Support Vector Machines

Géométrie, optimisation convexe et noyaux

Yoann Pull

Laboratoire d'Economie d'Orléans
Square Research Center

Version : 23 octobre 2025

À propos du cours

Ce cours est dispensé au sein du Master ESA de l'Université d'Orléans. Il s'adresse à des étudiants de niveau Master avec pour prérequis : analyse, algèbre linéaire et un peu d'optimisation. Certaines sections mobilisent des développements mathématiques plus exigeants ; des ressources complémentaires sont signalées au fil du texte.

Contact — Pour toute coquille, erreur ou suggestion, merci d'écrire à yoann.pull.pro@gmail.com.

Code des graphiques — Le code utilisé pour générer les figures est disponible sur GitHub : https://github.com/YoannPull/svm_courses.

Sections techniques.

Les sections précédées de ce pictogramme sont des passages plus techniques. À la fin de ces sections, un encadré intitulé *"Approfondir et mieux comprendre"* propose des pistes de lecture.

Références Les références principales utilisées pour l'écriture de ce cours sont :

- Bishop (2006)
- Hastie et al. (2009)
- Boyd and Vandenberghe (2004)
- Hurlin (2025)

Table des matières

1 Un bref historique des SVM et des classifieurs linéaires

Les premiers jalons remontent au **Perceptron** de Rosenblatt (1958), un algorithme itératif qui met à jour $(\boldsymbol{w}, b)$ par corrections successives. Très vite, Novikoff (1962) en établit la convergence sous l'hypothèse de séparabilité linéaire, fixant ainsi un premier cadre théorique pour les classifieurs linéaires.

Dans les années 1960–1970, Vapnik (1998) (avec Chervonenkis) déplacent le cœur du problème : plutôt que d'expliquer *comment* apprendre, ils cherchent *quand* l'apprentissage généralise. Avec la dimension VC et le principe de *Structural Risk Minimization* (SRM), ils proposent une boussole : contrôler la complexité du modèle pour garantir la performance hors échantillon. Ce glissement, de la simple mise à jour de poids vers des bornes de généralisation, prépare l'émergence d'une méthode d'optimisation à la fois géométrique et statistiquement fondée.

Au début des années 1990, la version moderne des Support Vector Machines s'impose. Boser et al. (1992) formulent la maximisation de la *marge* comme un programme quadratique convexe, et rendent naturelles les frontières non linéaires via l'*astuce du noyau*—déjà esquissée par Aizerman et al. (1964)—qui remplace explicitement les projections de caractéristiques par des produits scalaires implicites. Cortes and Vapnik (1995) généralisent au cas non séparables avec la *marge souple* (paramètre C) et la *hinge loss*, conciliant robustesse aux erreurs et contrôle de la complexité.

Sur le plan théorique, l'ancrage dans les espaces de Hilbert à noyau reproduisant (RKHS) et le *théorème de représentation* (Kimeldorf and Wahba, 1971; Schölkopf and Smola, 2002) expliquent pourquoi la solution optimale s'écrit comme une combinaison linéaire de noyaux centrés sur quelques observations seulement : les *vecteurs de support*. En bref, on passe d'une règle locale de mise à jour (Perceptron) à un principe global d'optimisation (SVM) qui marie géométrie, régularisation et fondements statistiques de la généralisation.

2 Rappels de géométrie euclidienne

Avant d'aborder les SVM, nous fixons les notations et rappelons quelques outils de géométrie vectorielle utilisés dans tout le cours.

2.1 Espace euclidien, produit scalaire et norme

Définition 2.1 (Espace euclidien)**.** Un espace euclidien est un espace vectoriel réel E de dimension finie muni d'un produit scalaire $\langle \cdot, \cdot \rangle : E \times E \to \mathbb{R}$ symétrique, bilinéaire et défini positif. La norme associée est $\|\boldsymbol{x}\| = \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$ et la distance $\mathrm{dist}(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|$.

On travaille dans $\mathbb{R}^d$ muni du **produit scalaire euclidien**

$$\langle \boldsymbol{x}, \boldsymbol{z} \rangle = \sum_{k=1}^{d} x_k z_k,$$

et de la **norme euclidienne** associée

$$\|\boldsymbol{x}\| = \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle} = \Big(\sum_{k=1}^{d} x_k^2 \Big)^{1/2}.$$

Définition 2.2 (Vecteurs orthogonaux, colinéaires)**.** Deux vecteurs $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$ sont *orthogonaux* si $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = 0$. Ils sont *colinéaires* s'il existe $\lambda \in \mathbb{R}$ tel que $\boldsymbol{u} = \lambda \boldsymbol{v}$.

Remarque 2.1 (Pythagore). Si $\boldsymbol{u} \perp \boldsymbol{v}$, alors $\|\boldsymbol{u} + \boldsymbol{v}\|^2 = \|\boldsymbol{u}\|^2 + \|\boldsymbol{v}\|^2$. C'est la généralisation du théorème de Pythagore.

2.2 Inégalités classiques

Théorème 2.1 (Cauchy–Schwarz)**.** *Pour tout* $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$*,*

$$|\langle \boldsymbol{u}, \boldsymbol{v} \rangle| \;\leq\; \|\boldsymbol{u}\|\,\|\boldsymbol{v}\|,$$

avec égalité ssi $\boldsymbol{u}$ *et* $\boldsymbol{v}$ *sont colinéaires.*

Théorème 2.2 (Inégalité triangulaire)**.** *Pour tout* $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$*,*

$$\|\boldsymbol{u} + \boldsymbol{v}\| \;\leq\; \|\boldsymbol{u}\| + \|\boldsymbol{v}\|.$$

Esquisse de preuve. *Par développement,* $\|\boldsymbol{u}+\boldsymbol{v}\|^2 = \|\boldsymbol{u}\|^2 + \|\boldsymbol{v}\|^2 + 2\langle \boldsymbol{u}, \boldsymbol{v}\rangle \leq (\|\boldsymbol{u}\| + \|\boldsymbol{v}\|)^2$ *via Cauchy–Schwarz, puis on prend la racine.*

Remarque 2.2 (Longueur d'un vecteur (Pythagore)). Dans la base canonique, $\|\boldsymbol{x}\| = \sqrt{x_1^2 + \cdots + x_d^2}$: c'est la « longueur » de $\boldsymbol{x}$, déduite du théorème de Pythagore en dimension d.

2.3 Sous-espaces affines, hyperplans et vecteurs normaux

Définition 2.3 (Sous-espace affine)**.** Un *sous-espace affine* est un ensemble de la forme

$$\mathcal{A} = \boldsymbol{x}_0 + \mathcal{V} \;=\; \{\boldsymbol{x}_0 + \boldsymbol{v} : \boldsymbol{v} \in \mathcal{V}\},$$

où $\boldsymbol{x}_0 \in \mathbb{R}^d$ et $\mathcal{V}$ est un sous-espace vectoriel.

Définition 2.4 (Hyperplan)**.** Un *hyperplan* de $\mathbb{R}^d$ est un sous-espace affine de dimension $d-1$, équivalent à un ensemble

$$H \;=\; \{\boldsymbol{x} \in \mathbb{R}^d : \langle \boldsymbol{w}, \boldsymbol{x}\rangle + b = 0\},$$

où $\boldsymbol{w} \in \mathbb{R}^d \setminus \{0\}$ et $b \in \mathbb{R}$. Le vecteur $\boldsymbol{w}$ est un **vecteur normal** à H (orthogonal à toute direction de H).

Remarque 2.3. Il est important de noter qu'un hyperplan sépare l'espace en deux. C'est cette propriété qui nous servira de critère de classification par la suite.

Définition 2.5 (Droite normale à un hyperplan par un point)**.** Soit $H = \{\boldsymbol{x} : \langle \boldsymbol{w}, \boldsymbol{x}\rangle + b = 0\}$ et $\boldsymbol{x}_0 \in \mathbb{R}^d$. La *droite normale* à H passant par $\boldsymbol{x}_0$ est

$$\mathcal{N}(\boldsymbol{x}_0) \;=\; \{\boldsymbol{x}(t) = \boldsymbol{x}_0 + t\,\boldsymbol{w} : t \in \mathbb{R}\}.$$

2.4 Projections : scalaire et vectorielle

Définition 2.6 (Projection scalaire)**.** La *projection scalaire* de $\boldsymbol{v}$ sur $\boldsymbol{u} \neq 0$ est

$$\mathrm{comp}_{\boldsymbol{u}}(\boldsymbol{v}) \;=\; \frac{\langle \boldsymbol{v}, \boldsymbol{u}\rangle}{\|\boldsymbol{u}\|}.$$

Définition 2.7 (Projection vectorielle sur une direction)**.** La *projection vectorielle* de $\boldsymbol{v}$ sur la droite dirigée par $\boldsymbol{u} \neq 0$ est

$$\mathrm{proj}_{\boldsymbol{u}}(\boldsymbol{v}) \;=\; \frac{\langle \boldsymbol{v}, \boldsymbol{u}\rangle}{\|\boldsymbol{u}\|^2}\,\boldsymbol{u}.$$

Proposition 2.1 (Projection orthogonale sur un hyperplan)**.** *Soit* $H = \{\boldsymbol{x} : \langle \boldsymbol{w}, \boldsymbol{x}\rangle + b = 0\}$ *avec* $\boldsymbol{w} \neq 0$ *et* $\boldsymbol{x}_0 \in \mathbb{R}^d$ *quelconque. La projection orthogonale* $\Pi_H(\boldsymbol{x}_0)$ *de* $\boldsymbol{x}_0$ *sur* H *est*

$$\Pi_H(\boldsymbol{x}_0) \;=\; \boldsymbol{x}_0 \;-\; \frac{\langle \boldsymbol{w}, \boldsymbol{x}_0\rangle + b}{\|\boldsymbol{w}\|^2}\,\boldsymbol{w}.$$

Justification. *On retranche à* $\boldsymbol{x}_0$ *la composante selon la normale* $\boldsymbol{w}$*.*

2.5 Distances point–hyperplan (signée et non signée)

Définition 2.8 (Distance point–hyperplan)**.** La distance (non signée) d'un point $\boldsymbol{x}_0$ à l'hyperplan $H = \{\boldsymbol{x} : \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = 0\}$ est

$$\operatorname{dist}(\boldsymbol{x}_0, H) \;=\; \frac{|\langle \boldsymbol{w}, \boldsymbol{x}_0 \rangle + b|}{\|\boldsymbol{w}\|}.$$

Définition 2.9 (Distance signée)**.** En fixant l'orientation par la normale $\boldsymbol{w}$, la *distance signée* est

$$\operatorname{sdist}_{\boldsymbol{w}}(\boldsymbol{x}_0, H) \;=\; \frac{\langle \boldsymbol{w}, \boldsymbol{x}_0 \rangle + b}{\|\boldsymbol{w}\|}.$$

Elle est positive du côté où $\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b > 0$ et négative de l'autre.

Remarque 2.4 (Distance entre deux hyperplans parallèles). Si $H_1 = \{\boldsymbol{x} : \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b_1 = 0\}$ et $H_2 = \{\boldsymbol{x} : \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b_2 = 0\}$ avec la *même* normale $\boldsymbol{w}$, alors

$$\operatorname{dist}(H_1, H_2) \;=\; \frac{|b_1 - b_2|}{\|\boldsymbol{w}\|}.$$

Proposition 2.2 (Décomposition orthogonale par rapport à un hyperplan)**.** *Pour tout $\boldsymbol{x} \in \mathbb{R}^d$,*

$$\boldsymbol{x} \;=\; \underbrace{\Big(\boldsymbol{x} - \frac{\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b}{\|\boldsymbol{w}\|^2}\, \boldsymbol{w}\Big)}_{\textit{projection sur } H} \;+\; \underbrace{\frac{\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b}{\|\boldsymbol{w}\|^2}\, \boldsymbol{w}}_{\textit{composante normale}} \quad .$$

Les deux composantes sont orthogonales.

