Université Paris Dauphine-PSL
Département de Mathématiques

Analyse en composantes principales à noyaux

Directeurs : M. Denis Pasquignon
M. Patrice Bertrand

Mémoire réalisé par :
Kévin Wardakhan
Erwan Ouabdesselam
Ibrahim Youssouf Abdelatif

Année académique 2024-2025

Table des matières

Remerciements

Nous tenons à exprimer notre profonde gratitude à nos encadrants M. Denis PASQUIGNON et M. Patrice BERTRAND pour leur accompagnement, leur disponibilité et leurs conseils tout au long de ce travail. Nous remercions également M. Yannick VIOSSAT pour la coordination de l'ensemble de cette unité d'enseignement. Aussi, à toutes les personnes qui nous ont soutenus tout le long de notre parcours académique, nous les remercions infiniment.

1 Introduction

Lorsqu'on manipule un grand nombre de variables quantitatives simultanément, la visualisation des données devient rapidement difficile. L'Analyse en Composantes Principales (ACP) permet de réduire la dimension tout en conservant l'essentiel de l'information, facilitant ainsi l'interprétation des structures internes du jeu de données. Toutefois, cette méthode repose sur l'hypothèse implicite que les relations entre les variables sont linéaires, ce qui n'est pas toujours adapté aux situations rencontrées dans la pratique.

Pour dépasser cette limite, l'ACP à noyaux propose une extension non linéaire de l'ACP classique. En exploitant le concept de noyau, elle permet de projeter les données dans un espace de Hilbert de dimension potentiellement infinie, sans jamais calculer explicitement cette projection. Cette approche ouvre la voie à l'analyse de structures complexes, difficilement accessibles par des méthodes linéaires.

Ce mémoire a pour objectif d'explorer les principes théoriques de l'ACP à noyaux, tout en en évaluant l'intérêt pratique à travers plusieurs applications concrètes. C'est ainsi que nous illustrons ses performances sur un ensemble de cas variés : une classification de sentiments à partir de critiques de films , une détection d'anomalies sur le jeu de données MNIST, ainsi qu'un débruitage de signaux ECG. Chaque application est l'occasion de discuter des choix de noyaux, de la sélection du nombre de composantes et de la qualité des résultats obtenus.

Notre approche théorique et expérimentale vise à fournir une compréhension globale de l'ACP à noyaux, de le comparer avec l'ACP mais aussi de ses limites dans le cas pratique.

Exemple illustratif : Afin d'illustrer concrètement l'apport de l'ACP à noyaux, nous présentons ci-dessous un exemple visuel particulièrement évocateur.

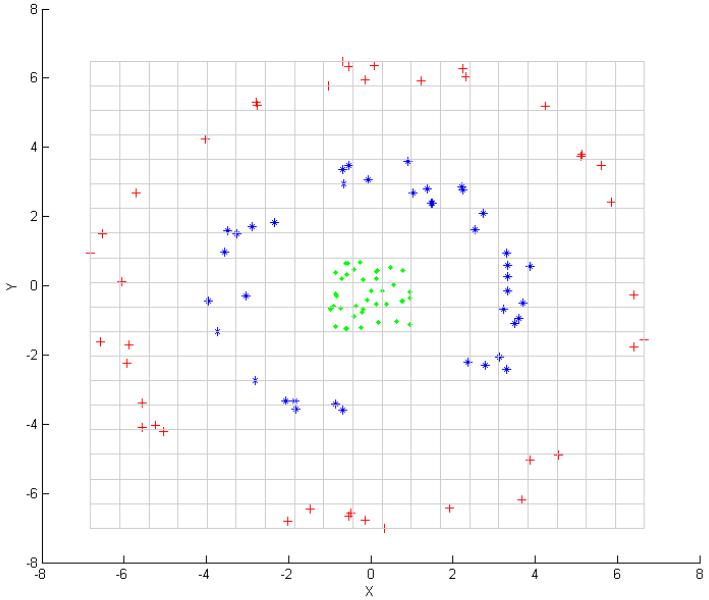

FIGURE 1 – Nuage de points. Source : Wikipédia, 2024

La figure 1 montre une structure de données initiales composée de trois classes réparties de façon non linéaire dans l'espace : une classe centrale (en vert), une classe intermédiaire en spirale (en bleu) et une classe périphérique (en rouge). Ce type de distribution est typiquement difficile à séparer ou à interpréter avec des techniques linéaires comme l'ACP classique.

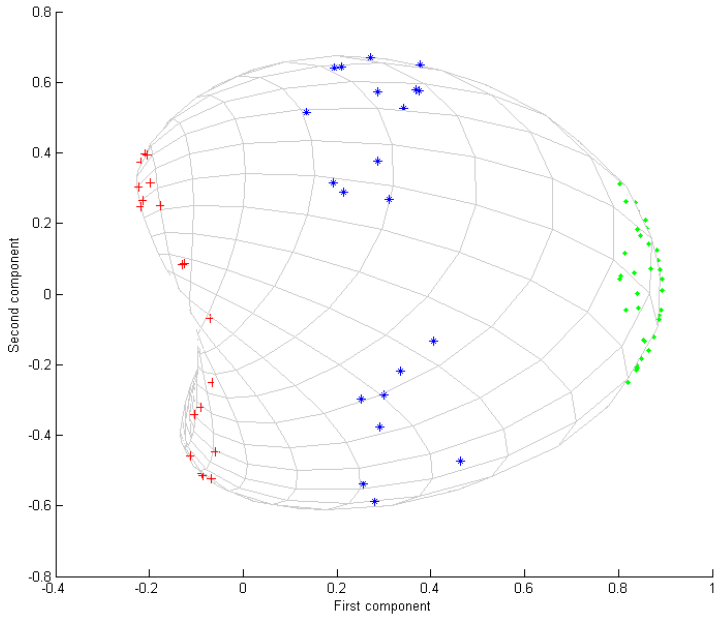

FIGURE 2 – Nuage de points après ACP à noyaux gaussien. Source : Wikipédia, 2024

En revanche, après application de l'ACP à noyaux avec un noyau gaussien (figure 2), les données sont projetées dans un espace transformé où les structures deviennent linéairement séparables. On observe alors un étalement clair des composantes principales, révélant des regroupements bien distincts, ce qui facilite grandement les tâches de classification ou de visualisation. Cette transformation met en lumière la capacité de l'ACP à noyaux à capturer des structures complexes invisibles dans l'espace original, justifiant ainsi son intérêt dans l'analyse exploratoire de données non linéaires.

Approche théorique

2 Analyse en Composantes Principales (ACP)

Le cadre théorique de l'ACP classique présenté ici suit principalement les développements du cours de M. Bertrand et M. Pasquignon [25].

Étant donné l'espace vectoriel $\mathbb{R}^p$, dans lequel on situe un nuage de N points muni chacun d'une masse, et dans lequel on définit une métrique. On souhaite calculer l'inertie totale de ce nuage, et déterminer ses composantes principales.

Les *entrées* d'une analyse factorielle sont donc dans tous les cas les suivantes : l'espace où vivent les individus, les points, les masses affectées aux points, la métrique utilisée.

Les *sorties* sont les composantes principales, les coordonnées des points sur ces axes, et diverses indications pour aider à la décision.

Entre différentes méthodes d'analyse factorielle, en particulier entre l'ACP classique et l'ACP à noyaux, seules les entrées varient. Voici donc des résultats globaux qui sont valides pour les deux méthodes.

2.1 Définitions et notations

Définition : Métrique

On considère un espace euclidien E de dimension finie $n \in \mathbb{N}$, muni du produit scalaire $\langle ., . \rangle$ Soit $\mathcal{B} = \{e_1, ..., e_n\}$ une base de E. Alors, la matrice $M = (\langle e_i, e_j \rangle)_{i,j \in I \times J}$ est appelée métrique de E.

2.1.1 Notations

Soit p variables statistiques réelles X_j $(j = 1, \ldots, p)$ observées sur n individus i $(i = 1, \ldots, n)$, auxquels sont associés des poids p_i :

$$\forall i = 1, \ldots, n : \quad p_i > 0 \quad \text{et} \quad \sum_{i=1}^{n} p_i = 1.$$

Pour chaque individu i, la mesure de la variable X_j est notée $x_i^j = X_j(i)$. Ces mesures sont rassemblées dans une matrice $\mathbf{X}$ de taille $(n \times p)$.

- **Espace des individus** : À chaque individu i est associé le vecteur $\mathbf{x}_i$, correspondant à la i-ème ligne de $\mathbf{X}$ écrite en colonne. Ce vecteur appartient à un espace vectoriel E de dimension p. Nous choisissons $\mathbb{R}^p$ muni de sa base canonique et d'une métrique définie par une matrice $\mathbf{M}$, conférant à E une structure d'espace euclidien. E est appelé **espace des individus**.
- **Espace des variables** : À chaque variable X_j est associé le vecteur $\mathbf{x}^j$, correspondant à la j-ème colonne de $\mathbf{X}$. Ce vecteur appartient à un espace vectoriel F de dimension n. Nous choisissons $\mathbb{R}^n$ muni de sa base canonique et d'une métrique définie par une matrice diagonale $\mathbf{D_p}$ des poids p_i, conférant à F une structure d'espace euclidien. F est appelé **espace des variables**.

Remarque : Ici, E et F sont considérés comme des espaces affines. Ainsi, on peut associer à chaque individu i, un point M_i tel que :

$$\forall i \in [\![1, n]\!], \quad \overrightarrow{OM_i} = \mathbf{x}_i.$$

Chaque axe de cet espace représente une variable. L'ensemble des points $\mathcal{N}(I) = \{M_i \mid 1 \leq i \leq n\}$ est appelé le **nuage des individus**.

De même, à chaque variable X_j, on peut associer un point N_j tel que :

$$\forall j \in [\![1, p]\!], \quad \overrightarrow{ON_j} = \mathbf{x}^j.$$

L'ensemble des points $\mathcal{N}(J) = \{N_j \mid 1 \leq j \leq n\}$ est appelé le **nuage des variables**.

2.1.2 Choix de la métrique M

Lors de notre étude, nous choisissons de considérer la métrique M comme étant la métrique euclidienne canonique. En effet, si l'on considère un espace muni d'une métrique différente de la métrique canonique, il est toujours possible d'effectuer une transformation linéaire sur le nuage de points qui permet d'utiliser la métrique canonique.

2.2 Théorème de l'analyse en composantes principales

Définition : Inertie

L'inertie du nuage $\mathcal{N}(I)$ par rapport à un point $A \in \mathbb{R}^p$ est définie par :

$$I_A\big(\mathcal{N}(I)\big) = \sum_{i=1}^{N} p_i \|M_i - A\|^2$$

Remarque :
En général, le point A sera choisi comme l'origine du repère, ou le barycentre G de $\mathcal{N}(I)$ affecté des poids p_i.

Définition : Centre de gravité du nuage des individus

Le centre de gravité g du nuages des individus admet pour j-ème coordonnée :

$$g_j = \sum_{i=1}^{n} p_i x_i^j$$

Ainsi, on a que :

$$g = \mathbf{X}'\mathbf{D}_p\mathbf{1}_n$$

si on note $\mathbf{1}_n$ le vecteur de $\mathbb{R}^n$ contenant que des 1.

