

The image features a central logo for 'SVM' (Support Vector Machine). The logo consists of a dark blue circle with a thick, light blue ring around it. The letters 'SVM' are written in white, bold, sans-serif font in the center of the dark circle. The background is a dark blue gradient with a complex network of light blue circuit lines, nodes, and small squares, giving it a high-tech, digital appearance.

SVM

Preparer par :
MOUHIHA Mohamed
EL MOUSAAIF Mohamed

Encadre par :
Ghazdali Abdelghani

Plan



- **Introduction**
 - Apprentissage supervisé / non supervisé
 - Définition de SVM
 - Les cas d'utilisation du SVM
- **Concept général**
 - Théoriquement
 - Mathématiquement
- **Cas non Linéairement séparable**
 - Overfitting / Underfitting
 - Kernel Trick
 - Kernel Functions
- **Avantages et inconvénients du SVM**
- **Exemple**
- **Démonstration**
- **Conclusions**




Introduction



Apprentissage supervisé

- C'est un axe du machine Learning où on supervise l'apprentissage d'un modèle avec des données qui sont déjà étiquetées avec la classe associée à chaque observation.
- Quelques algorithmes dans ce type sont : Les Arbres de décision, KNN, naïve bayes, SVM, ANN, Ensemble Learning, ...

Apprentissage non supervisé

- C'est un des autres types du machine Learning qui ne se base pas sur les libellés pour l'entraînement.
 - Ils essaient de classer des données en classes (ou clusters) où le nombre de ces derniers est donné comme paramètre.
 - Des implémentations de ce concept sont : la classification hiérarchique, DBSCAN, PAM, ...
- 

A white smartphone is shown vertically on the left side of the slide. The screen is white and displays the text 'SVM ?' in a bold, dark blue font. The phone has a thin black bezel and a circular home button at the bottom.

SVM ?

SVM

Les machines à vecteurs de support, ou support vector machine (SVM), sont des modèles de machine learning supervisés centrés sur la résolution de problèmes de discrimination et de régression mathématiques.

Ce modèle a été rapidement adopté en raison de sa capacité à travailler avec des données de grandes dimensions, ses garanties théoriques et les bons résultats réalisés en pratique. Requérant un faible nombre de paramètres, les SVM sont appréciées pour leur simplicité d'usage.

Dans l'algorithme SVM, nous représentons chaque élément de données comme un point dans un espace à n dimensions, la valeur de chaque caractéristique étant la valeur d'une coordonnée particulière. Ensuite, nous effectuons la classification en trouvant l'hyperplan qui différencie très bien les deux classes.

Les cas d'utilisation

Détection des visages

SVM classifie les parties de l'image en visages et en non-visages et crée une bordure carrée autour du visage

Reconnaissance de l'écriture manuscrite

Nous utilisons des SVM pour reconnaître les caractères manuscrits largement utilisés.



Classification des images

L'utilisation de SVM offre une meilleure précision de recherche pour la classification des images. Il offre une meilleure précision par rapport aux techniques de recherche traditionnelles basées sur des requêtes

Bio-informatique

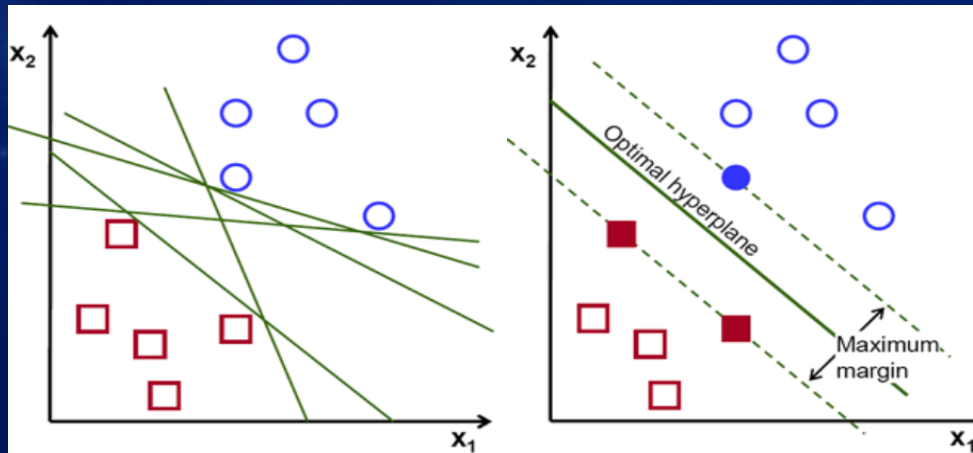
Comprend la classification des protéines et la classification du cancer. Nous utilisons SVM pour identifier la classification des gènes, des patients sur la base de gènes et d'autres problèmes biologiques.