3 Perceptron de Rosenblatt

Nous rappelons d'abord la règle de décision linéaire et la partition de l'espace induite par un hyperplan, puis nous introduisons le perceptron, sa fonction de perte, le problème d'optimisation associé, et enfin l'algorithme et son résultat de convergence classique.

Soit

$$\mathcal{T} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$$

nos données d'entraînement composées de N individus dans un espace $\mathcal{X} \subseteq \mathbb{R}^d$, telles que $\forall i \in \{1, \ldots, N\}$, $\boldsymbol{x}_i \in \mathcal{X}$ et $y_i \in \{-1, 1\}$. L'objectif est de séparer $\mathcal{T}$ en deux classes à l'aide d'un hyperplan

$$h(\boldsymbol{x}) \;=\; \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b,$$

où $\boldsymbol{w} \in \mathbb{R}^d$ est un vecteur normal à l'hyperplan et $b \in \mathbb{R}$ un biais.

Cet hyperplan induit deux demi-espaces

$$H^+ \;=\; \{\boldsymbol{x} \in \mathcal{X} :\; h(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b \geq 0\}, \qquad H^- \;=\; \{\boldsymbol{x} \in \mathcal{X} :\; h(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b < 0\},$$

auxquels on associe respectivement les classes $+1$ et -1, la Figure 1 illustre ce cas.

Définition 3.1 (Fonction sign et règle de décision)**.** Soit $h(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b$ l'équation d'un hyperplan dans $\mathcal{X}$. On définit la fonction sign : $\mathbb{R} \to \{-1, 1\}$ par

$$\operatorname{sign}(t) = \begin{cases} 1, & \text{si } t \geq 0, \\ -1, & \text{si } t < 0. \end{cases}$$

Pour un nouvel individu $\boldsymbol{x}_{\text{new}}$, la classe prédite est

$$\widehat{y} \;=\; \operatorname{sign}\big(h(\boldsymbol{x}_{\text{new}})\big) \;=\; \operatorname{sign}\big(\langle \boldsymbol{w}, \boldsymbol{x}_{\text{new}} \rangle + b\big).$$

Remarque. Par convention, les points $\boldsymbol{x}$ tels que $h(\boldsymbol{x}) = 0$ (sur l'hyperplan) sont rattachés à la classe $+1$. Si l'on souhaite distinguer ce cas, on peut utiliser $\operatorname{sign}_0 : \mathbb{R} \to \{-1, 0, 1\}$ avec $\operatorname{sign}_0(0) = 0$.

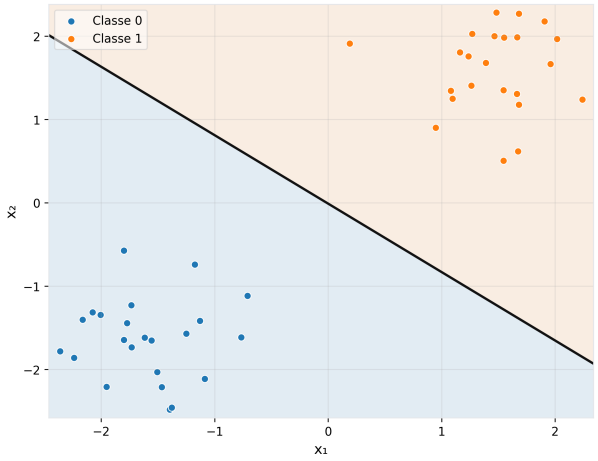


FIGURE 1 – Illustration d'un hyperplan séparateur et de la règle de décision.

3.1 Pertes et formulation du perceptron

Fil directeur. Le perceptron cherche un hyperplan qui rend *toutes* les marges signées $m_i = y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b)$ positives. La perte du perceptron ne « regarde » que les points mal classés (ou sur la frontière) et pousse $(\boldsymbol{w}, b)$ dans la direction qui corrige l'erreur courante.

Définition 3.2 (Score linéaire et marge signée)**.** Étant donné un hyperplan $h(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b$, on appelle *score linéaire* la quantité $h(\boldsymbol{x})$. Pour $(\boldsymbol{x}_i, y_i)$ avec $y_i \in \{-1, 1\}$, la *marge signée* est

$$m_i(\boldsymbol{w}, b) \;=\; y_i\, h(\boldsymbol{x}_i) \;=\; y_i\,(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b). \tag{3.1}$$

Alors $m_i > 0$ signifie que $\boldsymbol{x}_i$ est correctement classé, $m_i < 0$ qu'il est mal classé, et $m_i = 0$ qu'il est sur l'hyperplan.

Remarque 3.1 (Pourquoi le produit $y_i\, h(\boldsymbol{x}_i)$?). Sans le facteur y_i, le signe de $h(\boldsymbol{x}_i)$ s'interprète différemment selon la classe. En multipliant par $y_i \in \{-1, 1\}$, on *unifie* la contrainte de bonne classification :

$$(\boldsymbol{x}_i, y_i) \text{ correctement classé} \quad \Longleftrightarrow \quad y_i\, h(\boldsymbol{x}_i) > 0.$$

Cette écriture unique simplifie la modélisation et les mises à jour.

Définition 3.3 (Perte du perceptron)**.** La *perceptron loss* pénalise uniquement les observations mal classées (ou à marge nulle) :

$$\ell(z) = \max(0, -z), \qquad L(\boldsymbol{w}, b) = \sum_{i=1}^{N} \ell\big(y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b)\big) = \sum_{i=1}^{N} \max\big(0, -y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b)\big). \tag{3.2}$$

Proposition 3.1 (Problème d'optimisation convexe)**.** *Le perceptron résout*

$$\min_{\boldsymbol{w} \in \mathbb{R}^d,\, b \in \mathbb{R}} L(\boldsymbol{w}, b) = \min_{\boldsymbol{w}, b} \sum_{i=1}^{N} \max\big(0, -y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b)\big). \tag{3.3}$$

La fonction L est convexe en $(\boldsymbol{w}, b)$ (maximum de fonctions affines), mais non différentiable lorsque $y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) = 0$.

Proposition 3.2 (Sous-gradients)**.** *En notant $h_i = \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b$ et $\mathcal{M}(\boldsymbol{w}, b) = \{i : \; y_i h_i < 0\}$ l'ensemble des erreurs, un sous-gradient de L en $(\boldsymbol{w}, b)$ est*

$$\frac{\partial L(\boldsymbol{w}, b)}{\partial \boldsymbol{w}} = -\sum_{i \in \mathcal{M}(\boldsymbol{w}, b)} y_i\, \boldsymbol{x}_i, \qquad \frac{\partial L(\boldsymbol{w}, b)}{\partial b} = -\sum_{i \in \mathcal{M}(\boldsymbol{w}, b)} y_i. \tag{3.4}$$

Au point non différentiable $y_i h_i = 0$, tout vecteur entre $\mathbf{0}$ et $-y_i \boldsymbol{x}_i$ est un sous-gradient admissible du i-ième terme.

3.2 Algorithme et convergence

L'algorithme du perceptron est une descente de sous-gradient *en ligne* : on parcourt les exemples et l'on met à jour uniquement en cas d'erreur (ou marge nulle). L'écriture augmentée incorpore le biais dans le vecteur de poids.

Remarque 3.2 (Variables augmentées). Posons

$$\tilde{\boldsymbol{x}}_i = \begin{pmatrix} \boldsymbol{x}_i \\ 1 \end{pmatrix} \in \mathbb{R}^{d+1}, \qquad \tilde{\boldsymbol{w}} = \begin{pmatrix} \boldsymbol{w} \\ b \end{pmatrix} \in \mathbb{R}^{d+1},$$

de sorte que $h(\boldsymbol{x}_i) = \langle \tilde{\boldsymbol{w}}, \tilde{\boldsymbol{x}}_i \rangle$ et que les mises à jour s'écrivent de manière compacte.

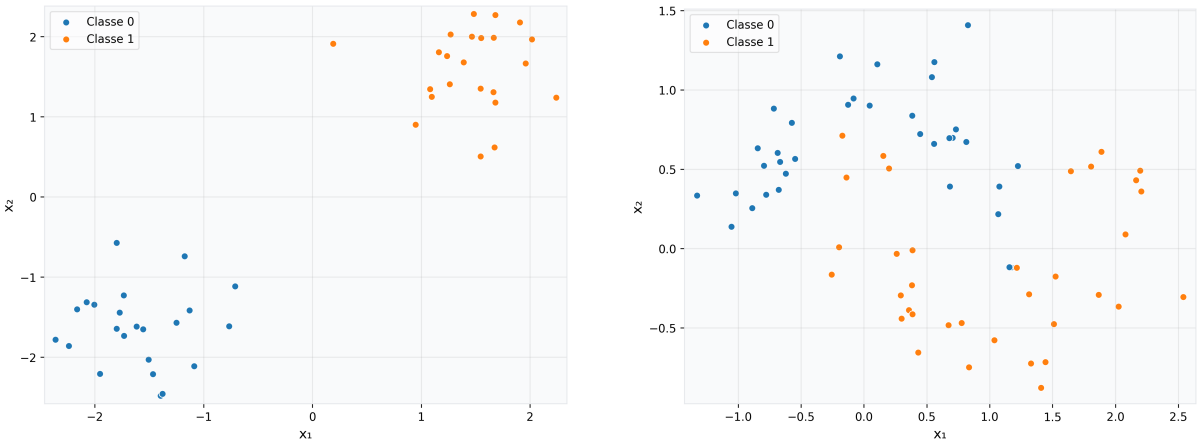
Algorithm 1 Algorithme du perceptron (descente de sous-gradient en ligne)

Require: Données $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$, pas $\eta > 0$, itérations T
Initialiser $\tilde{\boldsymbol{w}}^{(0)} = \mathbf{0} \in \mathbb{R}^{d+1}$ ▷ ou petite valeur aléatoire
for $t = 0, 1, \ldots, T-1$ **do**
  for $i = 1$ **à** N **do**
    Calculer $s_i = y_i \langle \tilde{\boldsymbol{w}}^{(t)}, \tilde{\boldsymbol{x}}_i \rangle$
    if $s_i \leq 0$ **then** ▷ erreur ou marge nulle
      $\tilde{\boldsymbol{w}}^{(t)} \leftarrow \tilde{\boldsymbol{w}}^{(t)} + \eta\, y_i\, \tilde{\boldsymbol{x}}_i$
    end if
  end for
end for
return $\tilde{\boldsymbol{w}}^{(T)}$ (donc $\boldsymbol{w}$ et b)

Définition 3.4 (Données linéairement séparables)**.** Un ensemble étiqueté $\mathcal{T} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ avec $\boldsymbol{x}_i \in \mathbb{R}^d$ et $y_i \in \{-1, 1\}$ est dit *linéairement séparable* s'il existe $\boldsymbol{w} \in \mathbb{R}^d$ et $b \in \mathbb{R}$ (et, de manière équivalente, une marge $M > 0$ après mise à l'échelle de $(\boldsymbol{w}, b)$) tels que

$$y_i \left(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b \right) \;\geq\; M \;>\; 0, \qquad \forall i = 1, \ldots, N. \tag{3.5}$$

Autrement dit, les deux classes sont strictement séparées par l'hyperplan $\{\boldsymbol{x} : \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = 0\}$.



(a) Données linéairement séparables (b) Données non linéairement séparables

FIGURE 2 – Comparaison visuelle entre cas séparable et non séparable.

