COURS DE DEUXIÈME ANNÉE
INGÉNIEUR ISIMA

Machine-Learning
Méthodes à noyau

Vincent Barra

Table des matières

1 Rappel sur les espaces de Hilbert

On suppose connues ici les notions de normes, d'espaces vectoriels normés et de produit scalaire.

Définition 1
Soit $\mathcal{H}$ un espace normé. Un sous-ensemble $\mathcal{M} \subset \mathcal{H}$ est dense dans $\mathcal{H}$ si $\mathcal{H} = \bar{\mathcal{M}}$, où $\bar{\mathcal{M}}$ est la fermeture de $\mathcal{M}$ dans $\mathcal{H}$.

Théorème 1
Théorème de projection
Soit $\mathcal{M}$ un sous-espace vectoriel fermé de l'espace vectoriel de $\mathcal{H}$. Soit $x \in \mathcal{H}$.
Alors il existe un unique $\tilde{x} \in \mathcal{M}$ tel que $\|x-\tilde{x}\| = \inf_{y\in\mathcal{H}} \|x-y\|$. $\tilde{x}$ est la projection orthogonale de x sur $\mathcal{M}$ et est noté $P_{\mathcal{M}}(x)$.

Exemples :
Soit $(x_t)_{t\in T}$ une famille d'un espace vectoriel $\mathcal{H}$ muni d'un produit scalaire $\langle .,. \rangle_{\mathcal{H}}$.

1. L'espace engendré par $(x_t)_{t\in T}$ est l'ensemble des combinaisons linéaires finies de $x_t, t \in T$ noté $Lin\{x_t, t \in T\}$.
2. L'espace fermé engendré par $(x_t)_{t\in T}$ est le plus petit sous-espace fermé vectoriel de $\mathcal{H}$ qui contient $(x_t)_{t\in T}$. On le note $\overline{Lin}\{x_t, t \in T\}$.
3. Un ensemble $(e_t)_{t\in T}$ d'éléments de $\mathcal{H}$ est dit orthonormal si et seulement si $\forall (s,t) \in T^2, \langle e_s, e_t \rangle_{\mathcal{H}} = \delta_{s,t}$.
4. Soit $\{e_1, \ldots, e_k\}$ un système orthonormal de $\mathcal{H}$.
   Soit $\mathcal{M} = \overline{Lin}\{e_1, \ldots, e_k\}$. Alors $\forall x \in \mathcal{H}$:
   — $P_{\mathcal{M}}(x) = \sum_{i=1}^{k} \langle x, e_i \rangle_{\mathcal{H}} e_i$
   — $\left\|P_{\mathcal{M}}(x)\right\|^2 = \sum_{i=1}^{k} \left|\langle x, e_i \rangle_{\mathcal{H}}\right|^2$
   — $\left\| x - \sum_{i=1}^{k} \langle x, e_i \rangle_{\mathcal{H}} e_i \right\| \leq \left\| x - \sum_{i=1}^{k} c_i e_i \right\| \forall (c_1, \ldots, c_k) \in \mathbb{R}^k$.
   On a égalité si $c_i = \langle x, e_i \rangle_{\mathcal{H}}$.

Définition 2
Un espace normé $\mathcal{H}$ est dit **séparable** *si et seulement si il contient une suite dénombrable dense. Il est dit* **complet** *si et seulement si toute suite de Cauchy de $\mathcal{H}$ converge dans $\mathcal{H}$.*

Définition 3
Un espace vectoriel A, muni d'un produit scalaire, est un espace de Hilbert s'il est complet. Dans ce cas, s'il est séparable, il possède une base dénombrable.

Dans la suite, nous considérerons des espaces de Hilbert séparables.

Exemples :

1. $\mathbb{R}^n$ muni de la norme euclidienne, et du produit scalaire canonique est un espace de Hilbert.
2. $(\mathbb{L}^2_n, \langle .,. \rangle)$ est un espace de Hilbert de dimension infinie.

2 ESPACES DE HILBERT À NOYAU REPRODUISANT

Les espaces de Hilbert à noyau reproduisant (EHNR) sont des espaces d'hypothèses ayant de bonnes propriétés vis à vis du problème d'apprentissage. La principale de ces propriétés est la propriété de reproduction, qui relie les normes dans l'espace de Hilbert à l'algèbre linéaire.

2.1 Écriture simplifiée du noyau reproduisant

Soit $\mathcal{H}$ un espace de Hilbert de fonctions : $f : E \to \mathbb{R}$.

Définition 4
(Noyau reproduisant de $\mathcal{H}$)
Une fonction $K : \begin{array}{ccc} E*E & \to & \mathbb{R} \\ (s,t) & \mapsto & K(s,t) \end{array}$ *est un noyau reproduisant de $\mathcal{H}$ si et seulement si :*

1. $\forall t \in E,\ K(\bullet,t) : \begin{array}{ccc} E & \to & \mathbb{R} \\ s & \mapsto & K(s,t) \end{array}$ *est un élément de $\mathcal{H}$.*
2. $\forall t \in E, \forall \varphi \in \mathcal{H}, \left\langle \varphi, K(\bullet,t) \right\rangle_{\mathcal{H}} = \left\langle \varphi(\bullet), K(\bullet,t) \right\rangle_{\mathcal{H}} = \varphi(t)$.
   On reproduit φ par produit scalaire.

Exemple : Orthogonalisation de Gram-Schmidt
Soit $\mathcal{H}$ un espace de Hilbert de dimension n finie avec une base $(f_1, \ldots, f_n)$.
Le produit scalaire sur $\mathcal{H}$ est alors défini par les nombres $g_{i,j} = \left\langle f_i, f_j \right\rangle_{\mathcal{H}}$ pour tout $i, j = 1 \ldots, n$.
La matrice G définit le produit scalaire et s'appelle la matrice de base de Gram.

Si $f = \sum_{i=1}^{n} a_i f_i$ et $g = \sum_{i=1}^{n} b_i f_i$, alors $\langle f, g \rangle = \sum_{i=1}^{n} \sum_{j=1}^{n} a_i b_j g_{i,j}$.
Prenons une base orthonormale $(e_1, \ldots, e_n)$ de $\mathcal{H}$.
La fonction $K : E * E \to \mathbb{R}$ donnée par $K(x,y) = \sum_{i=1}^{n} e_i(x) e_j(y)$ définit alors un noyau reproduisant de $\mathcal{H}$.

Corollaire 1
Tout espace de Hilbert de dimension finie admet un noyau reproduisant.

Nous introduisons les EHNR de deux manières, une première abstraite et une seconde plus intuitive.

2.2 Forme linéaire et noyau reproduisant

Définition 5
Soit $L : \mathcal{H} \to \mathbb{R}$ une forme linéaire. L est continue (ou bornée) s'il existe $M > 0$ tel que

$$(\forall f \in \mathcal{H}) \quad |L(f)| \leq M\|f\|$$

$\|.\|$ *étant la la norme de l'espace de Hilbert $\mathcal{H}$.*

Théorème 2
(Riesz) Soit $L : \mathcal{H} \to \mathbb{R}$ une forme linéaire. L bornée. Il existe $K \in \mathcal{H}$ telle que

$$(\forall f \in \mathcal{H}) \quad L(f) = \langle K, f \rangle$$

Définition 6
Soit $t \in X$. On appelle **fonctionnelle d'évaluation linéaire** L_t *(ou forme linéaire) une fonction :*

$$\begin{array}{rcl} L_t : \mathcal{H} & \to & \mathbb{R} \\ f & \to & f(t) \end{array}$$

linéaire par rapport à f.