Concept général

Fonctionnement

- Un **Support Vector** est toute observation qu'on utilise dans la recherche de notre Hyperplan, théoriquement, c'est l'ensemble de points les plus proches du hyperplan et qui appartient aux différentes classes
- **SVM** établit l'hyperplan optimal qui maximise la distance entre ce dernier et les bordures de la rue avec une largeur de cette rue (la marge).

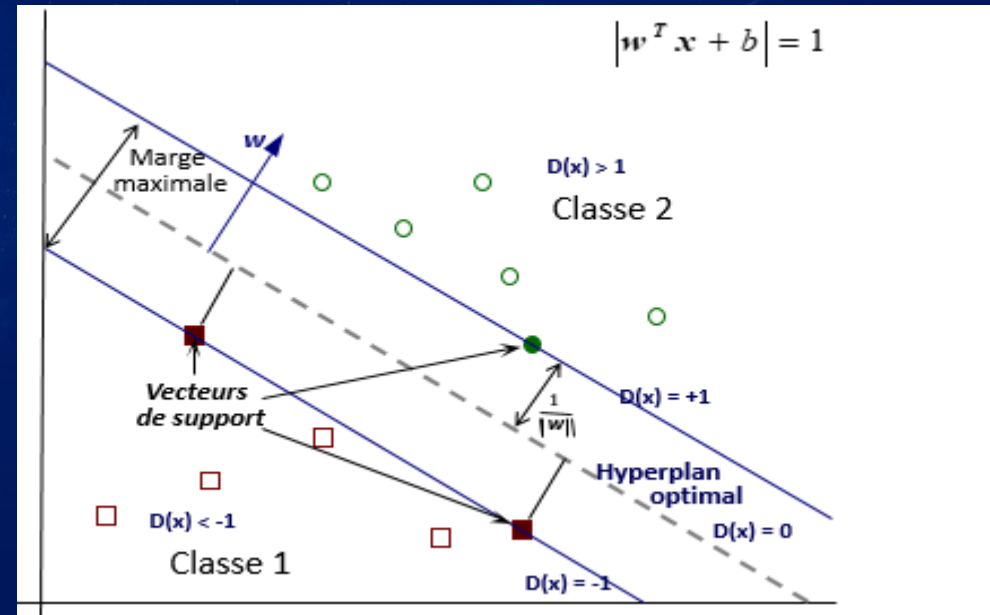


Mathématiquement

Supposons une fonction définie par : $y = ax + b \Leftrightarrow ax + b - y = 0$
On peut la définir par : $\langle W.X \rangle + b = 0$ avec $X = \{x, y\}$ & $W = \{a, -1\}$

$$\begin{cases} y_i = -1 : \langle W.X \rangle + b \leq -1 \\ y_i = 1 : \langle W.X \rangle + b \geq 1 \end{cases} \Leftrightarrow y_i(\langle W.X \rangle + b) - 1 \geq 0 \text{ avec } y_i = \{1, -1\}$$

- W est le vecteur normal avec de hyper-plan.
- y_i est la variable de décision.



Mathématiquement

$$H1: \langle W.X \rangle + b = -1$$

$$H2: \langle W.X \rangle + b = 1$$

Pour calculer la distance Point-Plane : $D = \frac{|ax_1+by_1-d|}{\|W\|}$

La distance entre le hyper-plan **H1** et l'origine **O(0,0)** : $d(H1, O) = \frac{|-1-b|}{\|W\|}$

La distance entre le hyper-plan **H2** et l'origine **O(0,0)** : $d(H2, O) = \frac{|1-b|}{\|W\|}$

Et donc la distance de la marge est : $M = d(H2, O) - d(H1, O) = \frac{1-b-(-1-b)}{\|W\|} = \frac{2}{\|W\|}$

D'où **2m = M** $\Leftrightarrow m = \frac{1}{\|W\|}$ qu'on doit la maximiser ou bien minimiser **$\|W\|$**

Sachant que minimiser **$\|W\|$** peut être écrite comme une minimisation de $\frac{\|W\|^2}{2}$

Mathématiquement

Donc le problème à optimiser est :

$$Z = \min \frac{\|W\|^2}{2} \text{ avec la contrainte } y_i(\langle W.X \rangle + b) - 1 \geq 0 \ (i = 1 \dots l)$$

Ce programme a une contrainte, alors l'algorithme le plus adéquat est Lagrange, car il élimine les contraintes en les insérant dans la fonction objective.

