Géométrie des Données et Apprentissage sur Variétés

Module 1 – Introduction enrichie avec exercices

Mastère Spécialisé HPC-AI

November 19, 2025

Plan

Module 1 : plan

- Notion de données en haute dimension
- Malédiction de la dimension et concentration de distances
- Variétés et distances géodésiques
- PCA
- Isomap
- Diffusion Maps
- t-SNE
- TP1 : Visualisation MNIST/Swiss Roll avec PCA, t-SNE, Isomap.
- TP2: Comparaison PCA / Isomap / Diffusion maps (scikit-learn, PyGSP).

Pourquoi une géométrie des données ?

- Données modernes : images, sons, textes $\rightarrow$ espaces $\mathbb{R}^d$ avec $d \gg 1$.
- Pourtant, elles possèdent souvent une **structure intrinsèque** de faible dimension.
- **Idée clé** : les données sont souvent concentrées sur une **variété** de dimension intrinsèque $k \ll d$.
- Exemple : images de chiffres manuscrits (784 dimensions) mais proches d'une *variété* de dimension *beaucoup* plus petite ≈ 10.

Explosion dimensionnelle : malédiction de la dimension

- Lorsque la dimension d augmente, le volume de l'espace croît **exponentiellement**.
- Dans un cube unité $[0,1]^d$, la majeure partie du volume se concentre près des **bords**.
- Les distances deviennent moins discriminantes :

Illustration mathématique

Soient n points tirés uniformément dans $[0,1]^d$. On définit :

$$d_{\min} = \min_{i \neq j} \|x_i - x_j\|,$$

$$d_{\max} = \max_{i \neq j} \|x_i - x_j\|.$$

On observe que :

$$\lim_{d \to \infty} \frac{d_{\max} - d_{\min}}{d_{\min}} \to 0.$$

Conséquence : toutes les distances deviennent presque égales !

Explosion dimensionnelle : la malédiction de la dimension (suite)

Intuition :
- Quand la dimension d augmente, les points aléatoires dans $[0,1]^d$ deviennent « équidistants ».
- La distance entre points se concentre autour d'une valeur moyenne.
- Cette perte de pouvoir discriminant est une des formes de la **malédiction de la dimension**.

Mise en place

Deux points $x, y \in [0,1]^d$ i.i.d. uniformes. Définir la distance au carré :

$$S_d := \|x - y\|_2^2 = \sum_{i=1}^{d} (x_i - y_i)^2 = \sum_{i=1}^{d} X_i,$$

où X_i sont i.i.d. copies de $X := (U - V)^2$ avec $U, V \sim \mathcal{U}[0,1]$ indépendants. On montrera : $\mathbb{E}[S_d] = d\mu$, $\mathrm{Var}(S_d) = d\sigma^2$ et la dispersion relative $\to 0$.

Cas $d = 1$: loi de la différence et moments

Posons $Z := U - V$. Alors $Z \sim$ loi triangulaire sur $[-1, 1]$ de densité $f_Z(z) = 1 - |z|$.
Comme $X = Z^2$:

$$\mathbb{E}[X] = \int_{-1}^{1} z^2(1 - |z|)\, dz = 2\int_{0}^{1} z^2(1 - z)\, dz = 2\left(\tfrac{1}{3} - \tfrac{1}{4}\right) = \tfrac{1}{6},$$

$$\mathbb{E}[X^2] = \int_{-1}^{1} z^4(1 - |z|)\, dz = 2\int_{0}^{1} z^4(1 - z)\, dz = 2\left(\tfrac{1}{5} - \tfrac{1}{6}\right) = \tfrac{1}{15}.$$

Donc $\mu = \mathbb{E}[X] = \frac{1}{6}$ et $\sigma^2 = \mathrm{Var}(X) = \frac{1}{15} - \frac{1}{36} = \frac{7}{180}$.

Vérification alternative (moments de l'uniforme)

Moments utiles pour $W \sim \mathcal{U}[0,1]$:
$\mathbb{E}[W] = \frac{1}{2},\ \mathbb{E}[W^2] = \frac{1}{3},\ \mathbb{E}[W^3] = \frac{1}{4},\ \mathbb{E}[W^4] = \frac{1}{5}$. Avec indépendance de U, V :

$$\mathbb{E}[(U-V)^4] = 2\mathbb{E}[U^4] - 4\mathbb{E}[U^3]\mathbb{E}[V] + 6\mathbb{E}[U^2]\mathbb{E}[V^2] - 4\mathbb{E}[U]\mathbb{E}[V^3] = \frac{1}{15}.$$

On retrouve donc $\mathbb{E}[X^2] = \frac{1}{15}$ et $\mathrm{Var}(X) = \frac{7}{180}$.

Étape de récurrence (somme de i.i.d.)

Supposer vrai pour d : $\mathbb{E}[S_d] = d\mu$, $\mathrm{Var}(S_d) = d\sigma^2$. Pour $d+1$:

$$S_{d+1} = S_d + X_{d+1} \Rightarrow \mathbb{E}[S_{d+1}] = (d+1)\mu, \quad \mathrm{Var}(S_{d+1}) = (d+1)\sigma^2,$$

par indépendance. Par récurrence simple, les formules valent pour tout $d \geq 1$.

Concentration relative et Chebyshev

Coefficient de variation :

$$\frac{\sqrt{\mathrm{Var}(S_d)}}{\mathbb{E}[S_d]} = \frac{\sqrt{d\,\sigma^2}}{d\,\mu} = \frac{C}{\sqrt{d}}, \qquad C = \frac{\sqrt{\sigma^2}}{\mu} = 6\sqrt{\tfrac{7}{180}}.$$

Chebyshev : pour tout $\varepsilon > 0$,

$$\Pr\left(|S_d - \mathbb{E}[S_d]| \geq \varepsilon\,\mathbb{E}[S_d]\right) \leq \frac{\sigma^2}{\varepsilon^2\mu^2} \cdot \frac{1}{d} \xrightarrow[d\to\infty]{} 0.$$

Donc **concentration relative** de S_d autour de sa moyenne.

Interprétation géométrique

- $\mathbb{E}[S_d] = d/6$ croît linéairement en d (donc $\mathbb{E}[\|x - y\|_2] \asymp \sqrt{d}$).
- L'écart-type est $\asymp \sqrt{d}$ aussi, mais **relativement** à la moyenne il vaut $O(1/\sqrt{d})$.
- En grande dimension, les distances se ressemblent **relativement** : perte de pouvoir discriminant.

