- \end{cases}$$

Déterminer la classe selon le signe(H(x))

- Aucun exemple ne se situe sur l'hyperplan, donc :

$$\begin{cases} Ck = +1 \quad \text{Si } H(X) >= 1 \\ Ck = -1 \quad \text{Si } H(X) <= -1 \end{cases}$$

=> Rechercher les meilleures valeurs de **W** et de **b**.

- SVM doit satisfaire : - i.e. doit correctement classifié tous les **Xi** (les datapoints dans le dataset) :

$$y_i (W^T x_i + b) \geq 1, i = 1..n$$

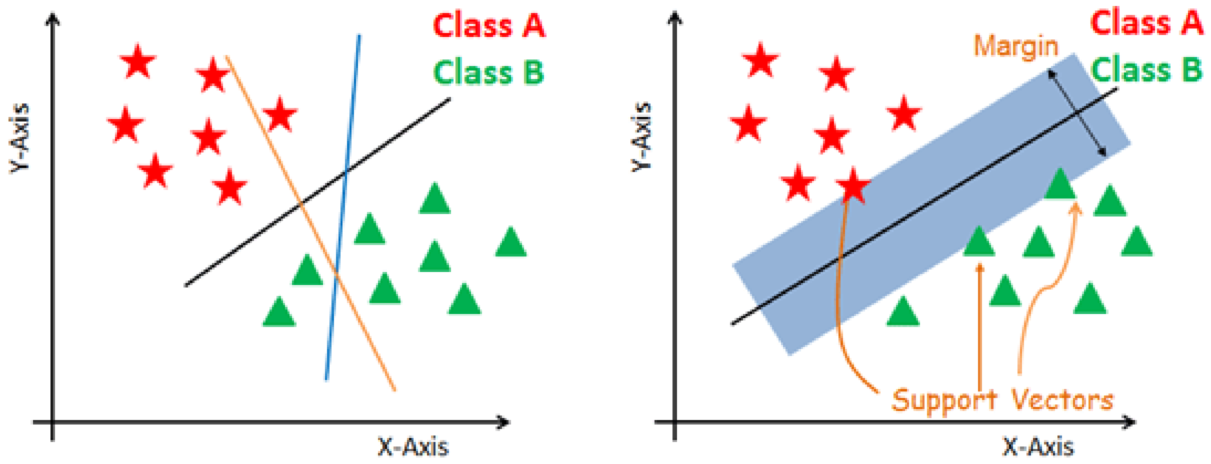
SVM Linéaire

<u>Cas de séparabilité linéaire</u>

- Etapes SVM:

1 - Générer et tester des hyperplans qui séparent les classes, en utilisant et testant différentes valeurs de W et de b, avec:

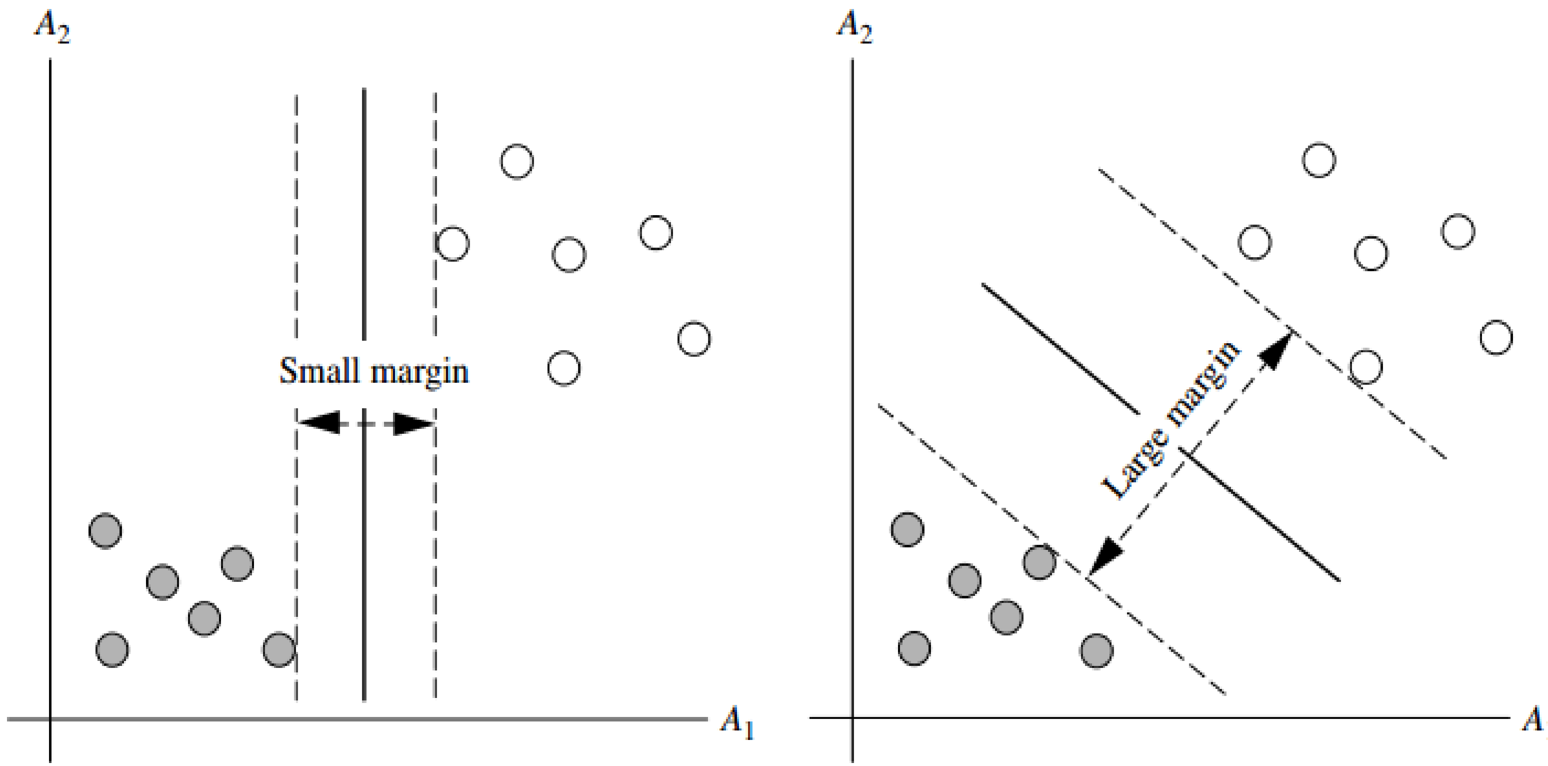
$$y_i(W^T x_i + b) \geq 1, i = 1..n$$



SVM Linéaire

Cas de séparabilité linéaire

- Quand s'arrêter avec l'étape 1 ? Quel est l'**hyperplan optimal** ? – i.e. qui sépare le mieux les données. Comment savoir que W et b trouvées sont les meilleures valeurs ?

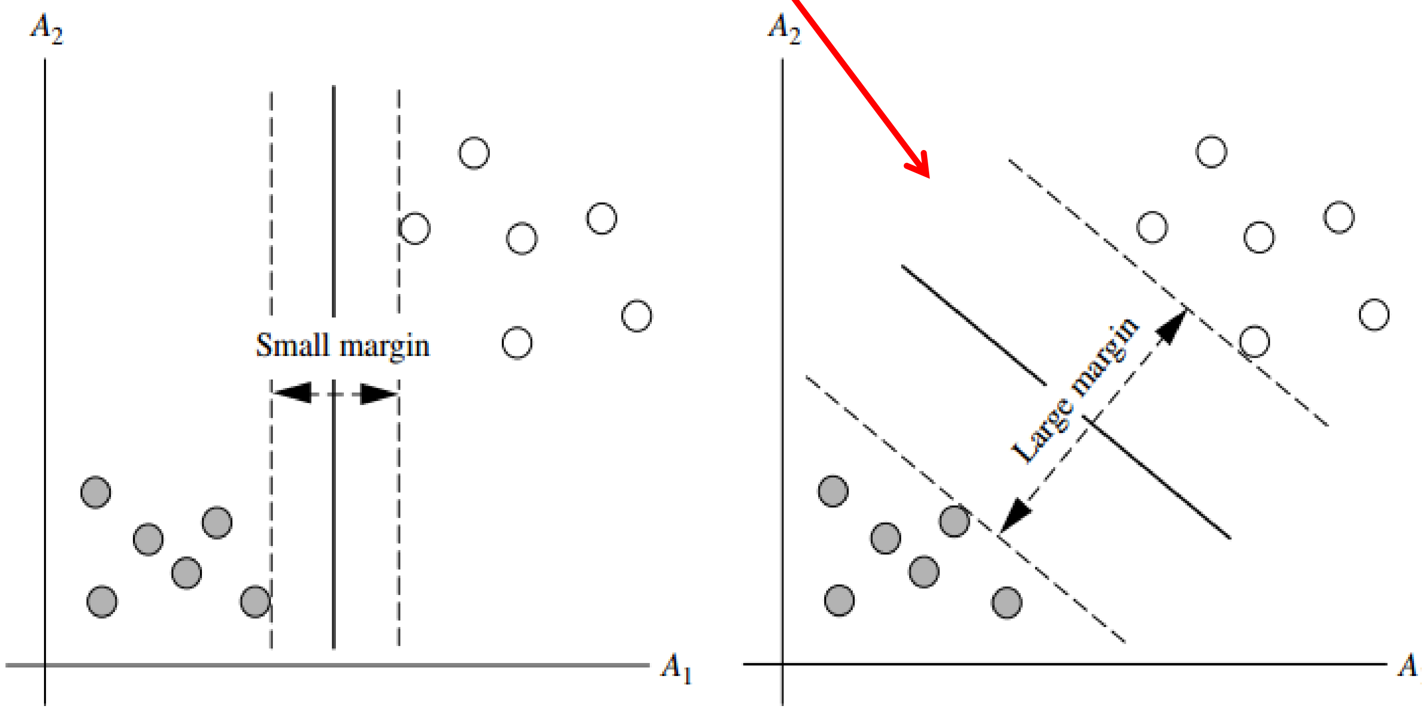


SVM Linéaire

<u>Cas de séparabilité linéaire</u>

- Quel est l'hyperplan **optimal** ? – i.e. qui sépare le mieux les données.