Explication de Lagrange :

C'est une méthode qui transforme un problème d'optimisation avec contraintes en un problème d'optimisation sans contraintes

Exemple :

Soit le problème :

$$\min_{x \in \mathbb{R}} f(x) \text{ avec } g(x) \geq 0 \quad \Rightarrow \quad \min_{\substack{x \in \mathbb{R} \\ \lambda \in \mathbb{R}^+, (\lambda \geq 0)}} f(x) - \lambda g(x)$$

ou x et λ sont inconnus

Mathématiquement

La fonction objective transformée :

$$l = \frac{\|W\|^2}{2} - \lambda (y_i(\langle W, X_i \rangle + b) - 1 \quad \forall i)$$

$$l = \frac{\|W\|^2}{2} - \sum_{i=1}^l \lambda_i (y_i(\langle W, X_i \rangle + b) - 1)$$

$$l = \frac{\|W\|^2}{2} - \sum_{i=1}^l \lambda_i (y_i(\langle W, X_i \rangle + b)) + \sum_{i=1}^l \lambda_i$$

Après la dérivation, on obtient :

$$\frac{\partial l}{\partial W} = W - \sum_{i=1}^l \lambda_i \cdot y_i \cdot X_i = 0$$

$$\frac{\partial l}{\partial \lambda} = \sum_{i=1}^l y_i (W \cdot X_i + b) + 1 = 0$$

$$\frac{\partial l}{\partial b} = \sum_{i=1}^l \lambda_i \cdot y_i = 0$$

Mathématiquement

A ce stade là on a trouvé l'expression de W :

$$W = \sum_{i=1}^l \lambda_i \cdot y_i \cdot X_i$$

Donc il suffit de trouver la valeur de λ pour trouver W , et après on aura la valeur de b .

Pour trouver λ on retourne à la fonction objective originale et on remplace W par son expression:

$$l = \frac{\|W\|^2}{2} - \sum_{i=1}^l \lambda_i (y_i (W \cdot X_i + b)) + \sum_{i=1}^l \lambda_i$$

Substitution de la valeur de W dans l'équation

$$l = \frac{\sum_{i=1}^l \|\lambda_i \cdot y_i \cdot X_i\|^2}{2} - \sum_{j=1}^l \sum_{i=1}^l \lambda_j y_j (\|\lambda_i \cdot y_i \cdot X_i\| \cdot X_j + b) + \sum_{i=1}^l \lambda_i$$

$$l = \sum_{i=1}^l \lambda_i - \frac{\sum_{i,j=1}^l \lambda_j \lambda_i y_j y_i X_i X_j}{2}$$

Mathématiquement

Pour simplifier ce problème, on définit un $K(i, j) = y_j y_i X_i X_j$, sous la forme vectorielle $K = y^T y \cdot X^T X$:

$$\max l = \sum_{i=1}^l \lambda_i - \frac{\lambda^T \lambda K}{2}$$

On résoudront ce problème, on trouve la valeur optimal de λ , après nous pouvons trouver la valeur de b :

Chaque Support vector est défini par : $y_s(W \cdot X_s + b) - 1 = 0$

$$y_s \left(\sum_{m \in s} \lambda_m y_m X_m \cdot X_s + b \right) - 1 = 0$$

Ou s dénote les indices des SV

$$y_s y_s \left(\sum_{m \in s} \lambda_m y_m X_m \cdot X_s + b \right) = y_s$$

Sachant que $y_s y_s = 1$

$$b = y_s - \sum_{m \in s} \lambda_m y_m X_m \cdot X_s$$

Mathématiquement

Modifiant l'équation pour utiliser la moyenne des SV :

$$b = \frac{\sum_{(s \in S)} (y_s - \sum_{m \in S} \lambda_m y_m X_m \cdot X_s)}{N_s}$$

En fin pour classifier les nouvelles données X_n :

$$\textit{sign}(\langle W \cdot X_n \rangle + b)$$

Mathématiquement

La marge

Les changements qu'on va effectuer pour accomplir un soft margin sont :