2.2.1 Centrage du nuage

Centrer un tableau de données consiste à soustraire la moyenne de chaque variable, ramenant ainsi le centre de gravité du nuage de points à l'origine. Cette opération permet de se concentrer sur la variabilité des données et d'éliminer l'influence des moyennes, facilitant l'interprétation et la comparaison des variables. De plus, centrer les données supprime tout arbitrage lié à la position du centre de gravité, garantissant que les analyses reflètent uniquement la structure intrinsèque des données.
Pour cela, on considère le tableau centré Y défini par :

$$\forall i \in [\![1, n]\!], \forall j \in [\![1, p]\!], \quad y_i^j = x_i^j - g_j$$

On a donc que :

$$\forall i \in [\![1, n]\!], \quad y^j = x^j - g_j\mathbf{1_n}$$

De plus, en identifiant chaque individu i à son point M_i, on a :

$$y_i = M_i - G.$$

Finalement,

$$Y = X - \mathbf{1_n}'g$$

2.3 Problème d'optimisation

Lors de l'ACP, notre objectif est de trouver les sous-espaces affines de dimension $k \in \mathbb{N}$ qui minimisent l'inertie du nuage de points par rapport à ces sous-espaces. Montrons d'abord que ces solutions doivent nécessairement passer par le centre de gravité G du nuage.
On considère le problème d'optimisation suivant :

$$A^* = \arg\min_{A \in \mathbb{R}^p} I_A(\mathcal{N}(I)),$$

Proposition

L'inertie vérifie :

$$I_A(\mathcal{N}(I)) = I_G(\mathcal{N}(I)) + \|G - A\|^2.$$

Preuve :

En développant la distance :

$$\|M_i - A\|^2 = \|(M_i - G) + (G - A)\|^2,$$

et en appliquant l'identité :

$$\|M_i - A\|^2 = \|M_i - G\|^2 + 2\langle M_i - G, G - A\rangle + \|G - A\|^2.$$

En sommant sur tous les individus :

$$I_A(\mathcal{N}(I)) = \sum_{i=1}^{n} p_i \|M_i - A\|^2.$$

On obtient :

$$I_A(\mathcal{N}(I)) = I_G(\mathcal{N}(I)) + 2\sum_{i=1}^{n} p_i \langle M_i - G, G - A\rangle + \|G - A\|^2.$$

Or, par définition du barycentre :

$$\sum_{i=1}^{n} p_i (M_i - G) = 0.$$

Ainsi, le second terme s'annule et il reste :

$$I_A(\mathcal{N}(I)) = I_G(\mathcal{N}(I)) + \|G - A\|^2.$$

Comme $\|G - A\|^2 \geq 0$ et s'annule uniquement pour $A = G$, on conclut que G est l'unique minimum. □

2.3.1 Formulation mathématique du problème

Nous cherchons maintenant un sous-espace affine $\mathcal{E}_k$ de dimension k qui minimise l'inertie du nuage par rapport à ce sous-espace. Un sous-espace affine de dimension k passant par G peut être écrit sous la forme :

$$\mathcal{E}_k = G + E_k$$

où E_k est un sous-espace vectoriel de dimension k.

Définition : Inertie par rapport à un sous-espace affine

L'inertie projetée sur un sous-espace E_k est donnée par :

$$I_{E_k}(\mathcal{N}(I)) = \sum_{i=1}^{n} p_i \|(I_d - P_{E_k})(M_i - G)\|^2 = p_i \|(I_d - P_{E_k})(y_i)\|^2$$

L'inertie totale du nuage est :

$$I_T := I_G(\mathcal{N}(I)) = \sum_{i=1}^{n} p_i \|M_i - G\|^2 = \sum_{i=1}^{n} p_i \|y_i\|^2$$

Proposition : Décomposition de l'inertie

Soit E un espace vectoriel associé à un sous-espace affine $\mathcal{E}$ passant par G. L'inertie totale du nuage se décompose comme suit :

$$I_T = I_E(\mathcal{N}(I)) + I_{E^\perp}(\mathcal{N}(I)),$$

où :

$$I_E(\mathcal{N}(I)) = \sum_{i=1}^{n} p_i \|(I_d - P_E)(y_i)\|^2,$$

$$I_{E^\perp}(\mathcal{N}(I)) = \sum_{i=1}^{n} p_i \|P_E(y_i)\|^2.$$

Preuve :
Par définition de la projection orthogonale, on peut écrire la décomposition suivante pour chaque point M_i :

$$y_i = P_E(y_i) + (Id - P_E)(y_i).$$

En prenant la norme au carré et en sommant sur tous les points pondérés :

$$\sum_{i=1}^{n} p_i \|y_i\|^2 = \sum_{i=1}^{n} p_i \left(\|P_E(y_i)\|^2 + \|(Id - P_E)(y_i)\|^2 \right).$$

D'où :

$$I_T = I_{E^\perp}(\mathcal{M}) + I_E(\mathcal{M}).$$

□

2.3.2 Interprétation : lien entre minimisation et maximisation

Comme l'inertie totale du nuage est constante, minimiser l'inertie $I_E(\mathcal{M})$ revient donc à maximiser l'inertie $I_{E^\perp}(\mathcal{M})$, ce qui est exactement le principe de l'Analyse en Composantes Principales (ACP). On maximise donc la part de l'inertie expliquée par la projection sur E_k, qui est donnée par les k directions principales de la matrice de covariance. Nous devons donc résoudre le problème d'optimisation suivant :

$$\arg\max_{E_k} I_{E_k^\perp}\left(\mathcal{N}(I)\right)$$

où E_k est un sous-espace vectoriel de dimension k.

2.4 Résolution du problème

Nous introduisons désormais la matrice V de **variance** qui est au centrale dans notre étude.

Définition : Matrice de variance

Le tableau Y étant centré, la matrice de variance V est définie par :

$$\forall j, j' \in [\![1, p]\!], \quad V_{j,j'} = \sum_{i=1}^{n} p_i \, y_i^j \, y_i^{j'},$$

Sous forme matricielle, cela s'exprime comme :

$$V = Y' D_p Y,$$

On introduit le résultat suivant :

Proposition (Courant-Fischer)

Soit $A \in S_d(\mathbb{R})$ une matrice symétrique définie sur $\mathbb{R}^d$, et soient $\lambda_1(M) \leq \lambda_2(M) \leq \cdots \leq \lambda_d(M)$ ses valeurs propres. Alors, on a :

$$\lambda_1(M) = \inf_{x \in \mathbb{S}^d} \langle Mx, x \rangle.$$

$$\lambda_d(M) = \sup_{x \in \mathbb{S}^d} \langle Mx, x \rangle.$$

Preuve : Soit $A \in S_d(\mathbb{R})$ une matrice symétrique définie sur $\mathbb{R}^d$. Nous allons démontrer que :

$$\lambda_d(A)\|x\|^2 \geq \langle Ax, x \rangle \geq \lambda_1(A)\|x\|^2, \quad \forall x \in \mathbb{R}^d.$$

L'ensemble $\mathbb{S}^d = \{x \in \mathbb{R}^d \mid \|x\| = 1\}$ est un fermé borné dans $\mathbb{R}^d$. En dimension finie, tout ensemble fermé et borné est compact. Ainsi, $\mathbb{S}^p$ est un ensemble compact.
Soit $\{u_i\}_{i=1,\ldots,d}$ une base orthonormale de vecteurs propres associés à A, tels que :

$$Au_i = \lambda_i(A)u_i, \quad \forall i.$$

Tout vecteur $x \in \mathbb{R}^d$ peut s'écrire dans cette base :

$$x = \sum_{i=1}^{d} \langle x, u_i \rangle u_i.$$

Appliquons A à cette expression :

$$Ax = \sum_{i=1}^{d} \lambda_i(A)\langle x, u_i \rangle u_i.$$

Prenons maintenant le produit scalaire avec x :

$$\langle Ax, x \rangle = \left\langle \sum_{i=1}^{d} \lambda_i(A)\langle x, u_i \rangle u_i, x \right\rangle.$$

Par linéarité et orthogonalité des u_i, cela donne :

$$\langle Ax, x \rangle = \sum_{i=1}^{d} \lambda_i(A)\langle x, u_i \rangle^2.$$

Puisque $\lambda_1(A) \leq \lambda_2(A) \leq \cdots \leq \lambda_d(A)$, on a :

$$\lambda_1(A) \sum_{i=1}^{d} \langle x, u_i \rangle^2 \leq \sum_{i=1}^{d} \lambda_i(A)\langle x, u_i \rangle^2 \leq \lambda_d(A) \sum_{i=1}^{d} \langle x, u_i \rangle^2.$$

- Comme $\sum_{i=1}^{d} \langle x, u_i \rangle^2 = \|x\|^2$, on obtient :

$$\lambda_1(A)\|x\|^2 \leq \langle Ax, x \rangle \leq \lambda_d(A)\|x\|^2.$$

□

Résultat : Sous-espace optimal

Le sous-espace affine de dimension k qui minimise l'inertie totale est donné par :

$$\mathcal{E}_k = G + \text{Vect}\{u_1, \dots, u_k\},$$

où $\{u_1, \dots, u_k\}$ sont les vecteurs propres associés aux k plus grandes valeurs propres de la matrice de covariance V.

Preuve :

On a par définition de l'inertie que :

$$I_{E_k^\perp}(\mathcal{N}(I)) = \sum_{i=1}^{n} p_i \|P_{E_k}(y_i)\|^2$$

Or, si on considère $(e_1, \dots, e_k)$ une base orthonormée de E_k, on a que :

$$\forall i \in \{1, \dots, n\}, \quad P_{E_k}(y_i) = \sum_{j=1}^{k} \langle y_i, e_j \rangle e_j$$

Ainsi,

$$I_{E_k^\perp}(\mathcal{N}(I)) = \sum_{i=1}^{n} p_i \left\| \sum_{j=1}^{k} \langle y_i, e_j \rangle e_j \right\|^2$$

Par orthonormalité :

$$= \sum_{i=1}^{n} p_i \sum_{j=1}^{k} \|\langle y_i, e_j \rangle e_j\|^2$$

$$= \sum_{j=1}^{k} \sum_{i=1}^{n} p_i \langle y_i, e_j \rangle^2$$

$$= \sum_{j=1}^{k} \langle V e_j, e_j \rangle$$

Or, par le principe de Courant-Fischer, cette quantité est maximale lorsque $(e_1, \dots, e_k)$ sont les vecteurs propres associés au k plus grandes valeurs propres de V.
Ainsi, si on note ces valeurs propres $\lambda_1 \geq \cdots \geq \lambda_k$, on a :

$$I_{E_k^\perp}(\mathcal{N}(I)) = \sum_{j=1}^{k} \lambda_j$$

□

Afin de disposer d'un critère quantitatif de détection des anomalies, nous introduisons la notion d'erreur de reconstruction, qui jouera un rôle central dans notre analyse expérimentale en prenant la forme d'un score de nouveauté.

Définition : Erreur de reconstruction en ACP

L'erreur de reconstruction d'un point $\mathbf{y}_i \in \mathbb{R}^p$ est définie par :

$$\mathcal{E}(\mathbf{y}_i, k) = \left\| y_i - \sum_{j=1}^{k} \langle y_i, e_j \rangle e_j \right\|^2, \tag{1}$$

où e_j est le j-ième vecteur propre associé à la j-ième plus grande valeur propre de la matrice de covariance V.
L'erreur totale de reconstruction sur l'ensemble des données est donnée par :

$$\mathcal{E}_{\text{totale}}(k) = \sum_{i=1}^{n} p_i \mathcal{E}(y_i, k) = \sum_{j=k+1}^{p} \lambda_j, \tag{2}$$

Cette erreur mesure la perte d'information induite par la réduction de dimension et correspond à la somme des variances non conservées par la projection.

3 Analyse en Composantes Principales à noyaux (ACP à noyaux)

3.1 Introduction et définitions

Pour poser les bases de cette section, nous commençons par définir la notion de noyau ainsi que celle de noyau défini positif, éléments essentiels à la compréhension de la suite.

Définition : Noyau

Un *noyau* est une application $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, avec $\mathcal{X}$ un ensemble quelconque. Celui-ci est *symétrique* si :

$$\forall x, y \in \mathcal{X},\ k(x, y) = k(y, x)$$

Définition : Noyau défini positif

Une fonction $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ est un noyau *défini positif* si et seulement si toutes les matrices noyaux résultant de ces fonctions noyaux sont *symétriques semi-définies positives*. i.e,

$$\forall n \in \mathbb{N}^*, \forall (x_1, .., x_n) \in \mathcal{X}^n, \forall \lambda \in \mathbb{R}^n,$$
$$\lambda^\intercal K \lambda \geq 0$$

où $K = (k(x_i, x_j))_{(i,j)\in[|1,n|]^2}$

$$et\ \lambda^\intercal K \lambda = 0\ ssi\ \lambda = \mathbf{0}$$

3.2 Théorème d'Aronszajn (1950)

Le théorème suivant assure la correspondance entre tout noyau défini positif et un espace de Hilbert reproduisant, formant ainsi la base théorique des méthodes à noyaux.

Théorème d'Aronszajn (1950)

Une fonction $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ est un noyau défini positif *si et seulement si*, il existe un espace de Hilbert $\mathcal{H}$ et une fonction $\varphi : \mathcal{X} \to \mathcal{H}$ telles que, pour tout $x, x' \in \mathcal{X}$,

$$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}.$$

Remarque :

1. Ici, K correspond à la matrice de Gram associée à l'échantillon $(x_1, .., x_n)$.
2. L'espace de Hilbert $\mathcal{H}$ est appelé *espace de redescription*, et cet espace est souvent de dimension plus grande que $\mathcal{X}$.

