Apprentissage de variété pour mieux appréhender les microscopes

Minh-Duy NGUYEN et Thanh-Chung NGUYEN

April 2023

1 Introduction

En imagerie, la qualité des photos ne peut être améliorée qu'avec une bonne connaissance de l'appareil. Et quand il s'agit de microscopie, c'est là où se joue la différence entre des images interprétables ou non.

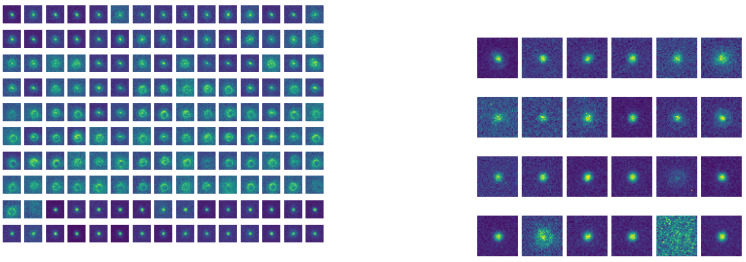
De manière simpliste, un microscope peut être modélisé par un opérateur de convolution dont le noyau (PSF) peut varier d'une acquisition à une autre. À l'aide d'une base de données de plusieurs noyaux de convolution, le but est de l'analyser et de paramétriser sa structure sous-jacente. L'objectif est double : obtenir une image déconvolée plus précise et être assuré de sa précision.

Dans ce projet, nous faisons l'analyse de structure sous-jacente de la base des données en utilisant la librairie sklearn.manifold en Python.

2 Modélisation du problème

2.1 La base de données

Nous prenons en entrées une base de données de 11850 images de taille 42x42. Il s'agit des cellules prises par le microscope.



(a) 150 premères images (b) 16 random images

Figure 1: Extraits de la base de données

On la modélise comme un tableau de 3 dimensions comportant de 11850 tableaux, qui sont 11850 images, de taille 42x42. Notons $n = 11850, K = 42$ et $D = 42 \times 42 = 1764$, nous allons rédimensionner les données en les transformant en n points d'images, répartis dans l'espace $\mathbb{R}^{K \times K}$. Ce que nous avons, tout d'abord, à faire est d'observer la structure de répartition de ces images dans un espace observable parce qur l'espace à 1784 dimensions n'est pas observable pour nous. Cela signifie que nous devons trouver une façon de réduire le nombre de dimensions de l'espace sur lequel sont distribués les images tout en maintenant la précision relative entre les points de données. Ainsi, il est naturel de poser le problème de la réduction dimensionnelle de l'espace de distribution des données.

Après avoir observé la variété de la structure des données dans un espace à 2 ou 3 dimensions, il nous apparait un autre problème que la qualité des images de la base de données est assez médiocre et faible. À partir de là, nous voudrons affiner l'image et les débruiter des données données. Pour le faire, nous allons considérer les images d'entrée comme les matrice de 2 dimensions de pixels de l'image y qui est une convolution d'une matrice de pixels de l'image nette x désirée mais dégradée par une fonction de répartion des points (PSF - point spread function) h. Sachant que les images sont de taille $K \times K$ Le modèle peut être décrit comme suivant :

$$y(n_1, n_2) = (x \circledast h)(n_1, n_2) \tag{1}$$

où $0 \leq n_1, n_2 \leq K - 1$. En utilisant la notation de matrice, le modèle se ramène en

$$\mathbf{y} = \mathbf{H}\mathbf{x} \tag{2}$$

où $\mathbf{y}$,$\mathbf{x} \in \mathbb{R}^D$ sont des vecteurs colonnes ordonnés représentant les tableaux y, x respectivement, et $\mathbf{H}$ est de taille $D \times D$ qui modèlise la fonction de répartition des points h. A partir de ces équations ci-dessus, on résout le problème inverse pour retrouver l'image $\mathbf{x}$ désirée.