- L'ajout d'une variable d'écart ξ_i qui va représenter la distance entre le support vector et le hyper-plane qui pose la bordure de la même classe.
- Le paramétrage du système avec une variable de cout C , qui est le nombre de supports vectors qui sont permis de dépasser le hyperplane approprié.

$$\begin{cases} W.X_i + b \geq 1 - \xi_i & \text{avec } y_i = 1 \\ W.X_i + b \geq -1 + \xi_i & \text{avec } y_i = -1 \end{cases} \Leftrightarrow y_i(W.X_i + b) - 1 + \xi_i \geq 0$$

$\text{avec } y_i = \{1, -1\} \quad \xi_i \geq 0 \text{ et } i = 1 \dots l$

$$\sum_{i=1}^l \xi_i = C$$

- Si $C = 0$, aucune observation aura une ξ (hard margin).
- Si $C > 0$, soft margin.

Mathématiquement

La marge

Donc notre problème devient :

$$l = \min \frac{\|W\|^2}{2} + C \sum_{i=1}^l \xi_i \quad \text{avec} \quad y_i(W \cdot X_i + b) - 1 + \xi_i \geq 0, \\ \xi_i \geq 0$$

Deuxième problème :

$$\max l = \sum_{i=1}^l \lambda_i - \frac{\lambda^T \lambda K}{2} \quad \text{avec} \quad 0 \leq \lambda \leq C, \sum_{i=1}^l \lambda_i \cdot y_i = 0$$



3

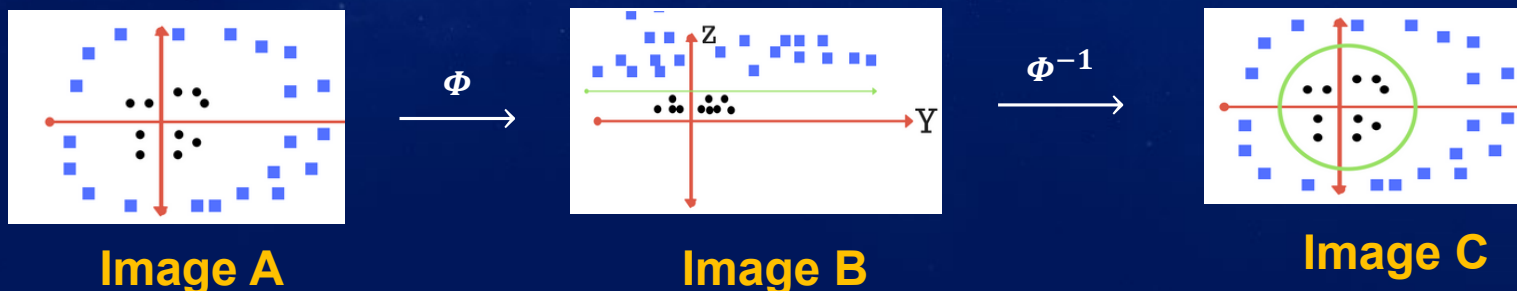
Cas non linéairement séparable

Cas non linéairement séparable

Considérons que se passe-t-il si nous avons des données comme indiqué dans l'image A? De toute évidence, aucune ligne ne peut séparer les deux classes dans ce plan x et y . Alors nous appliquons la transformation et ajoutons une dimension supplémentaire comme nous l'appelons $axe z$.

Supposons la valeur des points sur le plan z , $Z = X^2 + Y^2$. Maintenant, si nous traçons sur l'axe z , une séparation claire est visible et une ligne peut être dessinée (Image B). Lorsque nous reconvertissons cette ligne dans le plan d'origine, elle correspond à la frontière circulaire comme le montre l'image C.

Ces transformations sont appelées noyaux (*kernels*).