3.3 Principe général de l'ACP à noyaux

Soit $\phi : \mathcal{X} \to \mathcal{H}$ une application non linéaire, où $\mathcal{H}$ est un espace caractéristique (ou espace de représentation) et $x = (x_1, \ldots, x_n)$ un échantillon. Nous introduisons ici le principe de l'ACP dans l'espace caractéristique $\mathcal{H}$.

3.3.1 Cas 1

Supposons que la matrice $M = \mathcal{I}_n \in \mathcal{M}_{n,n}(\mathbb{R})$ et que l'échantillon x satisfait les conditions suivantes :

1. x est centré, c'est-à-dire $\sum_{i=1}^{n} x_i = 0$.
2. La matrice de poids est donnée par $\mathcal{D}_p = \frac{1}{n} I_n$.
3. Les données $\Phi(x)$ sont également centrées.

La matrice de variance-covariance dans $\mathcal{H}$ est alors définie par :

$$V = \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i)\Phi(x_i)'.$$

Soit $u \in \mathcal{H} \setminus \{0_{\mathcal{H}}\}$ un vecteur propre de V associé à la valeur propre $\lambda \in \mathbb{R}^*$. L'équation caractéristique s'écrit :

$$Vu = \lambda u. \tag{3}$$

Notre objectif est de résoudre l'équation (3) sans expliciter la matrice V. Puisque u est solution de (3), il appartient au sous-espace engendré par les images des données :

$$u \in \text{Vect}(\Phi(x_1), \ldots, \Phi(x_n)).$$

Il existe donc des coefficients $\mu_1, \ldots, \mu_n \in \mathbb{R}$ tels que :

$$u = \sum_{j=1}^{n} \mu_j \Phi(x_j). \tag{4}$$

L'équation (3) est alors équivalente à :

$$\langle \Phi(x_k) \cdot Vu \rangle = \lambda \langle \Phi(x_k) \cdot u \rangle, \quad \forall k = 1, \ldots, n. \tag{5}$$

En remplaçant u par son expression (4) dans (5), nous obtenons :

$$\frac{1}{n} \sum_{j=1}^{n} \mu_j \sum_{i=1}^{n} \langle \Phi(x_k) \cdot \Phi(x_i) \rangle \langle \Phi(x_i) \cdot \Phi(x_j) \rangle = \lambda \sum_{j=1}^{n} \mu_j \langle \Phi(x_k) \cdot \Phi(x_j) \rangle, \quad \forall k = 1, \ldots, n. \tag{6}$$

Nous introduisons la matrice de Gram K de format $n \times n$ et le vecteur $\mu \in \mathbb{R}^n$ définis par :

$$K_{ij} := \langle \Phi(x_i) \cdot \Phi(x_j) \rangle, \quad \text{et} \quad \mu = (\mu_1, \ldots, \mu_n)'. \tag{7}$$

L'écriture matricielle de l'équation (6) devient alors :

$$K\mu = n\lambda\mu. \tag{8}$$

Ainsi, pour réaliser l'ACP dans l'espace caractéristique $\mathcal{H}$, qui peut être de dimension arbitrairement grande (voire infinie), nous devons diagonaliser la matrice de Gram K, qui est semi-définie positive et possède donc des valeurs propres positives. Soient $\mu^1, \ldots, \mu^n$ les vecteurs propres de K associés respectivement aux valeurs propres ordonnées $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$.
Puisque u est unitaire, nous avons la relation :

$$\begin{aligned} 1 &= u'u \\ &= \sum_{i,j=1}^{n} \mu_i \mu_j \langle \Phi(x_i) \cdot \Phi(x_j) \rangle \\ &= \sum_{i,j=1}^{n} \mu_i^k \mu_j^k K_{ij} \\ &= \langle \mu^k \cdot K\mu^k \rangle \\ &= \langle \mu^k \cdot n\lambda_k \mu^k \rangle \\ &= n\lambda_k \|\mu^k\|^2. \end{aligned} \tag{9}$$

Nous obtenons alors la normalisation suivante :

$$\|\mu^k\| = \frac{1}{\sqrt{n\lambda_k}}.$$

3.3.2 Cas 2

Dans ce second cas, nous supposons que l'échantillon x satisfait les conditions 1 et 2 définies dans le **Cas 1**. L'objectif de cette partie de notre étude est de présenter la forme de la matrice de Gram K obtenue après qu'on ait centré les données dans l'espace caractéristique $\mathcal{H}$. C'est ainsi que l'observation centrée réalisée dans l'espace caractéristique $\mathcal{H}$ est donnée par :

$$\tilde{\Phi}(x_i) := \Phi(x_i) - \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i) \quad \forall i = 1, \ldots, n \tag{10}$$

De ce fait, l'ensemble de notation suivant : $\tilde{V}$,$\tilde{u}$,$\tilde{\lambda}$ correspond respectivement à la matrice de covariance-variance des observations centrées dans $\mathcal{H}$, au vecteur propre de $\tilde{V}$ et la valeur propre associée à $\tilde{u}$.Par conséquent, $\tilde{V}$,$\tilde{u}$ et $\tilde{\lambda}$ satisfont les équations (3),(4) et (5) pour les observations $\tilde{\Phi}(x)$.
Notons par $\tilde{\mu}$ le vecteur de $\mathbb{R}^n$ tel que : $\tilde{\mu} = (\tilde{\mu_1}, \ldots, \tilde{\mu_n})'$ et par $\tilde{K}$ la matrice de Gram définie par :

$$\tilde{K_{ij}} := \langle \tilde{\Phi}(x_i).\tilde{\Phi}(x_j) \rangle \quad \forall i, j = 1, \ldots, n \tag{11}$$

Donc les conditions définies dans le **Cas 1** étant toutes vérifiées,on diagonalisera la matrice de Gram $\tilde{K}$ c'est-à-dire chercher $\tilde{\lambda} \in \mathbb{R}^n$ tel que :

$$\tilde{K}\tilde{\mu} = \tilde{\lambda}\tilde{\mu} \tag{12}$$

Comme nous n'avons pas les données centrées $\tilde{\Phi}(x)$, il est nécessaire de chercher une forme explicite de $\tilde{K}$ en fonction de K. On a : $\forall\, l, s, i, j$

$$\begin{aligned} \tilde{K_{ij}} &:= (\tilde{\Phi}(x_i).\tilde{\Phi}(x_j)) \\ &= (\Phi(x_i) - \frac{1}{n}\sum_{s=1}^{n} \Phi(x_s))'(\Phi(x_j) - \frac{1}{n}\sum_{l=1}^{n} \Phi(x_l)) \\ &= \Phi(x_i)'\Phi(x_j) - \frac{1}{n}\sum_{l=1}^{n} \Phi(x_i)'\Phi(x_l)) - \frac{1}{n}\sum_{s=1}^{n} \Phi(x_s)'\Phi(x_j) + \frac{1}{n^2}\sum_{s,l=1}^{n} \Phi(x_s)'\Phi(x_l) \\ \tilde{K_{ij}} &= K_{ij} - \frac{1}{n}\sum_{l=1}^{n} K_{il} - \frac{1}{n}\sum_{s=1}^{n} K_{sj} + \frac{1}{n^2}\sum_{s,l=1}^{n} K_{sl} \end{aligned} \tag{13}$$

Introduisons les matrices A et $H \in \mathcal{M}_{n\times n}(\mathbb{R})$ définies par :

$$\mathbf{A} = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \end{pmatrix}$$

et $H = \mathcal{I}_n - A$.
En somme, nous obtenons l'écriture matricielle de $\tilde{K}$ suivante : $\tilde{K} = K - AK - KA + AKA = HKH$ qui permet de résoudre le problème de valeur propre défini dans (12) et par ailleurs nous substituons $\tilde{\mu}$ par $\frac{\tilde{\mu}}{\sqrt{\tilde{\lambda}_k}}$.

3.3.3 Calcul des composantes principales

Soit u_α^k un vecteur propre de V associé à la valeur propre λ_α. Les composantes principales sont définies comme la projection des nouvelles données $\Phi(x)$ sur l'espace propre engendré par u_α^k :

$$\psi_{\alpha,k} = \langle \Phi(x)' \cdot u_\alpha^k \rangle = \sum_{i=1}^{n} \mu_i^k \langle \Phi(x) \cdot \Phi(x_i) \rangle. \tag{14}$$

Afin de simplifier l'équation (14), nous introduisons le principe de l'analyse en composantes principales à noyaux.

3.4 Algorithme de l'ACP à noyaux

Soient x, y deux échantillons de $\mathcal{X}$. Le principe de l'ACP à noyaux consiste à réaliser une ACP dans un espace de représentation $\mathcal{H}$, tout en remplaçant le calcul explicite du produit scalaire $(\Phi(x) \cdot \Phi(y))$ par une fonction dite noyau $k(x, y)$. L'existence de cette fonction est garantie par le théorème d'Aronszajn (3.2). Ainsi, l'équation (14) devient :

$$\begin{aligned} \psi_{\alpha,k} &= (\Phi(x)' \cdot u_\alpha^k) \\ &= \sum_{i=1}^{n} \mu_i^k (\Phi(x) \cdot \Phi(x_i)) \\ &= \sum_{i=1}^{n} \mu_i^k k(x, x_i). \end{aligned} \tag{15}$$

Nous obtenons donc la relation vectorielle :

$$\psi_{\alpha,k} = \mu' k(x), \tag{16}$$

où :

$$k(x) = (k(x, x_1), \ldots, k(x, x_n))' \quad \text{et} \quad \mu = (\mu_1, \ldots, \mu_n)'.$$

Finalement, nous obtenons ce pseudo-code pour cet algorithme.

Algorithm 1: ACP à noyaux

Entrée : Données $\{x_1, \ldots, x_n\}$, noyau k, nombre de composantes m
Sortie : Composantes principales $\psi_{\alpha,k}(x)$

for $i = 1$ **to** n **do**
  for $j = 1$ **to** n **do**
    $K[i,j] \leftarrow k(x_i, x_j)$;
$A \leftarrow$ matrice $n \times n$ avec toutes les entrées égales à $\frac{1}{n}$;
$H \leftarrow I_n - A$;
$K_{\text{centré}} \leftarrow HKH$;
Calculer les valeurs propres λ_k et vecteurs propres μ_k de $K_{\text{centré}}$ tels que;
  $K_{\text{centré}}\mu_k = \lambda_k \mu_k$;
Garder les m plus grandes valeurs propres non nulles et leurs vecteurs propres associés;
for $k = 1$ **to** m **do**
  $\mu_k \leftarrow \mu_k / \sqrt{\lambda_k}$;
for *chaque donnée* x_{new} **do**
  Construire $k_x \leftarrow [k(x_{\text{new}}, x_1), \ldots, k(x_{\text{new}}, x_n)]^T$;
  Centrer k_x : $k_{x,\text{centré}} \leftarrow H\left(k_x - \frac{1}{n} K \mathbf{1_n}\right)$;
  for $k = 1$ **to** m **do**
    $\psi_{\alpha,k}(x_{\text{new}}) \leftarrow \mu_k^T k_{x,\text{centré}}$;

3.5 Exemples de noyaux

Un noyau à valeurs réelles $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ est dit **défini positif** s'il existe un espace caractéristique $\mathcal{H}$ et une application $\Phi : \mathcal{X} \to \mathcal{H}$ telles que :

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{H}}.$$

Dans certains cas, cette application Φ peut être construite explicitement. Mais dans la plupart des situations pratiques, cela n'est pas nécessaire : on travaille uniquement avec le noyau, ce qui est l'un des grands avantages de la méthode du noyau.
Dans ce qui suit, nous présentons plusieurs familles classiques de noyaux, avec une attention particulière à leur utilité en pratique et à la nature des structures qu'ils permettent de modéliser.

Noyau linéaire : On suppose $\mathcal{X} = \mathbb{R}^d$. Le noyau linéaire est défini par :

$$k(x, y) = x^\top y.$$

Dans ce cas l'application Φ est $\Phi(x) = x$, et a un espace de Hilbert isomorphe à $\mathbb{R}^d$. C'est le noyau utilisé en Analyse en Composantes Principales (ACP) d'où son utilisation dès lors les données présentent une structure linéaire.

