

CHAPITRE-3

∞ LES MACHINES A VECTEURS DE SUPPORT ∞

Introduction

Depuis quelques années, des nouvelles méthodes d'apprentissage se développent sur la base de la théorie de l'apprentissage statistique de Vapnik. L'une de ces méthodes, appelée Machines à vecteur de support (ou SVM : l'acronyme de Support Vector Machines en anglais), une méthode de classification qui fut introduite par Vapnik en 1995 [Vap, 98]. Cette méthode fortement basés sur la théorie. Il existe en effet un lien direct entre la théorie de l'apprentissage statistique et l'algorithme d'apprentissage de SVM. La formulation élégante de SVM laisse très peu de place aux paramètres utilisateurs. Mais ce fait véritablement sa force c'est le mécanisme de projection qui lui permet de changer d'espace pour réaliser l'apprentissage et aujourd'hui sont considérées comme une méthode les plus performantes sur nombreux problèmes réels, notamment pour les problèmes en grande dimension [Rey, 02]. Nous allons rappeler la théorie des SVMs de point de vue mathématique pour illustrer son mécanisme.

1. THEORIE DE VAPNIK-CHERVONENKIS

Vapnik-Chervonenkis ont défini les notions de VC dimension et VC-entropie, notions à partir des quelles ils ont établi des conditions nécessaires et suffisantes à la convergence du risque empirique vers le risque réel.

- La VC-dimension h , d'un modèle d'apprentissage est la taille maximum d'un échantillon (un ensemble d'exemples) qui peut être pulvérisé ou séparé par le modèle.
- La VC-entropie d'un modèle est l'espérance du l'algorithme de la diversité de l'ensemble des fonctions que le modèle peut réaliser (du nombre de séparation différentes possibles), sur un échantillon de taille donnée.

La seule façon de garantir le risque consiste à contrôler la VC-dimension h du modèle, Vapnik propose donc d'appliquer un nouveau principe qu'il nomme principe de minimisation du

risque structurel MRS. Ce principe est basé sur la minimisation conjointe des deux causes d'erreurs : le risque empirique et l'intervalle de confiance (h), qui est une fonction croissante de la VC dimension.

Considérons une famille imbriquée de classes de fonction $\Phi_1 \subset \dots \subset \Phi_k$, la minimisation du MRS consiste à choisir la classe i de sorte à ce qu'une borne supérieure de l'erreur de généralisation puisse être minimisée. Pour résoudre le problème, on choisit a priori un ensemble de fonctions paramétrées par θ et on cherche à minimiser le risque en fonction de θ . Le choix d'un θ adapté est une étape cruciale, puisqu'un ensemble trop contraint peut ne pas parvenir à séparer les données initiales, et au contraire un ensemble trop libre peut aboutir à l'incapacité de généraliser (Figure 3.1).

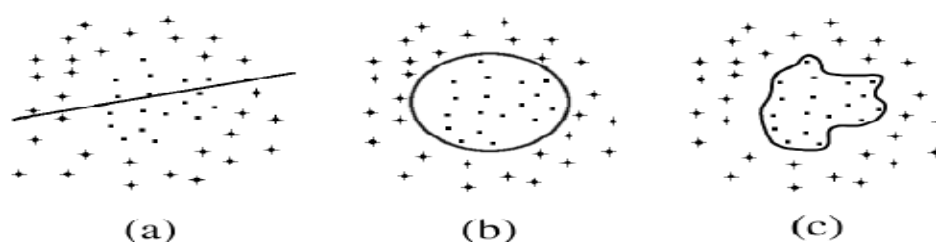


Figure 3.1 : Importance du choix de l'ensemble dans lequel est sélectionnée la fonction de décision.

▪ **Définition 3.1 (Dimension VC) [Cal, 03]**

Considérons la dimension VC d'un ensemble de fonction \mathcal{F} , notée h et supposons que la famille \mathcal{F} correspond aux droites $y = ax + y_0$ de \mathbb{R}^2 , la dimension VC de \mathcal{F} (voir figure 3.2) est 3 car on peut trouver une configuration de trois points séparables de toutes les façons possibles, par contre on ne peut trouver aucune configuration de 4 points (ou plus) rendant telle discrimination possible.

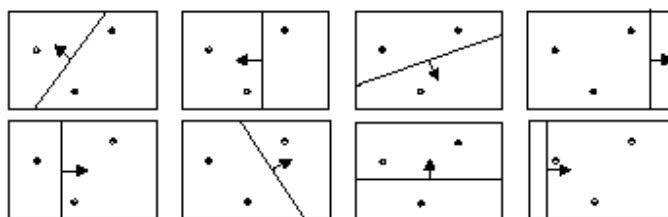


Figure 3.2 : La flèche représente le coté de la droite où les points seront classés positivement.