Le théorème de Riesz permet alors d'affirmer que tout espace de Hilbert ayant une fonctionnelle d'évaluation linéaire bornée possède un élément qui évalue tous ses vecteurs par simple produit scalaire.

Définition 7
Pour $t \in X$, soit L_t une fonctionnelle d'évaluation linéaire bornée, et $\mathcal{H}$ un espace de Hilbert. $\mathcal{H}$, muni de L_t est un **espace de Hilbert à noyau reproduisant (EHNR)***, noté $\mathcal{H}_K$*

La notation indicée K se réfère à la définition d'un noyau reproduisant K :

Définition 8
Le **noyau reproduisant** *est une fonction $K : E \times E \to \mathbb{R}$, symétrique, semi-définie positive, i.e. vérifiant pour tous réels a_i et tous vecteurs $t_i, t_j \in E$*

$$\sum_{i,j=1}^{n} a_i a_j K(t_i, t_j) \geq 0$$

La relation entre K et $\mathcal{H}$ se traduit par $K(s,t) = \langle K_s, K_t \rangle$ et $K(t,.) = K_t$ Il existe une relation très étroite entre un espace de Hilbert à noyau reproduisant et son noyau reproduisant associé, formalisée par le théorème suivant :

Théorème 3
(Aronszajn, 1950)

1. *Pour tout espace de Hilbert à noyau reproduisant, il existe un noyau reproduisant unique.*

2. *Réciproquement, étant donnée une fonction $K : E \times E \to \mathbb{R}$ symétrique, semi-définie positive, il est possible de construire un espace de Hilbert à noyau reproduisant ayant K pour noyau reproduisant.*

Nous présentons ici des éléments de preuve. Si $\mathcal{H}_K$ est un EHNR, il existe K_t, un représentant de l'évaluation de tout t. Définissons alors $K(s,t) = \langle K_s, K_t \rangle$. On a alors directement :

$$\begin{aligned} \left\| \sum_j a_j K_{t_j} \right\|^2 &\geq 0 \\ \left\| \sum_j a_j K_{t_j} \right\|^2 &= \sum_{i,j} a_i a_j \left\langle K_{t_i}, K_{t_j} \right\rangle \\ \sum_{i,j} a_i a_j K(t_i,t_j) &= \sum_{i,j} a_i a_j \left\langle K_{t_i}, K_{t_j} \right\rangle \end{aligned}$$

et K est semi définie positive.
Réciproquement, soit un noyau reproduisant $K(.,.)$, on définit pour tout $t \in E K_t(.) = K(t,.)$ On montre alors que l'on peut simplement constuire un espace de Hilbert à noyau reprosuisant $\mathcal{H}_K$ à partir de l'ensemble des fonctions formées par combinaison linéaire des fonctions K_{t_i}, muni du pruduit scalaire

$$\left\langle \sum_i a_i K_{t_i}, \sum_i a_i K_{t_i} \right\rangle = \sum_{i,j} a_i a_j \left\langle K_{t_i}, K_{t_j} \right\rangle = \sum_{i,j} a_i a_j K(t_i,t_j)$$

Puisque K est semi définie positive, le produit scalaire est bien défini et on peut vérifier que pour tout $f \in \mathcal{H}_K$, $\langle K_t, f \rangle = f(t)$.

2.3 Le noyau intégral

Soit $K : E \times E \to \mathbb{R}$ une fonction (noyau) symétrique continue. On définit pour $H \in L^2$ de dimension finie l'opérateur $L_K : H \to \mathbb{R}$ par :

$$L_K(f) = \int_E K(\bullet,t) f(t) dt$$

K s'appelle le noyau intégral. Il est semi-défini positif si et seulement si L_K l'est. Donc si K est semi-défini positif, les valeurs propres de L_K sont positives. Notons les $\lambda_1 \cdots \lambda_k$, et $\phi_1 \cdots \phi_k$ les vecteurs propres associés. On a en particulier $\langle \phi_i, \phi_j \rangle = \delta_{ij}$

Théorème 4
Théorème de Mercer
Étant donnés les éléments propres de l'équation intégrale L_K, définie par un noyau symétrique, défini positif K, on peut écrire

$$(\forall s,t \in E) K(s,t) = \sum_{j=1}^{k} \lambda_j \phi_j(s) \phi_j(t)$$

la convergence étant en norme L_2 sur E

On peut alors définir le EHNR comme l'espace des fonctions combinaisons linéaires des vecteurs propres de l'équation intégrale :

$$\mathcal{H}_K = \left\{ f, f(s) = \sum_j c_j \phi_j(s), \|f\|_{\mathcal{H}_K} < \infty \right\}$$

où $\|f\|_{\mathcal{H}_K}$ est définie par

$$\|f(s)\|^2_{\mathcal{H}_K} = \left\langle \sum_j c_j \phi_j(s), \sum_j c_j \phi_j(s) \right\rangle_{\mathcal{H}_K} = \sum_j \frac{c_j^2}{\lambda_j}$$

et où

$$\langle f, g \rangle_{\mathcal{H}_K} = \left\langle \sum_j c_j \phi_j(s), \sum_j d_j \phi_j(s) \right\rangle_{\mathcal{H}_K} = \sum_j \frac{c_j d_j}{\lambda_j}$$

2.4 Connexion avec les noyaux

On rencontre souvent le concept d'espace de Hilbert à noyau reproduisant sous le vocable "astuce du noyau'" (Kernel trick), en particulier lorsque l'on étudie les Machines à Vecteurs de Support (SVM, section4), ou plus généralement les méthodes à noyau. Les points $x \in E \subset \mathbb{R}^d$ sont projetés dans un espace de grande dimension par les éléments propres du noyau reproduisant (la dimension de l'espace est égale au nombre de valeurs propres non nulles de l'opérateur intégral) :

$$x \mapsto \Phi(x) = (\sqrt{\lambda_1}\Phi_1(x) \cdots \sqrt{\lambda_k}\Phi_k(x))$$

Le Kernel Trick sera abordé dans la section 5.

3 Complexité d'un espace d'hypothèses

Nous allons mesurer la complexité d'un espace d'hypothèses à partir de la norme d'un espace de Hilbert à noyau reproduisant. Dans la suite, nous nous intéressons aux espaces $\mathcal{H}_K$ de fonctions f vérifiant

$$\|f\|_{\mathcal{H}_K} \leq A$$

Restreindre la norme d'un EHNR (régulariser) permet d'imposer une contrainte de lissage sur les fonctions de l'espace d'hypothèse, et éviter par exemple le phénomène d'overfitting de données. Dans la suite, nous n'étudions que des méthodes de régularisation fondées sur la méthode de Tikhonov.