3 Appprentissage de variété

Manifold learning (Appprentissage de variété) est une classe de méthodes d'apprentissage automatique non linéaires utilisées pour résoudre les problèmes de réduction de dimensionnalité des données. **Variété** désigne un espace non linéaire. Une variété peut être décrite comme un ensemble de points de données, chacun étant représenté par un vecteur de caractéristiques. Cependant, ces points de données ne sont pas uniformément répartis sur un espace linéaire conventionnel, mais sont plutôt répartis sur un espace non linéaire.

La méthode "apprentissage de variété" vise à trouver la structure non linéaire des données, donnant une représentation des données dans l'espace linéaire pour réduire la dimensionnalité des données et minimiser le bruit. Les méthodes d'"apprentissage de variété" utilisent toutes des techniques mathématiques et statistiques pour résoudre des problèmes de recherche de la meilleure représentation de données non linéaires. L'Appprentissage de variété est le processus de découverte de la structure non linéaire des données et de son incorporation dans un espace linéaire pour réduire la dimensionnalité des données. Les méthodes courantes de manifold learning incluent:

1. Le Scalage multidimensionnel (MDS) : cette méthode réduit la dimensionnalité des données en recherchant un sous-espace de sorte que la distance entre les points dans ce sous-espace ressemble à la distance entre les points dans l'espace original.

2. L'Isomap: cette méthode réduit la dimensionnalité des données en identifiant des hypersurfaces sur le manifold et en calculant les distances entre les points de données en fonction de la longueur du chemin sur le manifold entre eux.

3. L'embedding linéaire local (LLE): cette méthode réduit la dimensionnalité des données en cherchant un sous-espace où la distance entre les points de données dans ce sous-espace est préservée de manière optimale.

Dans le cadre de ce projet, nous traitons la base de données de 11850 points d'images dont chacun est un vecteur à 1764 dimensions, où chaque caractéristique-dimension est un nombre de 0 à 255 représentant l'intensité des pixels. Nous tenterons de trouver une représentation spatiale bidimensionnelle qui transmet les similarités entre les points d'images et maintient les données dans les mêmes classes proches les unes des autres.

4 Réduction dimensionnelle

4.1 Multidimensional Scaling (MDS)

4.1.1 Introduction générale

Multidimensional scaling est un algorithme utilisé pour la réduction dimensionnelle. Il est régulièrement appliqué pour visualiser les similarités entre les

points de données dans un jeu de données, notamment pour des jeu de données linéaire. Ainsi, on le considère souvent comme une projection linéaire. Le MDS classique est équivalent à l'Analyse des Composantes Principales (ACP, ou PCA en anglais), qui est une technique très utile pour l'analyse des données et le traitement d'image. MDS est l'une des premières méthodes dévéloppées pour la réduction dimensionnelle. Il a de vastes applications pour les données de haute dimension en génétique, psychologie, sciences politiques, écologie et de nombreux autres domaines liés à la modélisation des relations structurelles entre objets et à la transmission de leurs similarités.

La différence entre MDS et des autres méthodes de réduction dimentionnelle est que MDS prend seulement en entrées la matrice de dissimilarité au lieu des vectors de position. En effet, étant donné une matrice de dissimilarités par paires $\boldsymbol{D}$ avec des entrées d_{ij} la distance/dissimilarité entre les observations i et j, puis on trouve $x_1, ..., x_n \in R^k$ tel que :

$$\underbrace{d_{ij}^2}_{\text{distance originale}} \quad \approx \quad \underbrace{\|x_i - x_j\|^2}_{\text{configuration de sortie}}$$

C'est-à-dire, nous sommes à trouver une configuration (typiquement celle de dimension inférieure dans $\mathbb{R}^2$) qui maintient les distances euclidiennes dans $\mathbb{R}^k$ aussi proches que possible de nos distances/similarité d'origine. Pourtant, ce fait limite la capacité de MDS à manipuler avec des structures non linéaires tandis que les autres méthodes qui cherchent à préserver les similarités de voisinage locales s'avèverent efficaces avec des données de répartition non linéaire.