Noyau polynomial : Pour un degré $p \in \mathbb{N}$ et une constante $c \geq 0$, le noyau polynomial est donné par :

$$k(x, y) = (x^\top y + c)^p.$$

le noyau polynomial prend ses valeurs dans un espace de dimension : $\dim(\mathcal{H}) = \binom{p+d}{p}$.
En effet lorsque les données obéissent à une relation non linéaire (par exemple, quadratique ou cubique), ce noyau permet de capturer cette complexité sans avoir à expliciter les transformations. Cependant, il présente une sensibilité aux valeurs aberrantes.

Noyau gaussien : Aussi appelé noyau à base radiale (RBF), il est défini par :

$$k(x, y) = \exp\left(-\gamma \|x - y\|^2\right), \quad \text{où } \gamma > 0.$$

Ce noyau est défini positif et prend ses valeurs dans un espace caractéristique de dimension infinie.De ce fait, il est appelé "noyau universel" du fait qu'avec des paramètres bien choisis, il peut approximer n'importe quelle fonction continue(**théorème d'approximation universelle**). Bien qu'il soit moins sensible aux outliers, le noyau gaussien peut présenter des limites telles que la non-distribution normale des données.

Noyau cosinus : Le noyau du cosinus est une adaptation normalisée du produit scalaire standard. Il est fondé sur le concept de similarité. La similarité cosinus est une métrique largement appliquée dans le traitement automatique du langage naturel (NLP) et l'exploration de texte. Cela permet de comparer deux vecteurs non pas en termes de distance mais en termes de direction dans l'espace vectoriel.
La formule du noyau cosinus est la suivante :

$$k(x, x') = \frac{\langle x, x' \rangle}{\|x\| \cdot \|x'\|},$$

où $\langle x, x' \rangle$ désigne le produit scalaire habituel et $\|x\|$ la norme euclidienne de x.

Ce noyau calcule donc le cosinus de l'angle entre deux vecteurs. Plus cet angle est petit, plus les vecteurs se trouvent dans la même direction et sont donc dits fortement similaires. Un angle proche de 90° présente peu ou pas de similitude.
Le noyau cosinus présente certains avantages théoriques et pratiques :

- Il ne varie pas en fonction de l'échelle : deux vecteurs ayant la même direction présenteront une similitude de cosinus de 1, quelle que soit leur norme.
- Il peut être considéré comme positivement défini dans les cas pratiques, notamment sur les vecteurs normalisés, ce qui est fréquent en NLP. Cela permet son usage efficace dans des méthodes à noyau comme l'ACP à noyau ou le SVM.
- De cette manière, nous pouvons traiter la structure directionnelle des données plutôt que le niveau de mesure, ce qui est particulièrement utile dans les situations où la norme peut être influencée par des facteurs extrinsèques (longueur du document, fréquence absolue, etc.).

Noyaux invariants par translation : Un noyau k est dit **invariant par translation** s'il existe une fonction $h : \mathbb{R}^d \to \mathbb{R}$ telle que :

$$k(x, y) = h(x - y).$$

Ce type de noyau ne dépend que de la différence $x - y$, ce qui le rend particulièrement adapté à des données où seule la différence entre les objets est significative (comme en traitement du signal ou en séries temporelles).
Exemples :

- Noyau gaussien : $h(u) = \exp(-\gamma\|u\|^2)$
- Noyau exponentiel : $h(u) = \exp(-\|u\|)$
- Noyau de Laplace : $h(u) = \exp(-\gamma\|u\|_1)$

Tableau récapitulatif

Nom du noyau	Forme	Espace de Hilbert	Utilité typique
Linéaire	$x^\top y$	$\mathbb{R}^d$	Données linéairement séparables
Polynomial	$(x^\top y + c)^p$	$\binom{p+d}{p}$	Relations non linéaires simples
Gaussien (RBF)	$\exp(-\gamma\|x - y\|^2)$	Dimension infinie	Approximation universelle, données complexes
Stationnaire (ex : Gaussien, Laplacien)	$h(x - y)$	Variable selon h	Données structurées : images, signaux, séries temporelles
Cosinus	$\frac{\langle x, y \rangle}{\|x\| \cdot \|y\|}$	Sous-espace directionnel de $\mathbb{R}^d$	Textes, NLP, données creuses ou normalisées

3.6 Conclusion de la partie théorique

3.6.1 Limitations de l'ACP à noyaux classique

L'ACP à noyaux souffre de limitations majeures :

- **Complexité computationnelle** : L'ACP à noyaux repose sur la diagonalisation de la matrice de Gram K, dont la complexité est en $\mathcal{O}(n^3)$ pour le temps et en $\mathcal{O}(n^2)$ pour la mémoire.
- **Impossibilité de mise à jour incrémentale** : Elle est conçue comme une méthode par lot ("batch"), elle requiert donc l'ensemble des données pour recalculer complètement la décomposition spectrale à chaque ajout d'observation ce qui est très couteux.

Ces limitations ont motivé l'introduction de l'ACP à noyaux incrémental.
Cette variante offre une alternative efficace en permettant :

- de mettre à jour la représentation sans recalculer intégralement K,
- de réduire considérablement la complexité à $\mathcal{O}(n)$ par ajout d'un point,
- de gérer la mémoire grâce à des techniques d'approximation par ensemble réduit (reduced set expansion).

3.7 ACP à noyaux incrémental

3.7.1 Principe général de l'ACP à noyaux incrémental

L'ACP à noyaux incrémental vise à mettre à jour les composantes principales sans avoir à recalculer l'ensemble de la décomposition spectrale à chaque ajout de nouvelles données. L'idée principale est d'appliquer l'ACP linéaire incrémentale dans l'espace de Hilbert reproduisant, sans expliciter la projection $\phi(x)$.

3.7.2 Algorithme de mise à jour

Soit un ensemble de données $X = \{x_1, ..., x_n\}$ dont on a calculé l'ACP à noyaux. On souhaite maintenant mettre à jour cette analyse avec un nouvel ensemble $X' = \{x'_1, ..., x'_m\}$.

1. Mise à jour de la moyenne et centrage : La nouvelle moyenne dans l'espace de Hilbert est mise à jour comme suit :

$$\mu' = \frac{n}{n+m}\mu + \frac{m}{n+m}\frac{1}{m}\sum_{i=1}^{m}\phi(x'_i) \tag{17}$$

Puis, les nouvelles données sont centrées en soustrayant cette nouvelle moyenne.

2. Mise à jour de la matrice de covariance noyau : La matrice de covariance est actualisée à l'aide de la matrice de Gram combinée :

$$K' = \begin{bmatrix} K & K_{new} \\ K_{new}^T & K'' \end{bmatrix} \tag{18}$$

où K_{new} représente les noyaux entre les anciennes et nouvelles données.

3. Construction d'une expansion réduite (Reduced-set expansion) : Afin d'éviter de stocker toutes les données précédentes, les composantes principales sont exprimées en fonction d'un sous-ensemble réduit :

$$\tilde{\phi}(x) = \sum_{i=1}^{r}\alpha_i\phi(x_i), \quad \text{avec } r \ll n \tag{19}$$

Cette approximation limite la mémoire et accélère les calculs.

4. Orthogonalisation et normalisation des vecteurs propres : Une fois les nouvelles composantes calculées, elles sont réorthogonalisées via une nouvelle décomposition en valeurs singulières pour garantir leur validité.

3.7.3 Comparaison avec l'ACP à noyaux classique

Critère	**ACP à noyaux standard**	**ACP à noyaux incrémental**
Complexité temporelle	$\mathcal{O}(n^3)$	$\mathcal{O}(n)$ par mise à jour
Complexité mémoire	$\mathcal{O}(n^2)$	$\mathcal{O}(r^2)$ avec $r \ll n$
Mise à jour	Recalcule toute la base	Mise à jour locale rapide
Application en ligne	Non	Oui

TABLE 1 – Comparaison entre ACP à noyaux standard et incrémental

L'ACP à noyaux incrémental est donc une méthode plus efficace pour traiter les données massives et les applications en ligne.

Partie expérimentale

Après avoir posé les fondements mathématiques de l'analyse en composantes principales à noyaux, nous sommes désormais en mesure d'évaluer empiriquement l'intérêt de cette méthode dans des contextes concrets.

La puissance de l'ACP à noyaux réside dans sa capacité à capter des structures complexes et non linéaires, inaccessibles à l'ACP classique. Pour valider cette intuition théorique, nous proposons une série d'expérimentations sur des jeux de données variés (texte, image, signal) aux structures intrinsèquement différentes.

L'objectif est double :

- D'une part, vérifier que le ACP à noyaux permet effectivement de mieux représenter les données que son équivalent linéaire, en améliorant les performances de modèles classiques de classification ou de reconstruction ;
- D'autre part, illustrer comment le choix du noyau, du nombre de composantes ou des paramètres influence les résultats obtenus.

Ces études de cas nous permettront de confronter la théorie à la pratique, en analysant la pertinence de l'ACP à noyaux pour la classification de sentiments dans des critiques textuelles, la détection d'anomalies sur des chiffres manuscrits (MNIST), ainsi que le débruitage de signaux ECG bruités.
Nous détaillons ci-après le protocole expérimental, les méthodes utilisées, ainsi que les résultats observés dans chacun de ces contextes.

4 Évaluation de l'impact de l'ACP à noyaux sur la classification de sentiments

4.1 Objectifs de l'expérimentation

Dans cette première étude expérimentale, l'efficacité de l'Analyse en Composantes Principales (ACP) à noyaux est évaluée dans le cadre d'une tâche emblématique du traitement automatique du langage naturel (NLP) : la classification de sentiments dans des critiques textuelles. L'objectif est d'analyser si une réduction de dimension non linéaire, réalisée par l'ACP à noyaux, peut améliorer les performances de modèles de classification supervisée.

L'ensemble de données considéré est le corpus IMDb introduit par Maas *et al.* [2], composé de 50 000 critiques de films annotées manuellement selon deux classes (*positive* ou *négative*). Afin de garantir une mise en œuvre expérimentale maîtrisée et reproductible, un sous-ensemble de 5 000 critiques est extrait, réparti équitablement entre les deux classes. La tâche à résoudre correspond ainsi à une classification binaire, visant à prédire automatiquement la polarité du sentiment exprimé par chaque critique.

L'expérimentation consiste ainsi à comparer systématiquement trois approches :

- une classification sans réduction de dimension ;
- une classification après réduction par ACP classique ;
- une classification après réduction par ACP à noyaux avec noyau cosinus.

Pour chacune de ces stratégies, trois modèles standards sont évalués : le k-plus proches voisins (k-NN), la régression logistique, et le SVM à noyau RBF. Les performances sont mesurées à l'aide de métriques adaptées telles que le F1-score et la courbe ROC. L'ensemble du pipeline expérimental est implémenté en Python, et suit une méthodologie rigoureuse incluant : prétraitement linguistique du texte, vectorisation via le modèle Bag-of-Words (BoW), normalisation, réduction de dimension, entraînement des modèles, validation croisée, et analyse comparative.

4.2 Prétraitement du texte

4.2.1 Objectif du prétraitement

Avant toute phase de modélisation ou de réduction de dimension, une étape de prétraitement est indispensable afin de nettoyer les textes et de les rendre compatibles avec les algorithmes d'apprentissage automatique. Les critiques utilisées dans cette étude, issues du jeu de données IMDb, présentent une grande variabilité lexicale, syntaxique et typographique. Elles sont également affectées par différentes sources de bruit susceptibles de dégrader la qualité de la vectorisation et, par conséquent, les performances des modèles supervisés [6].

Les principales nuisances observées dans les textes sont les suivantes :

- la présence de **balises HTML**, telles que `<br/>`, introduites par le format d'origine ;
- une **ponctuation excessive** et des **caractères spéciaux** peu informatifs ;
- une utilisation hétérogène des **majuscules et minuscules**, générant des doublons inutiles ;
- des **chiffres**, rarement pertinents dans l'analyse de sentiments ;
- des **contractions grammaticales** (ex. : *didn't*, *I've*) nécessitant d'être développées pour une meilleure cohérence lexicale ;
- la présence de **stopwords** [1], souvent peu discriminants pour la tâche de classification , souvent peu discriminants pour la tâche de classification ;
- l'**absence d'information grammaticale explicite**, compliquant la désambiguïsation lexicale [2].