3.1 Le théorème du représentant

Soit un problème d'apprentissage utilisant une base d'entraînement $\left\{(x_i, y_i), x_i \in \mathbb{R}^d, 1 \leq i \leq n\right\}$. Soit V une fonction de perte évaluant le coût induit par l'approximation de y_i par $f(x_i)$, f fonction apprise par le modèle. En minimisation de risque empirique, trois formes de régularisation sont couramment utilisées :

1. régularisation de Tikhonov : il s'agit ici de contraindre indirectement l'espace des hypothèses $\mathcal{H}$, en ajoutant un terme de pénalité

$$\min_{f\in\mathcal{H}}\left[\frac{1}{n}\sum_{i=1}^{n}V(f(x_i),y_i)+\lambda\Omega(f)\right]$$

2. régularisation d'Ivanov : il s'agit ici de contraindre directement $\mathcal{H}$:

$$\min_{f\in\mathcal{H}}\left[\frac{1}{n}\sum_{i=1}^{n}V(f(x_i),y_i)\right] \quad \text{sous } \Omega(f)\leq\tau$$

3. régularisation de Philips : là encore, il s'agit de contraindre directement $\mathcal{H}$:

$$\min_{f\in\mathcal{H}}\Omega(f) \quad \text{sous} \quad \left[\frac{1}{n}\sum_{i=1}^{n}V(f(x_i),y_i)\right]\leq\kappa$$

Dans la suite, nous utiliserons une norme EHNR comme fonctionnelle de régularisation : $\Omega(f)=\|f\|^2_{\mathcal{H}_K}$ et les problèmes d'optimisation qui vont nous intéresser sont donc

(P1) $\min_{f\in\mathcal{H}}\left[\frac{1}{n}\sum_{i=1}^{n}V(f(x_i),y_i)+\lambda\|f\|^2_{\mathcal{H}_K}\right]$

(P2) $\min_{f\in\mathcal{H}}\left[\frac{1}{n}\sum_{i=1}^{n}V(f(x_i),y_i)\right]$ sous $\|f\|^2_{\mathcal{H}_K}\leq\tau$

(P3) $\min_{f\in\mathcal{H}}\|f\|^2_{\mathcal{H}_K}$ sous $\left[\frac{1}{n}\sum_{i=1}^{n}V(f(x_i),y_i)\right]\leq\kappa$

En utilisant la seconde formulation de la construction des espaces de Hilbert à noyau reproduisant, on peut écrire

$$\mathcal{H}_K=\left\{f, f(x)=\sum_k\gamma_k\phi_k(x)\right\}$$

et les problèmes (P1),(P2),(P3) se ramènent alors un problème d'optimisation sur les coefficients γ_k. Le nombre de coefficients non nuls (possiblement très grand) définit la dimension de l'EHNR.

Théorème 5

Théorème du représentant

Soit X un ensemble, muni d'un noyau semi défini positif K, $\mathcal{H}_K$ l'EHNR associé, et $\{x_1\cdots x_n\}$ un ensemble de points de X.

Soit $g: X\to\mathbb{R}$ une fonction croissante monotone, et $c: X^n\to\mathbb{R}$ une fonction de coût quelconque.

Alors toute solution au problème

$$Min \quad c((f(x_1),y_1),\cdots,(f(x_n),y_n))+\lambda g(\|f\|_{\mathcal{H}_K})$$

admet une représentation de la forme

$$(\forall x\in X)f(x)=\sum_{i=1}^{n}c_iK(x,x_i)$$

On voit facilement que le problème (P1) est un cas particulier de ce théorème.
Ce théorème, qui découle de la représentation de f dans la base $\phi_1 \cdots \phi_k$, permet de transformer le problème d'optimisation en un problème à n variables réelles.

3.2 Equivalence des trois problèmes

Proposition 1
Soit une fonction de perte convexe. Si f_0 est solution de l'un des problèmes (P1)-(P2)-(P3), alors il existe des constantes λ, τ et κ telles que f_0 soit également solution des deux autres problèmes.

3.3 Régression de Tikhonov (Ridge regression)

Encore appelée machine à vecteurs de support aux moindres carrés, réseau de régularisation ou classification aux moindres carrés régularisée, cette méthode est fondée sur la régularisation de Tikhonov.
Soit la régularisation de Tikhonov où la fonctionnelle de régularisation est une norme EHNR :

$$\min_{f\in\mathcal{H}} \left[\frac{1}{n}\sum_{i=1}^{n} V(f(x_i), y_i) + \lambda \|f\|^2_{\mathcal{H}_K} \right]$$

En utilisant la fonction de perte $\frac{1}{n}\sum_{i=1}^{n} V(f(x_i), y_i) = \frac{1}{n}\sum_{i=1}^{n} (f(x_i) - y_i)^2$, le problème d'optimisation devient

$$\min_{f\in\mathcal{H}} \left[\frac{1}{n}\sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|^2_{\mathcal{H}_K} \right] \tag{1}$$

Par le théorème du représentant, un minimum est de la forme $f(x) = \sum_{i=1}^{n} c_i K(x, x_i)$, ce qui transforme un problème d'optimisation en dimension possiblement très grande, en un problème d'optimisation sur les n coefficients c_i.
Dans la suite, nous noterons indifféremment K la fonction noyau ou la matrice $n \times n$ d'élément (i, j) $K_{ij} = K(x_i, x_j)$. La fonction f, évaluée en x_j s'écrit alors

$$\begin{aligned} f(x_j) &= \sum_{i=1}^{n} K(x_i, x_j) c_i \\ &= [Kc]_j \end{aligned}$$

où $[Kc]_j$ est le j-ième élément du vecteur obtenu en multipliant K par le vecteur c. De même, $\|f\|^2_{\mathcal{H}_K} = c^T K c$ et le problème d'optimisation (1) devient

$$arg \min_{c\in\mathbb{R}^n} \left[g(c) \right]$$

avec

$$g(c) = \frac{1}{n}\|Kc - y\|^2 + \lambda c^T K c$$

g étant convexe et différentiable, la solution est obtenue en annulant la dérivée de g par rapport à c : $\frac{\partial g}{\partial c} = \frac{2}{n}K(Kc - y) + 2\lambda Kc = 0$, ce qui amène à résoudre le système linéaire (type gradient conjugué)

$$(K + \lambda nI)^{-1} y = c$$

Ce système possède certaines propriétés :

1. $(K + \lambda nI)$ est semi-définie positive, et bien conditionnée si λ n'est pas trop petit. Si $\lambda > 0$, elle est de plus inversible, mais en pratique, on utilisera plutôt un algorithme de résolution de systèmes linéaires.
2. Lorsque $\lambda \to 0$, la solution tend vers la solution gaussienne qui minimise le risque empirique. Lorsque $\lambda \to \infty$, la solution tend vers la fonction nulle.
3. En pratique, pour de grands ensembles d'apprentissage, cette méthode est inutilisable puisqu'elle nécessite de calculer et stocker toute la matrice K.

L'approche décrite ici peut être indifféremment utilisée en classification ou en régression.