Distance dans le cube unité : mise en place

Soient $x, y \in [0,1]^d$ deux points i.i.d. uniformes.

Définition :

$$S_d = \|x - y\|_2^2 = \sum_{i=1}^{d} (x_i - y_i)^2.$$

Objectif : Étudier la concentration de S_d autour de son espérance quand $d \to \infty$.

Cas $d = 1$: calcul des moments

Soient $U, V \sim \mathcal{U}[0,1]$ indépendants. Posons $X = (U - V)^2$.

- Espérance : $\mathbb{E}[X] = \frac{1}{6}$.
- Moment d'ordre 4 : $\mathbb{E}[(U - V)^4] = \frac{1}{15}$.
- Variance : $\mathrm{Var}(X) = \frac{7}{180}$.

Donc $\mu = 1/6$, $\sigma^2 = 7/180$.

Étape de récurrence

Soient $X_1, \ldots, X_d$ i.i.d. copies de X.

$$S_d = \sum_{i=1}^{d} X_i.$$

Hypothèse de récurrence : $\mathbb{E}[S_d] = d\mu$, $\mathrm{Var}(S_d) = d\sigma^2$.
Alors pour $d+1$:

$$S_{d+1} = S_d + X_{d+1}.$$

Par indépendance :

$$\mathbb{E}[S_{d+1}] = (d+1)\mu, \quad \mathrm{Var}(S_{d+1}) = (d+1)\sigma^2.$$

Donc la propriété est vraie pour tout d.

Concentration relative

Coefficient de variation :

$$\frac{\sqrt{\mathrm{Var}(S_d)}}{\mathbb{E}[S_d]} = \frac{\sqrt{d\sigma^2}}{d\mu} = \frac{C}{\sqrt{d}},$$

avec $C = \frac{\sqrt{\sigma^2}}{\mu} = 6\sqrt{\frac{7}{180}}$.

Donc l'écart relatif décroît comme $1/\sqrt{d}$. Les distances deviennent indiscernables en grande dimension.

Application de l'inégalité de Chebyshev

Pour tout $\varepsilon > 0$:

$$\Pr\left(|S_d - \mathbb{E}[S_d]| \geq \varepsilon \mathbb{E}[S_d]\right) \leq \frac{\sigma^2}{\varepsilon^2 \mu^2} \cdot \frac{1}{d}.$$

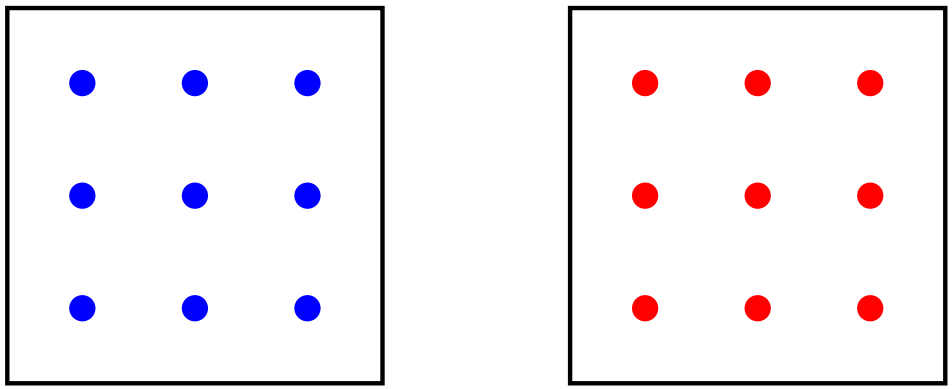
Ainsi, la probabilité d'un écart relatif décroît en $1/d$.

Interprétation géométrique

- La distance moyenne entre deux points est $\sim d/6$.
- L'écart-type est $\sim \sqrt{d}$.
- Donc les distances sont **concentrées autour d'une valeur moyenne**.
- Résultat : en grande dimension, **toutes les distances se ressemblent**.
  Conséquence : difficulté pour les méthodes basées sur les distances (kNN, clustering, etc.).

Exemple numérique : distances en haute dimension

- Simulation : on génère 10^4 points dans $[0,1]^d$.
- On calcule la distance moyenne $\bar{d}$ et son écart-type σ_d.
- Résultat : $\sigma_d/\bar{d} \to 0$ quand $d \to \infty$.



2D 3D (projection) iny Illustration : concentration des distances quand d augmente (ici 2D et projection 3D).

Loi triangulaire de la différence de deux uniformes

Objectif : Montrer que $Z = U - V$, avec $U, V \sim \mathcal{U}[0,1]$ i.i.d., suit une loi triangulaire sur $[-1,1]$.

Convolution

La densité de Z est :

$$f_Z(z) = \int_{-\infty}^{+\infty} f_U(x) f_V(x-z)\, dx.$$

Ici $f_U(x) = f_V(x) = 1$ pour $x \in [0,1]$, 0 sinon.

- Pour $z \in [-1,0]$:

$$f_Z(z) = \int_0^1 1_{0 \le x-z \le 1}\, dx = \int_0^1 1_{z \le x \le 1+z} dx = 1 + z.$$

- Pour $z \in [0,1]$:

$$f_Z(z) = \int_0^1 1_{0 \le x-z \le 1}\, dx = \int_z^1 dx = 1 - z.$$

Exercice : loi de la différence

Soient $U, V \sim \mathcal{U}[0,1]$ indépendants. Montrer que $Z = U - V$ a pour densité :

$$f_Z(z) = \begin{cases} 1+z, & -1 \leq z < 0, \\ 1-z, & 0 \leq z \leq 1, \\ 0, & \text{sinon.} \end{cases}$$

Indication : utiliser la convolution $f_Z(z) = \int f_U(x) f_V(x-z)dx$.

Corrigé de l'exercice

- Étape 1 : écrire $f_Z(z) = \int_0^1 1_{0 \leq x-z \leq 1} dx$.
- Étape 2 : séparer les cas $z < 0$ et $z \geq 0$.
- Étape 3 : calculer les intégrales :

$$f_Z(z) = 1 + z \text{ pour } z \in [-1, 0], \quad f_Z(z) = 1 - z \text{ pour } z \in [0, 1].$$

- Étape 4 : $f_Z(z) = 0$ sinon.