2. APPLICATION DES SVMs A LA CLASSIFICATION

En accord avec la théorie de l'apprentissage statistique, une fonction qui décrit correctement un ensemble d'apprentissage X et qui appartient à un ensemble de fonctions avec une dimension VC réduite (Vapnik-Chervonenkis), aura un bon pouvoir de généralisation, indépendamment de la dimension de l'espace de l'entrée. Basées sur ce principe, les SVM ont une approche systématique pour trouver une fonction linéaire, appartenant à un ensemble de fonctions avec une dimension VC basse [Men,02]. Les systèmes d'apprentissage appelés « Support Vector Machines », ou SVM en abrégé sont les algorithmes basés sur les trois principes mathématiques suivants :

- **Le principe de Fermat (1638) :** les point qui minimisent ou maximisent une fonction dérivable annulent sa dérivée. Ils sont appelés points stationnaires.
- **Le principe de Lagrange (1788) :** pour résoudre un problème d'optimisation sous contraintes, il suffit de rechercher un point stationnaire x_0 du Lagrangien L de la fonction f à optimiser, les f_i expriment les contraintes :

$$L(x, \alpha) = f(x) + \sum_{i=1}^k \alpha_i f_i(x) \quad (3.1)$$

où les a_i sont des constantes appelées coefficients (ou multiplicateurs) de Lagrange.

- **Le principe de Karush-Kuhn-Tucker KKT (1951) :** la relation de Kuhn-Tucker peuvent s'appliquer au cas qui nous intéresse. Avec des fonctions f et f_i convexes, il est même toujours possible de trouver un point-selle (x_0, α^*) qui vérifie :

$$\min_x L(x, \alpha^*) = L(x_0, \alpha^*) = \max_{\alpha \geq 0} L(x_0, \alpha) \quad (3.2)$$

Ces principes peuvent être appliqués à la recherche d'un hyperplan séparateur optimal, dans le cadre de la classification.

En effet, le principal objectif des SVM appliquées à la reconnaissance de formes est de construire un hyperplan séparateur optimal entre deux classes, c'est à dire, avec la plus grande marge (figure 3.3). Lorsqu'une solution linéaire n'est pas possible, la méthode réalise une projection de l'espace d'entrée X d'apprentissage dans un espace de caractéristiques Z de dimension plus importante, à travers une fonction noyau. Grâce à la liberté d'utiliser différents types de noyaux, l'hyperplan séparateur optimal correspond à des estimateurs non linéaires différents dans l'espace original.

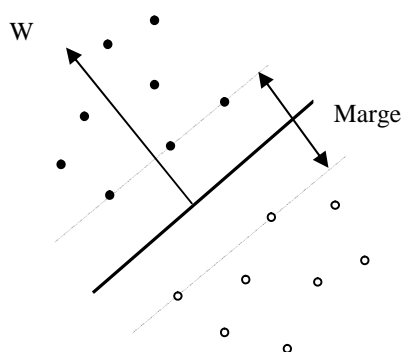


Figure 3.3 : Un hyperplan séparateur linéaire optimal et marge.

2.1 Cas linéairement séparable

Le modèle le plus simple de SVM est celui appelé linéaire de marge maximale. Il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables. Ce SVM cherche à séparer les deux classes de données par un hyperplan qui est équidistant des «frontières » de chaque classe.

2.1.1. Maximalisation de la marge [Ari, 04]

Reprenons la terminologie utilisée dans le cadre du perceptron, avec l'équation

$$w \cdot x + b = 0 \quad (3.3)$$

qui définit un hyperplan séparant deux classes. w correspond au vecteur normal à l'hyperplan.

La normale e de l'hyperplan séparateur est normalisée. Par conséquent, les vecteurs les plus proches de w , notés x_+ , x_- restent sur des hyperplans parallèles « canoniques » (figure 3.4).

$$\begin{aligned} \langle w, x_+ \rangle + b &= 1 \\ \text{et} \\ \langle w, x_- \rangle + b &= -1 \end{aligned} \quad (3.4)$$

On peut regarder x_+ , x_- comme projetés sur le vecteur unité $w/|w|$ (figure 3.5).