4 Machines à vecteurs de support

Les machines à vecteurs de support (SVM - Support vector Machines ou Séparateur à Vaste Marge) [**?**] ont été définies de nombreuses manières, et sont appliquées dans de nombreux domaines depuis quelques années. Nous proposons ici deux constructions, une reliée directement à la régularisation, et une autre plus classique issue d'un point de vue géométrique.

4.1 SVM et classification

Dans la suite, on se donne n exemples d'apprentissage $\{x_i, y_i\},\ i = 1,^T s, l\ ,\ x_i \in R^d$,et $(y_i \in \{-1, 1\})$.

4.2 Approche régularisation

L'approche consiste à partir d'un problème de régularisation de Tikhonov, utilisant une norme EHNR pour fonctionnelle de régularisation et pour fonction de perte $V(f(x_i), y_i) = \big(1 - y_i f(x_i)\big)_+$ (fonction de perte charnière). Le problème d'optimisation résultant est donc

$$\min_{f \in \mathcal{H}} \left[\frac{1}{n} \sum_{i=1}^{n} \big(1 - y_i f(x_i)\big)_+ + \lambda \|f\|^2_{\mathcal{H}_K} \right] \tag{2}$$

La fonction est non différentiable en $(1 - y_i f(x_i)) = 0$. On introduit alors les variables d'écart ξ_i et on transforme le problème (2) en un problème d'optimisation sous contraintes :

$$\begin{array}{lll} \min\limits_{f \in \mathcal{H}} & \frac{1}{n} \sum\limits_{i=1}^{n} \xi_i + \lambda \|f\|^2_{\mathcal{H}_K} & \\ \text{sous} & y_i f(x_i) \geq 1 - \xi_i & 1 \leq i \leq n \\ & \xi_i \geq 0 & 1 \leq i \leq n \end{array}$$

Le théorème du représentant permet de réécrire ce problème en un problème quadratique contraint

$$\begin{array}{lll} \min\limits_{c\in\mathbb{R}^n} & \frac{1}{n}\sum\limits_{i=1}^{n}\xi_i + \lambda c^T K c & \\ \text{sous} & y_i \sum\limits_{j=1}^{n} c_j K(x_i,x_j) \geq 1-\xi_i & 1\leq i\leq n \\ & \xi_i \geq 0 & 1\leq i\leq n \end{array}$$

Le modèle SVM autorise un biais b non régularisé, de sorte que le théorème du représentant donne

$$f(x) = \sum_{i=1}^{n} c_i K(x,x_i) + b$$

d'où la formulation primale du SVM

$$\begin{array}{lll} \min\limits_{c\in\mathbb{R}^n,\xi\in\mathbb{R}^n} & \frac{1}{n}\sum\limits_{i=1}^{n}\xi_i + \lambda c^T K c & \\ \text{sous} & y_i \left(\sum\limits_{j=1}^{n} c_j K(x_i,x_j) + b \right) \geq 1-\xi_i & 1\leq i\leq n \\ & \xi_i \geq 0 & 1\leq i\leq n \end{array}$$

Le Lagrangien est alors

$$\begin{aligned} L(c,\xi,b,\alpha,\eta) &= \frac{1}{n}\sum_{i=1}^{n}\xi_i + \lambda c^T K c \\ &- \sum_{i=1}^{n}\alpha_i \left(y_i \left[\sum_{j=1}^{n} c_j K(x_i,x_j) + b \right] - 1 + \xi_i \right) \\ &- \sum_{i=1}^{n}\eta_i\xi_i \end{aligned}$$

L'objectif est de minimiser L par rapport à c, b, ξ_i, et de maximiser ce Lagrangien par rapport à α, η, sous les contraintes du problème primal et des contraintes de positivité sur α, η. En annulant les dérivées de L par rapport à b et ξ, on obtient

$$\begin{aligned} \frac{\partial L}{\partial b} = 0 &\Rightarrow \sum_{i=1}^{n}\alpha_i y_i = 0 \\ \frac{\partial L}{\partial \xi_i} = 0 &\Rightarrow \frac{1}{n} - \alpha_i - \eta_i = 0 \Rightarrow 0 \leq \alpha_i \leq \frac{1}{n} \end{aligned}$$

Ces contraintes devront donc être satisfaites à l'optimalité, ce qui amène à considérer le lagrangien réduit :

$$L^R(c,\alpha) = \lambda c^T K c - \sum_{i=1}^{n}\alpha_i \left(y_i \sum_{j=1}^{n} c_j K(x_i,x_j) - 1 \right)$$

L'annulation de la dérivée partielle de L^R par rapport à c donne

$$2\lambda K c - K Y \alpha = 0 \Rightarrow c_i = \frac{\alpha_i y_i}{2\lambda}$$

où $Y = diag(y_i)$. En substituant dans l'expression de c, on obtient la formulation duale du SVM

$$\begin{array}{ll} \max\limits_{\alpha\in\mathbb{R}^n} & \sum\limits_{i=1}^{n}\alpha_i - \frac{1}{4\lambda}\alpha^T Q\alpha \\ \text{sous} & \sum\limits_{i=1}^{n} y_i\alpha_i = 0 \\ & 0 \le \alpha_i \le \frac{1}{n} \qquad 1 \le i \le n \end{array}$$

où $Q = yKy^T$.

Pour aboutir à la formulation classique des SVM, on remplace le paramètre de régularisation λ par un paramètre $C = \frac{1}{2\lambda n}$ de sorte que le problème d'optimisation initial s'écrit :

$$\min_{f\in\mathcal{H}} C\sum_{i=1}^{n} V(y_i, f(x_i)) + \frac{1}{2}\|f\|^2_{\mathcal{H}_K}$$

et les formulations primales et duales s'écrivent alors

$$\begin{array}{lll} \min\limits_{c\in\mathbb{R}^n,\xi\in\mathbb{R}^n} & C\sum\limits_{i=1}^{n}\xi_i + \frac{1}{2}c^T Kc & \\ \text{sous} & y_i\left(\sum\limits_{j=1}^{n} c_j K(x_i,x_j) + b\right) \ge 1-\xi_i & 1 \le i \le n \\ & \xi_i \ge 0 & 1 \le i \le n \end{array} \tag{3}$$

et

$$\begin{array}{ll} \max\limits_{\alpha\in\mathbb{R}^n} & \sum\limits_{i=1}^{n}\alpha_i - \frac{1}{2}\alpha^T Q\alpha \\ \text{sous} & \sum\limits_{i=1}^{n} y_i\alpha_i = 0 \\ & 0 \le \alpha_i \le C \qquad 1 \le i \le n \end{array} \tag{4}$$

L'interprétation des variables duales α_i permet de distinguer trois cas :

1. $\alpha_i = 0$ et x_i est intérieur à la classe
2. $\alpha_i < C$ et $y_i f(x_i) = 1$ et le point est sur la marge
3. $\alpha_i = C$ et $y_i f(x_i) = 1$ et le point est intérieur à la marge

Comme λ, la constante C contrôle le compromis entre précision de la classification et norme de la fonction.