On retrouve bien la loi triangulaire.

Conséquences pratiques

- Les méthodes classiques basées sur des distances euclidiennes deviennent inefficaces.
- Clustering et k-plus proches voisins perdent leur sens.
- Nécessité de méthodes tenant compte de la **structure intrinsèque** des données.
- D'où la géométrie des données : travailler dans une *dimension intrinsèque* $k \ll d$.

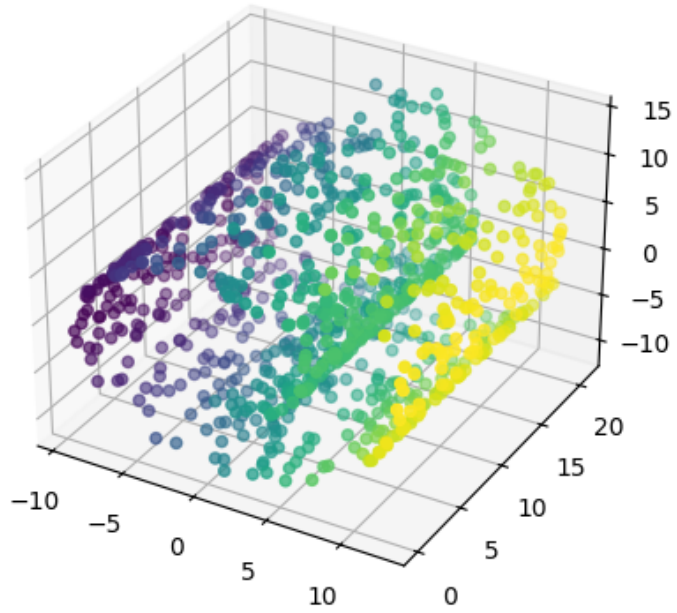
Explosion dimensionnelle

- Données modernes : images, sons, textes $\rightarrow$ espaces $\mathbb{R}^d$ avec $d \gg 1$.
- **Malédiction de la dimension** :
  - Volume croît exponentiellement avec d.
  - Distances deviennent peu discriminantes.
- Besoin de méthodes exploitant la structure « cachée ».

Exemple : MNIST

- Chaque image : $28 \times 28 = 784$ pixels $\rightarrow$ point dans $\mathbb{R}^{784}$.
- Mais les chiffres manuscrits vivent sur une structure de dimension **intrinsèque** $k \ll 784$.
- Question : comment trouver k et représenter les données sur $\mathbb{R}^k$?

Exemple visuel : Swiss Roll



Un nuage de points en 3D mais structurellement 2D.

Idée clé

Hypothèse de variété

Les données $x_i \in \mathbb{R}^d$ sont concentrées sur une variété $\mathcal{M}$ de dimension $k \ll d$.

$$x_i \in \mathcal{M} \subset \mathbb{R}^d, \quad \dim(\mathcal{M}) = k$$

Espace métrique

Un espace métrique est une paire (X, d) avec :

$$d(x, y) \geq 0 \qquad \text{(positivité)}$$
$$d(x, y) = 0 \iff x = y \qquad \text{(séparation)}$$
$$d(x, y) = d(y, x) \qquad \text{(symétrie)}$$
$$d(x, z) \leq d(x, y) + d(y, z) \qquad \text{(inégalité triangulaire)}$$

Distances usuelles

- Norme ℓ^2: $d(x,y) = \|x-y\|_2 = \sqrt{\sum_i (x_i - y_i)^2}$.
- Norme ℓ^1: $d(x,y) = \sum_i |x_i - y_i|$.
- Cosinus: $d(x,y) = 1 - \frac{\langle x,y \rangle}{\|x\|\|y\|}$.
- Distances adaptées aux graphes et variétés.

Distance cosinus et distance angulaire

Pour deux vecteurs $x, y \in \mathbb{R}^d$ non nuls :

$$\cos\theta = \frac{x \cdot y}{\|x\|\|y\|}.$$

- **Distance cosinus classique :**

$$d_{\cos}(x, y) = 1 - \cos\theta = 1 - \frac{x \cdot y}{\|x\|\|y\|}.$$

- Varies entre 0 et 2. - Très utilisée en NLP et apprentissage automatique. - Pas une vraie métrique : triangle inequality peut échouer.

- **Distance basée sur l'angle :**

$$d_{\text{angle}}(x, y) = \arccos\left(\frac{x \cdot y}{\|x\|\|y\|}\right).$$

- Angle réel entre les vecteurs. - Vraie métrique : satisfait la triangle inequality. - Correspond à distance géodésique sur la sphère unité S^{d-1}.

Variété

- Une variété $\mathcal{M}$ est un espace qui ressemble localement à $\mathbb{R}^k$.
- Exemple : sphère S^2 dans $\mathbb{R}^3$.

Définition simplifiée

Pour chaque $x \in \mathcal{M}$, il existe un voisinage U et une bijection $\varphi : U \to V \subset \mathbb{R}^k$.

Distance géodésique

Sur une variété $\mathcal{M}$, la distance naturelle est la longueur du plus court chemin γ contenu dans $\mathcal{M}$:

$$d_{\mathcal{M}}(x, y) = \inf_{\gamma:[0,1]\to\mathcal{M}} \int_0^1 \|\dot{\gamma}(t)\| dt$$

Exemple : distance sur la sphère = angle central multiplié par le rayon.

Variétés de données : définition détaillée

Définition informelle : Une variété $\mathcal{M}$ de dimension k est un sous-ensemble de $\mathbb{R}^d$ qui, localement autour de chaque point, ressemble à $\mathbb{R}^k$.

- Pour chaque point $x \in \mathcal{M}$, il existe un voisinage U et une application bijective **carte** $\varphi : U \to V \subset \mathbb{R}^k$ telle que φ et φ^{-1} soient continues (ou différentiables pour les variétés différentiables).
- k est la **dimension intrinsèque** de la variété.
- Exemple : le cercle $S^1 \subset \mathbb{R}^2$ est une variété 1D, la sphère $S^2 \subset \mathbb{R}^3$ est une variété 2D.

Notation : $\mathcal{M}^k \subset \mathbb{R}^d$ indique une variété de dimension k dans $\mathbb{R}^d$.