Solution : Maximiser la marge

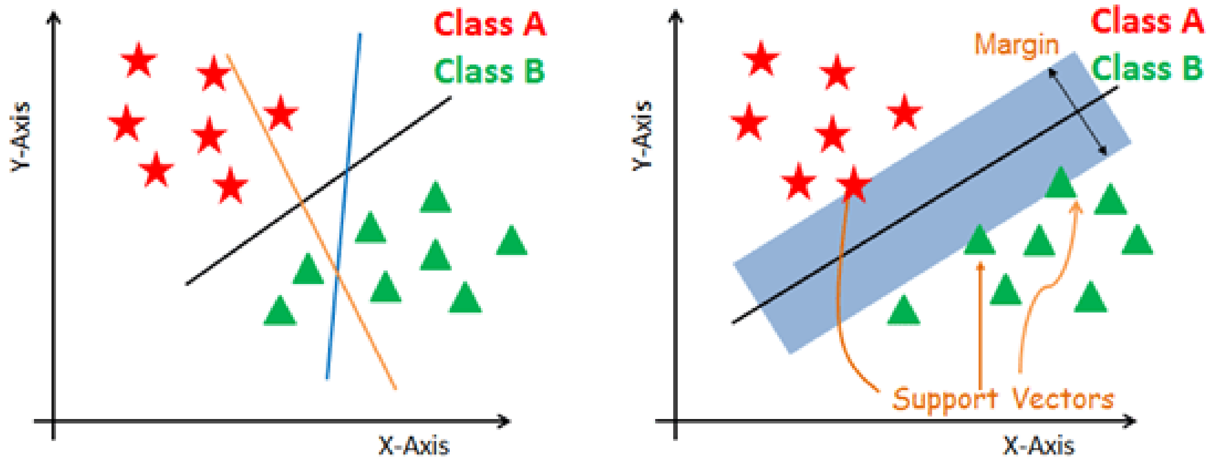


SVM Linéaire

Cas de séparabilité linéaire

➢ Etapes SVM:

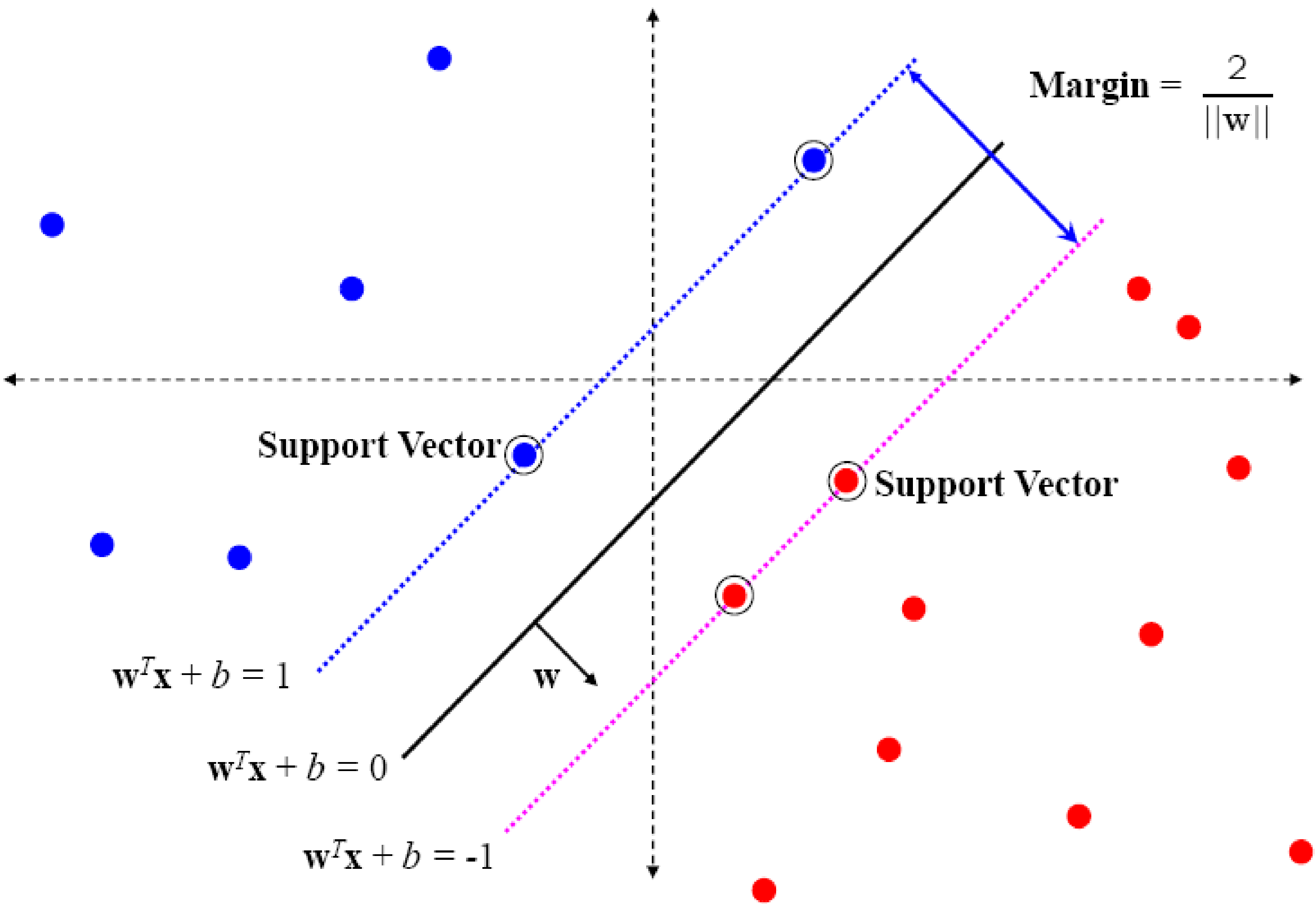
2 – Calculer la marge avec chaque hyperplan généré et sélectionné l'hyperplan qui a une **marge** maximale séparant le mieux les données (voire les **vecteurs supports** seulement):