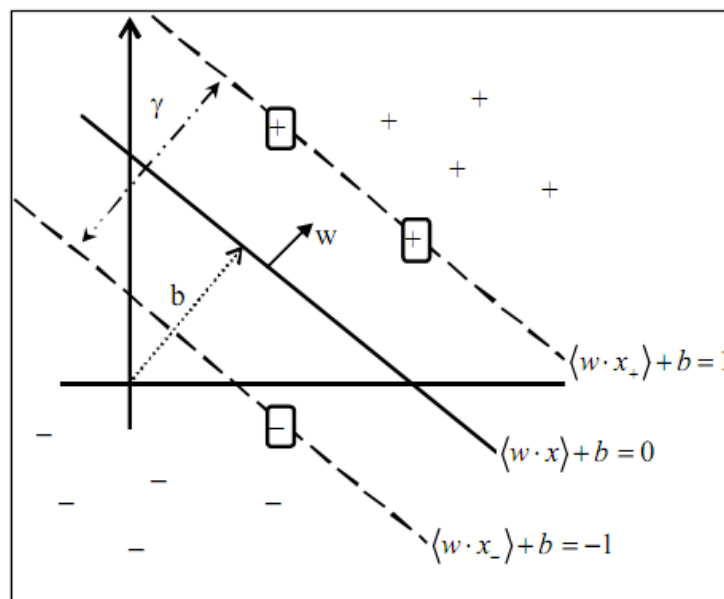


Figure 3.4: Hyperplan séparateur de deux classes (+) et (-). Il est défini comme de « Marge γ maximal », et situé au milieu des frontières entre classes.

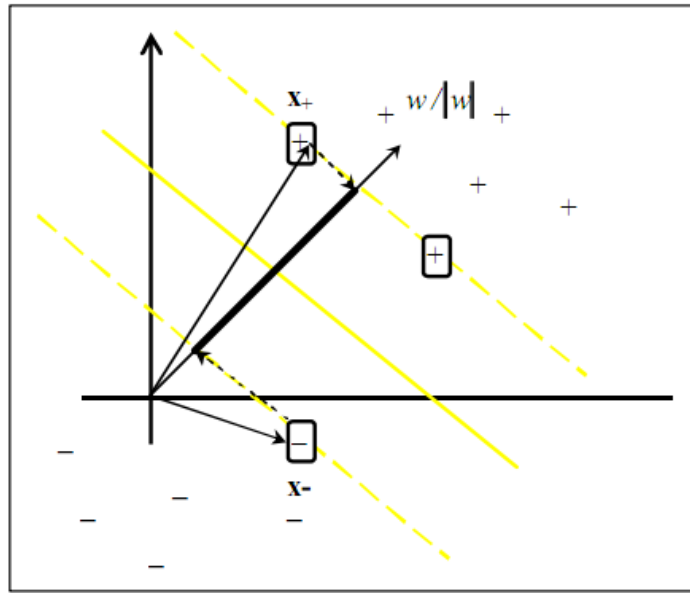


Figure 3.5 : La marge est calculée à partir du produit scalaire entre les vecteurs situés à la frontière de chaque classe et le vecteur unitaire normal de l'hyperplan séparateur.

De cette manière les deux classes sont définies comme suit :

$$\begin{cases} w \cdot x_i + b \geq +1 & \text{si } y_i = +1 \\ w \cdot x_i + b \leq -1 & \text{si } y_i = -1 \end{cases}$$

Ce qui s'écrit :

$$y_i(w \cdot x_i + b) - 1 \geq 0 \quad \forall i \quad (3.5)$$

Parmi tous les hyperplans de séparation possibles, le SVM considère l'hyperplan optimal comme étant celui qui maximise la marge (voir fig. 3.6). La marge du séparateur f et d'un point (x_i, y_i) est définie par $y f(x)$ (une marge négative correspond à une erreur de classification). Pour un ensemble d'exemples, la marge maximale est la distance aux points les plus proches. La distance de l'hyperplan définie précédemment à un point est donnée par [Sch 06] :

$$d(x_i) = \frac{|w x_i + b|}{\|w\|} \quad (3.6)$$

où $\|w\|$ est la norme euclidienne du vecteur caractéristique w , et b l'ordonnée à l'origine de l'hyperplan. Il est ainsi aisé de formuler la distance entre les deux hyperplans correspondant aux classes $\{-1, +1\}$:

$$d(wx + b = +1, wx + b = -1) = \frac{2}{\|w\|}. \quad (3.7)$$

2.1.2. Résolution du problème de minimisation

La solution à un problème de minimisation est généralement obtenue en annulant la dérivée de la fonction étudiée. En présence de contraintes exprimées sous la forme d'inégalités, le problème de minimisation peut se résoudre dans l'espace dual. Cette formulation du problème équivaut à injecter les contraintes dans la fonction objective. La formulation du problème dans l'espace dual peut être trouvée dans la littérature, nous n'allons pas l'explicitier ici. La fonction de décision peut se réécrire ainsi :

$$f(x) = \sum_t a_t y_t x_t \cdot x + b \quad (3.8)$$

avec $\sum_t a_t y_t = 0$ et $a_t \geq 0$.