Propriétés fondamentales d'une variété

- **Localement Euclidienne :** autour de chaque point, les distances et topologie se comportent comme dans $\mathbb{R}^k$.
- **Dimension intrinsèque :** le nombre minimal de coordonnées nécessaires pour paramétrer la variété.
- **Continuité et différentiabilité :** cartes et applications de transition doivent être continues ou différentiables.
- **Géodésiques :** la distance naturelle sur la variété est la longueur du plus court chemin restant dans la variété.
- **Courbure :** mesure locale de la déviation par rapport à un espace plat $\mathbb{R}^k$.

Exemples de variétés courantes

- Cercle S^1 dans $\mathbb{R}^2$ (dimension 1).
- Sphère S^2 dans $\mathbb{R}^3$ (dimension 2).
- Cylindre dans $\mathbb{R}^3$ (dimension 2).
- Variétés de matrices de rang fixe : $\{X \in \mathbb{R}^{m \times n} : \mathrm{rank}(X) = r\}$.
- Espace des rotations $SO(3)$ (dimension 3), utilisé en robotique et vision.

Carte et atlas

- **Carte :** fonction $\varphi : U \subset \mathcal{M} \to V \subset \mathbb{R}^k$ qui localement paramétrise la variété.
- **Atlas :** collection de cartes $\{(U_i, \varphi_i)\}$ recouvrant toute la variété.
- **Exemple :** sphère S^2, on peut utiliser coordonnées sphériques différentes pour couvrir les pôles et l'équateur.
- Les cartes permettent de définir des notions de dérivée, intégrale et courbure sur la variété.

Distances et géodésiques sur une variété

- La distance euclidienne $\|x - y\|$ dans $\mathbb{R}^d$ ne respecte pas toujours la structure intrinsèque de la variété.
- La **distance géodésique** $d_{\mathcal{M}}(x, y)$ est la longueur du chemin le plus court restant dans la variété.
- Exemples :
  - Cercle S^1: distance géodésique = arc le plus court entre deux points.
  - Sphère S^2: distance géodésique = longueur de l'arc de grand cercle.
- Les distances géodésiques sont fondamentales pour la réduction de dimension non linéaire et le clustering sur variétés.

Distance géodésique sur la sphère : arc de grand cercle

- Le plus court chemin sur la sphère est l'**arc de grand cercle** reliant A et B.
- Longueur de l'arc : $L = R \cdot \theta$, avec θ en radians.
- Angle au centre : $\theta = \arccos\left(\frac{x \cdot y}{R^2}\right)$ pour $x, y \in S^2$.
- Pour une sphère unité $(R = 1)$: $d_{S^2}(x, y) = \arccos(x \cdot y)$.

Applications des variétés en science des données

- Réduction de dimension non linéaire : Isomap, LLE, Diffusion Maps.
- Détection de structure intrinsèque dans des données haute dimension.
- Modélisation de données sur des espaces non Euclidiens : rotations, formes, graphes.
- Méthodes de machine learning adaptées aux données de faible dimension intrinsèque.

Laplacien de graphe

Pour un graphe $G = (V, E)$ avec poids w_{ij} :

$$D_{ii} = \sum_j w_{ij},$$
$$L = D - W.$$

Propriétés :

- L est semi-défini positif.
- L approxime le Laplacien sur la variété sous-jacente.

Exemple : chaleur sur un graphe

$$\frac{du}{dt} = -Lu$$

L joue le rôle de dérivée seconde discrète. Solution : $u(t) = e^{-tL}u(0)$.

Réduction de dimension

But : trouver $f : \mathbb{R}^d \to \mathbb{R}^k$ qui conserve la géométrie.

- PCA : directions de variance maximale.
- Isomap : distances géodésiques.
- Diffusion maps : diffusion de chaleur.

PCA en formule

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^T$$

vecteurs propres de $\Sigma \Rightarrow$ *directionsprincipales*.

Analyse en Composantes Principales (PCA) : Introduction

- PCA est une méthode linéaire de réduction de dimension.
- Objectif : trouver les directions principales (axes) qui capturent le plus de variance dans les données.
- Idée : projeter les données dans un sous-espace de dimension $k \ll d$ en minimisant la perte d'information.
- Utile pour visualisation, compression, prétraitement de machine learning.

Formulation mathématique de la PCA

- Soient $X = [x_1, \ldots, x_n]^\top \in \mathbb{R}^{n \times d}$ les données centrées ($\bar{x} = 0$).
- Matrice de covariance :

$$\Sigma = \frac{1}{n} X^\top X \in \mathbb{R}^{d \times d}.$$

- Cherchons vecteurs propres v_j et valeurs propres λ_j :

$$\Sigma v_j = \lambda_j v_j, \quad j = 1, \ldots, d.$$

- Les v_j sont les directions principales (composantes principales).

Projection sur les composantes principales

- Les k premières composantes principales correspondent aux k plus grandes valeurs propres $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k$.
- Projection :

$$y_i = V_k^\top x_i, \quad V_k = [v_1, \ldots, v_k] \in \mathbb{R}^{d \times k}.$$

- Reconstruction approchée :

$$\hat{x}_i = V_k y_i = V_k V_k^\top x_i.$$

- Erreur de reconstruction : minimisée par la PCA.

Propriété d'optimalité

La PCA est l'approximation linéaire optimale au sens des moindres carrés :

$$\min_{V_k \in \mathbb{R}^{d \times k}, V_k^\top V_k = I_k} \sum_{i=1}^{n} \|x_i - V_k V_k^\top x_i\|^2.$$

La solution est donnée par les k vecteurs propres associés aux k plus grandes valeurs propres de Σ.

Variance expliquée

- Variance totale : $\mathrm{Tr}(\Sigma) = \sum_{j=1}^{d} \lambda_j$.
- Variance capturée par les k premières composantes : $\sum_{j=1}^{k} \lambda_j$.
- Fraction de variance expliquée :

$$\mathrm{FVE}(k) = \frac{\sum_{j=1}^{k} \lambda_j}{\sum_{j=1}^{d} \lambda_j}.$$

- Permet de choisir le nombre optimal k pour la réduction de dimension.

Exemple numérique

- Données $X \in \mathbb{R}^{100\times 5}$ simulées.
- Calculer $\Sigma = X^\top X/100$.
- Calcul des valeurs propres et vecteurs propres.
- Projection sur $k = 2$ premières composantes principales pour visualisation.
- Observer la concentration des données le long des directions principales.