SVM Linéaire

$$H(X) = W^T X + b$$

<u>Cas de séparabilité linéaire</u>



SVM Linéaire

$$H(X) = W^T X + b$$

Cas de séparabilité linéaire

- Marge : distance entre deux hyperplans parallèles
- Marge = 2/||W||, avec ||W|| : norme du vecteur W (x1, x2) = sqrt($x1^2 + x2^2$)
- Maximiser la marge ➔ Maximiser $\frac{2}{||\mathbf{w}||}$

  avec ||W|| : la norme du vecteur W.

- Ceci équivaut à minimiser 1/2*$||W||^2$
- Pour trouver l'hyperplan séparateur qui maximise la marge :
  - déterminer le vecteur **W** qui possède la norme euclidienne minimale ;
  - et qui vérifie la contrainte de l'équation de bonne classification des exemples d'entrainement.

SVM Linéaire

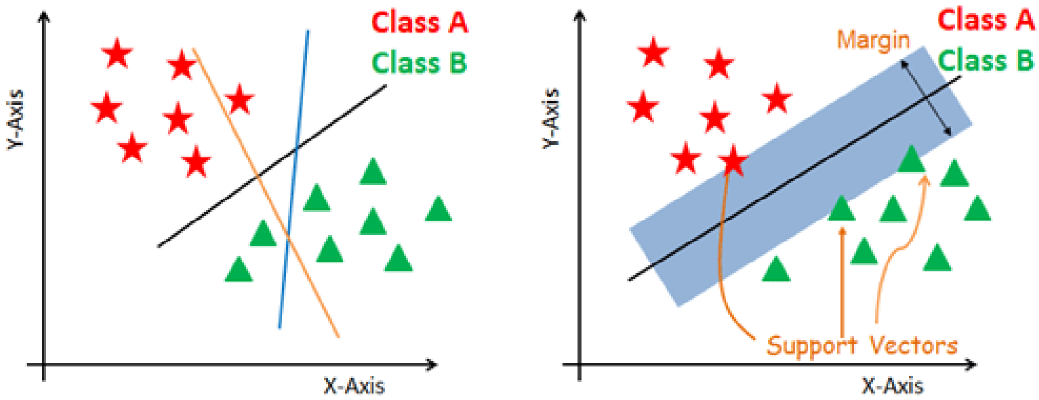
Cas de séparabilité linéaire

$$y_i(W^T x_i + b) \geq 1, i = 1..n$$

- Etapes SVM:

2 – Calculer la marge avec chaque hyperplan généré et sélectionné l'hyperplan qui a une **marge** maximale séparant le mieux les **vecteurs supports**, avec:

- W.Xi + b = 1, pour vecteurs supports de classe +1
- W.Xi + b = -1, pour vecteurs supports de classe -1



SVM Linéaire

Cas de séparabilité linéaire

- L'hyperplan séparateur optimal peut être obtenu en résolvant le problème de l'équation :

$$\begin{cases} \text{Minimiser } \; 1/2 \; ||W||^2 \\ \text{Sous contraintes de } \; y_i(W^T x_i + b) \geq 1, i = 1..n \end{cases}$$

  - yi (W.Xi + b) = 1, pour vecteurs supports de classe +1
  - yi (W.Xi + b) = -1, pour vecteurs supports de classe -1

- Le problème de l'équation est un problème de programmation quadratique avec contraintes linéaires.

- Convex optimization problem.

Cas de séparabilité linéaire

Minimize $\boxed{\frac{1}{2}\sum_{i=1}^{n} w_i^2}$ subject to $\boxed{y_i(\vec{w}\cdot\vec{x}_i + b) - 1 \geq 0}$ for $i = 1,\ldots,N$

Objective function   Constraints

- Utiliser l'équation de Lagrange et introduire ses multiplicateurs.
- L'hyperplan séparateur optimal peut être obtenu en maximisant le problème dual :

Maximiser $W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j$

Sous contraintes de: $\sum_{i=1}^{n} \alpha_i y_i = 0$

$$\alpha_i \geq 0$$

Cas de séparabilité linéaire

Minimize $\boxed{\frac{1}{2}\sum_{i=1}^{n} w_i^2}$ subject to $\boxed{y_i(\vec{w}\cdot\vec{x}_i + b) - 1 \geq 0}$ for $i = 1,\ldots,N$

Objective function   Constraints

- Utiliser l'équation de Lagrange et introduire ses multiplicateurs.
- On déduit que :

Avec
$$\begin{cases} W = \sum_{i=1}^{n} \alpha_i y_i x_i \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases}$$

SVM Linéaire

<u>Cas de séparabilité linéaire</u>

- Le problème de l'équation est un problème de programmation quadratique avec contraintes linéaires.