Cas non linéairement séparable

Mercer's

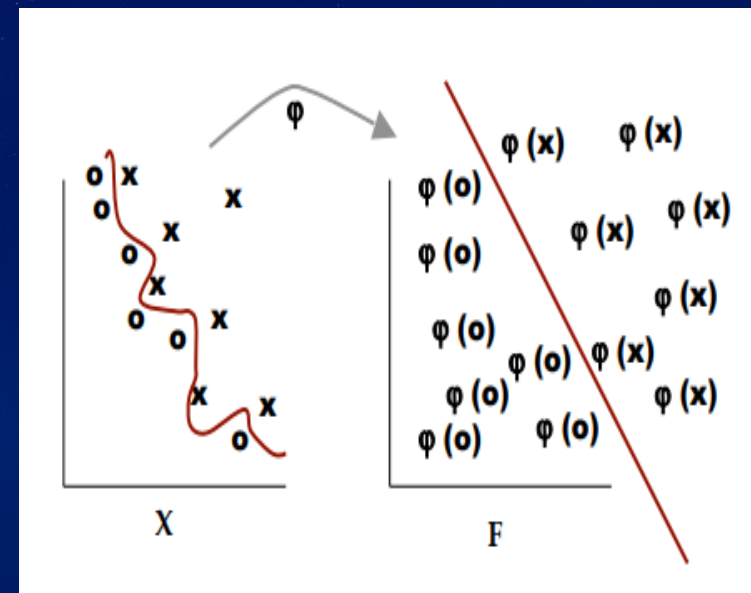
Le théorème de Mercer's dans l'analyse fonctionnel propose que si une fonction $K(a,b)$ satisfait tous les contraintes qui s'appellent les contraintes de Mercer's, donc il existe une fonction qui fait la projection de a et b sur une sur-dimension.

$$K(a,b) = \Phi(a)^T \cdot \Phi(b)$$

ou $\Phi(\cdot)$ est une kernel

Théoriquement on doit utiliser la fonction kernel pour transformer les inputs.

Mais dans la pratique en appliquant une kernel sur les données va coûter beaucoup d'espace mémoire et de temps d'exécution.



Kernels Functions

Les changements qu'on va effectuer pour accomplir un soft margin sont :

- L'ajout d'une variable d'écart ξ_i qui va représenter la distance entre le support vector et le hyper-plane qui pose la bordure de la même classe.
- Le paramétrage du système avec une variable de cout C , qui est le nombre de supports vectors qui sont permis de dépasser le hyperplane approprié.

$$\begin{cases} W \cdot X_i + b \geq 1 - \xi_i & \text{avec } y_i = 1 \\ W \cdot X_i + b \geq -1 + \xi_i & \text{avec } y_i = -1 \end{cases} \Leftrightarrow y_i(W \cdot X_i + b) - 1 + \xi_i \geq 0$$

$\text{avec } y_i = \{1, -1\} \quad \xi_i \geq 0 \text{ et } i = 1 \dots l$

$$\sum_{i=1}^l \xi_i = C$$

- Si $C = 0$, aucune observation aura une ξ (hard margin).
 - Si $C > 0$, soft margin.

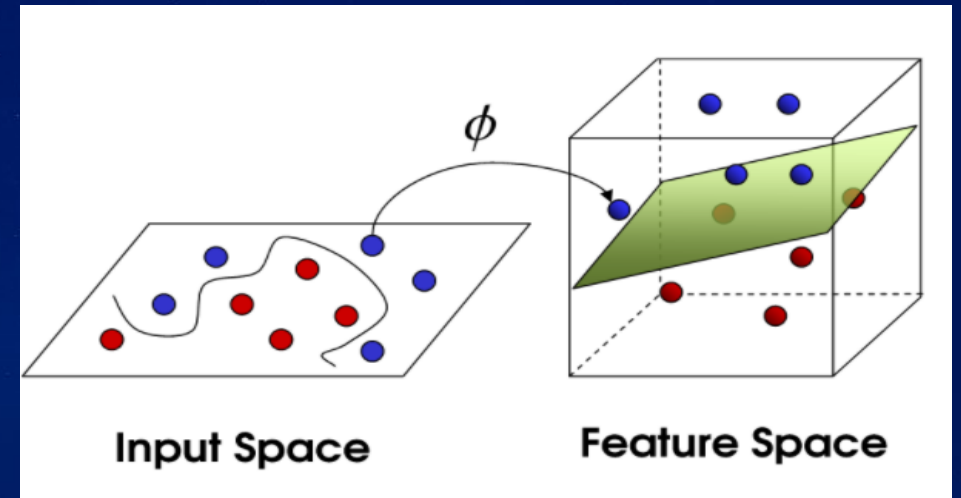
Kernels Functions

Trick

Donc on doit utiliser une Kernel Trick qui s'agit d'affecter la projection dans l'optimisation, pour garder la même fonction de décision :

$$K(X_i, X_j) = \langle \Phi(X_i), \Phi(X_j) \rangle \text{ donc } (X \in \mathbb{R}^2) \rightarrow (\Phi(X) \in \mathbb{R}^\infty)$$

Cette équation est un cas du noyaux linéaire.