Théorème 3.1 (Novikoff (1962))**,** borne sur le nombre d'erreurs)**.** *Supposons que* $\|\boldsymbol{x}_i\| \leq R$ *pour tout* i *et que les données sont linéairement séparables avec marge géométrique* $M > 0$*. Alors l'algorithme du perceptron effectue au plus* $(R/M)^2$ *mises à jour (erreurs), et s'arrête donc en un nombre fini d'étapes.*

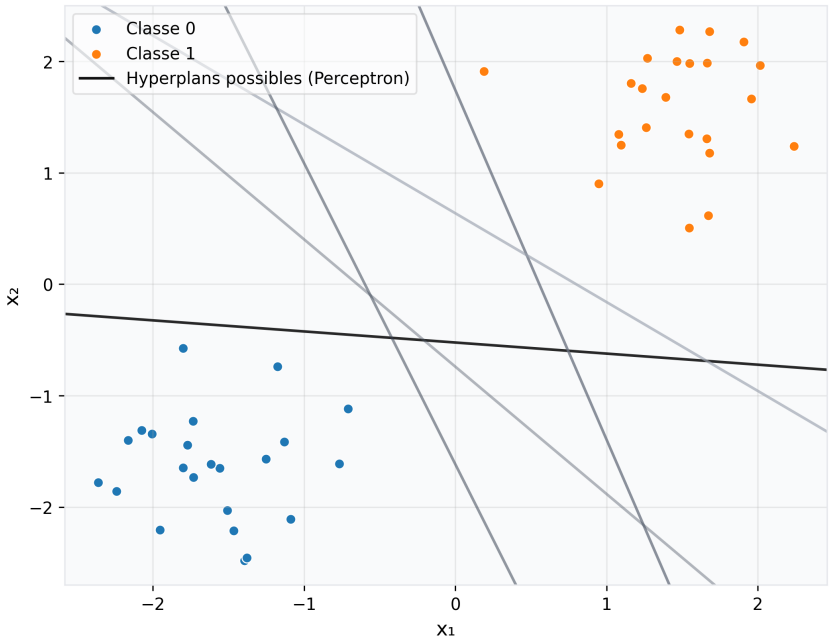


FIGURE 3 – Exemples d'hyperplans séparateurs obtenus par le perceptron selon l'initialisation et l'ordre de présentation : la solution n'est pas unique en cas de séparabilité.

3.3 Problèmes

L'algorithme du perceptron présente cependant plusieurs **limitations** (voir, par ex., Ripley (1996)) :

— **Non-unicité en cas de séparabilité.** Lorsque les données sont séparables, il existe en général *une infinité* d'hyperplans séparateurs (voir Figure 3). Le perceptron peut converger vers des solutions différentes selon l'*initialisation* et l'*ordre de présentation* des exemples.

— **Absence de convergence sur données non séparables.** Quand les données ne sont pas séparables linéairement, l'algorithme *ne converge pas* et peut entrer dans des *cycles* (oscillations d'updates), notamment en présence de bruit d'étiquetage ou d'outliers.

— **Temps de convergence potentiellement long.**Le Théorème 3.1 montre que le nombre de mises à jour du perceptron est majoré par $(R/M)^2$. Intuitivement, plus la marge M est petite (des points « frôlent » l'hyperplan), plus l'algorithme peut nécessiter de corrections avant de se stabiliser. Ainsi, *plus la marge est petite* (données « à peine » séparables), *plus la convergence peut être lente.*

Nous verrons par la suite comment le problème initial de l'équation 3.3 peut être modifié pour répondre à ces limitations.

4 Support Vector Classifier (Hard-Margin)

Dans la littérature plusieurs noms sont donnés à ce modèle, Support Vector Machines avec Hard-Margin, Optimal Separating Hyperplanes (Hastie et al. (2009)), Support Vector Classifier (SVC) avec perte Hard-Margin. Par la suite nous ferons référence à ce modèle par SVC Hard-Margin. Nous présentons ici l'intuition de la *grande marge* et la logique qui mène naturellement du problème « maximiser la marge » à la formulation convexe « minimiser $\frac{1}{2}\|\boldsymbol{w}\|^2$ » sous contraintes d'*apprentissage séparant*. Nous donnons ensuite le problème primal, son dual de Lagrange, les conditions KKT et l'équivalence primal–dual.

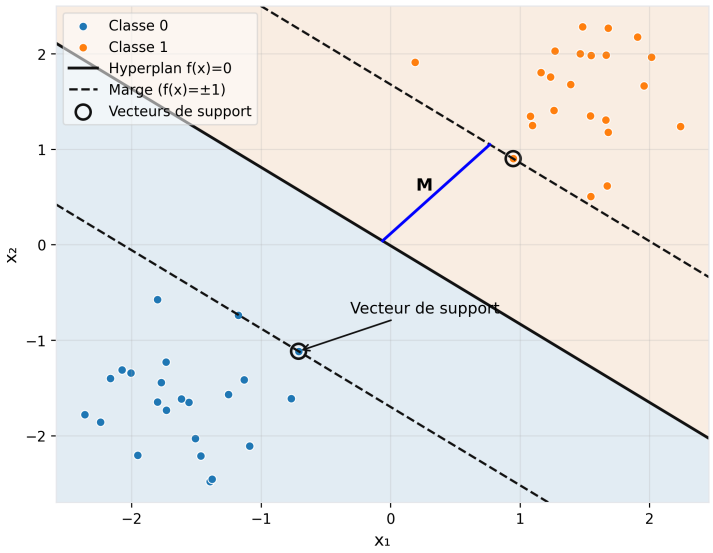


FIGURE 4 – Marge géométrique en SVC hard-margin : la bande délimitée par $\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = \pm 1$ a une largeur $\mathrm{M} = 1/\|\boldsymbol{w}\|$ et les points au contact sont les vecteurs de support.

4.1 Principe de la grande marge et formulation du SVC marge dure

Le perceptron veille à mettre chaque point du « bon côté » d'un hyperplan, sans se soucier de la distance à la frontière. Or, dès que les données sont séparables, il existe une infinité d'hyperplans corrects ; la solution dépend alors de l'initialisation et de l'ordre de présentation et peut être instable face à de petites perturbations. L'idée de la grande marge consiste à préférer, parmi tous les séparateurs corrects, celui qui laisse le plus grand « coussin de sécurité » autour de la frontière.

Pour un couple $(\boldsymbol{w}, b)$ et un exemple $(\boldsymbol{x}_i, y_i)$, on appelle

$$\widehat{M}_i(\boldsymbol{w}, b) := y_i \left(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b \right)$$

la *marge fonctionnelle* de cet exemple : elle est positive si l'exemple est bien classé, et d'autant plus grande que l'exemple est confortablement du bon côté. On agrège par le goulot d'étranglement,

$$\widehat{M}(\boldsymbol{w}, b) := \min_i \widehat{M}_i(\boldsymbol{w}, b),$$

mais cette quantité dépend de l'échelle, car $(\lambda \boldsymbol{w}, \lambda b)$ décrit le même hyperplan tout en multipliant $\widehat{M}$ par $\lambda > 0$.

On neutralise l'échelle en rapportant à la norme de la normale : la *marge géométrique*

$$M(\boldsymbol{w}, b) := \min_i \frac{\widehat{M}_i(\boldsymbol{w}, b)}{\|\boldsymbol{w}\|} = \min_i \frac{y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b)}{\|\boldsymbol{w}\|}$$

coïncide avec la distance signée minimale des points à l'hyperplan et ne dépend que de la position géométrique de la frontière. C'est cette quantité que l'on souhaite maximiser pour obtenir un séparateur robuste, la Figure 4 montre une illustration.

Comme $(\boldsymbol{w}, b)$ et $(\lambda \boldsymbol{w}, \lambda b)$ représentent le même hyperplan, on fixe l'échelle par la *normalisation canonique*

$$\min_i y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) = 1.$$

Les points au contact (vecteurs de support) satisfont alors $y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) = 1$, la marge vaut $M(\boldsymbol{w}, b) = 1/\|\boldsymbol{w}\|$ et la largeur de la bande de marge (entre $\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = \pm 1$) est $2/\|\boldsymbol{w}\|$.

Avec l'écriture canonique ci-dessus,

$$\max_{\boldsymbol{w},b} \; M(\boldsymbol{w},b) \quad \Longleftrightarrow \quad \max_{\boldsymbol{w},b} \; \frac{1}{\|\boldsymbol{w}\|} \quad \text{s.c.} \quad y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b) \geq 1 \;\; \forall i.$$

Maximiser $1/\|\boldsymbol{w}\|$ équivaut à *minimiser* $\|\boldsymbol{w}\|$, et par convenance analytique on prend l'objectif quadratique convexe $\frac{1}{2}\|\boldsymbol{w}\|^2$.

4.2 Formulation primale, dérivation du dual (hard-margin)

Sous la normalisation canonique $\min_i y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b) = 1$, maximiser la marge géométrique revient à minimiser $\|\boldsymbol{w}\|$. On obtient un programme quadratique convexe à contraintes affines.

Définition 4.1 (SVM marge dure — formulation primale)**.**

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^d,\; b\in\mathbb{R}} \quad & \frac{1}{2}\,\|\boldsymbol{w}\|^2 \\ \text{sous contraintes} \quad & y_i\big(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b\big) \;\geq\; 1, \qquad i = 1,\dots,N. \end{aligned} \tag{4.1}$$

Commentaire sur le primal. L'objectif est strictement convexe et les contraintes sont linéaires. En séparabilité stricte, il existe au moins une solution primalement faisable. La direction de $\boldsymbol{w}^\star$ est alors unique en position générale, le biais $b^\star$ se déduit des points au contact.

Passage au dual : construction et calcul de la fonction duale. Pour chaque contrainte $y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b) \geq 1$, on introduit un multiplicateur de Lagrange $\lambda_i \geq 0$. Le Lagrangien associé à (4.1) est

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\lambda}) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{N} \lambda_i \Big(y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b) - 1 \Big), \quad \text{avec } \boldsymbol{\lambda} = (\lambda_1,\dots,\lambda_N)^\top,\; \lambda_i \geq 0.$$

On définit la fonction duale $g(\boldsymbol{\lambda})$ comme l'infimum de $\mathcal{L}$ par rapport aux variables primales :

$$g(\boldsymbol{\lambda}) = \inf_{\boldsymbol{w}\in\mathbb{R}^d,\; b\in\mathbb{R}} \mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\lambda}).$$

On minimise d'abord en $\boldsymbol{w}$. Le terme en $\boldsymbol{w}$ s'écrit

$$\frac{1}{2}\|\boldsymbol{w}\|^2 \;-\; \Big\langle \boldsymbol{w},\; \sum_{i=1}^{N} \lambda_i y_i \boldsymbol{x}_i \Big\rangle,$$

qui est une forme quadratique strictement convexe en $\boldsymbol{w}$. Son unique minimiseur est donné par la condition de premier ordre

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = 0 \quad \Rightarrow \quad \boldsymbol{w}^\star = \sum_{i=1}^{N} \lambda_i y_i \boldsymbol{x}_i.$$

En remplaçant $\boldsymbol{w}$ par $\boldsymbol{w}^\star$, la contribution en $\boldsymbol{w}$ devient

$$\frac{1}{2}\|\boldsymbol{w}^\star\|^2 - \Big\langle \boldsymbol{w}^\star,\; \sum_{i=1}^{N} \lambda_i y_i \boldsymbol{x}_i \Big\rangle = \frac{1}{2}\|\boldsymbol{w}^\star\|^2 - \|\boldsymbol{w}^\star\|^2 = -\frac{1}{2}\|\boldsymbol{w}^\star\|^2 = -\frac{1}{2}\Big\| \sum_{i=1}^{N} \lambda_i y_i \boldsymbol{x}_i \Big\|^2.$$

On minimise ensuite en b. Le Lagrangien est affine en b via le terme $-b\sum_{i=1}^{N} \lambda_i y_i$. Si $\sum_i \lambda_i y_i \neq 0$, l'infimum par rapport à b est $-\infty$. Pour que $g(\boldsymbol{\lambda})$ soit fini, il faut donc imposer la contrainte d'égalité duale

$$\sum_{i=1}^{N} \lambda_i y_i = 0.$$

Dans ce cas, la dépendance en b disparaît, et l'on obtient

$$g(\boldsymbol{\lambda}) = -\frac{1}{2}\Big\| \sum_{i=1}^{N} \lambda_i y_i \boldsymbol{x}_i \Big\|^2 + \sum_{i=1}^{N} \lambda_i.$$

En développant la norme au carré, on arrive à la forme bilinéaire classique

$$g(\boldsymbol{\lambda}) = \sum_{i=1}^{N} \lambda_i \;-\; \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_i \lambda_j \, y_i y_j \, \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle.$$

Proposition 4.1 (SVM marge dure — formulation duale)**.**

$$\begin{aligned} \max_{\boldsymbol{\lambda} \in \mathbb{R}^N} \quad & \sum_{i=1}^{N} \lambda_i \;-\; \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_i \lambda_j \, y_i y_j \, \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle \\ \textit{sous contraintes} \quad & \lambda_i \geq 0, \quad i = 1, \dots, N, \\ & \sum_{i=1}^{N} \lambda_i y_i = 0. \end{aligned} \tag{4.2}$$

Commentaire sur le dual. Le dual est un programme quadratique concave posé sur un polyèdre (cône $\lambda_i \geq 0$ et hyperplan $\sum_i \lambda_i y_i = 0$). À l'optimum, les coefficients $\lambda_i^\star$ sont nuls pour les points qui n'influencent pas la frontière et strictement positifs pour les points qui « soutiennent » l'hyperplan. D'où l'appellation de vecteurs de support.