- En utilisant la formule de Lagrange, **la fonction de décision** devient :

Nombre de vecteurs support

$$H(x) = \sum_{i=1}^{l} \alpha_i y_i x_i x^T + b$$

Exemple à tester

Les classes des vecteurs supports

Supports vecteurs

Multiplicateurs de Lagrange

SVM Linéaire

<u>Cas de séparabilité linéaire</u>

- En utilisant la **formule de Lagrange**, la fonction de décision devient :

$$H(x) = \sum_{i=1}^{l} \alpha_i y_i x_i x^T + b$$

- La classe de x selon l'hyperplan optimal :

$$\begin{cases} \text{La classe de x = + 1} & \text{Si H(X) > 0} \\ \text{La classe de x = + 1} & \text{Si H(X) < 0} \end{cases}$$

$$i=1$$

$$= sign(\sum_{i=1}^{N} \alpha_i y_i \vec{x}_i \cdot \vec{x} + b)$$

- La zone - 1 < H(x) < 1 est appelée la zone de généralisation.

SVM Linéaire

Cas de séparabilité linéaire

- Implémentation des SVM: pour la classification binaire consiste à la résolution du problème dual de programmation quadratique.
- Résolution ➔ déterminer les multiplicateurs Lagrangiens optimaux.
- Si nombre d'exemples est important => problème de temps et de mémoire.
- Solution : Méthode de Shnuking. Entrainement sur quelques exemples choisis par une heuristique, puis ajout des autres itérativement.
- L'algorithme SMO : Sequential Minimal Optimisation. (*Platt et all., 1999*) – Librairie LibSVM

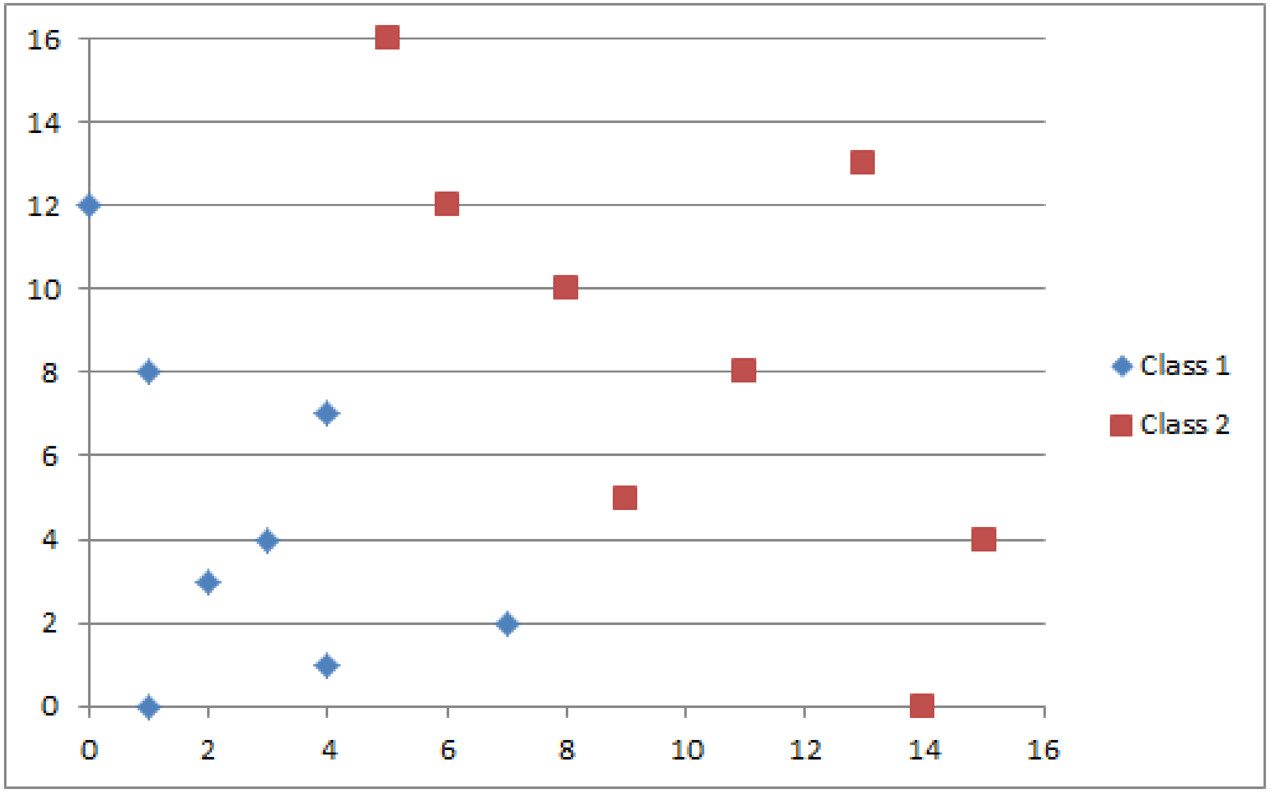
SVM Linéaire

<u>Cas de séparabilité linéaire</u>

$$H(x) = \sum_{i=1}^{l} \alpha_i y_i x_i x^T + b$$

- Exemple :

Le but est de trouver un hyperplan **H(x)** séparant les données en maximisant la marge.