La solution de ce problème se traduit par une somme pondérée de produits scalaires entre les exemples d'apprentissage. Les a_t sont les variables ajoutées pour l'expression du problème dans l'espace dual (multiplicateurs de Lagrange).

Les vecteurs supports sont les exemples d'apprentissage pour lesquels $a_t > 0$. La maximisation de la marge peut être vue comme une gestion de la complexité du problème, puisque le SVM ne mémorise que ces exemples pour discriminer les deux classes. Le paramètre a est nul pour les autres exemples. La fig.3.6 représente un séparateur linéaire et ses vecteurs supports associés et illustre le principe de la marge maximale.

Ce principe suit celui du Minimum Description Length postulant que la meilleure représentation des données, en terme de généralisation, est celle qui a nécessité la plus faible quantité d'information pour la décrire.

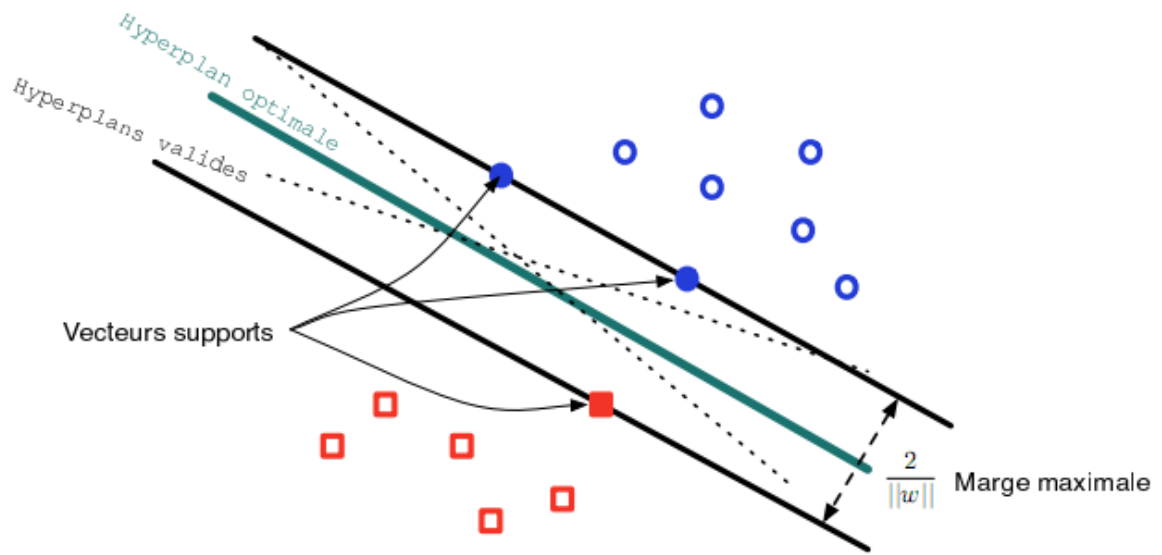


Figure 3.6 : Maximisation de la marge et vecteurs supports. La distance à maximiser est donnée par $2/\|w\|$. Les vecteurs supports sont représentés par les points sur la marge.

2.1.3. Vecteurs support [Ari, 04]

Sous les conditions de Karush-Kuhn-Tucker (KKT) pour un problème d'optimisation convexe, deux valeurs a_i sont possibles pour les vecteurs d'apprentissage : soit ils sont à l'extérieur de la contrainte ($y_i(\langle w, x_i \rangle + b) - 1 > 0$) soit ils se trouvent dans la frontière ($y_i(\langle w, x_i \rangle + b) - 1 = 0$). Les paramètres sont $a_i = 0$ pour le premier cas et $a_i > 0$ dans le deuxième. On en déduit la valeur de w , puis celle de b .

$$a_i[y_i(\langle w, x_i \rangle + b) - 1] = 0 \quad i = 1 \dots l \quad (3.9)$$

C'est un résultat où une condition implique que seuls les vecteurs x_i à une distance « 1 » de l'hyperplan de marge maximale possédant un paramètre $a_i > 0$ sont pris en compte pour le calcul de w . Ils sont appelés « vecteurs de support (sv) ». Les vecteurs qui ne sont pas de support n'ont aucune influence dans la solution.