Proposition 4.2 (Condition de Slater pour le SVC marge dure)**.** *Supposons que les données soient linéairement séparables, c'est-à-dire qu'il existe* $(\boldsymbol{w}_0, b_0)$ *tel que*

$$y_i(\langle \boldsymbol{w}_0, \boldsymbol{x}_i \rangle + b_0) > 0, \qquad i = 1, \dots, N.$$

Alors il existe $(\boldsymbol{w}, b)$ *tel que*

$$y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) > 1, \qquad i = 1, \dots, N.$$

En particulier, pour le problème primal (4.1), la condition de Slater est satisfaite. Le problème étant convexe avec contraintes affines, la dualité est forte : la valeur optimale du primal coïncide avec celle du dual, et les conditions KKT caractérisent l'optimalité.

Esquisse de preuve. Par séparabilité, définissons $m := \min_i y_i(\langle \boldsymbol{w}_0, \boldsymbol{x}_i \rangle + b_0) > 0$. Pour tout $c > 1/m$, posons $\boldsymbol{w} = c\,\boldsymbol{w}_0$ et $b = c\,b_0$. Alors, pour tout i,

$$y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) = c\, y_i(\langle \boldsymbol{w}_0, \boldsymbol{x}_i \rangle + b_0) \geq c\,m \;>\; 1.$$

On a donc exhibé un point primal strictement faisable. Par la condition de Slater (pour des inégalités affines dans un problème convexe), la dualité forte s'ensuit et les KKT sont nécessaires et suffisantes. □

Définition 4.2 (Conditions de Karush–Kuhn–Tucker (KKT))**.** Dans un problème convexe satisfaisant Slater, un triplet $(\boldsymbol{w}^\star, b^\star, \boldsymbol{\lambda}^\star)$ est optimal si et seulement si les quatre familles de conditions suivantes sont vérifiées :

1. *Faisabilité primale* : pour tout i, $y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) \geq 1$.
2. *Faisabilité duale* : pour tout i, $\lambda_i^\star \geq 0$, et $\sum_{i=1}^N \lambda_i^\star y_i = 0$.
3. *Stationnarité* : $\boldsymbol{w}^\star = \sum_{i=1}^N \lambda_i^\star y_i \boldsymbol{x}_i$. La condition $\sum_i \lambda_i^\star y_i = 0$ provient de la minimisation par rapport à b.

4. *Complémentarité* : pour tout i,

$$\lambda_i^\star \Big(y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) - 1 \Big) = 0.$$

La complémentarité se lit contrainte par contrainte. Pour un indice i donné, le produit entre le multiplicateur $\lambda_i^\star$ et la marge fonctionnelle « au seuil » $y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) - 1$ est nul. Deux cas exclusifs se présentent.

— Si $y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) > 1$, alors $\lambda_i^\star = 0$. Le point i est strictement en dehors de la bande de marge ; il ne contribue pas à $\boldsymbol{w}^\star$ dans la décomposition $\boldsymbol{w}^\star = \sum_j \lambda_j^\star y_j \boldsymbol{x}_j$.

— Si $y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) = 1$, la contrainte est active ; on observe alors typiquement $\lambda_i^\star > 0$. Le point i est au contact de la marge : c'est un *vecteur de support*. Ce sont uniquement ces points actifs qui déterminent $\boldsymbol{w}^\star$.

Reconstruction du classifieur à partir du dual. Une fois $\boldsymbol{\lambda}^\star$ optimaux calculés, on reconstitue la normale $\boldsymbol{w}^\star = \sum_i \lambda_i^\star y_i \boldsymbol{x}_i$. Pour déterminer le biais, on peut utiliser n'importe quel point k tel que $\lambda_k^\star > 0$ [1] (donc $y_k(\langle \boldsymbol{w}^\star, \boldsymbol{x}_k \rangle + b^\star) = 1$) et poser

$$b^\star = y_k - \langle \boldsymbol{w}^\star, \boldsymbol{x}_k \rangle.$$

En pratique, on moyenne cette valeur sur plusieurs vecteurs de support afin de limiter l'impact du bruit numérique.

Égalité primal–dual et valeur optimale. Sous Slater, l'écart de dualité est nul. La valeur optimale du primal est égale à la valeur optimale du dual. On a en particulier

$$\frac{1}{2}\|\boldsymbol{w}^\star\|^2 = \sum_{i=1}^N \lambda_i^\star \; - \; \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \lambda_i^\star \lambda_j^\star \, y_i y_j \, \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle.$$

Approfondir et mieux comprendre. Pour les bases de la dualité convexe, de la condition de Slater et des KKT, voir par exemple Boyd and Vandenberghe (2004).

5 Support Vector Classifier (Soft-Margin)

Le modèle à marge dure suppose une séparabilité parfaite : toutes les données peuvent être tenues à distance au moins $1/\|\boldsymbol{w}\|$ de l'hyperplan. Dans de nombreuses situations réelles, les données sont seulement *quasi-séparables* : on observe du bruit, quelques outliers, et parfois des classes qui se chevauchent légèrement. Il est alors souhaitable d'autoriser des *violations contrôlées* de la marge plutôt que de forcer une séparation irréaliste.

5.1 Du $\max M$ au soft-margin : intuition et convexité

Point de départ. En marge dure, on pose le problème directement comme une maximisation de la marge géométrique M, en figeant l'échelle par $\|\boldsymbol{w}\| = 1$ (normale unitaire) :

$$\max_{\boldsymbol{w}, b, M} \; M \qquad \text{s.c.} \qquad y_i\big(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b\big) \;\geq\; M, \quad i = 1, \ldots, N, \quad \|\boldsymbol{w}\| = 1.$$

Cette écriture signifie : avec une normale unitaire, la quantité $y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b)$ est exactement la distance signée de $\boldsymbol{x}_i$ à l'hyperplan, et M est la plus petite de ces distances.

Cas quasi-séparable. Lorsque les classes se chevauchent ou que du bruit est présent, on conserve l'idée « maximiser M » mais on autorise des violations contrôlées via des variables $\xi_i \geq 0$, Figure 5,t (une par point), tout en pénalisant leur somme. Deux façons naturelles d'assouplir la contrainte de marge existent. *Souplesse additive.* On autorise un *déficit absolu* ξ_i

1. Le lecteur attentif notera que k est donc un vecteur de support

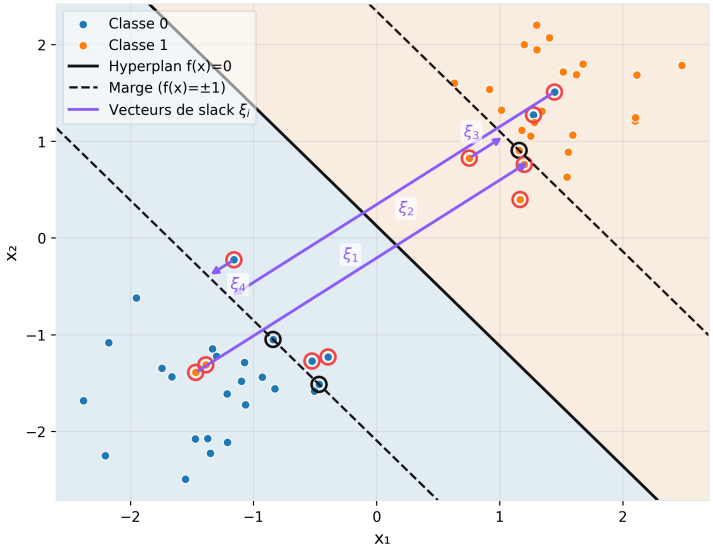


FIGURE 5 – Soft-margin avec variables d'écart ξ_i : $0 < \xi_i < 1$ pour un point dans la bande mais bien classé, $\xi_i \geq 1$ pour un point mal classé

sur la cible M :

$$y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) \;\geq\; M - \xi_i, \qquad \xi_i \geq 0, \qquad \max\; M - \alpha \sum_i \xi_i, \qquad \|\boldsymbol{w}\| = 1.$$

Intuition : ξ_i s'exprime « en mètres » (même unité que M), donc chaque point peut empiéter d'une certaine *distance* sur la marge globale. Caractère non convexe. Pour comprendre le problème, éliminons M en travaillant à échelle libre. Posons

$$\tilde{\boldsymbol{w}} = \frac{\boldsymbol{w}}{M}, \qquad \tilde{b} = \frac{b}{M}, \qquad \tilde{\xi}_i = \frac{\xi_i}{M} \quad (M > 0).$$

La contrainte devient alors

$$y_i(\langle \tilde{\boldsymbol{w}}, \boldsymbol{x}_i \rangle + \tilde{b}) \;\geq\; 1 - \tilde{\xi}_i,$$

c'est-à-dire *affine* en $(\tilde{\boldsymbol{w}}, \tilde{b}, \tilde{\xi})$. En revanche, l'objectif se réécrit

$$M - \alpha \sum_i \xi_i \;=\; \frac{1}{\|\tilde{\boldsymbol{w}}\|} \;-\; \alpha \sum_i \frac{\tilde{\xi}_i}{\|\tilde{\boldsymbol{w}}\|} \;=\; \frac{1 - \alpha \sum_i \tilde{\xi}_i}{\|\tilde{\boldsymbol{w}}\|},$$

qui couple $\tilde{\boldsymbol{w}}$ et $\tilde{\xi}$ de manière non linéaire (rapport linéaire/$\|\tilde{\boldsymbol{w}}\|$). Cette forme n'est ni concave (pour une maximisation) ni facilement convexifiable : le problème est *non convexe*. C'est la difficulté relevée dans ESL pour le choix « additif ».