PCA et SVD

- Alternative : utiliser la décomposition en valeurs singulières (SVD) : $X = U\Sigma V^\top$.
- Composantes principales : colonnes de V.
- Valeurs propres : carrés des valeurs singulières divisées par n.
- Utile pour $d \gg n$ ou pour stabilité numérique.

Exercice PCA

Soit un ensemble de données centrées $X \in \mathbb{R}^{10\times 3}$:

- Calculer la matrice de covariance Σ.
- Déterminer les vecteurs propres et valeurs propres.
- Projeter les données sur les 2 premières composantes principales.

Corrigé exercice PCA

- Étape 1 : $\Sigma = X^\top X/10$.
- Étape 2 : calcul des valeurs propres $\lambda_1 \geq \lambda_2 \geq \lambda_3$ et vecteurs propres v_1, v_2, v_3.
- Étape 3 : projection $y_i = [v_1, v_2]^\top x_i$.
- Étape 4 : visualisation et analyse de la variance expliquée.

Isomap

1. Construire graphe k-NN.
2. Approximer distances géodésiques par plus courts chemins.
3. Appliquer MDS (multidimensional scaling).

Isomap : Introduction

- Isomap (Isometric Mapping) est une méthode non-linéaire de réduction de dimension.
- Objectif : préserver les distances géodésiques entre les points d'une variété plongée dans un espace de haute dimension.
- Extension de MDS (Multidimensional Scaling) aux variétés non-linéaires.

Idée clé d'Isomap

1. Construire un graphe des plus proches voisins (k-NN ou ϵ-voisinage).
2. Approximater les distances géodésiques par des plus courts chemins dans le graphe.
3. Appliquer le MDS classique avec ces distances géodésiques approximées.

Étape 1 : Graphe de voisinage

- Pour chaque point x_i, on relie ses k plus proches voisins (ou tous les voisins dans une boule de rayon ϵ).
- Arêtes pondérées par la distance euclidienne :

$$w_{ij} = \|x_i - x_j\|_2.$$

- Graphe $G = (V, E)$ approximant la structure locale de la variété.

Étape 2 : Distances géodésiques

- La distance géodésique entre x_i et x_j est approximée par la longueur du plus court chemin dans G :

$$d_G(i,j) = \min_{chemins(i \to j)} \sum_{(u,v) \in chemin} w_{uv}.$$

- Utilisation d'algorithmes classiques : Dijkstra ou Floyd–Warshall.
- Matrice des distances géodésiques $D_G = (d_G(i,j))_{i,j}$.

Étape 3 : Application de MDS

- Objectif : trouver une configuration de points $Y_1, \ldots, Y_n \in \mathbb{R}^d$ telle que
$$\|Y_i - Y_j\|^2 \approx d_G(i,j)^2.$$
- On applique le MDS classique (Scaling multidimensionnel) :
  1. On construit la matrice des distances au carré $D_G^{(2)} = (d_G(i,j)^2)_{ij}$.
  2. On centre cette matrice :
$$B = -\tfrac{1}{2} H D_G^{(2)} H, \quad H = I - \tfrac{1}{n} 1 1^\top.$$
  3. B est une approximation de la matrice de Gram $YY^\top$.
  4. On diagonalise $B = V \Lambda V^\top$.
  5. Les coordonnées en dimension d sont données par :
$$Y = V_d \Lambda_d^{1/2},$$
où V_d contiennent les d vecteurs propres principaux et Λ_d les d plus grandes valeurs propres.
- Ainsi, on obtient une immersion en d dimensions qui préserve au mieux les distances géodésiques.

Étape 3 : Application de MDS

- On applique le MDS classique sur D_G pour obtenir une représentation en basse dimension.
- Centrage double :

$$B = -\frac{1}{2} H D_G^{(2)} H, \quad H = I - \frac{1}{n} 1 1^\top.$$

- Décomposition spectrale :

$$B = V \Lambda V^\top.$$

- Coordonnées en dimension d :

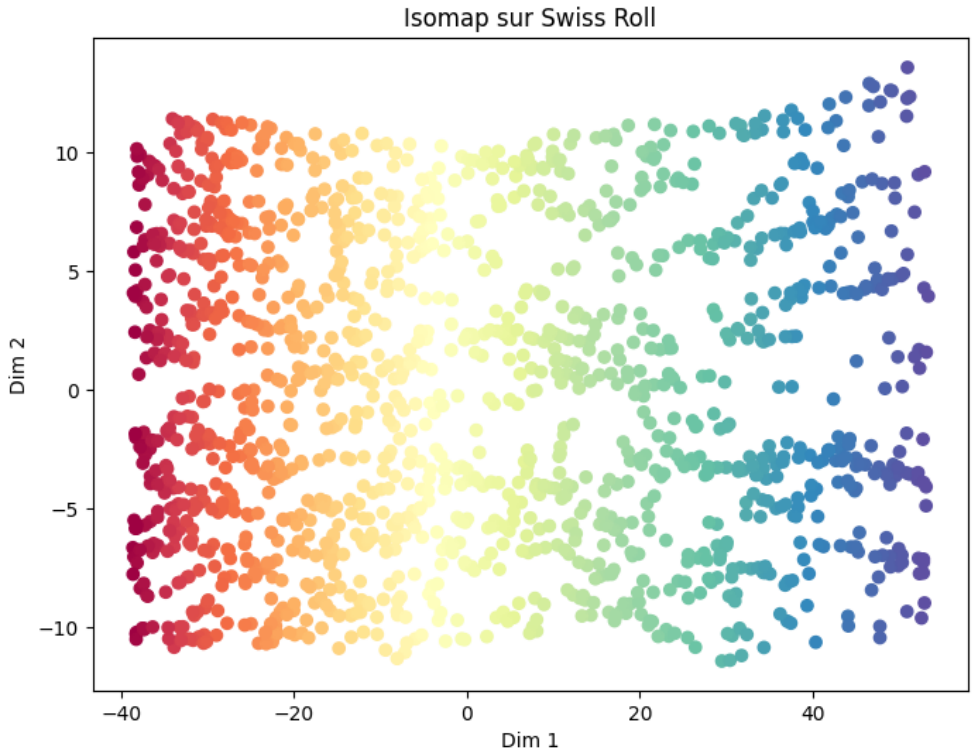
$$Y = V_d \Lambda_d^{1/2}.$$

Propriétés d'Isomap

- Préserve la géométrie globale de la variété.
- Gère des données fortement non-linéaires (spirale, Swiss roll).
- Consistance théorique : converge vers la métrique de la variété quand $n \to \infty$.
- Sensible aux choix de k ou ϵ.