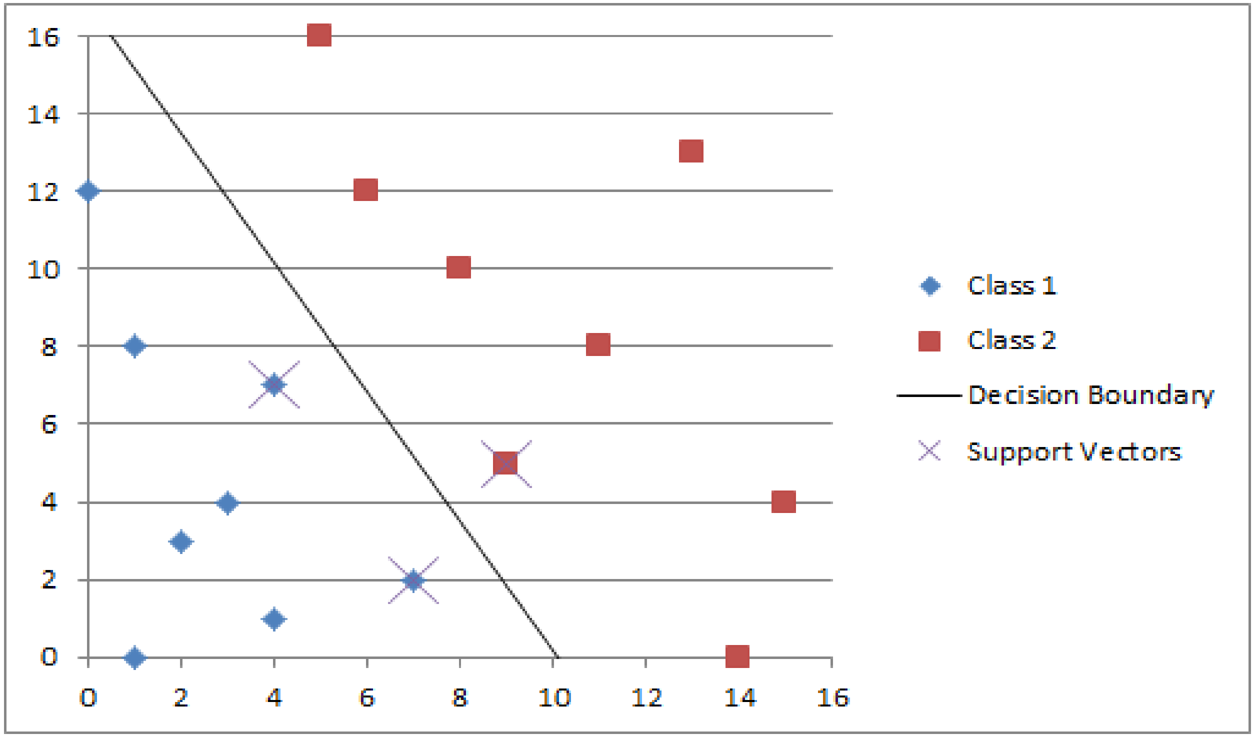
SVM Linéaire

Cas de séparabilité linéaire

$$H(x) = \sum_{i=1}^{l} \alpha_i y_i x_i x^T + b$$

- Exemple :