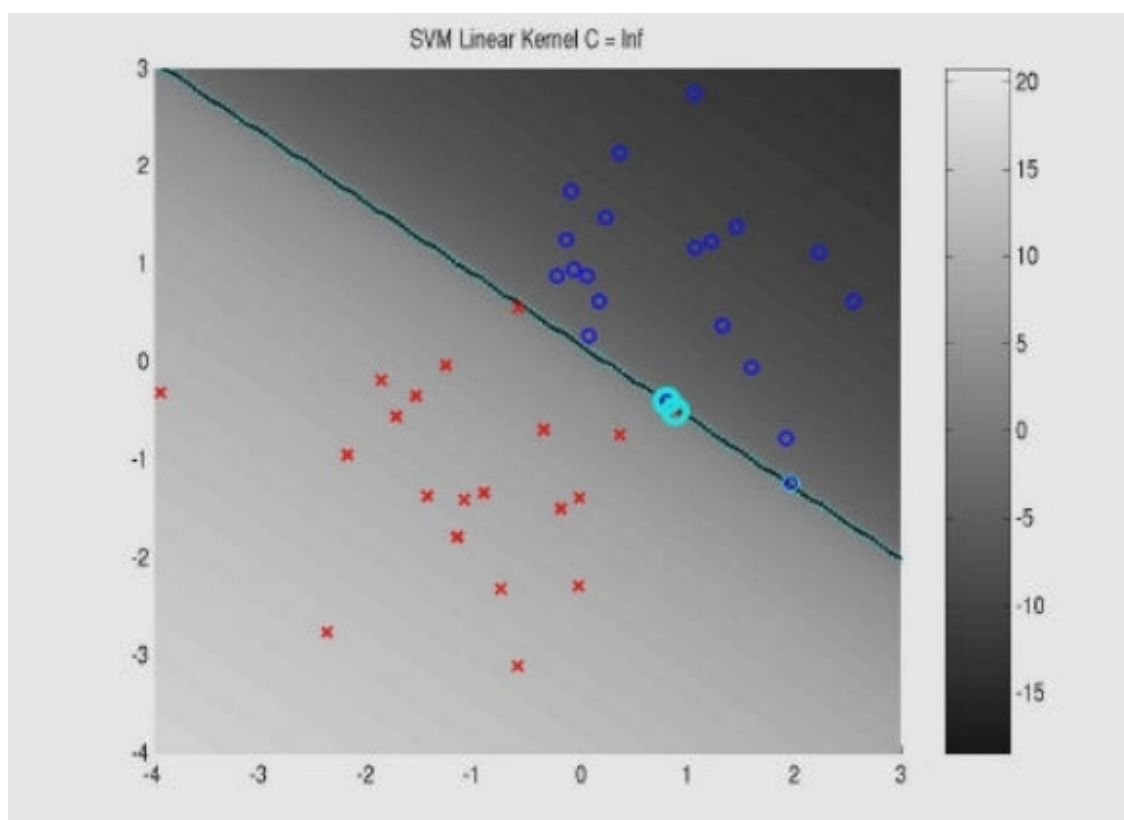


Figure 3.7 : Classification de deux classes de données avec un SVM linéaire.

Les vecteurs support ont été encerclés [Ari 04].

La fonction de décision pour la classification de vecteurs inconnus u est donnée par :

$$f(u) = \text{sign}\left(\sum_{i=1}^m a_i y_i x_i \cdot u\right) + b \quad (3.10)$$

où m est le nombre de vecteurs de support.

2.2. Cas linéairement non séparable [Ari, 04]

Le classificateur de marge maximale ne peut pas être utilisé dans la plupart des problèmes réels : si les données ont été affectées par le bruit, il n'y a pas de séparation linéaire entre elles. Dans ce cas, le problème d'optimisation ne peut pas être résolu.

Pour surmonter ces inconvénients, de nouvelles mesures de la marge ont été proposées. Ces mesures tolèrent le bruit et prennent en compte les données d'apprentissage en plus de celles qui sont dans les frontières de la classe.

Le problème d'optimisation initial était :

$$\text{Minimiser } \frac{1}{2} ||w||^2 \quad \text{avec } y_i(w \cdot x_i + b) - 1 \leq 0 \quad i = 1, \dots, l \quad (3.11)$$

Il s'agit dans ce nouveau cas (dit de « marges douces ») de relâcher les contraintes de la marge.