4.1.2 Algorithmes pour le MDS classique

Entrées: Distances/dissimilarités par paires pour n objets donnés par la matrice $D \in \mathbb{R}^{n \times n}$ avec des entrées d_{ij}, avec un paramètre k (dimension de sortie désirée).
Sortie: Vecteurs de configuration $x_1, ..., x_n \in \mathbb{R}^k$.
Démarche

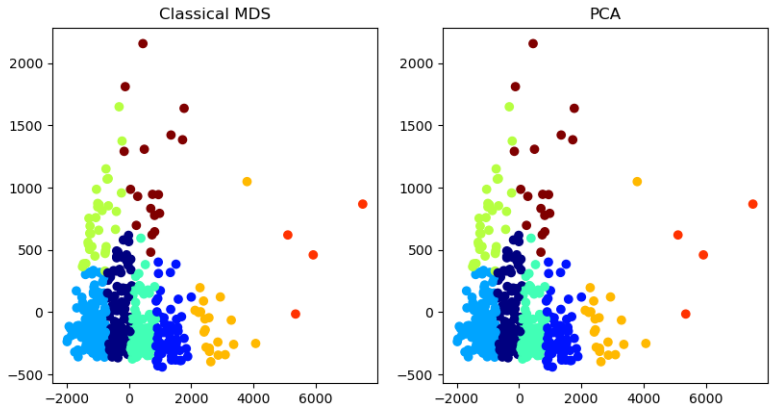
1. Calculer la matrice de Gram et le double centrage : à partir de la matrice de distances par paires, on trouve la matrice de Gram $B \in \mathbb{R}^{n \times n}$ où on fait doubler et centrer avec $B = \frac{-1}{2} C_n D^2 C_n$.

2. Trouver les valeurs propres et les vecteurs propres : Trouver les valeurs propres λ et les vecteurs propres v de la matrice de Gram et former les matrices $\Lambda_n = diag(\lambda_1, \ldots, \lambda_n)$ et $V_n = (v_1, \ldots, v_n)$

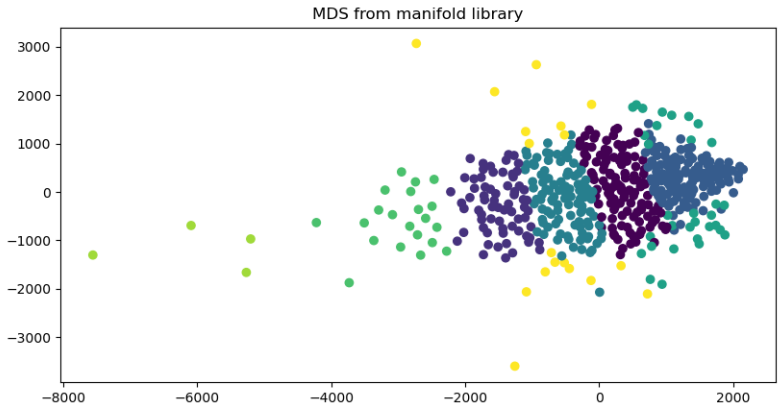
3. Scellage de dimension inférieure : utiliser les k vecteurs propres supérieurs $v_1, ..., v_k$ (associés plus grandes valeurs propres) et décomposition $X = V_k \Lambda \sqrt{k}$ pour former la matrice $X = (x_1, \ldots, x_n)T$ avec des vecteurs de sortie de dimension inférieure $x_i \in \mathbb{R}^k$

4.1.3 Application dans notre projet

Pour notre base de données, nous trouvons que la répartition des points d'images (ou bien des vecteurs d'images à taille 42x42) est aléatoire et ainsi non-linéaire évidemment. De ce fait, l'algorithme de MDS ne peut pas être efficace pour préserver les similarités entre les points de données dans la réduction dimensionnnelle.