Souplesse multiplicative. On demande à chaque point de respecter une *fraction* $(1 - \xi_i)$ de la marge globale :

$$y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) \;\geq\; M\,(1 - \xi_i), \qquad \xi_i \geq 0, \qquad \max\; M - \alpha \sum_i \xi_i, \qquad \|\boldsymbol{w}\| = 1.$$

Intuition : ξ_i est un *pourcentage* de « tolérance » ; par exemple $\xi_i = 0.1$ autorise le point i à se trouver à 90% de la marge visée. Passage à une forme convexe. Divisons de nouveau par M et posons

$$\tilde{\boldsymbol{w}} = \frac{\boldsymbol{w}}{M}, \qquad \tilde{b} = \frac{b}{M}.$$

On obtient la même contrainte affine que ci-dessus :

$$y_i(\langle \tilde{\boldsymbol{w}}, \boldsymbol{x}_i \rangle + \tilde{b}) \;\geq\; 1 - \xi_i,$$

mais cette fois ξ_i *n'est pas* rescalée. De plus, $\|\boldsymbol{w}\| = 1$ implique $M = 1/\|\tilde{\boldsymbol{w}}\|$, de sorte que maximiser $M - \alpha \sum_i \xi_i$ revient à maximiser

$$\frac{1}{\|\tilde{\boldsymbol{w}}\|} \;-\; \alpha \sum_i \xi_i.$$

Plutôt que de maximiser $1/\|\tilde{\boldsymbol{w}}\|$ (non convexe), on adopte l'objectif quadratique convexe $\frac{1}{2}\|\tilde{\boldsymbol{w}}\|^2$ qui est *monotone* par rapport à $1/\|\tilde{\boldsymbol{w}}\|$ (ces deux critères ordonnent les solutions de la même façon après choix d'un paramètre). On arrive ainsi au SVC standard :

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}\geq 0} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i \qquad \text{s.c.} \qquad y_i\big(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b\big) \;\geq\; 1-\xi_i, \quad i = 1,\dots,N. \tag{5.1}$$

Ici, tout est convexe : objectif quadratique strictement convexe en $\boldsymbol{w}$, contraintes affines, domaine $\xi_i \geq 0$. Le paramètre $C > 0$ joue le même rôle que α après mise à l'échelle et règle le compromis entre largeur de marge et quantité d'écarts.

Lecture géométrique. Avec la contrainte $y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b) \geq 1 - \xi_i$, on retrouve la même bande de marge que pour la marge dure : $H_\pm : \ \langle \boldsymbol{w}, \boldsymbol{x}\rangle + b = \pm 1$. Un point a $\xi_i = 0$ s'il est hors de la bande (ou sur ± 1) ; il a $0 < \xi_i < 1$ s'il est à l'intérieur de la bande mais du bon côté ; il a $\xi_i \geq 1$ s'il est mal classé. Augmenter C réduit les violations mais contracte la marge ; diminuer C élargit la marge mais accepte davantage d'erreurs.

5.2 Formulation primale, dérivation du dual (soft-margin)

À l'échelle canonique (marge-cible fixée à 1), autoriser des violations contrôlées de la marge conduit à introduire des variables d'écart $\xi_i \geq 0$ et à pénaliser leur somme. On obtient un programme quadratique convexe à contraintes affines.

Définition 5.1 (SVM marge souple — formulation primale)**.**

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^d,\ b\in\mathbb{R},\ \boldsymbol{\xi}\in\mathbb{R}^N_+} \quad & \frac{1}{2}\,\|\boldsymbol{w}\|^2 \;+\; C\sum_{i=1}^{N}\xi_i \\ \text{sous contraintes} \quad & y_i\big(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b\big) \;\geq\; 1-\xi_i, \qquad i = 1,\dots,N, \\ & \xi_i \;\geq\; 0, \qquad i = 1,\dots,N. \end{aligned} \tag{5.2}$$

Remarque 5.1 (Commentaire sur le primal). L'objectif est strictement convexe en $\boldsymbol{w}$; les contraintes sont linéaires. Le paramètre $C > 0$ règle le compromis entre largeur de marge (via $\|\boldsymbol{w}\|$) et quantité de violations (via $\sum_i \xi_i$). Géométriquement, $\xi_i = 0$ signifie point hors de la bande (ou sur la marge), $0 < \xi_i < 1$ point dans la bande mais du bon côté, $\xi_i \geq 1$ point mal classé.

Passage au dual : construction et calcul de la fonction duale. Pour les contraintes $y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b) - 1 + \xi_i \geq 0$, on introduit des multiplicateurs $\lambda_i \geq 0$; pour $\xi_i \geq 0$, des multiplicateurs $\nu_i \geq 0$. Le Lagrangien est

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\lambda_i\Big(y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b) - 1 + \xi_i\Big) - \sum_{i=1}^{N}\nu_i\,\xi_i.$$

La fonction duale $g(\boldsymbol{\lambda}) = \inf_{\boldsymbol{w},b,\boldsymbol{\xi}} \mathcal{L}$ s'obtient en minimisant successivement :
• En $\boldsymbol{w}$:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = 0 \implies \boldsymbol{w}^\star = \sum_{i=1}^{N}\lambda_i y_i \boldsymbol{x}_i, \qquad \inf_{\boldsymbol{w}}\Big(\tfrac{1}{2}\|\boldsymbol{w}\|^2 - \langle \boldsymbol{w}, \sum_i \lambda_i y_i \boldsymbol{x}_i\rangle\Big) = -\tfrac{1}{2}\Big\|\sum_i \lambda_i y_i \boldsymbol{x}_i\Big\|^2.$$

• En b :

$$\frac{\partial \mathcal{L}}{\partial b} = -\sum_{i=1}^{N} \lambda_i y_i = 0 \implies \sum_{i=1}^{N} \lambda_i y_i = 0,$$

sinon l'infimum en b vaut $-\infty$.

• En ξ_i :

$$\frac{\partial \mathcal{L}}{\partial \xi_i} = C - \lambda_i - \nu_i = 0 \implies \nu_i = C - \lambda_i \ (\geq 0) \implies \boxed{0 \leq \lambda_i \leq C}.$$

La dépendance en $\boldsymbol{\xi}$ disparaît alors. On obtient

$$g(\boldsymbol{\lambda}) = \sum_{i=1}^{N} \lambda_i \ - \ \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_i \lambda_j \, y_i y_j \, \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle,$$

valable sous $\sum_i \lambda_i y_i = 0$ et $0 \leq \lambda_i \leq C$.

Proposition 5.1 (SVM marge souple — formulation duale)**.**

$$\begin{aligned} \max_{\boldsymbol{\lambda} \in \mathbb{R}^N} \quad & \sum_{i=1}^{N} \lambda_i \ - \ \frac{1}{2} \boldsymbol{\lambda}^\top (YKY) \boldsymbol{\lambda} \\ \textit{sous contraintes} \quad & 0 \leq \lambda_i \leq C, \quad i = 1, \ldots, N, \\ & \boldsymbol{y}^\top \boldsymbol{\lambda} = 0. \end{aligned} \tag{5.3}$$

Ici $K \in \mathbb{R}^{N \times N}$ est la matrice de Gram, $K_{ij} = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$, et $Y = \mathrm{diag}(y_1, \ldots, y_N)$. Le dual ne dépend donc des données qu'au travers des produits scalaires $\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$ (structure clé pour le kernel trick (astuce du noyau)).

Remarque 5.2 (Lecture du dual). Il s'agit d'un programme quadratique concave sur le polyèdre $\{ \boldsymbol{\lambda} : \ 0 \leq \lambda_i \leq C, \ \boldsymbol{y}^\top \boldsymbol{\lambda} = 0 \}$. À l'optimum, seules quelques composantes $\lambda_i^\star$ sont non nulles : les indices actifs correspondent aux vecteurs de support.

Proposition 5.2 (Condition de Slater pour le SVC marge souple)**.** *Pour tout ensemble de données, il existe un point primal strictement faisable. Par exemple,* $\boldsymbol{w} = \boldsymbol{0}$*,* $b = 0$*,* $\xi_i = 2$ *vérifient* $y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) - 1 + \xi_i = 1 > 0$ *et* $\xi_i > 0$ *pour tout* i*. La condition de Slater est donc satisfaite ; la dualité est forte et les KKT caractérisent l'optimalité.*

Définition 5.2 (Conditions de Karush–Kuhn–Tucker (KKT) — marge souple)**.** Un quintuplet $(\boldsymbol{w}^\star, b^\star, \boldsymbol{\xi}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\nu}^\star)$ est optimal si et seulement si :

1. *Faisabilité primale* : $y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) \geq 1 - \xi_i^\star$ et $\xi_i^\star \geq 0$ pour tout i ;
2. *Faisabilité duale* : $\lambda_i^\star \geq 0$, $\nu_i^\star \geq 0$ pour tout i, et $\sum_i \lambda_i^\star y_i = 0$;
3. *Stationnarité* : $\boldsymbol{w}^\star = \sum_i \lambda_i^\star y_i \boldsymbol{x}_i$ et $C - \lambda_i^\star - \nu_i^\star = 0$ pour tout i ;
4. *Complémentarité* : pour tout i,

$$\lambda_i^\star \Big(y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) - 1 + \xi_i^\star \Big) = 0, \qquad \nu_i^\star \, \xi_i^\star = 0.$$

Interprétation. Si $0 < \lambda_i^\star < C$, alors $\nu_i^\star > 0$ donc $\xi_i^\star = 0$, et la contrainte est active : $y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) = 1$ (vecteur de support « libre », sur la marge). Si $\lambda_i^\star = C$, alors $\nu_i^\star = 0$ et $y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) = 1 - \xi_i^\star \leq 1$ (vecteur de support « à la borne », dans la bande ou mal classé si $\xi_i^\star \geq 1$). Si $\lambda_i^\star = 0$, alors $\nu_i^\star = C$ et $\xi_i^\star = 0$, d'où $y_i(\langle \boldsymbol{w}^\star, \boldsymbol{x}_i \rangle + b^\star) \geq 1$ (point hors bande, non support).

Reconstruction du classifieur à partir du dual. On a $\boldsymbol{w}^\star = \sum_{i=1}^{N} \lambda_i^\star y_i \boldsymbol{x}_i$. Pour le biais, choisir un k tel que $0 < \lambda_k^\star < C$ (support « libre ») et poser

$$b^\star = y_k - \langle \boldsymbol{w}^\star, \boldsymbol{x}_k \rangle,$$

puis moyenner sur plusieurs supports « libres » pour la stabilité numérique. La fonction de décision s'écrit

$$f(\boldsymbol{x}) = \operatorname{sign}\left(\sum_{i=1}^{N} \lambda_i^\star y_i \langle \boldsymbol{x}_i, \boldsymbol{x}\rangle + b^\star\right).$$

Égalité primal–dual et valeur optimale. Sous Slater, l'écart de dualité est nul ; en particulier,

$$\frac{1}{2}\|\boldsymbol{w}^\star\|^2 + C\sum_{i=1}^{N}\xi_i^\star = \sum_{i=1}^{N}\lambda_i^\star \;-\; \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i^\star\lambda_j^\star\, y_i y_j\, \langle \boldsymbol{x}_i, \boldsymbol{x}_j\rangle.$$

Approfondir et mieux comprendre. Pour la dualité convexe (condition de Slater, conditions KKT), voir par exemple Boyd and Vandenberghe (2004). Une présentation didactique du SVM à marge souple (problème primal et dual, perte *hinge*) est proposée par Hastie et al. (2009). Pour l'origine et un traitement plus théorique, consulter Cortes and Vapnik (1995) et Vapnik (1998).

6 Support Vector Machine (SVM) : Kernel Trick

L'extension du SVC à marge souple aux frontières non linéaires consiste à remplacer le produit scalaire euclidien $\langle x, x'\rangle$ par un *noyau* $K(x, x')$ induisant implicitement un plongement ϕ dans un espace (souvent de grande, voire infinie, dimension) où la séparation redevient linéaire. Cette idée est appuyée par le théorème de Cover (1965) : *« Projeter non linéairement des données dans un espace de dimension plus élevée augmente, en général, la probabilité de séparabilité linéaire. »* Dans l'espace de caractéristiques, la complexité du classifieur est contrôlée par la norme de la normale (ou, en RKHS, $\|f\|_{\mathcal{H}_K}$), tandis que le paramètre C maintient la régularisation du compromis marge/violations. Les travaux fondateurs de Boser et al. (1992); Cortes and Vapnik (1995) ont formalisé ce cadre que nous détaillons ci-après.