4.3 Approche géométrique

Une approche plus traditionnelle pour introduire les SVM est de partir du concept d'hyperplan séparateur des exemples positifs et négatifs de l'ensemble d'apprentissage. On définit alors la marge comme la distance du plus proche exemple à cet hyperplan, et on

espère intuitivement que plus grande sera cette marge, meilleure sera la capacité de généralisation de ce séparateur linéaire.
Un hyperplan de $\mathbb{R}^d$ est défini par

$$w^T x + b = 0$$

w étant le vecteur normal à l'hyperplan. La fonction

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

permet, si elle sépare les données d'apprentissage, de les classifier correctement. Un tel hyperplan ,représenté par (w,b) peut également être exprimé par $(\lambda w, \lambda b), \lambda \in \mathbb{R}$. Il est donc nécessaire de définir l'hyperplan canonique comme étant celui éloigné des données d'une distance au moins égale à 1. En fait, on impose qu'un exemple au moins de chaque classe soit à distance égale à 1. On considère alors le couple (w,b) tel que :

$$w^T x_i + b \geq +1 \text{ si } y_i = +1$$
$$w^T x_i + b \leq -1 \text{ si } y_i = -1$$

ou de manière plus compacte

$$\forall i \quad y_i(w^T x_i + b) \geq 1$$

Puisque l'on cherche à avoir la marge la plus grande possible, il est alors intéressant de calculer la distance, au sens de la norme euclidienne, d'un point x_i à cet hyperplan. Cette distance est la longueur du vecteur reliant x_i à sa projection sur l'hyperplan, et est donnée par :

$$d\Big((w,b)\ ,\ x_i\Big) = \frac{y_i(w^T x_i + b)}{\| w \|} \geq \frac{1}{\| w \|}$$

Intuitivement, on veut trouver l'hyperplan qui maximise la cette distance, pour les x_i les plus proches. L'équation précédente permet d'affirmer que cela est réalisé en minimisant $\| w \|$, sous les contraintes de bonnes classification.
Le problème s'écrit alors comme un problème de minimisation sous contraintes :

$$\begin{aligned} \min_{w \in \mathbb{R}^d} \quad & \| w \|^2 \\ \text{sous} \quad & y_i(w^T x_i) \geq 1, \quad 1 \leq i \leq n \end{aligned}$$

En introduisant les multiplicateurs de Lagrange, le problème dual s'écrit :

$$\begin{aligned} \textit{min} \quad & W(\alpha) = -\sum_{i=1}^{n} \alpha_i + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j ({x_i}^T x_j) \\ \textit{sous} \quad & \sum_{i=1}^{n} y_i \alpha_i = 0 \\ (\forall 1 \leq i \leq n) \quad & 0 \leq \alpha_i \leq C \end{aligned}$$

où α est le vecteur des n multiplicateurs de Lagrange à déterminer, et C est une constante. En définissant la matrice $(H)_{ij} = y_i y_j (x_i^T x_j)$ et $\mathbf{1}$ le vecteur de $\mathbb{R}^n$ dont toutes les composantes sont égales à 1, le problème se réécrit comme un problème de programmation quadratique (QP) :

$$\begin{aligned} \textit{min} \quad & W(\alpha) = -\alpha^T \mathbf{1} + \tfrac{1}{2}\alpha^T H \alpha \\ \text{sous} \quad & \alpha^T y = 0 \\ & 0 \leq \alpha \leq C\mathbf{1} \end{aligned}$$

pour lequel de nombreuses méthodes de résolution ont été développées.
En dérivant l'équation précédente, il est possible de montrer que l'hyperplan optimal (canonique) peut être écrit comme

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

et w est donc juste une combinaison linéaire des exemples d'apprentissage.
On peut également montrer que

$$(\forall 1 \leq i \leq n) \quad \alpha_i (y_i (w^T x_i + b) - 1) = 0$$

ce qui exprime que lorsque $y_i(w^T x_i + b) > 1$, alors $\alpha_i = 0$: seuls les points d'apprentissage les plus proches de l'hyperplan (tels que $\alpha_i > 0$) contribuent au calcul de ce dernier, et on les appelle les vecteurs de support.
En supposant avoir résolu le problème QP, et donc en disposant du α qui permet de calculer le vecteur w optimal, il reste à déterminer le biais b. Pour cela, en prenant un exemple positif x^+ et un exemple négatif x^- quelconques, pour lesquels

$$\begin{aligned} (w^T x^+ + b) &= +1 \\ (w^T x^- + b) &= -1 \end{aligned}$$

on a

$$b = -\frac{1}{2}(w^T x^+ + w^T x^-)$$

L'hyperplan ainsi défini a besoin de très peu de vecteurs de support (méthode éparse). La figure 1 montre deux jeux de points à respectivement 50 et 500 points par classe, tirées selon les mêmes lois. Dans les deux cas, l'hyperplan est défini par un très faible nombre de vecteurs support (en vert).
Il reste à préciser le rôle de la contrainte $0 \leq \alpha \leq C\mathbf{1}$. Lorsque $C \to \infty$, l'hyperplan optimal est celui qui sépare totalement les données d'apprentissage (si tant est qu'il existe). Pour des valeurs de C "raisonnables", des erreurs de classification peuvent être acceptées par le classifieur (soft margin). Pour cela on introduit des variables d'écart ξ_i :

$$\forall 1 \leq i \leq n \quad y_i (w^T x_i + b) > 1 - \xi_i$$

Les vecteurs de support vérifient l'égalité, et les anciennes contraintes peuvent être violées de deux manières :

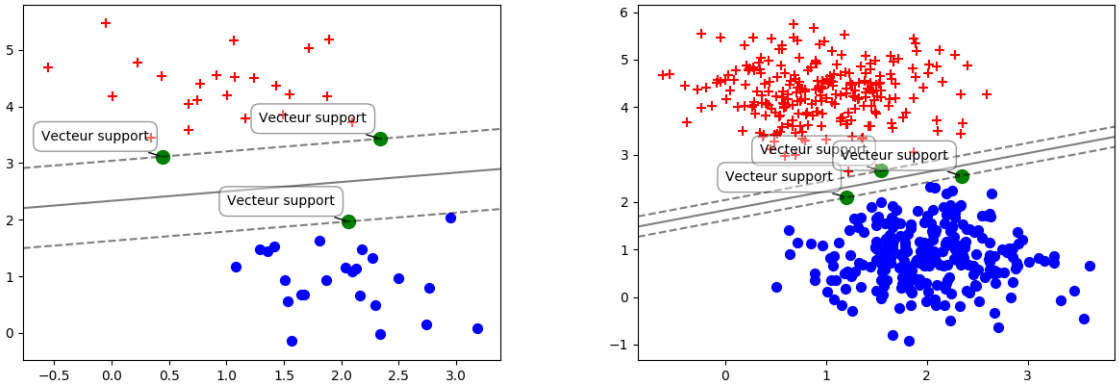


Fig. 1: Séparation linéaire par SVM

1. (x_i, y_i) est à distance inférieure à la marge, mais du bon côté de l'hyperplan
2. (x_i, y_i) est du mauvais côté de l'hyperplan

L'objectif est alors de minimiser la moyenne des erreurs de classification $\sum_{i=1}^{n} \mathbf{1}_{\xi_i > 0}$. Ce problème étant NP-complet (fonction non continue et dérivable), on lui préfère le problème suivant

$$\begin{aligned} Min \quad & \frac{1}{2} w^T w + C \sum_{i}^{n} \xi_i \\ sous \quad & y_i \left(w^T x_i + b \right) = 1 - \xi_i \end{aligned}$$

C représente alors un compromis entre la marge possible entre les exemples et le nombre d'erreurs admissibles. Nous illustrons dans la suite deux situations influencées par C :

— La figure 2 présente une première illustration du rôle de C : dans le cas de données linéairement séparables, un C faible autorisera des vecteurs à rentrer dans la marge (vert). Plus C devient grand, plus le nombre de vecteurs support diminue, pour ne laisser aucun vecteur à distance inférieure à la marge de l'hyperplan optimal

— La figure 3 présente un ensemble de données non linéairement séparables. La valeur de C contrôle le nombre d'erreurs de classification dans le résultat final.