(a) MDS classique et PCA



(b) MDS de la librairie manifold

Figure 2: Variété obtennue par les techniques linéaire

Ici, nous avons utilisé un algorithme de regroupement pour diviser les données en grappes plus petites pour faciliter l'observation. Nous avons choisi le nombre de grappes = 8 car après un examen préliminaire des données, nous avons con-

Type	1	2	3	4	5	6	7	8
Cellule	1	1	1	1	2	2	2	2
Anneau	oui	oui	non	non	oui	oui	non	non
Env	clair	bruité	clair	bruité	clair	bruité	clair	bruité

staté (de manière incertaine) qu'il existe 8 types différents d'images cellulaires qui se distinguent à l'œil nu.

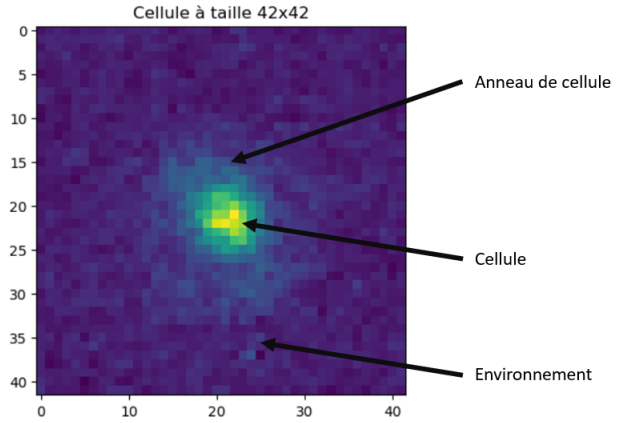


Figure 3: Structure d'une image de cellule générale

Ici, le critère "cellule" est à 1 ou 2, c'est-à-dire qu'il y a 1 ou 2 cellule dans cette image. Le critère "anneau" est à oui ou non, c'est-à-dire que l'anneau autour de cellule apparait ou non. Le critère "env" est à clair ou vague, c'est-à-dire l'environnement est bien à observé ou bruité par les points verts. Revenir aux graphiques, nous pouvons constater que les résultats sont les mêmes pour PCA et MDS classique, bien qu'ils puissent être décalés par un degré de liberté de rotation en raison du caractère aléatoire sous-jacent des implémentations. Pendant ce temps, l'implémentation scikit-learn utilisant une variante de l'algorithme de métrique MDS appelée SMACOF semble mieux fonctionner que l'implémentation MDS classique précédente. Il y a une division plus claire des pixels en différents clusters.

4.2 Locally linear embedding (LLE)

4.2.1 Introduction générale

La méthode Locally Linear Embedding (LLE) est une méthode de réduction de dimensionnalité non linéaire qui maintient les relations de connectivité entre les points de données dans l'espace d'origine. Ainsi, cette méthode répond à

l'objectif de maintenir la précision des relations entre les points d'image après leur déconvolution.

L'utilité de la méthode LLE est de réduire la dimensionnalité des données pour faciliter l'analyse et l'exploitation des données, minimiser la complexité de calcul et éviter le surajustement dans le modèle. Elle est également utilisée pour découvrir les relations spatiales entre les points de données, afin de mieux comprendre les propriétés des données.

4.2.2 Algorithmes pour LLE

La méthode LLE fonctionne en trouvant des poids linéaires pour décrire les relations entre les points de données dans l'espace d'origine. Ensuite, les points de données sont transformés dans un nouvel espace de dimension réduite, dans lequel les relations de connexion entre les points de données sont conservées. Cela aide à conserver les caractéristiques importantes des données, à réduire le bruit et à accélérer le traitement des données.

Les étapes de la méthode LLE pour réduire la dimension des données sont les suivantes:

Comme nous le savons, nos données sont de forme (11850, 42, 42), ce qui signifie que nous avons 11850 images et chaque image est une matrice carrée de 42x42.

Heureusement, dans la bibliothèque "sklearn.manifold" de Python, il existe une fonction "Locally Linear Embedding" qui nous permet de réaliser rapidement la méthode décrite ci-dessus. L'important est de trouver le paramètre k ("n_neighbors") qui convient le mieux à nos données pour obtenir le résultat le plus approprié.