Exemple : Swiss Roll

- Données 3D enroulées en spirale (« rouleau suisse »).
- PCA : incapacité à « dérouler » la structure.
- Isomap : reconstitue correctement la structure 2D sous-jacente.



Exercice Isomap

- Générer un jeu de données 3D de type Swiss roll.
- Construire le graphe k-NN avec $k = 10$.
- Calculer les plus courts chemins (algorithme de Dijkstra).
- Appliquer MDS sur la matrice de distances.
- Visualiser la représentation 2D obtenue.

Corrigé Exercice Isomap

- Étape 1 : génération des points (x, y, z).
- Étape 2 : construction du graphe k-NN.
- Étape 3 : distances géodésiques via Dijkstra.
- Étape 4 : matrice B et décomposition spectrale.
- Résultat : une carte 2D déroulant le Swiss roll.

Résumé intuitif d'Isomap

- But : représenter les données dans un espace de dimension réduite d (typiquement 2 ou 3).
- On calcule d'abord les distances géodésiques $d_G(i,j)$ entre points dans l'espace original (via le graphe de voisinage).
- Puis on cherche des points $Y_i \in \mathbb{R}^d$ tels que :

$$\|Y_i - Y_j\| \approx d_G(i,j).$$

- Ainsi :
  - Les distances locales sont respectées.
  - La géométrie intrinsèque de la variété est préservée.
  - On obtient une carte en basse dimension reflétant la structure réelle des données.
- Différence clé avec PCA : Isomap ne se base pas uniquement sur les distances euclidiennes globales, mais sur les distances intrinsèques (géodésiques).

Diffusion Maps

- Construire matrice de transition $P = D^{-1}W$.
- Valeurs propres $\lambda_1, \lambda_2, \ldots$ et vecteurs propres ψ_i.
- Représentation des données :

$$x \mapsto (\lambda_1^t \psi_1(x), \ldots, \lambda_k^t \psi_k(x))$$

Diffusion Maps : introduction

- Diffusion Maps (Coifman & Lafon, 2006) : méthode spectrale non-linéaire de réduction de dimension.
- Idée : utiliser la dynamique de diffusion (marche aléatoire) sur le graphe de voisinage pour révéler la géométrie intrinsèque.
- Résultat : coordonnées multi-échelle (robustes au bruit et à l'échantillonnage irrégulier).

Diffusion Maps : idée générale

- Construire un graphe de similarité entre les points (par ex. noyau gaussien).
- Normaliser la matrice de similarité K pour obtenir une matrice de transition stochastique P d'une marche aléatoire.
- Les puissances P^t décrivent la diffusion des probabilités après t pas.
- Les coordonnées réduites sont données par les vecteurs propres dominants de P associés aux valeurs propres $\{\lambda_k\}$.

Pseudo-code : Diffusion Maps (notations corrigées)

```
Entrée : Données {x_i}, paramètre $\epsilon$ (bande du noyau),
1. Construire la matrice de similarité : $K(i,j) = \exp(-\|x_i
2. Normaliser : D(i,i) = somme_j K(i,j).
3. Matrice de transition : P = D^{-1} K.
4. Calcul spectral : trouver les paires (\lambda_k, \phi_k) ave
5. Embedding diffusion à temps t : y_i = (\lambda_1^t \phi_1(i)
Sortie : Points {y_i} en dimension réduite.
```

Remarques sur les notations

- K : matrice de similarité (symétrique positive).
- D : matrice diagonale des degrés $D(i,i) = \sum_j K(i,j)$.
- $P = D^{-1}K$: matrice de transition stochastique (les lignes somment à 1).
- (λ_k, ϕ_k) : valeurs propres et vecteurs propres droits de P.
- L'embedding utilise les d plus grandes valeurs propres (hors $\lambda_0 = 1$ trivial).

Construction : noyau à noyau gaussien

- Noyau usuel (gaussien) :

$$W_{ij} = \exp\Big(- \frac{\|x_i - x_j\|^2}{\varepsilon} \Big).$$

- $\varepsilon > 0$ est la largeur du noyau (contrôle la localité).
- W est souvent construit sur un graphe k-NN (matrice creuse) pour l'efficacité.
- Noter le rôle de la densité d'échantillonnage : $q_i = \sum_j W_{ij}$.

Normalisation : corriger la densité (paramètre α)

- Pour neutraliser l'effet d'une densité non uniforme, on normalise :

$$\widetilde{W}_{ij} = \frac{W_{ij}}{(q_i q_j)^{\alpha}}, \qquad \alpha \in [0,1].$$

- Interprétations pratiques :
  - $\alpha = 0$: pas de correction (sensitif à la densité).
  - $\alpha = 1$: corrige l'échantillonnage pour approcher le Laplace–Beltrami (annule l'effet de densité).
  - $\alpha \in (0,1)$: compromis ; souvent $\alpha = 1/2$ est utilisé empirique.
- Ensuite : construire $\widetilde{D} = \mathrm{diag}(\widetilde{d}_i)$ avec $\widetilde{d}_i = \sum_j \widetilde{W}_{ij}$.

Opérateur de Markov et symétrisation

- Opérateur de transition (ligne-stochastique) :

$$P = \widetilde{D}^{-1}\widetilde{W}, \qquad P_{ij} = \Pr(x_i \to x_j \text{ en 1 pas}).$$

- Pour une décomposition numérique stable, on passe à la forme symétrique :

$$\widehat{W} = \widetilde{D}^{-1/2}\widetilde{W}\ \widetilde{D}^{-1/2}.$$

- Si $\widehat{W} = \Phi\Lambda\Phi^\top$ (diagon.), alors

$$\Psi = \widetilde{D}^{-1/2}\Phi$$

contient les vecteurs propres de P et $P\Psi = \Psi\Lambda$.

Lien continu : noyau de chaleur et Laplace–Beltrami

- Dans la limite $n \to \infty$, $\varepsilon \to 0$ (avec un bon régime), P approxime un opérateur de diffusion :

$$P^t \approx e^{t\Delta},$$

où Δ est le Laplace–Beltrami (ou un opérateur lié selon α).

