Projet Manifold Learning

Cong Anh Khann LE, Dan-Azoumi Lawan, Julien Nique

Novembre 2020

1 Introduction

A l'ère du Big Data, les données sont de tout genre : texte, vidéo, images, séries temporelles etc., et celles-ci sont représentées dans des espaces de grande dimension. De tels espaces de représentation posent des problèmes de comparaison et d'interprétation des écarts entre ces données, celles-ci étant plus éparses, et ont tendance à fausser les dissimilarités entre elles : c'est le fléau de la dimension (curse of dimensionality) introduit en 1961 par Bellman.
Une des solutions consiste à remplacer les données originales par des données dans un espace de dimension inférieure ($d < D$), tout en conservant l'essentiel des caractéristiques de celles-ci. Figure 1 montre deux ensembles de données dont la dimension intrinsèque respective est évidente . Les objectifs de ce projet sont d'appliquer des techniques de réduction de dimension linéaire ou non sur des jeux de données simulées, des les calibrer et de les comparer à l'aide de mesures de qualité de conservation de leur structure locale et globale. Enfin nous traiterons le cas d'un jeu de données réelles labellisées sur lequel nous appliquerons, après réduction de la dimension, une méthode d'apprentissage supervisée. Nous comparerons ensuite les scores obtenus.

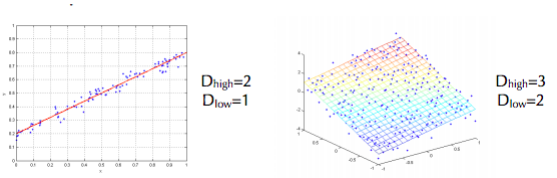

FIGURE 1 – Exemples de sous-variétés linéaires

2 Méthodes de réduction de la dimension

2.1 Classical MDS ou PCA

La méthode du positionnement multidimensionnel, en anglais **Multi Dimensional Scaling** est une méthode de réduction linéaire de la dimension. Étant donnés n points x_i dans un espace euclidien de dimension D, on cherche à les représenter par n points y_i dans un espace de dimension inférieure d en conservant les proximités. En pratique, dans sa formulation classique, cela revient à minimiser la fonction de stress :

$$S(y_1, y_2, ..., y_N) = \sum_{i \neq j} (b_{ij} - \langle y_i, y_j \rangle)^2 \quad (1)$$

où $b_{ij} =< x_i - \overline{x}, x_j - \overline{x} >$ est le terme général de la matrice de similarité B. Cette matrice symétrique peut-être obtenue aussi à partir de la matrice des distances D où $d_{ij} = ||x_i - x_j||$ par double centrage suivant la formule : $B = (I - \frac{1}{N}J)D^2(I - \frac{1}{N}J)$ où J est une matrice carrée de taille N ne contenant que des 1.
Une solution pour le positionnement multidimensionnel est alors de prendre les d colonnes de la matrice $Y = \Lambda_d^{1/2} {E_d}^T$ où ${E_d}^T$ est la transposée de la matrice des d vecteurs propres de B associés aux d plus grandes valeurs propres et Λ_d la matrice diagonale de ces valeurs propres.

2.2 Isomap

Isomap est une méthode non linéaire de réduction de la dimension. Elle est utilisée pour obtenir en basse dimension une représentation quasi isométrique d'un ensemble de points appartenant à un espace de grande dimension. C'est une extension de l'algorithme **MDS** qui tient compte de la géométrie locale de la variété en incorporant un graphe de proximité pondéré par les distances euclidiennes entre les points. La méthode **Isomap** dépend de deux hyperparamètres : le nombre de voisins k à utiliser pour construire le graphe de proximité et la dimension d de l'espace cible. Le nombres de voisins k peut être calibré à l'aide d'une mesure de qualité (voir paragraphe 4). L'algorithme de la méthode **Isomap** est le suivant :

Algorithm 1 Isomap

1. Déterminer les k plus proches voisins de chaque point.
2. Construire un graphe de voisinage où chaque point est connecté à un autre s'il fait parti de ses k plus proches voisins et où chaque arête est pondérée par la distance euclidienne.
3. Calculer le plus court chemin d_{ij} entre deux points i et j quelconques à l'aide de l'algorithme de Dijkstra.
4. Utiliser l'algorithme MDS à partir de la matrice des distances $D = (d_{ij})$ pour obtenir une représentation en dimension m.

2.3 LLE

La méthode **LLE**, acronyme de **Local Linear Embedding**, est une méthode non linéaire de réduction de la dimension. Elle s'appuie sur une représentation locale linéaire des points de l'espace des données initiales. Plus précisément, chaque point x_i est décrit comme une combinaison linéaire de ses k plus proches voisins :

$$x_i = \sum_{j \in \mathcal{V}_i^k} w_{ij} x_j \quad \text{avec} \quad \sum_{j \in \mathcal{V}_i^k} w_{ij} = 1$$

La matrice carrée des poids W s'obtient en minimisant la quantité :

$$\sum_{i=1}^{n} ||x_i - \sum_{j \in \mathcal{V}_i^k} w_{ij} x_j||^2$$

La représentation Y des points en faible dimension d s'obtient ensuite en minimisant la fonction :

$$f(Y) = \sum_{i=1}^{n} ||y_i - \sum_{j=1}^{k} w_{ij} y_{i_j}||^2$$

avec $Y^T \times Y = 1$.
La solution est donnée analytiquement en calculant les d vecteurs propres associés aux d plus petites valeurs propres non nulles de la matrice $(I - W)^T (I - W)$.