4.4 SVM et régression

Il est également possible, en changeant les fonctions de perte, d'utiliser les SVM en régression non paramétrique (SVR : Support Vector Regression)[**?**], i.e. approcher une fonction de $\mathbb{R}^d$ dans $\mathbb{R}^p$ par les mêmes mécanismes d'optimisation.

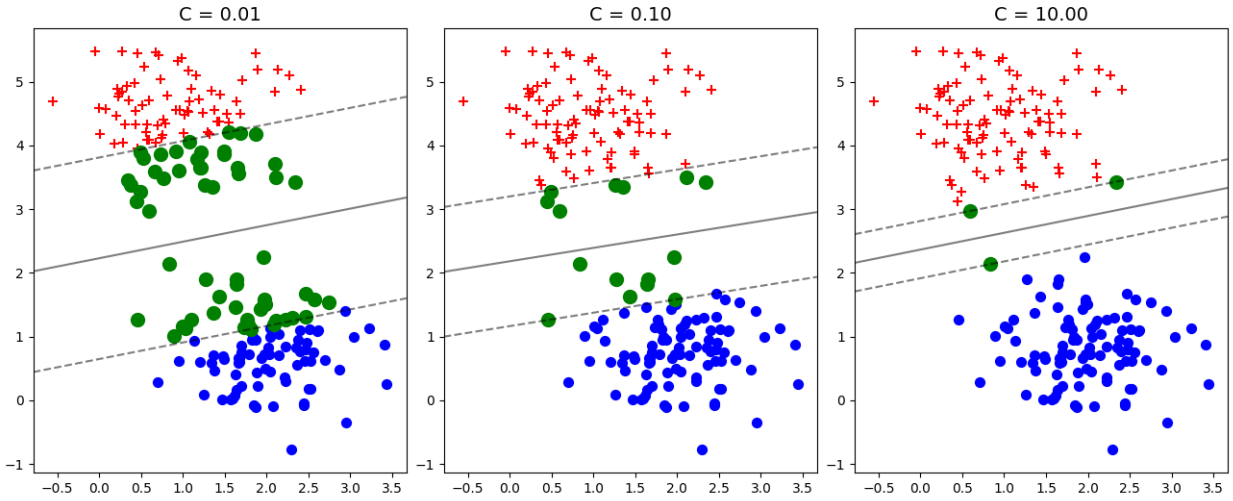


Fig. 2: Séparation linéaire par SVM. Influence de C sur le nombre et la localisation des vecteurs support (en vert)

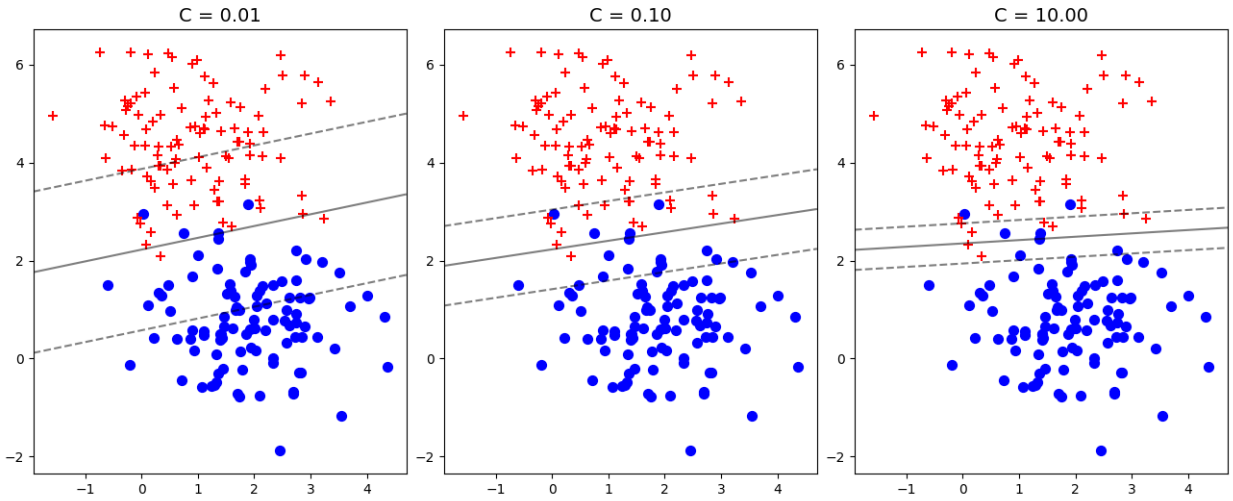
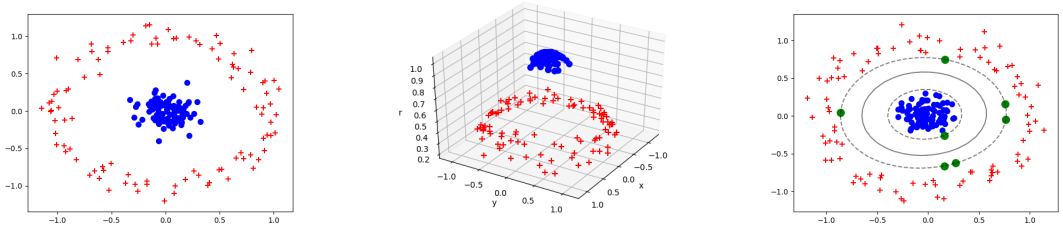


Fig. 3: Séparation non linéaire par SVM linéaire, et influence de C sur les mauvaises classifications.



(a) Données initiales. (b) Mapping des données en 3D. (c) Classification.

Fig. 4: Données non linéairement séparables en 2D, mais séparables en 3D.

5 Astuce du noyau et séparation non linéaire

En utilisant l'ensemble des sections précédentes, il est possible de construire des séparateurs non linéaires très performants, en introduisant l'astuce du noyau dans des algorithmes type SVM.
De manière très macroscopique, il s'agit de transformer les données de $\mathbb{R}^d$ en des vecteurs $z \in \mathbb{R}^{d'}, d' > d$, via une fonction $z = \phi(x)$ en choisissant ϕ de sorte que les données d'entraînement $\{\phi(x_i), y_i\}$ soient linéairement séparables dans $\mathbb{R}^{d'}$. La figure 4 présente un exemple simple de données concentriques 2D (figure 4(a)), non linéairement séparables dans le plan, mais séparables en 3D (figure 4(c)) lorsque ϕ est une simple fonction à base radiale (figure 4(b)).