On introduit alors des variables d'écart (normalisées par rapport

à w) $\xi_i \geq 0$, $i = 1, \dots, l$ dans la définition des contraintes :

$$\begin{cases} w \cdot x_i + b \geq 1 - \xi_i & \text{si } y_i = +1 \\ w \cdot x_i + b \leq -1 + \xi_i & \text{si } y_i = -1 \end{cases} \quad (3.12)$$

ce qui s'écrit :

$$y_i w \cdot x_i + b \geq 1 - \xi_i \quad i = 1 \dots l \quad (3.13)$$

Quand une erreur de classification intervient, la variable ξ_i a une valeur plus grande que 1, donc ξ_i est une borne supérieure du nombre d'erreurs à l'apprentissage. De là, un moyen naturel pour pénaliser les erreurs est de remplacer la fonction précédente à minimiser par $\frac{1}{2} ||w||^2 + C \sum \xi_i$. D'où, le fait de choisir une valeur pour le paramètre C revient à définir une valeur pour w en minimisant x pour cette valeur de w .

$$\begin{aligned} L &= \frac{1}{2} ||w||^2 + C \sum_{i=1}^l \xi_i = \sum_{i=1}^l a_i [y_i (w \cdot x_i + b) - 1 + \xi_i] - \sum_{i=1}^l r_i \xi_i \\ \frac{\partial L}{\partial w} &= w - \sum_{i=1}^l a_i y_i x_i = 0 \\ \frac{\partial L}{\partial b} &= \sum_{i=1}^l a_i y_i = 0 \\ \frac{\partial L}{\partial \xi_i} &= C - a_i - r_i = 0 \end{aligned} \quad (3.14)$$

$$L_D = \sum_{i=1}^l a_i - \frac{1}{2} \sum_{i,j=1}^l a_i a_j y_i y_j \langle x_i \cdot x_j \rangle$$

La seule différence avec le cas linéairement séparable est que

$$\begin{aligned} C - a_i - r_i &= 0 \\ r_i &> 0 \\ a_i &\leq C \quad i = 1 \dots l \end{aligned} \quad (3.15)$$

Le problème d'optimisation est équivalent au cas de la marge maximale, avec une contrainte additionnelle (3.15). Cette formulation est connue comme « contrainte de boîte », car chaque valeur a_i est limitée par 0 d'un côté et par C de l'autre. C s'est révélé être un compromis entre la précision et la régularisation (le contrôle de l'erreur).

$$\begin{aligned} a_i [y_i (w \cdot x_i + b) - 1 + r_i] &= 0 \quad i = 1 \dots l \\ \xi_i (a_i - C) &= 0 \quad i = 1 \dots l \end{aligned} \quad (3.16)$$

Ces conditions impliquent que les variables d'écart soient différentes de zéro quand, c'est à dire, quand leur marge est moins de $1 / |w|$. Les vecteurs pour lesquels $0 < a_i < C$, sont considérés comme vecteurs support.

La fonction de décision pour la classification de vecteurs inconnus u reste :

$$f(u) = \text{sign} \left(\sum_{i=1}^m a_i y_i x_i \cdot u \right) + b \quad (3.17)$$