- Les vecteurs propres de P les fonctions propres de Δ : portent la géométrie multi-échelle de la variété.

Embedding : coordonnées de diffusion

- Diagonaliser P: $P\psi_j = \lambda_j \psi_j$, avec $1 = \lambda_0 \geq \lambda_1 \geq \cdots$.
- Embedding à l'échelle t :

$$\Psi_t(x_i) = \left(\lambda_1^t \psi_1(i),\ \lambda_2^t \psi_2(i),\ \ldots,\ \lambda_k^t \psi_k(i)\right) \in \mathbb{R}^k.$$

- t joue le rôle d'échelle temporelle : plus t grand $\rightarrow$ relations plus globales sont privilégiées.
- On supprime la composante constante associée à $\lambda_0 = 1$.

Distance de diffusion

- Distance de diffusion (au pas t) :

$$D_t^2(x_i, x_j) = \sum_{\ell \geq 1} \lambda_\ell^{2t} \big(\psi_\ell(i) - \psi_\ell(j)\big)^2.$$

- Propriété clé : D_t est la distance euclidienne dans l'espace Ψ_t :

$$D_t(x_i, x_j) = \|\Psi_t(x_i) - \Psi_t(x_j)\|_2.$$

- Interprétation : D_t compare la distribution des positions après t pas de marche aléatoire démarrée en x_i et x_j.

Algorithme pratique (récapitulatif)

1. Construire k-NN (ou graphe complet) et $W_{ij} = \exp(-\|x_i - x_j\|^2/\varepsilon)$.
2. Calculer q_i, choisir α et normaliser $\widetilde{W}_{ij} = W_{ij}/(q_i q_j)^\alpha$.
3. Construire $\widetilde{D}$ et $P = \widetilde{D}^{-1}\widetilde{W}$ (ligne-stochastique).
4. Calculer quelques premières valeurs propres et vecteurs propres (Lanczos/ARPACK sur la forme creuse ou sur $\widehat{W}$).
5. Embedding Ψ_t pour t et réduction sur k premières coordonnées utiles.

Choix pratiques et pièges

- Choix de ε : heuristiques — médiane des distances au carré, ou adaptation locale (bandwidth local).
- Choix de k pour k-NN : trop petit $\rightarrow$ graphe non connecté ; trop grand $\rightarrow$ raccourcis artificiels.
- α corrige la densité : tester 0, 1/2, 1 selon but (estimation de Laplace–Beltrami $\rightarrow$ α proche de 1).
- Diagonalisation : utiliser routines creuses (eigs / eigsh) ; calculer seulement premiers vecteurs.

Diffusion Maps : intuition spectrale

- La matrice de transition P est diagonalisable :

$$P\phi_k = \lambda_k \phi_k, \quad k = 0, 1, \ldots, n-1.$$

- Comme P est stochastique, on a $|\lambda_k| \leq 1$ et $\lambda_0 = 1$ associé au vecteur propre constant.
- En élevant P à la puissance t :

$$P^t \phi_k = \lambda_k^t \phi_k.$$

- Les puissances de λ_k contrôlent la persistance de chaque mode de diffusion.

Convergence de la diffusion

- Lorsque $t \to \infty$, seuls les modes dominants (valeurs propres proches de 1) subsistent.
- Les modes associés à $|\lambda_k| < 1$ décroissent géométriquement :

$$\lambda_k^t \to 0 \quad \text{si } |\lambda_k| < 1.$$

- Ainsi, la diffusion filtre progressivement le bruit et ne garde que les structures globales de la variété.
- Cela justifie l'utilisation des vecteurs propres dominants pour l'embedding.

Coordonnées de diffusion

- On définit les coordonnées réduites à temps t :

$$\Psi_t(x_i) = (\lambda_1^t \phi_1(i), \ldots, \lambda_d^t \phi_d(i)).$$

- Interprétation :
  - Chaque coordonnée capture une échelle de diffusion donnée.
  - Les valeurs propres pondèrent la contribution en fonction de la durée t.
- Résultat : les distances euclidiennes dans l'espace Ψ_t approximent la distance de diffusion sur la variété.

Valeurs propres de P stochastique

- P est ligne-stochastique : $\sum_j P_{ij} = 1$, $P_{ij} \geq 0$.
- Le vecteur constant 1 est vecteur propre : $P1 = 1$, donc $\lambda_0 = 1$.
- Théorème de Perron-Frobenius : pour une matrice stochastique irréductible et apériodique,

$$|\lambda_k| < 1 \quad \text{pour } k \geq 1.$$

- Interprétation : modes non dominants décroissent avec le temps :

$$P^t = \sum_k \lambda_k^t \psi_k \phi_k^\top, \quad \lambda_k^t \to 0 \text{ si } k \geq 1.$$

- Conséquence pour Diffusion Maps : les petites valeurs propres sont filtrées par la diffusion, ne restent que les modes lents (grande échelle) dans l'embedding.

Exemple schématique : décroissance de λ_k^t avec t

t=0: $|\lambda_1^0| = 1$, $|\lambda_2^0| < 1$, ...

t=5: $|\lambda_1^5| < 1$, $|\lambda_2^5| \ll 1$, ...

t=10: $|\lambda_1^{10}| \ll 1$, ...

Comparaison synthétique

- **Isomap** : préserve distances géodésiques (MDS sur d_G) — sensible à raccourcis et trous.
- **Diffusion Maps** : préserve similarités multi-échelle via diffusion — robuste au bruit et à l'échantillonnage non uniforme (avec normalisation).
- **t-SNE** : optimisé pour visualisation locale (ne définit pas une métrique globale stable comme D_t).