Afin de corriger ces effets, un pipeline de prétraitement structuré en plusieurs étapes a été mis en œuvre :

1. **Développement des contractions** : transformation des formes contractées en leur forme explicite (*didn't* $\rightarrow$ *did not*) ;
2. **Nettoyage brut** : suppression des balises HTML, de la ponctuation, des chiffres et conversion en minuscules ;
3. **Tokenisation** : découpage des textes en séquences de mots ;
4. **Étiquetage grammatical (POS-tagging)** : attribution d'une catégorie grammaticale à chaque mot, réalisée avec la bibliothèque NLTK [15] ;
5. **Lemmatisation** : réduction des mots à leur forme canonique en fonction de leur rôle syntaxique, en utilisant `WordNetLemmatizer` [15] ;
6. **Filtrage des stopwords** : suppression des mots peu informatifs, en conservant cependant les marqueurs de négation et d'intensité (ex. : *not*, *never*, *very*), essentiels pour la détection fine du sentiment exprimé. Ce choix est appuyé par des travaux récents montrant que l'élimination systématique des stopwords peut nuire aux performances en analyse de sentiments [4].

L'ordre de ces étapes est déterminé de manière à assurer une cohérence linguistique optimale : la lemmatisation est réalisée après le POS-tagging pour garantir un appariement syntaxique correct, et le filtrage des stopwords intervient en toute fin pour préserver l'information contextuelle durant les étapes précédentes.

1. Les *stopwords* désignent des mots très fréquents, souvent considérés comme peu informatifs pour les tâches de classification textuelle (exemples : *the*, *and*, *but*).
2. La *désambiguïsation lexicale* consiste à déterminer le sens précis d'un mot en fonction de son rôle syntaxique ou de son contexte dans la phrase.

4.2.2 Illustration sur un exemple

L'effet du pipeline de prétraitement est illustré sur l'exemple suivant, extrait du corpus initial :

> *"I REALLY didn't like this movie!! It was
 TOO long, with absolutely no plot. Just boring dialogues, and the acting? Terrible. I've seen better in student films..."*

Ce commentaire présente diverses formes de bruit : balise HTML, ponctuation excessive, majuscules, mots de négation (*didn't, no*), intensificateurs (*really, too, absolutely*), formes conjuguées (*was, seen*), ainsi que de nombreux stopwords.
Après application du prétraitement, le texte est transformé en :

> *"really not like movie too long absolutely no plot just boring dialogue acting terrible seen better student film"*

Le résultat final est un texte réduit, lexicalement normalisé et focalisé sur les éléments sémantiques essentiels. Les marqueurs de négation, d'intensité et les termes porteurs de sens sont conservés, ce qui est indispensable pour préserver correctement la polarité exprimée dans l'analyse de sentiments.

4.2.3 Analyse du corpus après prétraitement

L'application du pipeline conduit à une réduction drastique de la redondance lexicale et à une structuration plus homogène du corpus, avec plusieurs effets mesurables sur les données :

Statistiques globales avant et après prétraitement :

- **Nombre total de mots avant prétraitement** : 1 156 177
- **Nombre total de mots après prétraitement** : 612 404
- **Nombre de mots uniques après prétraitement** : 47 166
- **Réduction globale du corpus** : 47.03 %

Ces chiffres mettent en évidence l'efficacité du pipeline : près de la moitié des occurrences sont éliminées, traduisant l'élimination des éléments bruités et la consolidation des formes lexicales.

Analyse du vocabulaire et couverture des occurrences [6] : La figure suivante représente la courbe de couverture cumulative des mots du corpus. Chaque point de cette courbe indique quelle proportion du total des mots utilisés est capturée en conservant les i mots les plus fréquents. Autrement dit, la courbe mesure l'accumulation progressive des occurrences lorsque l'on parcourt le vocabulaire trié par fréquence décroissante. On observe ainsi combien de mots sont nécessaires pour couvrir une certaine fraction du corpus.
Formellement, si $f_1, f_2, \ldots, f_k$ désignent les fréquences décroissantes des k mots les plus fréquents, la couverture cumulative au rang i est définie par :

$$C(i) = \frac{\sum_{j=1}^{i} f_j}{\sum_{j=1}^{k} f_j},$$

où $C(i) \in [0, 1]$ indique la proportion cumulée des occurrences expliquée par les i mots les plus fréquents.

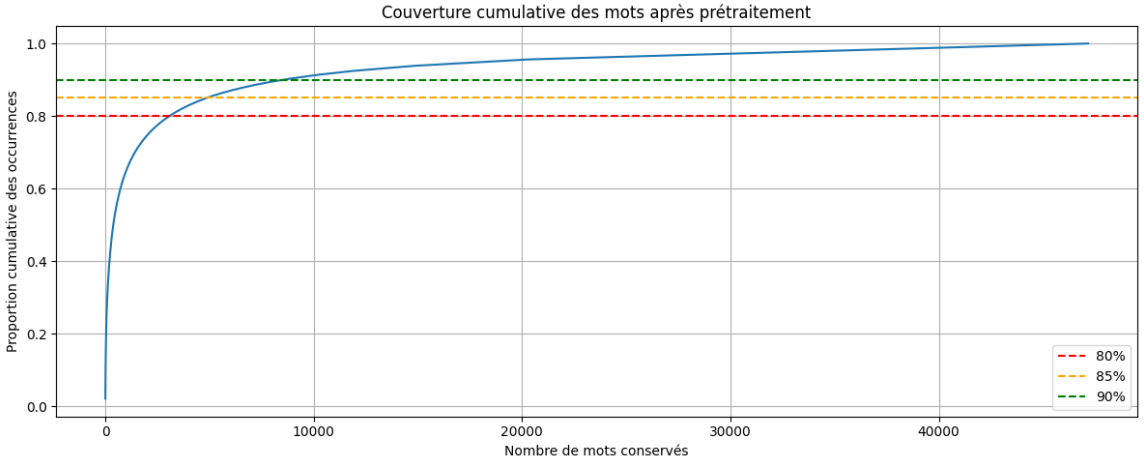

FIGURE 3 – Couverture cumulative des mots après prétraitement

Cette analyse montre que la majorité des occurrences est concentrée sur un nombre limité de mots :

- 3 074 mots suffisent à couvrir 80 % des occurrences ;
- 4 865 mots couvrent 85 % ;
- 8 518 mots couvrent 90 %.

Cette distribution suit la loi de Zipf [3], selon laquelle la fréquence d'apparition d'un mot est inversement proportionnelle à son rang. Ce phénomène explique pourquoi, comme observé ici, une fraction réduite du vocabulaire suffit à couvrir la majorité des occurrences du corpus.

4.3 Vectorisation des textes

4.3.1 Objectif de la vectorisation

Afin de rendre les critiques textuelles exploitables par des algorithmes d'apprentissage supervisé, il est nécessaire de les transformer en représentations numériques adaptées. La vectorisation a pour objectif d'associer à chaque document une représentation mathématique sous forme de vecteur, qui capture l'information lexicale tout en étant compatible avec les algorithmes statistiques utilisés dans la suite du traitement. Dans cette étude, le modèle du sac de mots (*Bag-of-Words*, BoW) est retenu pour effectuer cette transformation, en raison de sa simplicité, de son efficacité et de sa large adoption dans les tâches de classification textuelle [6].

4.3.2 Explication de la méthode et implémentation

Chaque critique est représentée par un vecteur $x_i \in \mathbb{R}^k$, où k est la taille du vocabulaire retenu. Afin de limiter la dimensionnalité, de réduire le temps de calcul et d'éviter d'apprendre sur un espace trop large et creux, le paramètre `max_features` du `CountVectorizer` est fixé à 3 074. Ce choix est motivé par l'analyse de couverture réalisée précédemment, qui montre qu'environ 80 % des occurrences du corpus sont couvertes par les 3 074 mots les plus fréquents.

La vectorisation est réalisée à l'aide de la classe `CountVectorizer` de `scikit-learn` [16]. Le vocabulaire est appris exclusivement sur l'ensemble d'entraînement, conformément aux bonnes pratiques en apprentissage supervisé, afin de garantir l'indépendance entre apprentissage et test et d'éviter tout risque de *data leakage*. La transformation aboutit à deux matrices :

- $X_{\text{train}} \in \mathbb{R}^{3500\times3074}$ pour l'ensemble d'entraînement ;
- $X_{\text{test}} \in \mathbb{R}^{1500\times3074}$ pour l'ensemble de test.

Ces matrices présentent une structure typique des représentations BoW : elles sont de très grande dimension mais fortement creuses, chaque document n'activant qu'une fraction réduite des termes possibles. Cette

forte sparsité complique l'apprentissage supervisé, en amplifiant la malédiction de la dimensionnalité [3] et en rendant la mesure de distance euclidienne peu fiable.

Dans ce contexte, l'application d'une méthode de réduction de dimension s'avère nécessaire pour rendre l'apprentissage plus efficace. En particulier, le recours à une approche non linéaire, comme l'Analyse en Composantes Principales à noyaux, apparaît particulièrement pertinent : la structure des vecteurs BoW, dominée par la présence/absence de mots et par des relations angulaires, suggère que les similarités entre documents ne se réduisent pas à des relations purement linéaires. L'ACP à noyaux permet ainsi de mieux capturer ces structures latentes, en exploitant des mesures de similarité adaptées, et d'extraire une représentation plus compacte et mieux alignée avec la nature intrinsèque des données.

4.3.3 Exemple illustratif

Pour illustrer le fonctionnement du modèle BoW, on peut reprendre l'exemple que nous avons utilisé pour le prétraitement :

> *"really not like movie too long absolutely no plot just bore dialogue act terrible see well student film"*

Un vocabulaire restreint de 16 mots est considéré à titre d'exemple :

```
["excellent", "bad", "movie", "plot", "boring", "great", "terrible", "dialogue",
     "predictable", "not", "like", "fun", "slow", "awful", "love", "just"]
```

La critique est alors représentée par un vecteur de dimension 16, où chaque coordonnée correspond au nombre d'occurrences du mot associé dans le document :

```
[0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1]
```

Le tableau suivant récapitule cette correspondance :

Mot	excellent	bad	movie	plot	...	slow	awful	love	just
Occur.	0	0	1	1	...	0	0	0	1

Malgré sa simplicité, la représentation BoW présente une limite importante : en très haute dimension, la sparsité des vecteurs dégrade la pertinence des mesures de distance, ralentit l'apprentissage supervisé, et augmente le risque de surapprentissage [14]. Ce constat motive l'étude de techniques de réduction de dimension adaptées, présentées dans la suite de ce travail.

4.4 Réduction de dimension et modélisation supervisée

4.4.1 Objectif de la réduction de dimension

Pour d'atténuer les difficultés mentionnées précédemment, une réduction de dimension est appliquée avant la modélisation, avec l'objectif de condenser l'information discriminante dans un espace plus compact et mieux structuré.

Afin de mesurer l'impact de cette réduction, trois stratégies sont ainsi comparées :

- apprentissage sur vecteurs BoW sans réduction ;
- réduction linéaire par Analyse en Composantes Principales (ACP) classique ;
- réduction non linéaire par Analyse en Composantes Principales à noyaux (ACP à noyaux) avec noyau cosinus.

Le but est ainsi de projeter les données dans un espace plus compact, où l'information discriminante est concentrée, afin de faciliter l'apprentissage supervisé tout en réduisant les coûts computationnels.

3. La *malédiction de la dimensionnalité* désigne les difficultés spécifiques rencontrées en haute dimension, telles que la dispersion des données et la perte de pertinence des mesures de distance, comme expliqué dans [14]. Ces effets compliquent l'apprentissage supervisé en rendant la notion de proximité moins fiable.

4.4.2 Justification du choix du noyau cosinus dans l'ACP à noyaux

Le noyau cosinus est retenu pour plusieurs raisons spécifiques aux propriétés des données considérées :

- Les représentations Bag-of-Words (BoW) produisent des vecteurs de grande dimension, creux, et dont l'information essentielle est portée par la direction du vecteur plutôt que par sa norme.
- Le noyau cosinus mesure la similarité angulaire entre documents, indépendamment de leur longueur absolue, ce qui est particulièrement adapté pour capturer les proximités de sens dans des vecteurs textuels normalisés [9].
- Contrairement à des noyaux paramétriques comme le noyau gaussien (RBF) ou polynomial, le noyau cosinus ne nécessite aucun réglage d'hyperparamètre supplémentaire, garantissant une implémentation simple et robuste sans risque d'ajustement excessif.