Kernels Functions

Les algorithmes SVM utilisent un ensemble de fonctions mathématiques qui sont définies comme le noyau.

La fonction du noyau est de prendre des données en entrée et de les transformer dans la forme requise. Les différents algorithmes SVM utilisent différents types de fonctions de noyau.

Ces fonctions peuvent être de différents types.

Par exemple:

- Polynomial Kernel
- Gaussian Kernel
- Gaussian Radial Basis Function (RBF)
- Sigmoid Kernel
- Linear Spline Kernel in 1D

Kernels Functions

No	Kernel function	Formula	Optimization parameter
1	Dot-product	$K(x_n, x_i) = (x_n, x_i)$	C
2	RBF	$K(x_n, x_i) = \exp(-\gamma \ x_n - x_i\ ^2 + C)$	C and γ
3	Sigmoid	$K(x_n, x_i) = \tanh(\gamma(x_n, x_i) + r)$	C, γ , and r
4	Poly-nomial	$K(x_n, x_i) = (\gamma(x_n, x_i) + r)^d$	C, γ , r, d

Overfitting/Underfitting

Overfitting

L'Overfitting (sur-apprentissage) désigne le fait que le modèle prédictif produit par l'algorithme de Machine Learning s'adapte bien au Training Set. Cela représente un modèle qui a appris par cœur ses données d'entraînement, qui fonctionne donc bien sur le jeu d'entraînement mais pas de validation. Il effectue alors de mauvaises prédictions sur de nouvelles, car elles ne sont pas exactement les mêmes que celle du jeu d'entraînement. Pour y remédier, il faut améliorer la flexibilité du modèle, et donc jouer sur des concepts de régularisation par exemple, ou encore d'early stopping.

Overfitting/Underfitting

Underfitting

L'Underfitting (sous-apprentissage), sous entend que le modèle prédictif généré lors de la phase d'apprentissage, s'adapte mal au Training Set.

❑ Autrement dit, le modèle prédictif n'arrive même pas à capturer les corrélations du Training Set. Par conséquent, le coût d'erreur en phase d'apprentissage reste grand. Bien évidemment, le modèle prédictif ne se généralisera pas bien non plus sur les données qu'il n'a pas encore vu. Finalement, le modèle ne sera viable car les erreurs de prédictions seront grandes.

❑ On dit également qu'il souffre d'un grand Bias (biais).



Avantages et inconvénients

Avantages et inconvénients du SVM

Inconvénients

- ✓ Les inconvénients sont que la théorie ne couvre réellement que la détermination des paramètres pour une valeur donnée des paramètres de régularisation et du noyau et le choix du noyau. D'une certaine manière,
- ✓ le SVM déplace le problème du sur-ajustement de l'optimisation des paramètres à la sélection du modèle. Malheureusement, les modèles de noyau peuvent être assez sensibles au sur-ajustement du critère de sélection du modèle, voir GC Cawley et NLC Talbot, Sur-ajustement dans la sélection des modèles et biais de sélection subséquent dans l'évaluation des performances.
- ✓ On peut conclure que le SVM ne permet pas de se prémunir contre le sur-ajustement si les points sont indépendant (à cause des paramètres des noyaux) (pas de choix adéquat général)

Avantages et inconvénients du SVM

Avantages

Il y a quatre avantages principaux:

- ✓ Il a un paramètre de régularisation, ce qui incite l'utilisateur à éviter les sur-ajustements.
- ✓ Il utilise l'astuce du noyau, vous pouvez donc acquérir des connaissances d'experts sur le problème via l'ingénierie du noyau.
- ✓ Le fait que SVM est une des méthodes les plus robustes le rend très performant dans le cas où on a un volume minimal des données.