6.1 Intuition et exemples de noyaux

Point de départ : pour rendre une séparation linéaire possible lorsque les frontières sont non linéaires dans $\mathbb{R}^d$, on imagine un *plongement* $\phi : \mathbb{R}^d \to \mathcal{H}$ vers un espace de Hilbert $(\mathcal{H}, \langle\cdot,\cdot\rangle_{\mathcal{H}})$ où les données deviennent (presque) séparables. Le SVC marge souple « dans $\mathcal{H}$ » s'écrit

$$\min_{\boldsymbol{w}\in\mathcal{H},\; b\in\mathbb{R},\; \boldsymbol{\xi}\geq 0} \; \frac{1}{2}\|\boldsymbol{w}\|_{\mathcal{H}}^2 + C\sum_{i=1}^{N}\xi_i \quad \text{s.c.} \quad y_i(\langle \boldsymbol{w}, \phi(\boldsymbol{x}_i)\rangle_{\mathcal{H}} + b) \geq 1 - \xi_i.$$

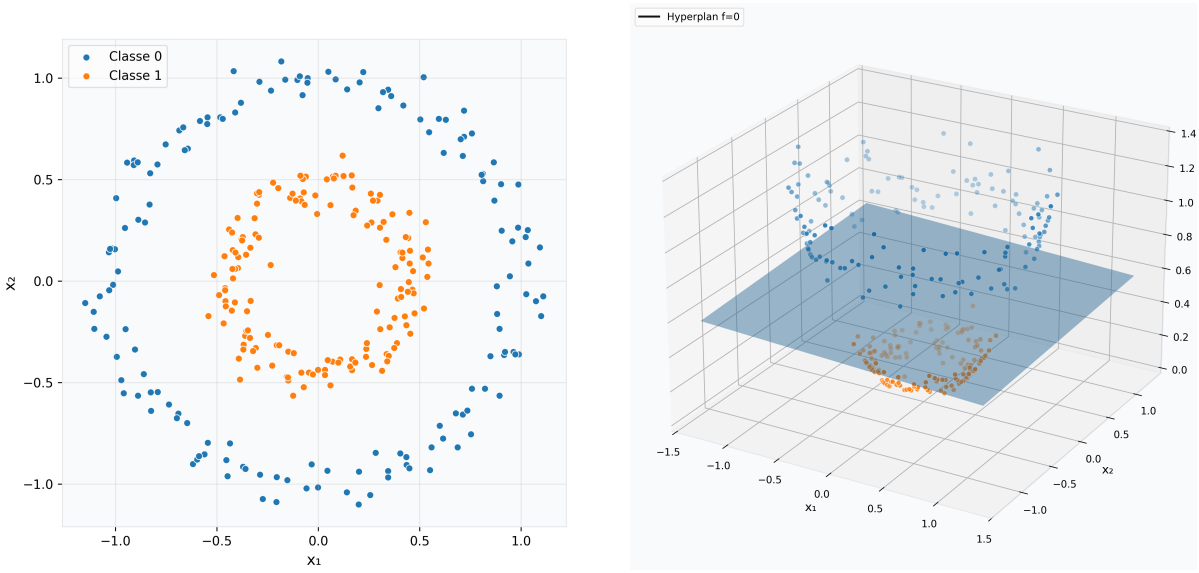
Le théorème de Cover (1965), formalise l'intuition : un plongement non linéaire en dimension plus élevée augmente, en général, la probabilité de séparabilité linéaire (Figure 6). *Mais* estimer directement $\boldsymbol{w} \in \mathcal{H}$ devient vite impraticable : pour un noyau gaussien, $\mathcal{H}$ est de dimension *infinie* ; même pour un noyau *polynomial* de degré p, la dimension du plongement (tous les monômes jusqu'au degré p) explose combinatoirement avec d. Le kernel trick contourne cette difficulté : on n'a jamais besoin de construire ϕ ni de stocker $\boldsymbol{w}$ dans $\mathcal{H}$, car le *dual* du SVC ne dépend des données qu'au travers de produits scalaires de la forme

$$\langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j)\rangle_{\mathcal{H}} \;=:\; K(\boldsymbol{x}_i, \boldsymbol{x}_j),$$

où K est une fonction dite *noyau*. En remplaçant partout $\langle \boldsymbol{x}_i, \boldsymbol{x}_j\rangle$ par $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, on résout l'optimisation comme si l'on travaillait linéairement dans $\mathcal{H}$, *sans* calculer ϕ.

Définition informelle : un *noyau* $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ est une fonction telle qu'il existe un espace de Hilbert $\mathcal{H}$ et un plongement ϕ vérifiant

$$K(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}')\rangle_{\mathcal{H}}.$$



(a) Données dans $\mathbb{R}^2$ non linéairement séparable (b) Données dans $\mathbb{R}^3$ linéairement séparable

FIGURE 6 – Plongement de $\mathbb{R}^2$ dans $\mathbb{R}^3$

La décision apprise prend alors la forme

$$f(\boldsymbol{x}) = \operatorname{sign}\left(\sum_{i=1}^{N} \lambda_i^\star y_i \, K(\boldsymbol{x}_i, \boldsymbol{x}) \; + \; b^\star\right),$$

où seuls les *vecteurs de support* (indices i tels que $\lambda_i^\star > 0$) interviennent : on n'a jamais manipulé $\boldsymbol{w}$ ni $\phi(\boldsymbol{x})$.

Quelques noyaux usuels :

— **Linéaire** : $K(\boldsymbol{x}, \boldsymbol{x}') = \langle \boldsymbol{x}, \boldsymbol{x}' \rangle$ (cas de base, ϕ identité).

— **Polynomial (degré p)** : $K(\boldsymbol{x}, \boldsymbol{x}') = (\langle \boldsymbol{x}, \boldsymbol{x}' \rangle + c)^p$, $p \in \mathbb{N}$, $c \geq 0$ (plongement fini : monômes jusqu'au degré p).

— **Gaussien / RBF** : $K(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\gamma \|\boldsymbol{x} - \boldsymbol{x}'\|^2)$, $\gamma > 0$ (plongement *infini*).

— **Sigmoïde (type réseau)** : $K(\boldsymbol{x}, \boldsymbol{x}') = \tanh(\kappa \, \langle \boldsymbol{x}, \boldsymbol{x}' \rangle + \theta)$ (p.s.d. seulement pour certaines plages de paramètres).

Exemple explicite — noyau polynomial degré 2 *avec* $c = 1$ *en dimension* 2. Prenons $\boldsymbol{x} = (x_1, x_2)$ et $\boldsymbol{x}' = (x_1', x_2') \in \mathbb{R}^2$, et

$$K(\boldsymbol{x}, \boldsymbol{x}') = (\langle \boldsymbol{x}, \boldsymbol{x}' \rangle + 1)^2 = \left(x_1 x_1' + x_2 x_2' + 1\right)^2.$$

En développant,

$$(\langle \boldsymbol{x}, \boldsymbol{x}' \rangle + 1)^2 = (x_1 x_1' + x_2 x_2')^2 + 2(x_1 x_1' + x_2 x_2') + 1 = x_1^2 x_1'^2 + 2x_1 x_2 x_1' x_2' + x_2^2 x_2'^2 + 2x_1 x_1' + 2x_2 x_2' + 1.$$

On peut écrire cette quantité comme un *produit scalaire euclidien* en dimension 6 :

$$K(\boldsymbol{x}, \boldsymbol{x}') = \big\langle \phi(\boldsymbol{x}), \; \phi(\boldsymbol{x}') \big\rangle_{\mathbb{R}^6},$$

où l'application de caractéristiques (une réalisation de l'espace de caractéristiques associé à K) est

$$\phi : \mathbb{R}^2 \to \mathbb{R}^6, \qquad \phi(\boldsymbol{x}) = \left(x_1^2, \; \sqrt{2}\, x_1 x_2, \; x_2^2, \; \sqrt{2}\, x_1, \; \sqrt{2}\, x_2, \; 1\right)$$

Avec ce choix de normalisation, on vérifie directement que

$$\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle = x_1^2 x_1'^2 + 2x_1 x_2 x_1' x_2' + x_2^2 x_2'^2 + 2x_1 x_1' + 2x_2 x_2' + 1 = \left(\langle \boldsymbol{x}, \boldsymbol{x}' \rangle + 1\right)^2 = K(\boldsymbol{x}, \boldsymbol{x}').$$

Remarque 6.1. La fonction ϕ n'est pas unique (toute transformation orthogonale de ϕ convient) ; ce qui est déterminé par K, c'est la classe d'équivalence de l'espace de caractéristiques (isomorphe au RKHS associé).

Sans kernel trick, « travailler en degré 2 » imposerait donc d'estimer un vecteur normal $\boldsymbol{w}$ dans $\mathbb{R}^6$ (et, pour des degrés plus grands et/ou d plus élevé, dans une dimension combinatoirement explosive) ; avec le noyau, on se contente d'évaluer $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ et d'optimiser le dual en dimension N via la matrice de Gram noyautée $K = [K(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{i,j}$.

Le raisonnement précédent éclaire aussi le choix algorithmique. Dans le primal « noyauté », le vecteur normal $\boldsymbol{w}$ vit dans $\mathcal{H}$, dont la dimension peut être énorme (voire *infinie* pour un RBF) : estimer directement $\boldsymbol{w}$ devient vite impraticable. En passant par le Lagrangien, les conditions KKT donnent au contraire la représentation

$$\boldsymbol{w}^\star \;=\; \sum_{i=1}^{N} \lambda_i^\star \, y_i \, \phi(\boldsymbol{x}_i), \qquad 0 \le \lambda_i^\star \le C, \quad \sum_{i=1}^{N} \lambda_i^\star y_i = 0,$$

et mènent au dual

$$\max_{\boldsymbol{\lambda} \in [0,C]^N} \; \sum_{i=1}^{N} \lambda_i \;-\; \frac{1}{2} \sum_{i,j=1}^{N} \lambda_i \lambda_j y_i y_j \underbrace{\langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}}_{K(\boldsymbol{x}_i, \boldsymbol{x}_j)} \quad \text{s.c.} \; \sum_{i=1}^{N} \lambda_i y_i = 0.$$

Autrement dit, toute la difficulté numérique se concentre dans la *matrice de Gram noyautée* $K = [K(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{i,j}$: on ne manipule que des produits scalaires dans $\mathcal{H}$, sans jamais construire ϕ ni stocker $\boldsymbol{w}$. C'est précisément pour cela qu'on préfère le dual : la complexité dépend essentiellement de N (taille de l'échantillon) et non de $\dim(\mathcal{H})$, qui peut exploser dans le primal.

6.2 Formalisation du kernel trick

Définition 6.1 (Espace de Hilbert)**.** Un espace de Hilbert est un espace vectoriel réel $\mathcal{H}$ muni d'un produit scalaire $\langle \cdot, \cdot \rangle_{\mathcal{H}}$, complet pour la norme induite $\|u\|_{\mathcal{H}} = \sqrt{\langle u, u \rangle_{\mathcal{H}}}$. La complétude signifie que toute suite de Cauchy (pour la norme) converge dans $\mathcal{H}$.