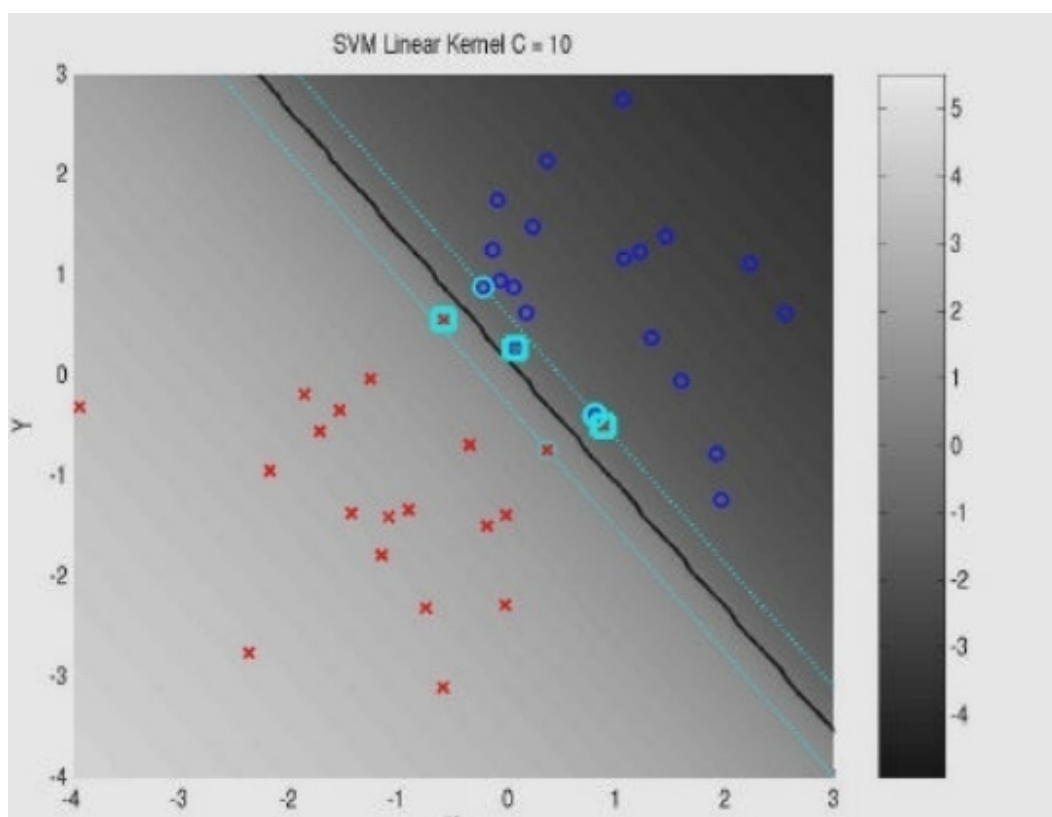


Figure 3.8 : Classification de deux classes de données par une SVM linéaire de « marges douces ». La valeur de $C = 10$. Des erreurs de classification et de vecteurs à l'intérieur de la marge (points encadrés) sont permises. Les points encadrés sont les vecteurs de support [Ari 04].

2.3 Les fonctions noyau, le changement de dimension et le cas non linéaire

Le classificateur à marge maximale que nous venons de présenter, permet d'obtenir de très bons résultats lorsque les données sont linéairement séparables. L'intérêt principal d'un classificateur de ce type réside dans le fait que l'on en contrôle facilement la capacité et donc le pouvoir de généralisation. Naturellement, un grand nombre de jeux de données sont non-linéairement séparables. Pour classer ce genre de données on pourrait utiliser une fonction de décision non-linéaire. Géométriquement, cela reviendrait à avoir une (hyper)courbe qui marquerait la frontière entre les exemples positifs et négatifs. Les fonctions de décision dites noyau (kernel en anglais) ont été proposées pour pouvoir construire des algorithmes non-linéaire

à partir d'algorithmes linéaires en calculant le produit vectoriel non plus dans l'espace de caractéristiques est donc définie par une projection non-linéaire :

$$\begin{aligned} \Phi : \mathcal{R}^N &\rightarrow F \\ x &\mapsto \Phi(x) \quad \text{où } N \ll \dim(F) \end{aligned} \quad (3.18)$$

Permettant d'obtenir un nouvel ensemble d'apprentissage :

$$(\Phi(x_1), y_1), \dots, (\Phi(x_l), y_l) \in F \times \{\pm 1\} \quad (3.19)$$

Une fois les données modifiées nous les utilisons pour faire l'apprentissage. Dans la figure (3.8) nous pouvons voir un exemple simple de cette transformation

$$\begin{aligned} \Phi : \mathcal{R}^2 &\rightarrow \mathcal{R}^3 \\ (x_1, x_2) &\mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}x_1x_2, x_2^2) \end{aligned} \quad (3.20)$$

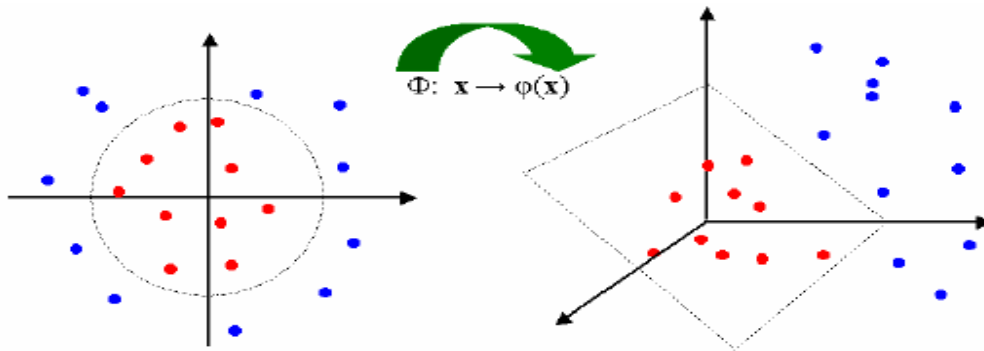


Figure 3.9 : Illustration de l'effet du changement d'espace (mapping) par une fonction noyau [Lad].