Evaluation du modèle

Métriques de classification

Lors de l'exécution de prédictions de classification, il est nécessaire d'évaluer les modèle quatre types de résultats peuvent se produire :

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Evaluation du modèle

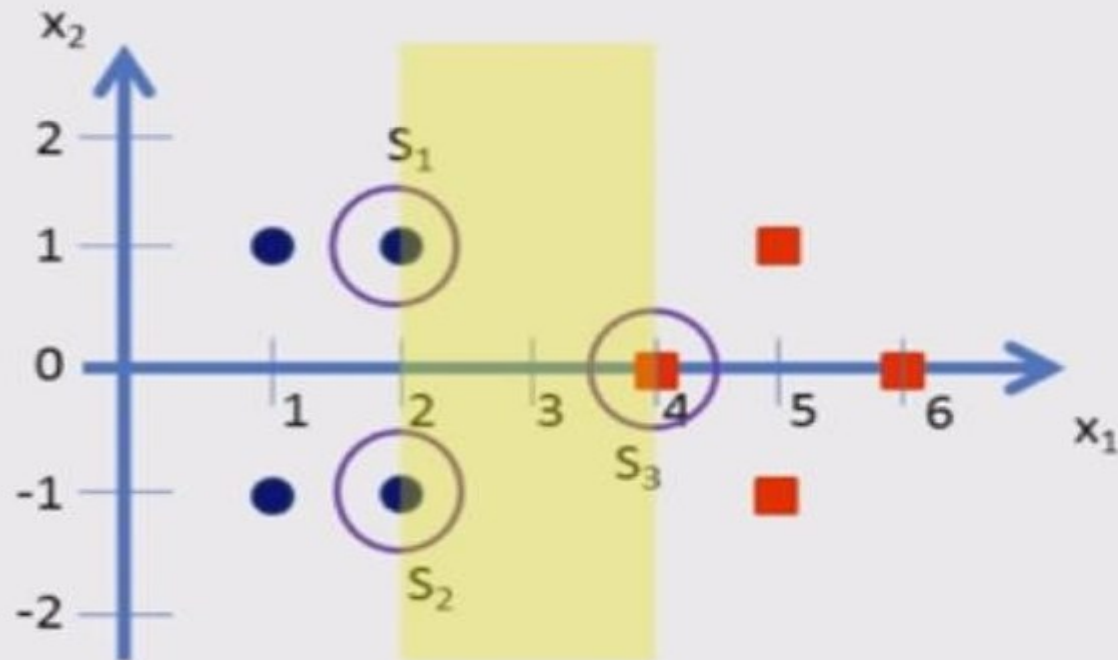
- ✓ True Positive (TP) : sont lorsque vous prédisez qu'une observation appartient à une classe et qu'elle appartient en fait à cette classe.
- ✓ True Negative (TN) : sont lorsque vous prédisez qu'une observation n'appartient pas à une classe et qu'elle n'appartient en fait pas à cette classe.
- ✓ False Positive (FP) : se produisent lorsque vous prédisez qu'une observation appartient à une classe alors qu'en réalité ce n'est pas le cas.
- ✓ False Negative (FN) : se produisent lorsque vous prédisez qu'une observation n'appartient pas à une classe alors qu'en fait c'est le cas.



Example

Exemple

Dans ce exemple, nous sélectionnons 3 vecteurs de support pour commencer.
ce sont S_1, S_2 et S_3 .



$$S_1 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

$$S_2 = \begin{pmatrix} 2 \\ -1 \end{pmatrix}$$

$$S_3 = \begin{pmatrix} 4 \\ 0 \end{pmatrix}$$

Exemple

ici, nous utiliserons des vecteurs augmentés d'un 1 comme entrée de biais, et pour plus de clarté

$$s_1 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

$$s_2 = \begin{pmatrix} 2 \\ -1 \end{pmatrix}$$

$$s_3 = \begin{pmatrix} 4 \\ 0 \end{pmatrix}$$

$$\tilde{s}_1 = \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix}$$

$$\tilde{s}_2 = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}$$

$$\tilde{s}_3 = \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix}$$

il faut maintenant trouver 3 paramètres basés sur les 3 équations linéaires suivantes :

$$\alpha_1 \tilde{s}_1 \cdot \tilde{s}_1 + \alpha_2 \tilde{s}_2 \cdot \tilde{s}_1 + \alpha_3 \tilde{s}_3 \cdot \tilde{s}_1 = -1 \text{ (-ve class)}$$

$$\alpha_1 \tilde{s}_1 \cdot \tilde{s}_2 + \alpha_2 \tilde{s}_2 \cdot \tilde{s}_2 + \alpha_3 \tilde{s}_3 \cdot \tilde{s}_2 = -1 \text{ (-ve class)}$$

$$\alpha_1 \tilde{s}_1 \cdot \tilde{s}_3 + \alpha_2 \tilde{s}_2 \cdot \tilde{s}_3 + \alpha_3 \tilde{s}_3 \cdot \tilde{s}_3 = +1 \text{ (+ve class)}$$