Définition 6.2 (Noyau p.s.d. et RKHS, (Aronszajn, 1950))**.** Soit $\mathcal{X}$ un ensemble non vide. Une fonction $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ (symétrique) est *positive semi-définie* (p.s.d.) si, pour tout $n \in \mathbb{N}$ et tout choix $(x_1, \ldots, x_n) \subset \mathcal{X}$,

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \, K(x_i, x_j) \;\ge\; 0 \quad \text{pour tous} \;\; (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n,$$

i.e. la matrice de Gram $K_n = [K(x_i, x_j)]_{i,j}$ est p.s.d. Par le théorème de Moore–Aronszajn, à tout noyau p.s.d. K est associé un *espace de Hilbert à noyau reproduisant* (RKHS) $(\mathcal{H}_K, \langle \cdot, \cdot \rangle_{\mathcal{H}_K})$ et une application canonique

$$\Phi : \; \mathcal{X} \to \mathcal{H}_K, \qquad \Phi(x) := K_x := K(x, \cdot),$$

telles que, pour tous $x, z \in \mathcal{X}$,

$$K(x, z) \;=\; \langle \Phi(x), \Phi(z) \rangle_{\mathcal{H}_K} \qquad \text{et} \qquad \forall f \in \mathcal{H}_K, \;\; f(x) = \langle f, \, K_x \rangle_{\mathcal{H}_K} \;\; \text{(propriété de reproduction)}.$$

Remarque 6.2 (Deux lectures équivalentes d'un noyau). Dire que K est un noyau p.s.d. revient à dire (i) qu'il existe un *plongement* Φ dans un Hilbert $(\mathcal{H}_K, \langle \cdot, \cdot \rangle)$ tel que $K(x, z) = \langle \Phi(x), \Phi(z) \rangle$, *ou* (ii) que toutes les matrices de Gram formées avec K sont p.s.d. La première lecture est géométrique (*feature map*) ; la seconde est algorithmique (QP bien posé dans le dual).

Remarque 6.3 (Exemples et stabilité des noyaux). Exemples p.s.d. usuels : $K_{\text{lin}}(x,z) = \langle x, z\rangle$, $K_{\text{poly}}(x,z) = (\langle x, z\rangle + c)^p$ ($c \geq 0$, $p \in \mathbb{N}$), $K_{\text{RBF}}(x,z) = \exp(-\gamma\|x - z\|^2)$ ($\gamma > 0$). Stabilité : si K_1, K_2 sont p.s.d., alors $aK_1 + bK_2$ ($a, b \geq 0$), $K_1 \cdot K_2$, $x \mapsto \phi(x)^\top A\, \phi'(x)$ avec $A \succeq 0$ sont encore p.s.d. (utile pour construire des noyaux adaptés au domaine).

Théorème 6.1 (Théorème de représentation, Kimeldorf and Wahba (1971); Schölkopf and Smola (2002))**.** *Soit un RKHS $(\mathcal{H}_K, \|\cdot\|_{\mathcal{H}_K})$, une fonction de régularisation $\Omega : \mathbb{R}_+ \to \mathbb{R}$ strictement croissante et une perte $L : \boldsymbol{Y} \times \mathbb{R} \to \mathbb{R}_+$. Pour le problème*

$$\min_{f \in \mathcal{H}_K} \; \Omega(\|f\|_{\mathcal{H}_K}) \; + \; \sum_{i=1}^{N} L(y_i,\; f(x_i)),$$

tout minimiseur admet une représentation finie

$$f^\star(\cdot) = \sum_{i=1}^{N} \alpha_i\, K(x_i, \cdot).$$

Idée de preuve. Décomposer $f = g + h$ avec $g \in \operatorname{span}\{K_{x_i}\}_{i=1}^N$ et $h \perp \operatorname{span}\{K_{x_i}\}$. Par reproduction, $f(x_i) = g(x_i)$ pour tout i ; la partie « données » ne dépend donc que de g. Comme Ω est croissante et $\|f\|^2 = \|g\|^2 + \|h\|^2$, on ne gagne jamais à garder $h \neq 0$: à l'optimum $h = 0$, d'où la forme finie. □

Remarque 6.4 (Lien direct avec le SVM noyauté). En prenant $\Omega(t) = \frac{1}{2}t^2$ et la perte hinge $L(y,u) = \max(0, 1 - yu)$, on obtient le SVC en RKHS :

$$\min_{f \in \mathcal{H}_K,\; b \in \mathbb{R}} \; \frac{1}{2}\|f\|^2_{\mathcal{H}_K} + C \sum_{i=1}^{N} \max\left(0,\; 1 - y_i(f(x_i) + b)\right).$$

Par le théorème 6.1, $f(\cdot) = \sum_i \alpha_i K(x_i, \cdot)$. En introduisant des ξ_i et en écrivant le dual comme dans le cas linéaire, *tous* les produits scalaires $\langle x_i, x_j\rangle$ s'y remplacent par $K(x_i, x_j)$:

$$\max_{\boldsymbol{\lambda}} \; \sum_{i=1}^{N} \lambda_i \; - \; \frac{1}{2} \sum_{i,j=1}^{N} \lambda_i \lambda_j\, y_i y_j\, K(x_i, x_j) \quad \text{s.c.} \quad 0 \leq \lambda_i \leq C,\; \sum_i \lambda_i y_i = 0,$$

et la décision

$$f(x) = \sum_{i=1}^{N} \lambda_i^\star y_i\, K(x_i, x) + b^\star.$$

Le noyau permet donc de travailler comme si l'on était linéaire dans un (éventuellement infini-)dimensional $\mathcal{H}_K$, sans jamais expliciter Φ ni estimer $\boldsymbol{w}$ dans cet espace.

Remarque 6.5 (Rôle de la norme RKHS et contrôle de la complexité). Dans un SVM noyauté, la grandeur $\|f\|_{\mathcal{H}_K}$ joue le rôle de *norme de la normale* dans l'espace de caractéristiques : minimiser $\frac{1}{2}\|f\|^2_{\mathcal{H}_K}$ revient à *maximiser la marge* dans $\mathcal{H}_K$. Le paramètre C équilibre cette marge et les violations de type hinge. On concrétise ainsi l'intuition du théorème de Cover (meilleure séparabilité après plongement) tout en évitant le sur-apprentissage par une régularisation explicite dans le RKHS.

Remarque 6.6 (Pourquoi kerneliser résout l'infini-dimensionnel). Si l'on tentait d'optimiser directement sur $\boldsymbol{w} \in \mathcal{H}$ (par exemple avec un noyau RBF), on aurait une variable de décision *infinie-dimensionnelle*. Le dual n'expose jamais $\boldsymbol{w}$: il ne manipule que la matrice de Gram noyautée $K = [K(x_i, x_j)]$, de taille $N \times N$, et les coefficients duals λ_i. La faisabilité et la concavité du dual reposent précisément sur la p.s.d.-té de K.

Approfondir et mieux comprendre. Pour une introduction claire au kernel trick et aux cartes de caractéristiques $\varphi(\cdot)$, voir Hastie et al. (2009) et Bishop (2006). Pour une présentation rigoureuse via les espaces de Hilbert à noyau reproduisant (RKHS), l'axiome de définie-positive et le *representer theorem*, voir Schölkopf and Smola (2002).

7 Compléments : sorties probabilistes et variantes SVM

7.1 Sorties probabilistes : de la marge à une probabilité

Le SVM renvoie un *score marginal*

$$h(\boldsymbol{x}) \;=\; \sum_{i=1}^{N} \lambda_i^\star \, y_i \, K(\boldsymbol{x}_i, \boldsymbol{x}) \;+\; b^\star \qquad \text{(linéaire si } K(\boldsymbol{x}_i, \boldsymbol{x}) = \langle \boldsymbol{x}_i, \boldsymbol{x} \rangle),$$

qui quantifie la distance signée à la frontière de décision. Ce score est bien ordonné (plus $h(\boldsymbol{x})$ est grand, plus la confiance en la classe $+1$ augmente), mais ce n'est *pas* une probabilité. Pour obtenir une estimation de $\Pr(Y = 1 \mid \boldsymbol{x})$, on applique une *calibration* au score.

Définition 7.1 (Calibration logistique)**.** On utilise la sigmoïde $\sigma(t) = \dfrac{1}{1 + \mathrm{e}^{-t}}$ pour transformer $h(\boldsymbol{x})$ en probabilité :

$$\Pr(Y = 1 \mid \boldsymbol{x}) \;=\; \sigma\big(\theta_1 + \theta_2 \, h(\boldsymbol{x})\big).$$

Le cas « sigmoïde simple » correspond à des paramètres *fixés* $(\theta_1, \theta_2) = (0, 1)$, soit $\Pr(Y = 1 \mid \boldsymbol{x}) = \sigma(h(\boldsymbol{x}))$. En pratique, on préfère *apprendre* (θ_1, θ_2) par maximum de vraisemblance sur un jeu de validation (ou via validation croisée), à partir des paires $\big(h(\boldsymbol{x}_i), t_i\big)$ avec $t_i = \frac{1+y_i}{2} \in \{0, 1\}$:

$$\min_{\theta_1, \theta_2} \; -\sum_{i=1}^{N} \Big[t_i \log \sigma\big(\theta_1 + \theta_2 h(\boldsymbol{x}_i)\big) + (1 - t_i) \log\big(1 - \sigma(\theta_1 + \theta_2 h(\boldsymbol{x}_i))\big) \Big].$$

Remarque 7.1 (Intuition et mise en pratique). Des marges très positives ($h(\boldsymbol{x}) \gg 0$) donnent des probabilités proches de 1, des marges très négatives ($h(\boldsymbol{x}) \ll 0$) proches de 0, et $h(\boldsymbol{x}) \approx 0$ une probabilité voisine de $1/2$. L'estimation de (θ_1, θ_2) doit se faire *hors-échantillon* pour éviter l'optimisme (hold-out ou k-fold). Lorsque le lien entre h et la probabilité est plus complexe, une alternative non paramétrique est la *calibration isotone* (plus flexible mais plus sensible au sur-ajustement). Ces techniques s'appliquent de la même manière aux SVM noyautés (seule l'expression de h change).

7.2 Least-Squares SVM

Least-Squares SVM (LS-SVM) (Suykens and Vandewalle (1999); Suykens et al. (2002)) est une variante du SVM à marge souple qui remplace la perte *hinge* et les contraintes d'inégalité par une pénalisation quadratique des résidus de marge et des contraintes d'égalité. Le problème reste convexe et l'apprentissage se ramène à la résolution d'un système linéaire, avec un schéma de noyautage identique à celui du SVM.

Définition 7.2 (LS-SVM — formulation primale (classification))**.** Soit un espace de caractéristiques $(\mathcal{H}, \langle \cdot, \cdot \rangle_{\mathcal{H}})$, un plongement $\phi : \mathbb{R}^d \to \mathcal{H}$, des étiquettes $y_i \in \{-1, 1\}$ et un paramètre $\gamma > 0$. Le problème primal du LS-SVM s'écrit

$$\begin{aligned} \min_{\boldsymbol{w} \in \mathcal{H},\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^N} \quad & \frac{1}{2} \|\boldsymbol{w}\|_{\mathcal{H}}^2 \;+\; \frac{\gamma}{2} \sum_{i=1}^{N} \xi_i^2 \\ \text{sous contraintes} \quad & y_i\big(\langle \boldsymbol{w}, \phi(\boldsymbol{x}_i) \rangle_{\mathcal{H}} + b\big) \;=\; 1 - \xi_i, \qquad i = 1, \dots, N. \end{aligned} \tag{7.1}$$

Remarque 7.2 (Commentaires). (i) Les variables d'écart ξ_i ne sont pas contraintes en signe ; la pénalisation est symétrique autour de la cible de marge 1. (ii) L'objectif est strictement convexe en $\boldsymbol{w}$ et les contraintes sont affines ; le problème est convexe bien posé. (iii) Le paramètre γ règle le compromis entre régularisation et ajustement des résidus de marge.