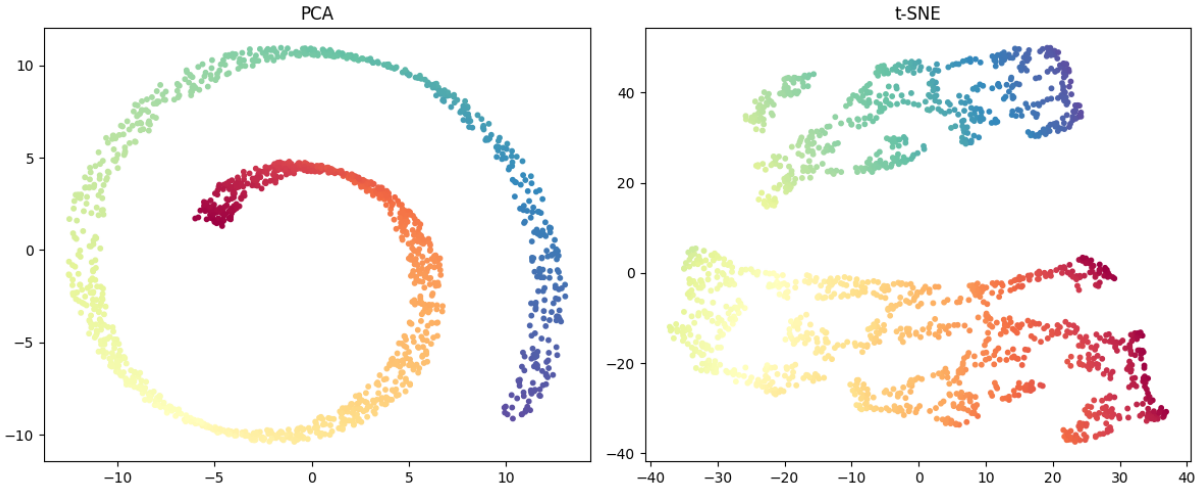
Exercice (pratique)

- Implémenter Diffusion Maps sur un Swiss-roll bruité.
- Tester : différents ε, α (0, 0.5, 1) et t (1,5,10).
- Tracer Ψ_t (2D) et mesurer la corrélation entre D_t et une estimation des distances géodésiques.

Esquisse de code (Python)

```
# pseudo-code (numpy/scipy)
# X : n x D data
# 1) k-NN graph -> sparse distances
# 2) W_ij = exp(-dist^2 / eps)
# 3) q = W.sum(axis=1)
# 4) Wt = W / (q[:,None]*q[None,:])**alpha
# 5) dtilde = Wt.sum(axis=1); Dtil = diag(dtilde)
# 6) P = sparse_diag(1/dtilde) @ Wt
# 7) compute top-k eigenpairs of P (or symmetrized)
# 8) Psi_t = (lambdas[1:k]**t) * psi[1:k,:]  # per-point vectors
```

Exemple de visualisation



PCA vs t-SNE vs UMAP sur MNIST.

t-SNE : idée générale

- Transformer les distances locales en probabilités de voisinage :

$$p_{j|i} = \frac{\exp(-\|x_i - x_j\|^2/2\sigma_i^2)}{\sum_{k\neq i}\exp(-\|x_i - x_k\|^2/2\sigma_i^2)}$$

- Symétrisation :

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$$

- Objectif : trouver un embedding $\{y_i\}$ en 2D/3D tel que les probabilités q_{ij} dans l'espace réduit soient proches de p_{ij} :

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k\neq l}(1 + \|y_k - y_l\|^2)^{-1}}$$

t-SNE : fonction de coût

- Mesure de similarité entre distributions : divergence de Kullback-Leibler

$$C = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

- Minimiser C par descente de gradient pour obtenir l'embedding $\{y_i\}$
- Résultat : les points proches dans l'espace original restent proches dans l'espace réduit, les distances globales sont moins fiables

t-SNE vs Diffusion Maps

- Diffusion Maps :
  - Spectral : vecteurs propres de P (ou L)
  - Distances globales approximatives
  - Capture multi-échelle
- t-SNE :
  - Probabiliste local : p_{ij} et q_{ij}
  - Préserve très fidèlement les voisins proches
  - Distances globales moins significatives
- Conclusion : Diffusion Maps = structure globale, t-SNE = visualisation intuitive

t-SNE : paramètres clés

- **Perplexité** : nombre effectif de voisins pris en compte
- **Learning rate / itérations** : stabilité et convergence
- **Initialisation / Random seed** : influence l'embedding final
- Astuce : tester plusieurs perplexités pour explorer différentes structures locales

Applications pratiques

- Visualisation de données haute dimension.
- Compression (codage parcimonieux).
- Classification et clustering améliorés.
- IA générative : échantillonnage sur variétés.

Exercice 1 – Distance en haute dimension

Soit $x, y \in \mathbb{R}^{100}$ deux vecteurs aléatoires de coordonnées $\sim \mathcal{U}[0,1]$.

- Estimer $\mathbb{E}[\|x - y\|_2]$.
- Comparer avec $\mathbb{E}[\|x - y\|_1]$.

Correction Exercice 1

- Chaque coordonnée $x_i - y_i$ suit une loi $\mathcal{U}[-1, 1]$ de variance $\frac{1}{3}$.
- Espérance de la norme ℓ^2 : $\sqrt{d \cdot \mathbb{E}[(x_i - y_i)^2]} = \sqrt{100 \cdot \frac{1}{3}} \approx 5.77$.
- Espérance de la norme ℓ^1 : $d \cdot \mathbb{E}[|x_i - y_i|] = 100 \cdot \frac{1}{3} \approx 33.3$.

Exercice 2 – Distance géodésique sur la sphère

Soient $x, y \in S^2$ (sphère unité dans $\mathbb{R}^3$).

- Montrer que la distance géodésique est :

$$d(x, y) = \arccos(\langle x, y \rangle)$$

Correction Exercice 2

- Sur S^2, le plus court chemin est un arc de grand cercle.
- L'angle entre x et y est $\theta = \arccos(\langle x, y \rangle)$.
- Longueur de l'arc $= \theta$ (car rayon $= 1$).
- Donc : $d(x, y) = \arccos(\langle x, y \rangle)$.

Exercice 3 – Laplacien de graphe

Graphe à 3 nœuds reliés en ligne : $1 - 2 - 3$ avec poids $w_{ij} = 1$.

- Écrire la matrice de poids W.
- Calculer le Laplacien L.
- Vérifier que $L1 = 0$.

Correction Exercice 3

$$W = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$L = D - W = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

Vérification : $L1 = 0$.

Résumé

Idée clé

Les données haute dimension vivent souvent sur une variété de faible dimension. La géométrie des données permet d'exploiter cette structure via distances, graphes et spectre.

Prochain module

Réduction de dimension non linéaire (Isomap, Diffusion Maps, etc.).