5.1 Choix de ϕ

Première question à se poser, ϕ existe t elle ? La réponse est donnée par le théorème de Cover

Théorème 6

- *Dans un espace de dimension d, la probabilité que deux classes quelconques de d exemples ne soient pas linéairement séparables tend vers 0 lorsque* $d \to \infty$.
- *Si* $d > n$ *: on peut toujours trouver un hyperplan séparant les exemples (ne garantit pas la capacité de généralisation)*

Deuxième question : comment choisir ϕ ? Il ne s'agit bien sur pas de construire explicitement cette fonction. Il s'agit également de prendre garde à taille de l'espace d'arrivée (d'), les calculs (de la matrice H) pouvant devenir prohibitifs et le sur apprentissage pouvant également apparaître.
Etant donné $z = \phi(x)$, on remplace alors x par z dans la formulation du problème QP :

$$\text{minimize :} \quad W(\alpha) = -\alpha^T 1 + \tfrac{1}{2}\alpha^T H \alpha$$

avec $(H)_{ij} = y_i y_j (\phi(x_i)^T \phi(x_j))$. La détermination de w est alors

$$w = \sum_{i=1}^{n} \alpha_i y_i \phi(x_i)$$

et la fonction de décision

$$\begin{aligned} f(x) = & \quad \text{sign}(w^T\phi(x)+b) \\ = & \quad \text{sign}([\sum_{i=1}^{n}\alpha_i y_i \phi(x_i)]^T\phi(x)+b) \\ = & \quad \text{sign}(\sum_{i=1}^{n}\alpha_i y_i (\phi(x_i)^T\phi(x))+b) \end{aligned}$$

Chaque occurrence d'un $\phi(x_i)$ est reliée à un produit scalaire avec un $\phi(x_j)$. Ainsi, en définissant un noyau K, tel que

$$K(x_i,x_j) = \phi(x_i)^T\phi(x_j) \tag{5}$$

il n'est pas nécessaire de définir explicitement ϕ, ni d'ailleurs l'espace d'arrivée, pour adresser le problème QP. La matrice H s'écrit alors simplement $(H)_{ij} = y_i y_j (K(x_i,x_j))$, et le classifieur

$$f(x) = \text{sign}(\sum_{i=1}^{n}\alpha_i y_i (K(x_i,x))+b)$$

La recherche de l'hyperplan optimal se fait donc dans l'espace d'arrivée, les données originales $x_1 \cdots x_n$ étant séparées par une frontière non linéaire, image réciproque de l'hyperplan par ϕ.

5.2 Choix de K

5.2.1 Noyau polynomial

Considérons $K(x_i,x_j) = \left(x_i^T x_j + \theta\right)^p$ En développant on obtient $\binom{d+p-1}{p}$ termes, chacun d'entre eux étant un polynôme de degré variable des vecteurs d'entrée. K peut donc être vu comme le produit scalaire de deux vecteurs z de très grande taille, d'autant plus importante que p est grand.

5.2.2 Noyau gaussien

Un noyau très utilisé est le noyau gaussien :

$$K(x_i,x_j) = exp(-\frac{\| x_i - x_j \|^2}{2\sigma^2}) \tag{6}$$

où σ est un paramètre (bande passante). Le classifieur est alors

$$f(x) = \text{sign}[\sum_{i=1}^{n}\alpha_i y_i exp(-\frac{\| x - x_i \|^2}{2\sigma^2})+b]$$

qui est une fonction de base radiale (RBF), dont les centres sont les vecteurs supports. Le SVM est alors implicitement utilisé pour trouver le nombre et la position des centres nécessaires au RBF pour une performance en généralisation maximale.

5.2.3 Construction de noyaux

Pour construire un noyau K, il faut que ce noyau calcule un produit scalaire dans l'espace d'arrivée, pour une certaine fonction ϕ. Pour ce faire, deux stratégies sont possibles :

1. poser une fonction ϕ et en déduire K. Hormis dans des cas très simples, cela n'est pas réalisable
2. construire un noyau K, et vérifier qu'il remplit les conditions de Mercer (voir théorème 4).

Ainsi par exemple, on peut construire par exemple :

— un noyau sigmoïde $K(x_i,x_j) = tanh(\eta x_i^T x_j + \theta)$ qui ne satisfait les conditions de Mercer que pour des valeurs particulières de ses paramètres η, θ

— un noyau pour des ensembles $\mathbb{K}(X,Y) = \sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} K(x_i,y_j)$

5.3 Noyaux de données non quantitatives

Il est également possible de définir des noyaux sur des données non numériques, telles que des chaînes de caractères, des graphes. Il suffit pour cela de disposer d'une mesure de similarité entre deux objets. Par exemple :

— si s,s' sont deux chaînes de caractères, $K(s,s')$ peut quantifier la différence de longueur, le nombre de lettres différentes, la distance d'édition minimale,...

— si G,G' sont deux graphes, un noyau K peut être défini à partir de distances calculées entre des sacs de plus courts chemins décrivant G et G'.

5.4 Généralisation d'algorithmes

L'astuce du noyau se généralise à tout algorithme fondé sur un produit scalaire. On peut citer par exemple :

— le perceptron linéaire
— l'analyse discriminante linéaire
— l'ACP
— les K-means
— ...

Remplacer le produit scalaire par la valeur du noyau permet de rendre non linéaire ces méthodes.

A titre d'exemple, l'ACP à noyau (Kernel PCA) permet une réduction de dimension non linéaire, et il est facile de montrer que cette méthode agit comme un multimensional scaling dans l'espace d'arrivée de ϕ.

6 Partie pratique

6.1 SVM

Le module `sklearn.svm` implémente de nombreux algorithmes de SVM. A partir de la documentation, il vous est demandé de classer des données générées par la fonction `make_moons` à l'aide :

— d'un SVM à noyau polynomial, en faisant varier ses paramètres
— d'un noyau gaussien, en faisant varier ses paramètres

Pour chacune des expériences, il vous est demandé d'étudier l'influence des paramètres (figure 5).

6.2 SVR

6.2.1 Données de synthèse

Générez un ensemble de points bruités, échantillons d'une fonction que vous choisirez. A l'aide de la classe `SVR`, tester la régression par noyau (figure 6), en utilisant les noyaux suivants :

— noyau linéaire
— noyau polynomial
— noyau gaussien

Pour chacun des noyaux, vous évaluerez l'importance de leurs paramètres, et en particulier du paramètre C dans la précision de la régression.
Comparez ces résultats à une régression par moindres carrés linéaires (`LinearRegression`) et à une régression de Tikhonov (`Ridge`).

6.2.2 Interpolation d'images

On se propose ici d'utiliser les SVR sur des images pour reconstituer les pixels manquants. A partir d'une image quelconque extraite du jeu de données MNIST (figure 7(a)), supprimer $P\%$ des pixels (pour $10 \leq P \leq 80$, figure 7(b)), et utiliser un SVR sur les pixels restants pour reconstruire le niveau de gris ces pixels manquants (figure 7(c)). Étudier l'influence du noyau et sélectionner ses paramètres par validation croisée (`model_selection`).