En résumé, le noyau cosinus permet d'exploiter efficacement la structure directionnelle intrinsèque des représentations BoW normalisées, tout en simplifiant le processus expérimental, ce qui en fait un choix naturel pour l'ACP à noyaux appliquée à la classification textuelle.

4.4.3 Normalisation préalable des vecteurs

Afin d'utiliser efficacement le noyau cosinus dans le cadre de l'ACP à noyaux, il est nécessaire que la similarité mesurée entre deux vecteurs soit uniquement dépendante de leur orientation, et non de leur norme. Pour cela, une normalisation ℓ^2 est appliquée à chaque vecteur x_i, selon :

$$\forall i \in \{1, \ldots, n\}, \quad x_i \leftarrow \frac{x_i}{\|x_i\|_2}$$

La similarité cosinus entre deux vecteurs x et y est alors donnée par :

$$\cos(\theta) = \frac{\langle x, y \rangle}{\|x\|_2 \|y\|_2}$$

et se simplifie, pour des vecteurs normalisés, en un simple produit scalaire :

$$\cos(\theta) = \langle x, y \rangle$$

Ainsi, après normalisation, les vecteurs sont projetés sur la sphère unité $\mathbb{S}^{k-1}$, et le noyau cosinus revient à calculer un produit scalaire standard entre ces vecteurs normalisés. La projection par ACP à noyaux devient ainsi directement sensible à l'angle entre documents.

Bien que ce processus rapproche formellement le noyau cosinus d'un noyau linéaire, il est essentiel de souligner que l'ACP à noyaux et l'ACP classique demeurent fondamentalement différentes :

- **ACP classique** : maximise la variance euclidienne globale dans l'espace initial $\mathbb{R}^k$ en cherchant les directions principales de dispersion [18].
- **ACP à noyaux** : maximise la variance dans un espace de Hilbert reproduit $\mathcal{H}$, induit par la fonction noyau choisie [17], sans construire explicitement cet espace.

Formellement, soit K la matrice noyau définie par $K_{ij} = k(x_i, x_j)$, où k est ici le noyau cosinus. L'ACP à noyaux procède alors par décomposition spectrale de K, tandis que l'ACP classique procède par décomposition de la matrice de covariance $X^T X$. Ce changement de géométrie implique que, même en cas de normalisation préalable, les sous-espaces extraits par les deux méthodes diffèrent fondamentalement, notamment en raison de la nature non linéaire de la projection implicite réalisée par le noyau.

4.4.4 Présentation des modèles supervisés :

Pour évaluer l'impact de la réduction de dimension, trois modèles classiques d'apprentissage supervisé ont été retenus :

- **KNN** (k-plus proches voisins) : algorithme local, très sensible à la dimensionnalité ;
- **Régression logistique** : modèle linéaire discriminatif rapide et robuste ;

- **SVM à noyau RBF** : modèle non linéaire puissant, capable de modéliser des frontières complexes.

Chaque modèle est évalué selon trois configurations :

- Sans réduction de dimension (BoW brut) ;
- Après réduction par ACP classique ;
- Après réduction par ACP à noyaux (noyau cosinus).

4.4.5 Optimisation et protocole expérimental :

Le choix du nombre optimal de composantes principales est déterminé empiriquement via une validation croisée. Pour chaque modèle supervisé (KNN, régression logistique, SVM), un `GridSearchCV` est utilisé afin d'optimiser simultanément :

- le nombre de composantes principales ;
- les hyperparamètres spécifiques à chaque modèle (par exemple : k pour KNN, C et γ pour SVM).

Le critère d'évaluation est le F1-score moyen mesuré sur 5 plis de validation croisée, garantissant une sélection robuste et adaptée à la tâche binaire considérée.

Définition : F1-score (moyenne harmonique entre précision et rappel [14])

Le F1-score est défini par :

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

où :

$$\text{Precision} = \frac{TP}{TP + FP} \quad \text{et} \quad \text{Recall} = \frac{TP}{TP + FN}$$

avec : TP : vrais positifs, TN : vrais négatifs, FP : faux positifs, FN : faux négatifs.

Dans ce contexte, le F1-score est particulièrement adapté lorsqu'on souhaite équilibrer l'impact des faux positifs et des faux négatifs, ce qui est crucial pour la classification binaire des sentiments. L'ensemble de cette méthodologie assure une comparaison rigoureuse de l'impact de la réduction de dimension sur différents types de modèles supervisés, et permet d'évaluer précisément l'apport spécifique de l'ACP à noyaux par rapport aux approches traditionnelles.

4.4.6 Résultats expérimentaux :

Le tableau suivant présente les meilleurs scores F1 obtenus, ainsi que les aires sous la courbe ROC (AUC) associées :

Modèle	Réduction	n_composantes	F1-score	AUC
KNN	Aucune	–	0.5177	0.702
KNN	ACP	300	0.6320	0.712
KNN	**ACP à noyaux (cosinus)**	**100**	**0.6645**	**0.731**
LogReg	Aucune	–	0.8247	0.914
LogReg	ACP	700	0.8235	0.912
LogReg	**ACP à noyaux (cosinus)**	**700**	**0.8280**	**0.928**
SVM	Aucune	–	0.8267	0.926
SVM	ACP	2000	0.8263	0.924
SVM	**ACP à noyaux (cosinus)**	**1000**	**0.8375**	**0.932**

TABLE 2 – Scores F1 obtenus par les modèles selon la méthode de réduction de dimension appliquée.

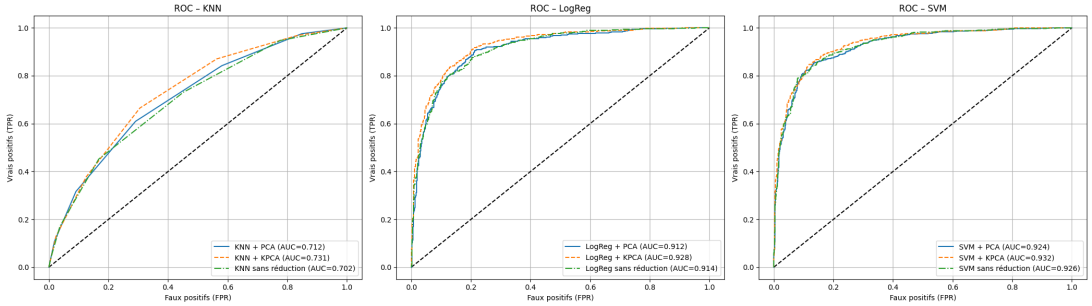

FIGURE 4 – Courbes ROC des modèles KNN, LogReg et SVM selon la réduction utilisée.

4.4.7 Interprétation des résultats obtenus

Les résultats expérimentaux mettent en évidence plusieurs tendances significatives.

Tout d'abord, l'application d'une réduction de dimension améliore systématiquement les performances des trois modèles étudiés par rapport à l'utilisation directe des vecteurs BoW sans réduction. Cette observation confirme que la très forte dimensionnalité et la sparsité des représentations initiales nuisent à l'efficacité des algorithmes d'apprentissage supervisé.

Ensuite, l'Analyse en Composantes Principales à noyaux (ACP à noyaux) avec noyau cosinus surpasse systématiquement l'ACP classique, quelle que soit l'architecture du modèle. Cette supériorité peut s'expliquer par la capacité du noyau cosinus à mieux capturer les relations de similarité directionnelle entre documents textuels, ce qui est crucial dans des espaces creux où les distances euclidiennes perdent de leur pertinence.

Le gain est particulièrement marqué pour le k-plus proches voisins (KNN), dont les performances sont fortement influencées par la notion de proximité locale. En effet, la réduction de dimension non linéaire via l'ACP à noyaux permet de projeter les données dans un espace où la structure de voisinage est mieux conservée, favorisant ainsi la performance de ce type de modèle.

Pour la régression logistique, modèle linéaire par nature, l'amélioration apportée par l'ACP à noyaux est plus modérée. Cela s'explique par le fait que la réduction de dimension n'induit pas de transformation radicale de la frontière de décision dans le cas de représentations déjà adaptées à une séparation linéaire.

Enfin, le SVM avec noyau RBF bénéficie également de la projection par ACP à noyaux, atteignant les meilleurs résultats globaux, avec un F1-score de **0.8375** et une aire sous la courbe ROC (AUC) de **0.932**. Cette amélioration est cohérente avec la nature du SVM, capable d'exploiter efficacement des représentations non linéaires pour construire des séparations optimales.

4.5 Conclusion de la phase expérimentale

Cette première étude expérimentale démontre l'efficacité de l'Analyse en Composantes Principales à noyaux pour la classification de sentiments sur des critiques de films vectorisées par modèle Bag-of-Words. L'application de l'ACP à noyaux avec noyau cosinus permet d'améliorer de manière tangible les performances supervisées sur l'ensemble des modèles testés. L'amélioration est particulièrement nette pour les algorithmes sensibles à la structure locale des données, tels que le KNN. Le SVM associé à la réduction par ACP à noyaux atteint les meilleures performances globales.
Ces résultats valident l'intérêt de la réduction de dimension non linéaire dans des contextes où les représentations textuelles sont creuses et de grande dimension.

5 Détection d'anomalies sur MNIST

Nous allons implémenter une détection d'anomalies entre deux classes utilisant uniquement l'ACP à noyaux, et montrer que cette application est plus performante que celle effectuée avec l'ACP classique.

5.1 Prétraitement des images MNIST

Dans cette première étape, nous avons exploité le dataset MNIST à l'aide de la bibliothèque Keras contenant 70,000 images de chiffres manuscrits. Les images originales, de taille 28×28, sont redimensionnées en images de 8×8 pixels afin de simplifier la représentation tout en conservant des informations essentielles. Ce redimensionnement est réalisé à l'aide de la fonction `resize` de la bibliothèque `skimage`. Un floutage gaussien a ensuite été appliqué à ces images redimensionnées à l'aide de `skimage.filters.gaussian`, avec un paramètre `sigma = 0.5`, afin de réduire leur caractère binaire et d'obtenir des variations locales moins abruptes. Cette opération permet d'atténuer le bruit tout en conservant les contours significatifs. Elle rend les données plus adaptées à la modélisation via des méthodes de projection, tout en réduisant la dimension des vecteurs d'entrée à 64 (au lieu de 784), ce qui diminue considérablement la charge mémoire et le coût computationnel.

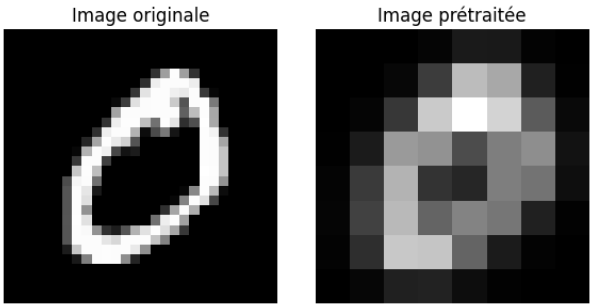

FIGURE 5 – Exemple d'une image brute et de son exemple prétraité.

5.2 Application de l'ACP à noyaux

L'objectif de cette étape est de modéliser la structure des images correspondant au chiffre « 0 » dans un espace de caractéristiques non-linéaire. Pour cela, nous utilisons la classe `ACP à noyaux` de scikit-learn avec un noyau Gaussien (RBF). Le paramètre `gamma` du noyau est défini selon la relation $\gamma = \frac{1}{2\sigma^2}$, où σ représente la largeur du noyau. Nous allons pour l'instant choisir arbitrairement $\sigma^2 = 1$. Ainsi, seules les images de la classe « 0 » issues du jeu d'entraînement sont utilisées pour ajuster le modèle, permettant d'extraire un sous-espace principal caractérisant cette classe. Nous allons choisir ici $q = 20$, le nombre de composantes principales retenues. Ces deux paramètres sont temporaires, et seront optimisés par la suite à l'aide de la fonction **GridSearchCV**.