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$



SVM Linéaire

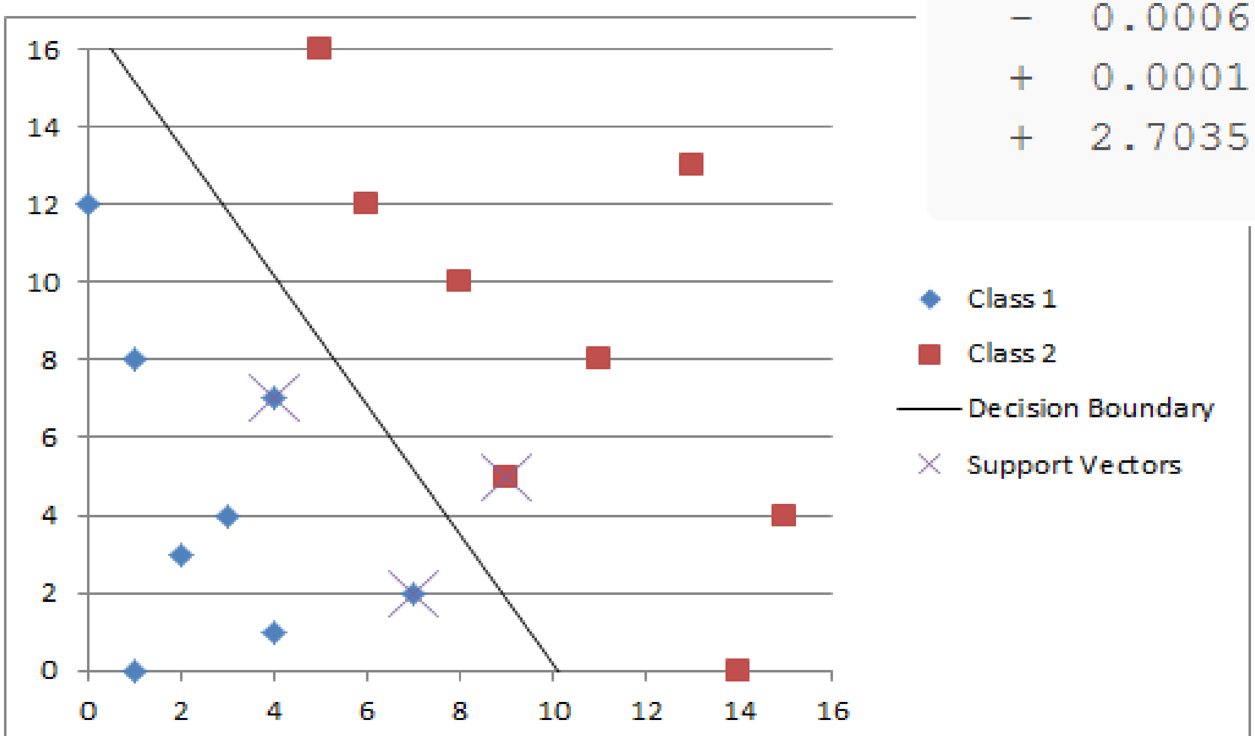
$$H(x) = \sum_{i=1}^{l} \alpha_i y_i x_i x^T + b$$

<u>Cas de séparabilité linéaire</u>

➢ Exemple :

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

```
    0.0005 * <7 2 > * x]
 -  0.0006 * <9 5 > * x]
 +  0.0001 * <4 7 > * x]
 +  2.7035
```



SVM Linéaire

Cas de séparabilité linéaire

- Exemple :

$$H(x) = \sum_{i=1}^{l} \alpha_i y_i x_i x^T + b$$

```
    0.0005 * <7 2 > * x]
-   0.0006 * <9 5 > * x]
+   0.0001 * <4 7 > * x]
+   2.7035
```

$$(1)(0.0005)(7x_1+2x_2)^2+(-1)(0.0006)(9x_1+5x_2)^2+(1)(0.0001)(4x_1+7x_2)^2+2.7035$$

SVM Linéaire

Cas de séparabilité linéaire

- Exemple :

- Introduire dans Google Search Bar :