Commentaires Nous avons constaté que la qualité de la réduction de dimension dépendait fortement du paramètre "k". **Si "k" est trop petit**, certains des relations importantes entre les points de données peuvent être omises, ce qui peut entraîner une perte d'information importante. **Si "k" est trop grand**, la matrice de pondération devient trop complexe et peut contenir du bruit, ce qui peut également entraîner une perte d'information importante.

Il existe certaine méthodes pour déterminer la valeur appropriée de k, telles que la méthode de la validation croisée, etc. Cependant, à ce stade, nous avons utilisé une méthode manuelle en testant plusieurs valeurs de k pour trouver la valeur la plus appropriée.

Nous avons exécuté des tests avec le nombre de voisins k allant de **5 à 30** et nous avons constaté que les valeurs de k autour de 16 nous permettent d'afficher les résultats de la manière la plus raisonnable.

Voici quelques exemples lorsque k est trop petit, k est trop grand et k est approprié :

k=6

k=30

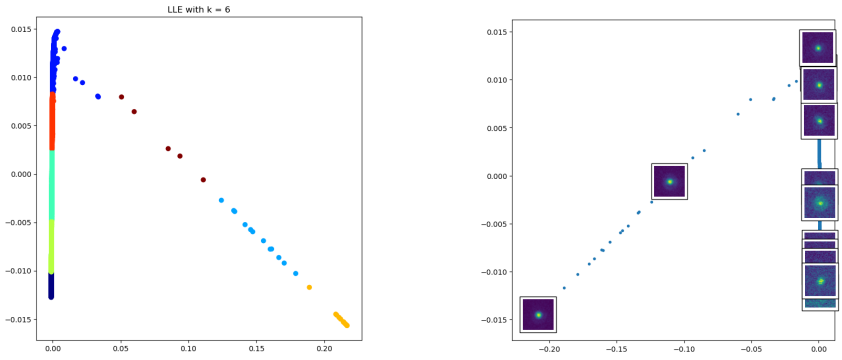


Figure 4: LLE avec k=6

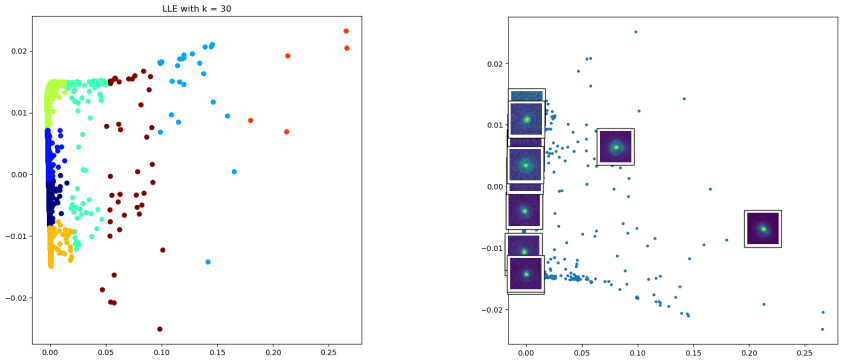


Figure 5: LLE avec k=30

Nous constatons que lorsque k = 5 (faible), les données similaires à des positions éloignées dans le nouveau système de coordonnées, ce qui signifie que certaines caractéristiques importantes des données ont été ignorées ou pas préservées dans le nouveau système, conduisant à une perte de précision de l'ensemble de données d'origine dans le nouveau système.