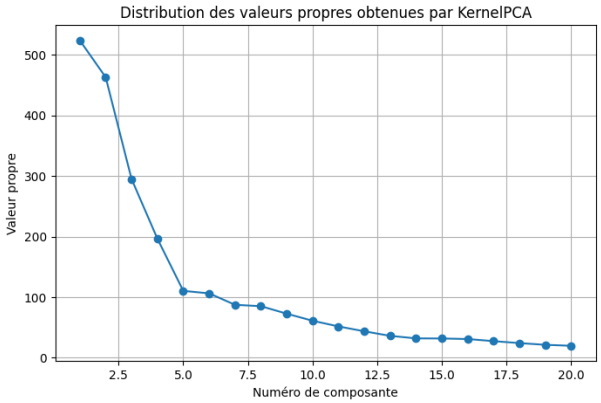

FIGURE 6 – Distribution des valeurs propres extraites par ACP à noyaux lors de l'entraînement sur les images du chiffre 0.

La décroissance rapide des valeurs propres montre que la majorité de la variance est capturée par un nombre réduit de composantes, justifiant notre choix initial de $q = 20$.

5.3 Projection et calcul de l'erreur de reconstruction

Une fois le modèle ACP à noyaux entraîné sur les images de la classe « 0 », les images de l'ensemble de test sont projetées dans l'espace de caractéristiques non linéaire défini par le noyau RBF. Cette projection s'effectue à l'aide de la méthode `transform`, qui permet de représenter les données dans un sous-espace de dimension réduite.
Une reconstruction inverse des images projetées est ensuite réalisée via la méthode `inverse_transform`. Cette étape est rendue possible par le paramètre `fit_inverse_transform=True` activé lors de l'entraînement du modèle. Elle consiste à approximer, dans l'espace original, l'image initiale à partir de ses coordonnées projetées. Il est important de noter que cette reconstruction n'est qu'une approximation : ACP à noyaux ne permet pas une inversion exacte, mais offre une reconstruction par projection inverse dans le sous-espace des composantes principales. L'erreur de reconstruction d'une image $\mathbf{x} \in \mathbb{R}^{64}$ est définie par :

$$\mathcal{E}(\mathbf{x}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2 = \sum_{i=1}^{64} (x_i - \hat{x}_i)^2, \tag{20}$$

où $\hat{\mathbf{x}}$ désigne la reconstruction de $\mathbf{x}$ à partir de sa projection dans l'espace des composantes principales. Cette quantité représente la distance quadratique entre l'image initiale et sa reconstruction, et sert ici de **score de nouveauté** : plus l'erreur est grande, plus l'image est considérée comme atypique par rapport aux données d'entraînement (ici, les chiffres « 0 »). Cette mesure peut être interprétée comme un *score de nouveauté* : plus une image est atypique par rapport à la classe « 0 », plus son erreur de reconstruction sera élevée. Cette approche repose sur l'hypothèse que les images similaires à celles du jeu d'entraînement (les « 0 ») seront correctement reconstruites, tandis que les images d'autres chiffres seront mal reconstruites, car elles ne respectent pas la structure apprise par le modèle.
Empiriquement, cette hypothèse est vérifiée : la moyenne des erreurs de reconstruction calculées sur les images de la classe « 0 » s'élève à environ 2 %, tandis qu'elle atteint 33 % pour les autres chiffres. Cette différence marquée constitue la base de la détection d'anomalies mise en œuvre dans notre approche. rence marquée constitue la base de la détection d'anomalies mise en œuvre dans notre approche.

5.4 Définition du seuil et classification

Pour classifier les images comme « normales » (chiffre 0) ou comme anomalies (non-0), nous fixons un seuil de détection. Ce seuil est défini comme le 95$^{\text{e}}$ percentile des erreurs de reconstruction calculées sur les images de la classe « 0 ». Ce choix garantit qu'au moins 95 % des images issues de la classe normale sont correctement reconnues, ce qui permet de limiter le taux de faux positifs.
Une fois ce seuil déterminé, chaque image de l'ensemble de test est évaluée individuellement : une erreur de reconstruction supérieure ou égale à ce seuil entraîne une classification de l'image comme anomalie. À l'inverse, une erreur inférieure au seuil conduit à la considérer comme normale.
Dans notre configuration expérimentale, nous considérons la classe positive comme étant celle des anomalies, c'est-à-dire les chiffres différents de « 0 ». Ainsi, un *vrai positif* (TP) correspond à une image d'un chiffre non-0 correctement détectée comme anomalie, tandis qu'un *faux positif* (FP) désigne une image appartenant à la classe « 0 » qui a été incorrectement classée comme anomalie.
Sur l'ensemble de test, seules 5 images ont été mal classées, toutes issues de la classe « 0 ». Elles constituent donc des faux positifs, confirmant le bon comportement du système dans le cas où la majorité des données normales sont correctement identifiées.

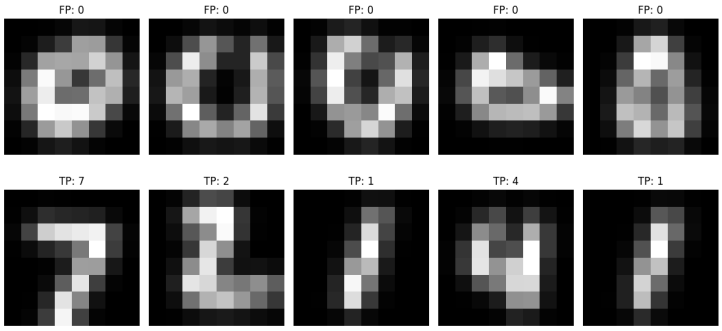

FIGURE 7 – Exemples de fausses détections (false positives, FP) sur la ligne supérieure et de vraies détections (true positives, TP) sur la ligne inférieure. Les labels indiquent le chiffre réel associé à l'image ; par exemple, « FP : 0 » signifie qu'une image du chiffre « 0 » a été à tort classée comme anomalie, tandis que « TP : 7 » indique qu'une image du chiffre « 7 » a été correctement détectée comme anomalie.

Remarque : Les cinq chiffres « 0 » classés comme anomalies correspondent à des exemples particulièrement mal dessinés du jeu de données MNIST.

5.5 Comparaison avec l'ACP

Dans le but d'évaluer l'intérêt de l'approche non-linéaire, nous avons également mis en œuvre une ACP classique. Cette méthode linéaire est entraînée sur les mêmes images de la classe « 0 » (après aplatissage), puis utilisée pour projeter et reconstruire l'ensemble de test. L'erreur de reconstruction est ensuite calculée de manière analogue à l'ACP à noyaux. Une comparaison est réalisée en traçant les courbes ROC pour chacune des méthodes et en comparant l'aire sous la courbe (AUC). Une meilleure séparation entre les images normales et anormales se traduira par une AUC élevée.

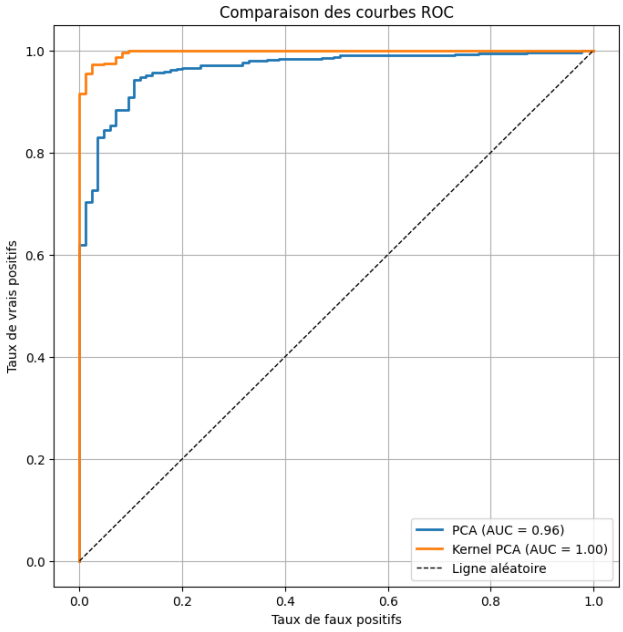

FIGURE 8 – Courbes ROC comparant l'ACP classique (en bleu) et l'ACP à noyaux (en orange) pour la détection du chiffre « 0 » sur MNIST. On observe une meilleure séparation des anomalies (non-0) par l'ACP à noyaux, avec une AUC égale à 1, contre 0,96 pour l'ACP . La ligne pointillée indique la performance aléatoire (AUC = 0,5).

5.6 Recherche de paramètres optimaux via GridSearchCV

Enfin, pour optimiser la performance de la détection de nouveauté, nous effectuons une recherche par grille sur deux paramètres clés : le nombre de composantes principales q et la largeur du noyau σ (via le paramètre `gamma`). Pour chaque combinaison de (q, σ), le modèle ACP à noyaux est entraîné sur les images de la classe « 0 », puis évalué sur l'ensemble de test en calculant l'AUC de la courbe ROC. Cette démarche permet d'identifier la combinaison de paramètres offrant la meilleure séparation entre les images de la classe normale et celles des anomalies. Les paramètres optimaux obtenus sont $q = 30$ et $\sigma^2 = 1$.

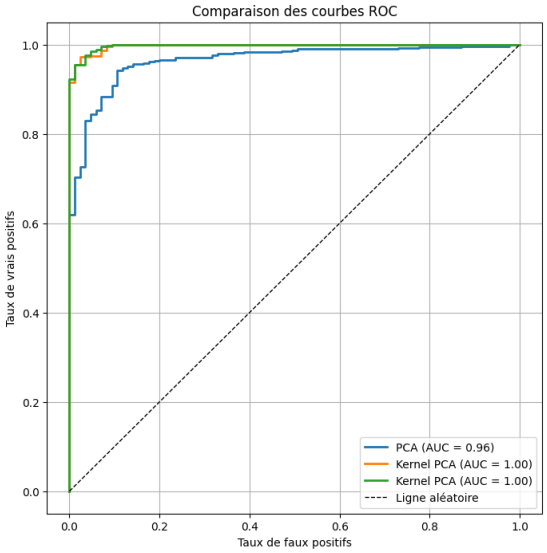

FIGURE 9 – Courbes ROC comparant l'ACP classique (en bleu) et l'ACP à noyaux (en orange) et ACP à noyaux avec les paramètres optimaux (en vert) pour la détection du chiffre « 0 » sur MNIST.

6 Débruitage d'un ECG

Introduction

Les signaux ECG sont souvent corrompus par différents types de bruits d'origine physiologique ou environnementale. Ces bruits à l'exemple du bruit musculaire (ma), les mouvements d'électrodes (em) ou le bruit blanc (wn) peuvent compliquer l'analyse automatique ou visuelle de l'ECG.
Dans ce contexte, il est essentiel d'appliquer des méthodes de débruitage pour retrouver un signal fidèle à l'activité cardiaque réelle. C'est ainsi que nous nous servons dans ce cas de figure d'utiliser une étude comparative entre l'ACP et l'ACP à noyaux sur des données réelles issues des bases MIT-BIH et NSTDB. En effet, la base de données MIT-BIH correspond à une collection de signaux ECG enregistrés à une fréquence de 360 Hz.Quant à la base NSTDB (Noise Stress Test Database), elle fournit des enregistrements de bruits physiologiques indépendants comme ceux cités ci-haut. Ces bruits sont utilisés pour contaminer artificiellement les signaux ECG propres issus de la base MIT-BIH afin d'évaluer les performances de différentes méthodes de débruitage dans un cadre expérimental.
Dans cette étude, nous appliquons une contamination contrôlée avec un rapport signal sur bruit (SNR) fixé à 5 dB, puis nous comparons les performances de l'Analyse en Composantes Principales (ACP) et de l'Analyse en Composantes Principales à noyaux (ACP à noyaux ou KPCA) en termes de capacité à reconstruire le signal propre. L'évaluation repose sur l'erreur quadratique moyenne (RMSE) entre le signal d'origine et le signal reconstruit après débruitage.

6.1 Prétraitement des données

Pour notre étude, nous nous intéressons aux signaux 100,105 et 116 qui, d'après la documentation de la base de données, correspondent à des signaux ECG pris dans les conditions normales. La durée de chaque enregistrement varie, mais nous avons sélectionné les 10 premières secondes de chaque signal, soit 3600 échantillons, pour normaliser les données. Les enregistrements sont échantillonnés à une fréquence de 360 Hz, ce qui nous permet d'analyser les détails temporels du rythme cardiaque.

1. **Chargement du signal ECG** : Le signal est extrait de la base de données au format WFDB (WaveForm DataBase), qui est un format standard pour les enregistrements de signaux physiologiques. Seules les 12 premières secondes du signal sont extraites pour garantir une durée constante de traitement à travers tous les enregistrements.