La dimension de l'espace de caractéristiques (feature space) est généralement très élevée. Cela ne pose pas de problème pour notre classificateur à marge maximale vu que sa formulation duale fixe le nombre de variables à déterminer en fonction de la taille de l'ensemble d'apprentissage (training set). Les nouveaux axes (figure 3.8) contiennent une sur-génération d'informations par rapport aux précédents. Ce qui permet idéalement d'effectuer une discrimination linéaire là où auparavant ce n'était pas possible.

2.3.1 Mesure de la similarité

De manière générale, il peut-être utile de savoir à quel point un exemple est similaire à un autre. Pour faire cela, on utilise souvent en mathématique le produit scalaire qui moyennant une normalisation, correspond au cosinus de l'angle entre deux vecteurs. En utilisant le mapping Φ introduit à la section précédente, on peut définir une mesure de similarité dans le feature space :

$$K(x, y) = \langle \Phi(x), \Phi(y) \rangle \quad (3.21)$$

La fonction $k(x, y)$ est appelée noyau (kernel).

▪ Condition de Mercer

La matrice contenant les similarités entre tous les exemples du training set :

$$G = \begin{pmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{pmatrix} \quad (3.22)$$

est appelée matrice de Gram.

Théorème (Condition de Mercer) [Cal, 03]

La fonction : $k(x, z) : X \times X \rightarrow \mathbb{R}$ est un noyau SSI:

$$G = (K(x_i, x_j))_{i,j=1}^n \quad (3.23)$$

est définie positive possède les trois propriétés fondamentales du produit scalaire :

- ✓ Positivité : $k(x_i, x_i) \geq 0$.
- ✓ Symétrie : $k(x_i, x_j) = k(x_j, x_i)$.
- ✓ Inégalité de Cauchy-Schwartz : $|k(x_i, x_j)| \leq \sqrt{k(x_i, x_i) k(x_j, x_j)}$.

La condition de Mercer nous indique si une fonction est un noyau mais nous n'avons aucun renseignement sur le mapping Φ (et donc sur le feature space) induit par ce noyau.

2.3.2 Les fonctions Noyau (Kernel)

La solution s'exprime sous la forme :		
$f(x) = \sum \alpha_i^* y_i . K(x_i, x_j) + b^*$		
Fonction de Kernel (noyau)	Forme fonctionnelle	Commentaire
- Polynomiale	$K(x, y) = (x.y + c)^n$	La puissance n est déterminée <i>a priori</i> par l'utilisateur
- Fonctions gaussiennes RBF	$K(x, y) = \exp\left(-\frac{\ x - y\ ^2}{2\sigma^2}\right)$	L'écart type σ^2 , commun à tous les noyaux, est spécifié <i>a priori</i> par l'utilisateur
- Fonctions sigmoïdes	$K(x, y) = \tanh((a(x.y) - b))$	Le théorème de Mercer n'est vérifié que pour certaines valeurs de a et b .

Tableau. 3.1 Les fonctions noyau les plus courantes avec leurs paramètres [Lad].

3. LES AVANTAGES ET LES INCONVENIENTS DES SVMs

➤ **Avantage**

SVM est une méthode de classification intéressante car le champ de ses applications est large, parmi ses avantages nous avons :

- Un grand taux de classification et de généralisation par rapport aux méthodes classiques.
- Elle nécessite moins d'effort pour designer l'architecture adéquate (petit nombre de paramètre à régler ou à estimer).
- La résolution du problème est convertie en résolution d'un problème quadratique convexe dont la solution est unique et donnée par des méthodes mathématiques classiques de programmation quadratique.

➤ **Inconvénients**

L'inconvénient majeur du classificateur SVM est qu'il est désigné ou conçu pour la classification binaire (la séparation entre deux classes une +1 et l'autre -1).

Conclusion

Dans ce chapitre, on a tenté de présenter d'une manière simple et complète le concept de système d'apprentissage introduit par Vladimir Vapnik, les «Machine à Vecteur de Support». Cette méthode de classification est basée sur la recherche d'un hyperplan qui permet de séparer au mieux des ensembles de données. On a exposé les cas linéairement séparable et les cas non linéairement séparables qui nécessitent l'utilisation de fonction noyau (Kernel) pour changer d'espace.