Proposition 7.1 (Conditions stationnaires et système linéaire)**.** *En introduisant des multiplicateurs d'égalité $\lambda_i \in \mathbb{R}$ et la Lagrangienne*

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}) = \frac{1}{2}\|\boldsymbol{w}\|_{\mathcal{H}}^2 + \frac{\gamma}{2}\sum_{i=1}^{N}\xi_i^2 - \sum_{i=1}^{N}\lambda_i\Big(y_i(\langle \boldsymbol{w}, \phi(\boldsymbol{x}_i)\rangle_{\mathcal{H}} + b) - 1 + \xi_i\Big),$$

les conditions de stationnarité donnent

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = 0 \iff \boldsymbol{w} = \sum_{i=1}^{N}\lambda_i y_i\,\phi(\boldsymbol{x}_i), \qquad \frac{\partial \mathcal{L}}{\partial b} = 0 \iff \sum_{i=1}^{N}\lambda_i y_i = 0,$$

$$\frac{\partial \mathcal{L}}{\partial \xi_i} = 0 \iff \gamma\,\xi_i - \lambda_i = 0 \quad \textit{pour tout } i \quad \textit{donc} \quad \xi_i = \lambda_i/\gamma.$$

En substituant dans les contraintes de (7.1) et en notant $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j)\rangle_{\mathcal{H}}$ et $\Omega_{ij} = y_i y_j\, K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, on obtient le système bloc

$$\begin{bmatrix} 0 & \boldsymbol{y}^\top \\ \boldsymbol{y} & \Omega + \gamma^{-1} I_N \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{1}_N \end{bmatrix}, \qquad \boldsymbol{y} = (y_1, \dots, y_N)^\top, \quad \boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_N)^\top. \tag{7.2}$$

Définition 7.3 (Fonction de décision)**.** Soit $(b, \boldsymbol{\lambda})$ solution de (7.2). La fonction décisionnelle s'écrit

$$f(\boldsymbol{x}) = \sum_{i=1}^{N}\lambda_i\, y_i\, K(\boldsymbol{x}_i, \boldsymbol{x}) + b, \qquad \widehat{y} = \operatorname{sign}\big(f(\boldsymbol{x})\big).$$

Proposition 7.2 (Formulation duale)**.** *L'élimination de $(\boldsymbol{w}, b, \boldsymbol{\xi})$ dans la Lagrangienne conduit au problème dual*

$$\max_{\boldsymbol{\lambda} \in \mathbb{R}^N} \quad \mathbf{1}_N^\top \boldsymbol{\lambda} \;-\; \frac{1}{2}\,\boldsymbol{\lambda}^\top(\Omega + \gamma^{-1} I_N)\boldsymbol{\lambda} \quad \textit{sous contrainte} \quad \boldsymbol{y}^\top \boldsymbol{\lambda} = 0.$$

Remarque 7.3 (Lecture géométrique, atouts et limites). En posant $f(\boldsymbol{x}_i) = \sum_{j=1}^{N}\lambda_j y_j K(\boldsymbol{x}_j, \boldsymbol{x}_i) + b$, les contraintes primales donnent $\xi_i = 1 - y_i f(\boldsymbol{x}_i)$. On a $\xi_i \le 0$ pour des points au-delà de la marge du bon côté, $0 < \xi_i < 1$ pour des points à l'intérieur de la bande du bon côté, et $\xi_i \ge 1$ en cas de mauvaise classification. Atouts : apprentissage par un système linéaire $(N{+}1) \times (N{+}1)$; noyautage identique au SVM. Limites : solution généralement *dense* (moins de parcimonie), sensibilité accrue aux *outliers* du fait de la pénalisation quadratique.

7.3 Support Vector Regression :

Le Support Vector Regression (SVR) (Smola and Schölkopf (2004)) vise à estimer une fonction affine dans l'espace de caractéristiques, $f(\boldsymbol{x}) = \langle \boldsymbol{w}, \phi(\boldsymbol{x})\rangle_{\mathcal{H}} + b$, qui tolère des écarts jusqu'à un seuil $\epsilon > 0$ sans pénalisation (*perte ϵ-insensible*). Géométriquement, il s'agit d'ajuster un *tube* de largeur 2ϵ autour du prédicteur (voir Figure 7) : les points à l'intérieur du tube ne sont pas pénalisés, les points situés en dehors engendrent des variables d'écart.

Définition 7.4 (SVR — formulation primale)**.** Soit un espace de Hilbert de caractéristiques $(\mathcal{H}, \langle\cdot,\cdot\rangle_{\mathcal{H}})$, un plongement $\phi : \mathbb{R}^d \to \mathcal{H}$, des observations $(\boldsymbol{x}_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, et un paramètre $C > 0$. Le problème primal de la SVR s'écrit

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathcal{H},\, b\in\mathbb{R},\, \boldsymbol{\xi}\in\mathbb{R}^N_+,\, \boldsymbol{\xi}^\star\in\mathbb{R}^N_+} \quad & \frac{1}{2}\,\|\boldsymbol{w}\|_{\mathcal{H}}^2 \;+\; C\sum_{i=1}^{N}(\xi_i + \xi_i^\star) \\ \text{sous contraintes} \quad & y_i - (\langle \boldsymbol{w}, \phi(\boldsymbol{x}_i)\rangle_{\mathcal{H}} + b) \;\le\; \epsilon + \xi_i, \qquad i = 1, \dots, N, \\ & (\langle \boldsymbol{w}, \phi(\boldsymbol{x}_i)\rangle_{\mathcal{H}} + b) - y_i \;\le\; \epsilon + \xi_i^\star, \qquad i = 1, \dots, N, \\ & \xi_i \ge 0, \quad \xi_i^\star \ge 0, \qquad i = 1, \dots, N. \end{aligned} \tag{7.3}$$

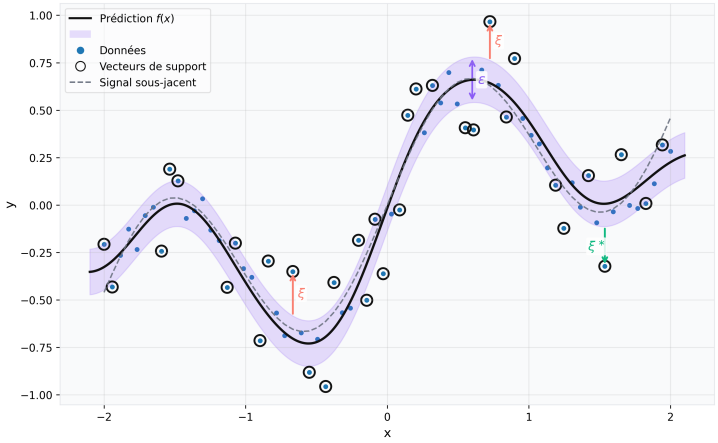


FIGURE 7 – **SVR et tube ϵ-insensible.** Courbe estimée $f(\boldsymbol{x})$ (trait plein), tube ϵ (bandes violettes). Les points situés hors du tube engendrent des variables d'écart $(\xi_i, \xi_i^\star)$ et deviennent des *vecteurs de support* de la régression.

Proposition 7.3 (Conditions stationnaires et boîtes duales)**.** *Soient $\lambda_i, \lambda_i^\star \geq 0$ les multiplicateurs d'inégalités associés respectivement aux deux familles de contraintes dans (7.3), et $\mu_i, \mu_i^\star \geq 0$ ceux associés à $\xi_i \geq 0$, $\xi_i^\star \geq 0$. En posant $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}$, les conditions de stationnarité de la Lagrangienne donnent*

$$\boldsymbol{w} = \sum_{i=1}^{N} (\lambda_i^\star - \lambda_i)\, \phi(\boldsymbol{x}_i), \qquad \sum_{i=1}^{N} (\lambda_i^\star - \lambda_i) = 0.$$

On a également $C - \lambda_i - \mu_i = 0$ et $C - \lambda_i^\star - \mu_i^\star = 0$ pour tout i, avec complémentarité $\mu_i \xi_i = 0$ et $\mu_i^\star \xi_i^\star = 0$. Il en résulte les bornes en boîte

$$0 \;\leq\; \lambda_i \;\leq\; C, \qquad 0 \;\leq\; \lambda_i^\star \;\leq\; C, \qquad i = 1, \ldots, N.$$

Proposition 7.4 (Dual noyauté)**.** *L'élimination de $(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\xi}^\star, \boldsymbol{\mu}, \boldsymbol{\mu}^\star)$ dans la Lagrangienne associée à (7.3) conduit au problème dual*

$$\begin{aligned} \max_{\boldsymbol{\lambda}, \boldsymbol{\lambda}^\star} \quad & -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\lambda_i - \lambda_i^\star)(\lambda_j - \lambda_j^\star)\, K(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ & - \epsilon \sum_{i=1}^{N} (\lambda_i + \lambda_i^\star) \;+\; \sum_{i=1}^{N} y_i (\lambda_i - \lambda_i^\star) \\ \textit{sous contraintes} \quad & \sum_{i=1}^{N} (\lambda_i - \lambda_i^\star) = 0, \qquad 0 \leq \lambda_i, \lambda_i^\star \leq C, \quad i = 1, \ldots, N. \end{aligned} \tag{7.4}$$

Définition 7.5 (Prédicteur et calcul du biais)**.** Pour une solution $(\boldsymbol{\lambda}, \boldsymbol{\lambda}^\star)$ de (7.4), la fonction de prédiction s'écrit

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} (\lambda_i - \lambda_i^\star)\, K(\boldsymbol{x}_i, \boldsymbol{x}) + b.$$

Le terme b se déduit des conditions KKT en sélectionnant un indice i tel que $0 < \lambda_i < C$ ou $0 < \lambda_i^\star < C$. Par exemple,

$$b \;=\; y_i - \epsilon - \sum_{j=1}^{N} (\lambda_j - \lambda_j^\star)\, K(\boldsymbol{x}_j, \boldsymbol{x}_i) \quad \text{si } 0 < \lambda_i < C,$$

et

$$b \;=\; y_i + \epsilon - \sum_{j=1}^{N} (\lambda_j - \lambda_j^\star)\, K(\boldsymbol{x}_j, \boldsymbol{x}_i) \quad \text{si } 0 < \lambda_i^\star < C.$$

En pratique, on moyenne b sur plusieurs indices admissibles pour une meilleure stabilité numérique.

Remarque 7.4 (Lecture géométrique et rôle des hyperparamètres). Les points vérifiant $0 < \lambda_i < C$ ou $0 < \lambda_i^\star < C$ se trouvent *sur* le tube et sont des vecteurs de support ; ceux vérifiant $\lambda_i = C$ ou $\lambda_i^\star = C$ sont *hors* tube. Le paramètre ϵ contrôle l'épaisseur du tube : une valeur plus grande induit un modèle plus lisse et moins de vecteurs de support, mais un biais plus élevé. Le paramètre C règle la pénalisation des dépassements : une valeur grande favorise un ajustement plus serré (variance plus élevée), une valeur petite autorise davantage de tolérance (variance plus faible). Le noyautage intervient uniquement via $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, de la même manière qu'en classification.

Références

Mark A. Aizerman, Eduard M. Braverman, and Lev I. Rozonoer. Theoretical foundations of the potential function method in pattern recognition. In *Automation and Remote Control*, volume 25, pages 821–837, 1964.

Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3) :337–404, 1950.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2006.

Bernhard E. Boser, Isabelle Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory (COLT)*, pages 144–152, 1992.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3) : 273–297, 1995.

Thomas M. Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Transactions on Electronic Computers*, EC-14(3) : 326–334, 1965.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, New York, 2 edition, 2009.

C. Hurlin. Support vector machine, September 2025. URL https://doi.org/10.5281/zenodo.17115854.

George Kimeldorf and Grace Wahba. Some results on tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33(1) :82–95, 1971.

A. B. J. Novikoff. On convergence proofs on perceptrons. *Proceedings of the Symposium on the Mathematical Theory of Automata*, 12 :615–622, 1962.

Brian D Ripley. *Pattern recognition and neural networks*. Cambridge university press, 1996.

Frank Rosenblatt. The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6) :386–408, 1958.

Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

Alex J. Smola and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3) :199–222, 2004.

Johan A. K. Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3) :293–300, 1999.

Johan A. K. Suykens, Tony Van Gestel, Jos De Brabanter, Bart De Moor, and Joos Vandewalle. *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 1995.

Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.