2. **Prétraitement du signal ECG** : Le signal brut est soumis à un nettoyage pour enlever les artefacts fréquents dans les signaux ECG :

   - *Suppression de la dérive de ligne de base* : La dérive de la ligne de base correspond à de lentes variations du signal ECG causées par la respiration ou des mouvements du patient. Il est considéré comme un bruit. Nous appliquons une décomposition en ondelettes discrètes (filtres locaux) dit en anglais (**DWT** : *Discrete Wavelet Transform*) qui est une technique qui analyse le signal dans différentes intervalles de fréquence. En efet le premier niveau de ce filtre capture les très basses féquences que nous annulons puis le signal sans la variation lente es reconstruit.
   - *Suppression de l'interférence secteur (PLI)* : la PLI (power line interference) est un type bruit issu des champs élctromagnétiques produit par l'éléctricité. Il est aussi éliminé par la même méthode précitée.

3. **Reconstruction du signal** : Le signal est reconstruit à partir des coefficients modifiés de la DWT, et il est redimensionné pour correspondre à la longueur d'origine, soit 3600 échantillons. Le signal est segmenté en battements cardiaques individuels.

   Chaque battement est extrait à partir du signal sous la forme d'une fenêtre temporelle de 311 échantillons : ici, un **échantillon** correspond à une mesure instantanée de la tension cardiaque enregistrée par l'appareil. Par exemple, si l'appareil enregistre 360 mesures par seconde (360 Hz), un échantillon représente environ 2,78 millisecondes.

   En segmentant ainsi, on construit une **matrice** où :

   - chaque **ligne** représente un battement cardiaque,
   - chaque **colonne** correspond à une valeur de l'ECG à un instant donné à l'intérieur du battement (parmi les 311 instants).

   Autrement dit, si l'on extrait n battements dans les 10 secondes de signal, on obtient une matrice de dimension $n \times 311$.

   Par exemple, si dans 10 secondes il y a 15 battements, la matrice aura la taille 15×311.

Cette matrice segmentée est ensuite utilisée comme entrée pour l'algorithme d'ACP à noyaux (KPCA), permettant de traiter les relations non linéaires entre les différents battements cardiaques et de débruiter le signal ECG.

6.2 Ajout des bruits

Afin de simuler des conditions réelles de collecte de signaux ECG, nous avons ajouté trois types de bruits courants présents dans les enregistrements physiologiques : le bruit musculaire (ma), le bruit d'électrode en mouvement (em) et le bruit blanc (wn). Ces bruits ont été extraits de la base de données NSTDB (Noise Stress Test Database), qui contient des enregistrements de bruit typiques rencontrés dans les signaux ECG.

1. **Ajout des bruits** : Chaque type de bruit est ajouté séparément au signal ECG prétraité, conformément à un rapport signal/bruit (SNR) de 5 dB. Ce rapport est calculé en fonction des puissances respectives du signal et du bruit. Le bruit est généré avec la même longueur que le signal ECG et est ensuite ajouté au signal après l'avoir mis à l'échelle pour obtenir le SNR souhaité.

2. **Calcul du SNR** : Le rapport signal/bruit (SNR) est défini comme suit :

$$\text{SNR} = 10 \log_{10} \left(\frac{\text{puissance du signal}}{\text{puissance du bruit}} \right)$$

   où la puissance du signal et du bruit est calculée comme la moyenne des carrés des échantillons :

$$P_{\text{signal}} = \frac{1}{N} \sum_{n=0}^{N-1} |x[n]|^2 \quad \text{et} \quad P_{\text{bruit}} = \frac{1}{N} \sum_{n=0}^{N-1} |b[n]|^2$$

où $x[n]$ et $b[n]$ sont les valeurs respectives du signal et du bruit à l'échantillon n, et N est le nombre total d'échantillons dans le signal. Afin d'obtenir un SNR de 5 dB, nous avons ajusté l'amplitude du bruit ajouté en fonction du signal, de manière à respecter ce seuil. Ce processus garantit que le bruit ajouté au signal ECG reste significatif, mais ne l'impacte pas de manière excessive.

3. **Visualisation des signaux bruités** : Après l'ajout du bruit, nous visualisons les signaux ECG bruités comparés aux signaux originaux. Cela permet de mieux comprendre l'impact de chaque type de bruit sur la qualité du signal ECG. Les signaux bruités sont utilisés pour tester l'efficacité des méthodes de débruitage, comme l'ACP avec et sans noyaux.

FIGURE 10 – Comparaison des signaux avec et sans bruit.

6.3 Méthodes de débruitage des signaux ECG

6.3.1 Choix des hyperparamètres pour l'ACP à noyaux

Pour identifier les meilleurs hyperparamètres du modèle en fonction des enregistrements ECG et des types de bruit, nous avons mis en place une recherche en grille manuelle. Cette méthode consiste à tester systématiquement différentes configurations de noyaux (RBF, polynomial) ainsi que leurs paramètres spécifiques, comme le *gamma* pour le noyau RBF et le *degré* pour le noyau polynomial. Pour chaque combinaison, nous avons ajouté un bruit artificiel au signal propre, puis segmenté le signal en battements cardiaques. Ensuite, nous avons appliqué l'ACP à noyaux pour tenter de débruiter les segments et reconstruire le signal complet. La performance de chaque configuration est évaluée à l'aide de l'erreur quadratique moyenne (RMSE) entre le signal reconstruit et le signal original. Le couple d'hyperparamètres qui minimise la RMSE est alors retenu comme le meilleur pour chaque cas. Cette approche nous a permis d'adapter finement le modèle aux caractéristiques spécifiques de chaque enregistrement et de chaque type de bruit, ce qui est essentiel dans notre contexte.
De ce fait, une méthode analogue a permis le choix du nombre de composantes $n = 8$ pour l'ACP à noyaux.

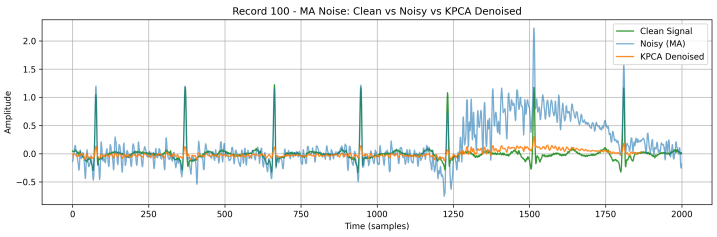

FIGURE 11 – Comparaison des signaux pour le bruit ma.

6.3.2 Comparaison des méthodes de débruitage

Nous comparons les performances des ACP avec et sans noyaux en termes d'erreur quadratique moyenne (RMSE) sur des signaux ECG bruités par différents types de bruit. Les résultats montrent que l'ACP à noyaux surpasse généralement l'ACP classique dans le cas de bruits non linéaires, offrant une meilleure qualité de débruitage pour des signaux contaminés par des artefacts musculaires,des mouvements d'électrodes et le bruit gaussien. Dans la section suivante, nous présentons les résultats expérimentaux obtenus, y compris les valeurs de RMSE pour chaque méthode et chaque type de bruit.

Record	Noise	PCA RMSE	KPCA RMSE	Params
100	ma	0.28	**0.14**	{kernel: rbf, gamma: 0.1}
100	em	4.23	**0.18**	{kernel: rbf, gamma: 0.01}
100	wn	8.75	**0.26**	{kernel: rbf, gamma: 0.01}
105	ma	0.19	**0.17**	{kernel: rbf, gamma: 0.01}
105	em	2.52	**0.27**	{kernel: rbf, gamma: 0.01}
105	wn	5.21	**0.29**	{kernel: rbf, gamma: 0.01}
116	ma	0.29	**0.15**	{kernel: rbf, gamma: 0.1}
116	em	1.29	**0.50**	{kernel: rbf, gamma: 0.1}
116	wn	2.45	**0.53**	{kernel: rbf, gamma: 0.01}

TABLE 3 – Comparaison des performances des différentes ACP en termes de RMSE pour différents enregistrements ECG et types de bruit.

Conclusion : Le tableau montre que l'ACP à noyaux surpasse systématiquement l'ACP pour le débruitage des signaux ECG contaminés par différents types de bruit. Les gains sont particulièrement importants en présence de bruit de mouvement d'électrode (em) et de bruit blanc (wn), où l'erreur est réduite considérablement. Pour le bruit d'origine musculaire (ma), l'amélioration reste notable mais plus modérée. L'utilisation d'un noyau RBF avec un réglage adapté du paramètre gamma s'avère déterminante pour atteindre ces performances. Un gamma de 0.01 est optimal dans la plupart des cas de bruit em et wn, tandis qu'un gamma de 0.1 est préférable pour certains cas de bruit ma. Ces résultats soulignent l'intérêt d'utiliser l'ACP à noyaux, associé à une sélection fine des hyperparamètres, pour des tâches de débruitage complexes. L'ACP à noyaux permet ainsi une reconstruction beaucoup plus fidèle des signaux que l'ACP classique.

Université Paris Dauphine - Année 2025

Références

[1] Hoffmann, H. (2007). ACP à noyaux for novelty detection. *Pattern Recognition*, 40(3), 863–874. Preprint disponible sur : `https://heikohoffmann.de/documents/hoffmann_kpca_preprint.pdf`

[2] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies* (pp. 142–150).

[3] Zipf, G. K. (1949). *Human behavior and the principle of least effort*. Addison-Wesley.

[4] Schmidt, A., & Wiegand, M. (2017). A Survey on Hate Speech Detection using Natural Language Processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media* (pp. 1–10). Association for Computational Linguistics. `https://doi.org/10.18653/v1/W17-1101`

[5] Harris, Z. S. (1954). Distributional structure. *Word*, 10(2–3), 146–162.

[6] Manning, C. D., Raghavan, P., & Schütze, H. (2009). *Introduction to Information Retrieval*. Cambridge University Press.

[7] Turney, P. D., & Pantel, P. (2010). From frequency to meaning : Vector space models of semantics. *Journal of Artificial Intelligence Research*.

[8] Aggarwal, C. C., & Zhai, C. (2012). *Mining Text Data*. Springer.

[9] Huang, A. (2008). Similarity measures for text document clustering. *Proceedings of the 6th New Zealand Computer Science Research Student Conference*.

[10] Shawe-Taylor, J., & Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge University Press.

[11] Mika, S., Schölkopf, B., Smola, A. J., Müller, K.-R., Scholz, M., & Rätsch, G. (1999). ACP à noyaux and De-Noising in Feature Spaces. *Advances in Neural Information Processing Systems*.

[12] Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis : a review and recent developments. *Philosophical Transactions of the Royal Society A : Mathematical, Physical and Engineering Sciences*, 374(2065).

[13] Joachims, T. (1998). *Text categorization with support vector machines : Learning with many relevant features*. In *Proceedings of the 10th European Conference on Machine Learning (ECML)*, pp. 137–142. Springer.

[14] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[15] Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media Inc.

[16] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn : Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.

[17] B. Schölkopf, A. Smola, et K.-R. Müller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[18] I. T. Jolliffe, *Principal Component Analysis*, Springer, 2^{e} édition, 2002.

[19] Chollet, F., et al. (2015). *Keras*. GitHub repository : `https://keras.io`

[20] Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... & Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362.

[21] Hunter, J. D. (2007). Matplotlib : A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 90–95.

[22] Salton, G., Wong, A., & Yang, C. S. (1975). *A Vector Space Model for Automatic Indexing*. *Communications of the ACM*, 18(11), 613–620. `https://doi.org/10.1145/361219.361220`

[23] Van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., ... & Yu, T. (2014). scikit-image : Image processing in Python. *PeerJ*, 2, e453. `https://scikit-image.org`

[24] Wikipédia, *Analyse en composantes principales à noyaux*, 2024. Disponible à l'adresse : `https://fr.wikipedia.org/wiki/Analyse_en_composantes_principales_%C3%A0_noyaux`.

[25] P. Bertrand et D. Pasquignon, *Analyse des données*, Département MIDO, M1, Université Paris-Dauphine, version du 16 septembre 2023.

[26] `https://www.researchgate.net/publication/317729132_ECG_Signal_Denoising_through_Kernel_Principal_Components`

[27] OpenAI, *ChatGPT, modèle de langage utilisé pour l'aide à la rédaction, l'assistance dans les idées de code, et pour répondre à des questions pendant la réalisation du mémoire*, version 2025, disponible sur `https://openai.com/chatgpt`.