Et lorsque k = 30 (élevé), les données se chevauchent dans le nouveau système de coordonnées, car avec une valeur élevée de k, nous avons évalué trop de caractéristiques des données, dont certaines sont inutiles et peuvent être supprimées, rendant l'évaluation de l'ensemble de données dans le nouveau système difficile en raison du grand nombre de points similaires qui se chevauchent.

k=16

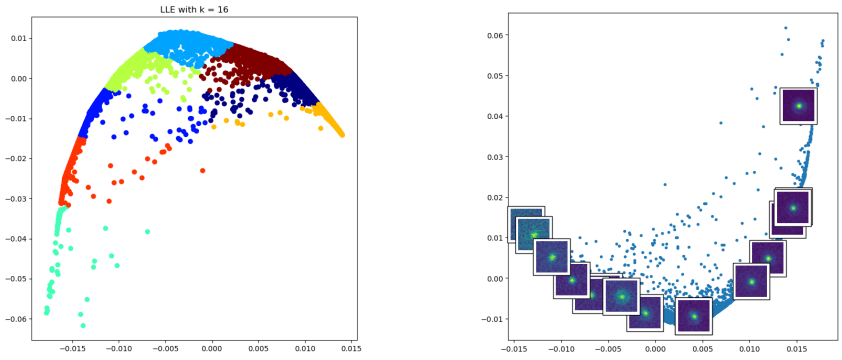


Figure 6: LLE avec k=16

4.3 Isomap

4.3.1 Introduction générale

Isomap (Isometric mapping) est un autre des algorithmes fondamentaux de la réduction de la dimensionnalité non linéaire et de l'apprentissage de variété. Isomap introduit une approche unique pour apprendre la structure géométrique globale, qui est basée sur la représentation d'ensembles de données de grande dimension avec des graphes de voisins les plus proches. L'idée est que les distances que vous parcourez le long des bords du graphique se rapprochent des distances le long de la variété.

En apprenant la métrique intrinsèque définie le long de la variété au lieu de la métrique définie par un système de coordonnées externe, isomap peut apprendre des relations géométriques non linéaires très complexes. De plus, isomap fournit alors une transformation d'intégration dimensionnelle inférieure qui préserve la structure dimensionnelle supérieure.

4.3.2 Algorithme

Entrées : Vecteurs de données $x_1, ..., x_n \in \mathbb{R}^d$ avec le paramètre k-nearest neighbors (les voisinages les plus proches) et $m < d$ (dimension désirée).
Output: Vecteurs après embedding dimensionnel $y_i, ..., y_n \in \mathbb{R}^m$.

Démarche

1. Trouver les k plus proches voisins pour chaque point et créer le graphe G de ces voisinages avec les points de données x_i considérés comme les noeuds et arêtes reliant les voisins les plus proches.

2. Calculer la matrice de similarité géodésique D avec des distances de chemin le plus court par paires le long du graphique (algorithme de Dijkstra)

3. Appliquer la mise à l'échelle multidimensionnelle à la matrice D pour former l'embedding dimensionnel.

4.3.3 Application dans notre projet

Nous importons nos données qui sont les 11850 vecteurs d'images à 1764 dimensions. Par ISOMAP, nous les ramènons à 2 dimensions comme suivant :

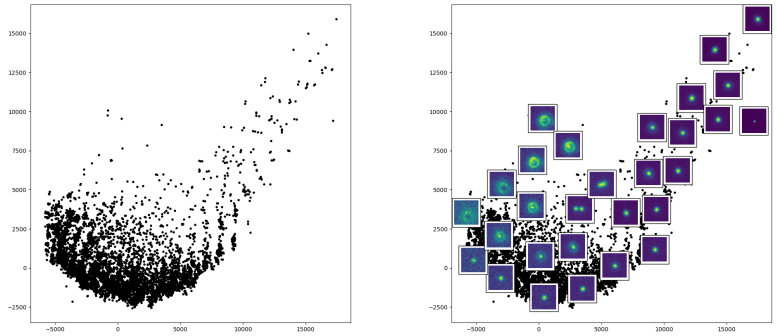


Figure 7: Réduction dimensionnelle par ISOMAP

Nous pouvons trouver que les points des données sont mieux répartis avec ISOMAP que le méthode de LLE car les images s'avèrent bien classées. Les images qui ont les mêmes similarités se rapprochent les une aux autres. Il semble qu'Isomap réussisse assez bien à préserver certaines relations dimensionnelles plus élevées.

References