6.3 Challenge

Le fichier `vins.txt` contient les résultats d'analyses chimiques de trois classes de vins italiens, provenant de la même région mais de viticulteurs différents. Les analyses portent sur la quantification de 13 indices quantitatifs continus reliés aux vins : degré d'alcool, acide malique, présence de cendres, alcalinité des cendres, magnésium, phénols totaux, flavonoïdes, phénols non flavanoïdes, proanthocyanidines, intensité de la couleur du vin, teinte, OD280/OD315 des vins dilués et proline. Le fichier décrit un vin par ligne (sa classe et ses 13 attributs).
Proposez un algorithme, fondé sur les SVM, de classification multiclasse de ces données.

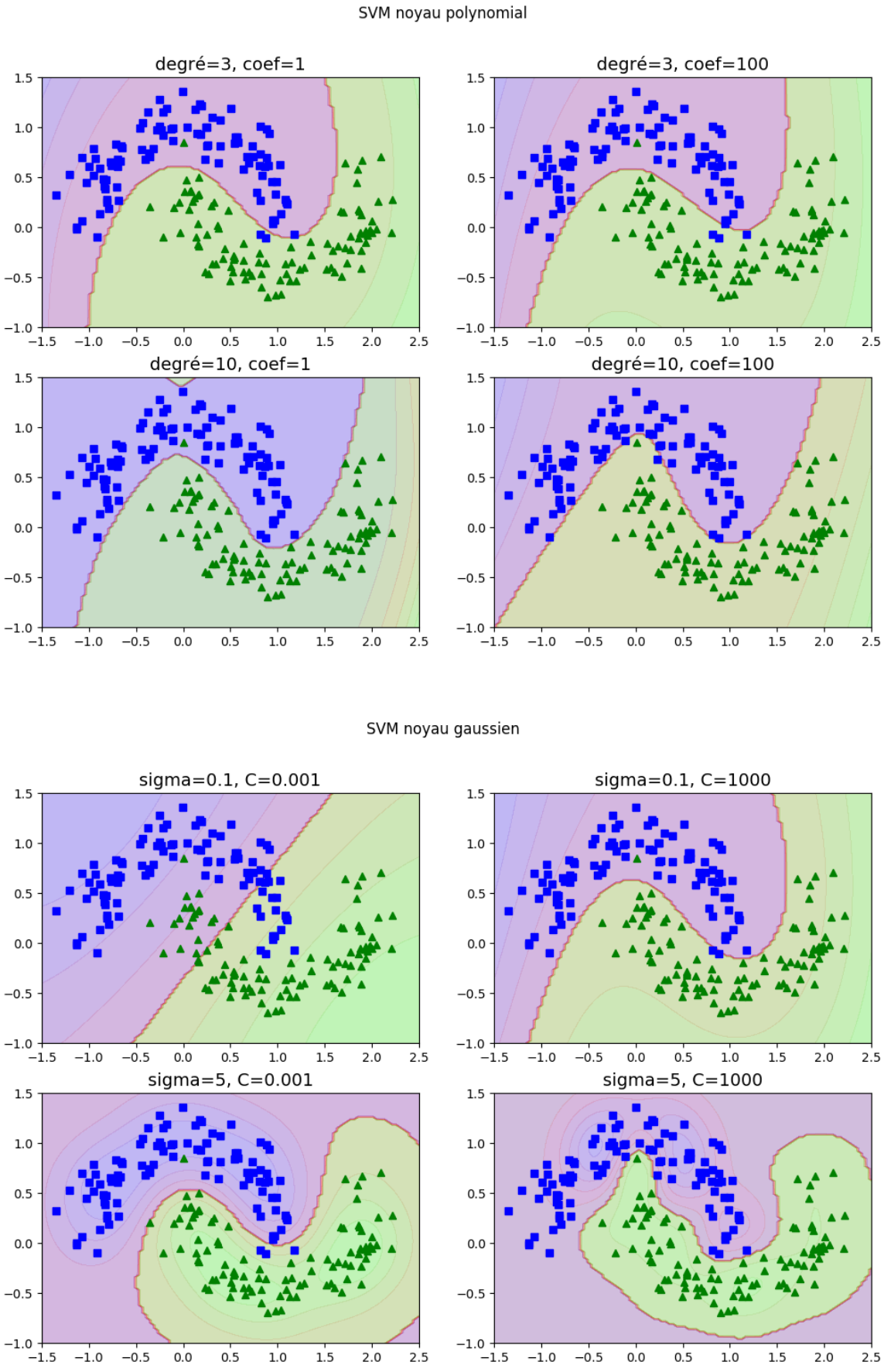


Fig. 5: Noyaux polynomial et gaussien sur des données non linéairement séparables. Influence des paramètres

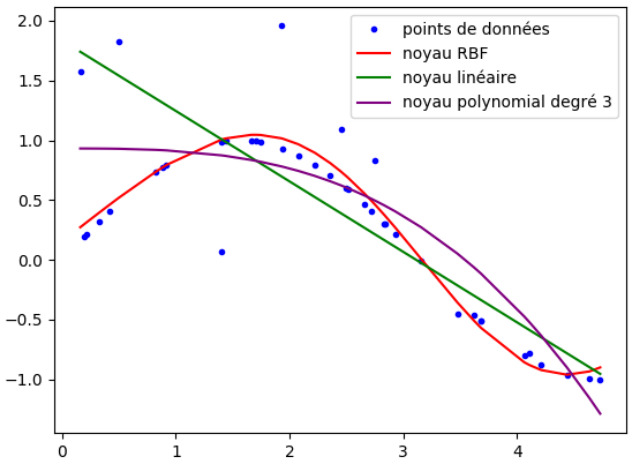
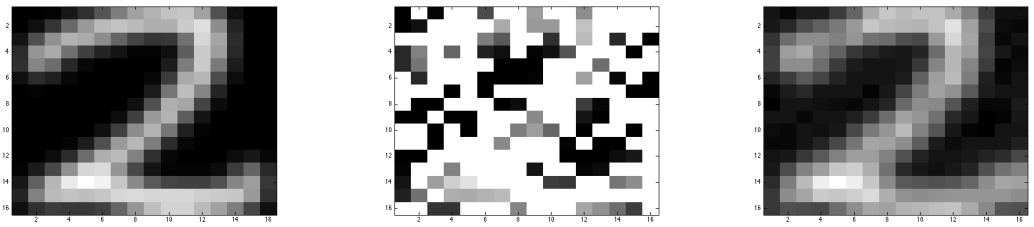


Fig. 6: Régression par SVR, en utilisant différents noyaux



(a) Données initiales. (b) 60% de pixels supprimés. (c) Reconstruction par SVR.

Fig. 7: Reconstruction de données manquantes par SVR, $P = 60$.

Vous construirez un ensemble d'apprentissage et un ensemble de test en fonction des données qui vous sont fournies (faible quantité pour chaque classe, classes non équilibrées). Vous explorerez les stratégies un contre un et un contre tous. Dans les deux cas, vous expliquerez la démarche retenue et évaluerez les performances. Sélectionner le meilleur noyaux, et les paramètres correspondants par validation croisée (`sklearn.model_selection`).