Example

$$\widetilde{S}_1 = \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \quad \widetilde{S}_2 = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \quad \widetilde{S}_3 = \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix}$$

$$\alpha_1 \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} + \alpha_3 \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} = -1$$

$$\alpha_1 \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} + \alpha_3 \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} = -1$$

$$\alpha_1 \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} + \alpha_3 \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} = +1$$

Exemple

Après simplification nous obtenons

$$6\alpha_1 + 4\alpha_2 + 9\alpha_3 = -1$$

$$4\alpha_1 + 6\alpha_2 + 9\alpha_3 = -1$$

$$9\alpha_1 + 9\alpha_2 + 17\alpha_3 = +1$$

l'hyperplan qui discrimine la classe positive de la classe négative est donné par :

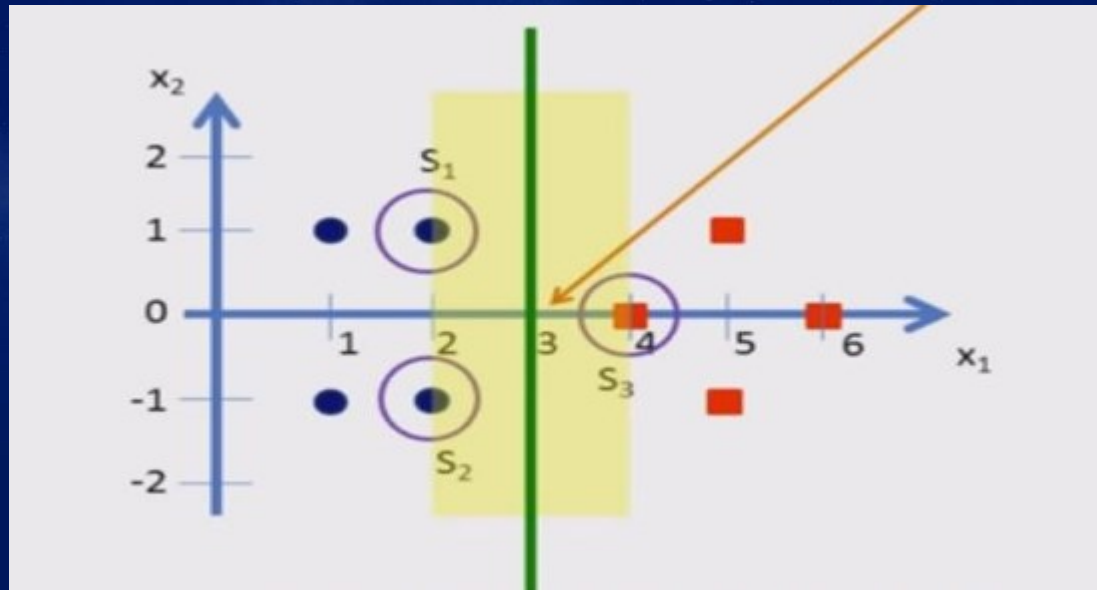
$$W = \sum_{i=1}^l \lambda_i \cdot y_i \cdot X_i$$

en remplaçant les valeurs que nous obtenons

$$\begin{aligned} \tilde{w} &= \alpha_1 \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} + \alpha_2 \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} + \alpha_3 \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} \\ \tilde{w} &= (-3.25) \cdot \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} + (-3.25) \cdot \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} + (3.5) \cdot \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ -3 \end{pmatrix} \end{aligned}$$

Exemple

Alors : $\alpha_1 = \alpha_2 = -3,25$ et $\alpha_3 = 3,5$ et $b = -3$
nos vecteurs sont augmentés d'un biais.
par conséquent, nous pouvons assimiler l'entrée de w
à l'hyperplan avec un décalage b .
Sachant que $y = w \cdot x + b$ avec $w = (1, 0)$ et $b = -3$.





Démonstration



Conclusions

Conclusions

- ✓ Recherche d'un hyperplan, dit de marge optimale (vaste), pour la séparation de deux classes dans un espace hilbertien défini par un noyau reproduisant associé au produit scalaire de cet espace. Estimation de l'hyperplan dans le cas linéaire et séparable.
- ✓ les contraintes actives du problème d'optimisation déterminent les vecteurs supports. Extension au cas non séparable par pénalisation.
- ✓ Extension au cas non linéaire par plongement dans un espace hilbertien à noyau reproduisant

MERCI