plot z = 0.0005(7x + 2y)^2-0.0006(9x+5y)^2+0.0001(4x+7y)^2+2.7035

Pour (x, y) : (11, 8) ➔ H(x) = -3.5646
➔ classe : -1

Pour (x, y) : (4, 1) ➔ H(x) = +2.1978
➔ classe : +1

SVM Linéaire

Cas de séparabilité linéaire

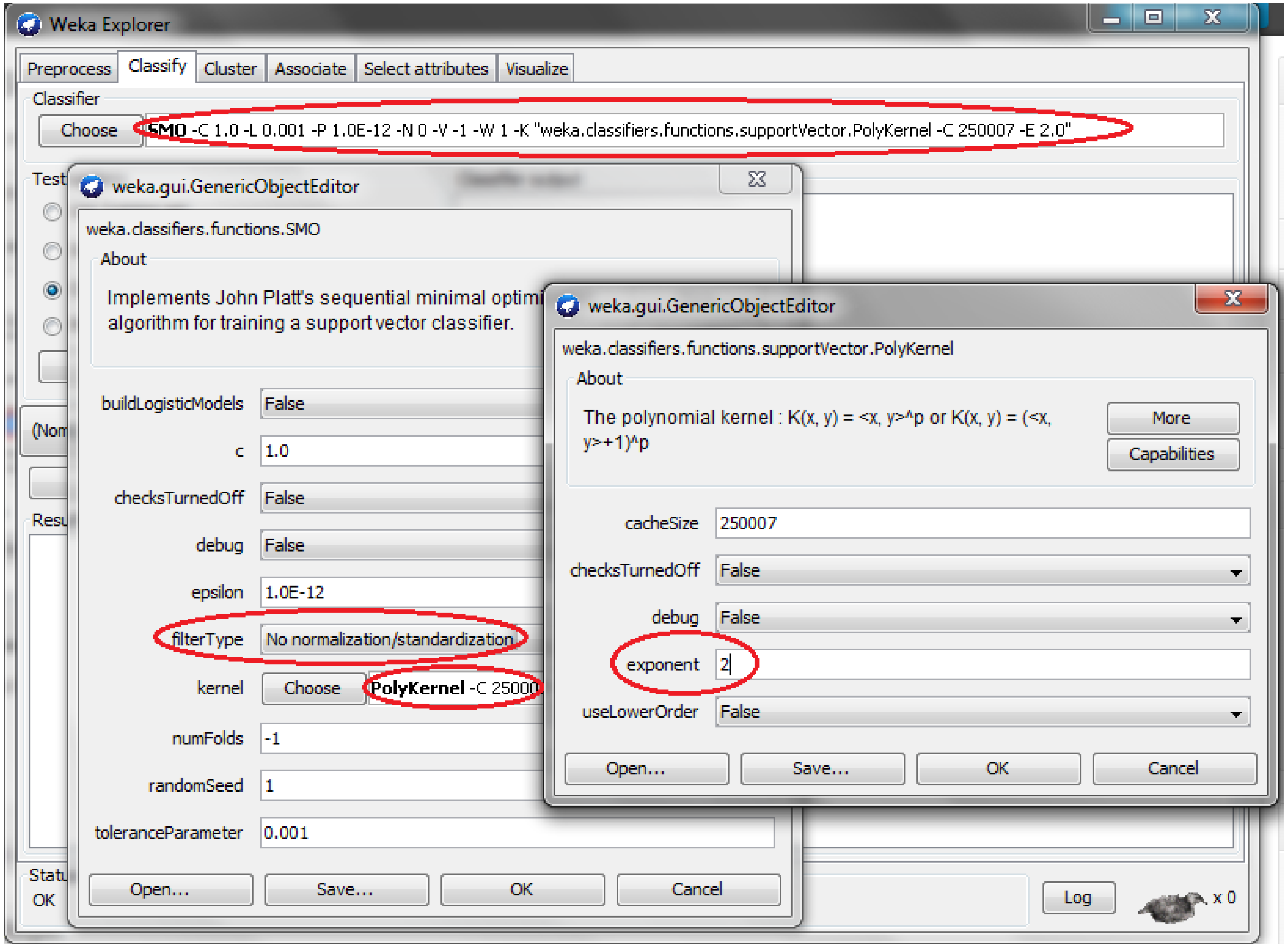
- Exemple : tester sur **Weka**

- Charger les données (CSV ou arff)
- Sélectionner le classifieur SMO : Classify -> Choose -> classifiers -> functions ->SMO
- Sélectionner Use training set
- Paramétrer comme suit :
- Clic sur Start

SVM Linéaire

Cas de séparabilité linéaire

➢ Exemple : tester sur **Weka**

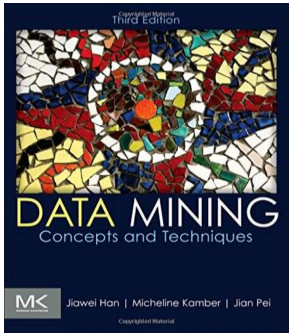


Cas de séparabilité linéaire

- Exemple : tester sur **Weka**

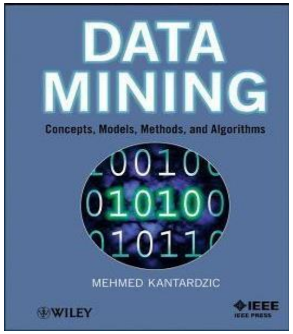
```
=== Classifier model (full training set) ===

SMO

Kernel used:
  Poly Kernel: K(x,y) = <x,y>^2.0

Classifier for classes: A, B

BinarySMO

      0.0005 * <7 2 > * X]
 -       0.0006 * <9 5 > * X]
 +       0.0001 * <4 7 > * X]
 +       2.7035

Number of support vectors: 3
```

Ressources



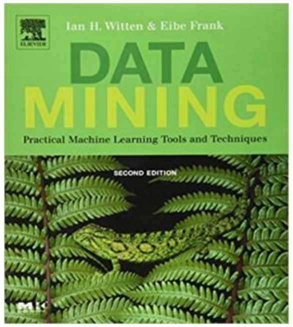
Data Mining : concepts and techniques, 3rd Edition

- ✓ Auteur : Jiawei Han, Micheline Kamber, Jian Pei
- ✓ Éditeur : Morgan Kaufmann Publishers
- ✓ Edition : Juin 2011 - 744 pages - ISBN 9780123814807



Data Mining : concepts, models, methods, and algorithms

- ✓ Auteur : Mehmed Kantardzi
- ✓ Éditeur : John Wiley & Sons
- ✓ Edition : Aout 2011 – 552 pages - ISBN : 9781118029121



Data Mining: Practical Machine Learning Tools and Techniques

- ✓ Auteur : Ian H. Witten & Eibe Frank
- ✓ Éditeur : Morgan Kaufmann Publishers
- ✓ Edition : Juin 2005 - 664 pages - ISBN : 0-12-088407-0

Ressources

Cours – Abdelhamid DJEFFAL – Fouille de données avancée

- ✓ www.abdelhamid-djeffal.net

WekaMOOC – Ian Witten – Data Mining with Weka

- ✓ https://www.youtube.com/user/WekaMOOC/featured

Cours - Phugo Larochelle – Apprentissage Automatique

✓https://www.youtube.com/playlist?list=PL6Xpj9I5qXYFD_rc1tttugXLfE2TcKyiO

Chris McCormick - SVM Example

- ✓ http://mccormickml.com/2013/04/16/trivial-svm-example/