2.4 t-SNE

L'intégration de voisin stochastique (SNE ; Hinton et Roweis, 2003) est une technique qui minimise la divergence de Kullback-Leibler des similitudes mises à l'échelle des points i et j dans un espace dimensionnel élevé, p_{ij}, et dans un espace de dimension réduite, q_{ij} :

$$KL(P||Q) = \sum_{i,j=1}^{n} p_{ij} \log \left(\frac{p_{ij}}{q_{ij}} \right) \qquad (2)$$

SNE utilise un noyau gaussien pour calculer les similitudes. L'intégration stochastique de voisinage t-distribuée (t-**SNE** ; van der Maaten et Hinton, 2008) améliore **SNE** en utilisant une distribution de Student t comme noyau dans l'espace de faible dimension. En raison de la distribution t à queue épaisse, la méthode t-**SNE** maintient mieux les voisinages locaux des données et pénalise les incursions

erronées de différents points. Cette propriété la rend particulièrement appropriée pour représenter en clusters les données et structures complexes en faible dimension. La méthode t-**SNE** a un paramètre, la perplexité, à régler qui est une estimation du nombre de voisins proches pour chaque point.

2.5 Le package dimRed

Le package R **dimRed** implémente la plupart des techniques classiques de réduction ainsi que des indicateurs de qualité de la conservation des distances, utiles pour la calibration et la comparaison de ces techniques (voir section 4). Nous utiliserons ce package pour les expérimentations.

3 Données

3.1 Données artificielles

Les quatre jeux de données que nous avons choisis de simuler sont des sous-variétés de l'espace euclidien. La représentation 3D de ces données est illustrée dans Figure 2. Le premier, **Carpet**, a une dimension intrinsèque égale à 1 alors que les trois suivants : **Swissroll**, **Cup** et **Heart** ont une dimension intrinsèque égale à 2. Chaque jeu de données contient 1000 ou 2000 points. Nous les avons choisis pour la diversité de leur topologie, et leur structure globale plus ou moins complexe. Par exemple le dataset **Carpet** a une dimension intrinsèque plus petite que les autres dataset.

3.2 Données réelles

Pour le jeu de données réelles, nous avons choisi le dataset **Iris**. Ce dataset, certes classique, est bien adapté à notre problématique tant pour la réduction de sa dimension que pour la classification.

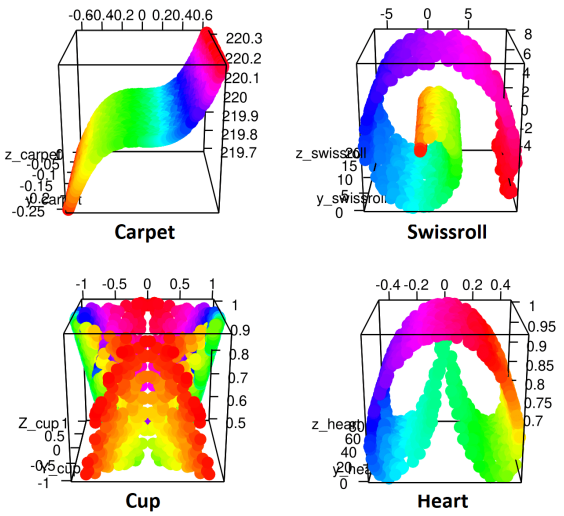

FIGURE 2 – Données artificielles en dimension $D = 3$

Il contient quatre variables descriptives : largeur et longueur du pétale et du sépale de l'espère iris et une variable catégorielle cible décrivant l'espèce d'iris auquel appartient chacune des 150 observations.

4 Outils pour la calibration et la comparaison des méthodes de réduction

Afin de calibrer les hyperparamètres de chaque méthode de réduction de la dimension pour un jeu de données, nous utiliserons :

1. une estimation de la dimension intrinsèque du dataset ;
2. une ou plusieurs mesures de qualité associées à la conservation de la structure locale ou globale du dataset.

Les mesures de qualité seront aussi utilisées pour comparer entre-elles les techniques de réduction sur les jeux de données correctement calibrés.

4.1 Dimension intrinsèque

La dimension intrinsèque est déterminée par la méthode décrite par P. Desmartines (1994). Cette méthode est basée sur le fait que dans une distribution uniforme, un nombre $C(\epsilon)$ de points à l'intérieur d'une sphère de rayon ϵ est proportionnel au volume, donc à ϵ^d, où d est la dimension locale de la distribution (dimension fractale). Le principe de cette méthode se résume comme suit :

1. Calculer les distances pour toutes les paires de points.
2. Créer une liste de distances dans l'ordre croissant ϵ dont la valeur est entre la distance minimum et maximum entre les points.
3. Déterminer la nombre C_i de distances qui sont inférieures ϵ_i.
4. Tracer la caractéristique $logC$ en fonction de $log\epsilon$.
5. La pente p trouvée par régression linéaire de cette caractéristique correspond à la dimension intrinsèque.

4.2 Mesures de qualité basées sur la matrice de co-classement

La matrice de co-classement $\mathcal{Q}$, en anglais **co-ranking matrix**, permet de capturer les changements de distances ordinales entre les espaces de haute et basse dimension. Son terme général q_{kl} est donné par :

$$q_{kl} = \text{card}\{(i,j) : \hat{r}_{i,j} = k \text{ et } r_{i,j} = l\}$$

q_{kl} comptabilise le nombre de points à distance de rang l devenant de rang k après réduction de la dimension. On peut calculer la métrique suivante qui mesure le nombre de points appartenant aux k plus proches voisins à la fois en haute dimension et en basse dimension :

$$Q(k) = \frac{1}{kn} \sum_{i=1}^{k} \sum_{j=1}^{k} q_{ij}$$

Une mesure de la préservation locale des distances peut être alors définie par :

$$Q_{local} = \frac{1}{k_{max}} \sum_{k=1}^{k_{max}} Q(k)$$

où k_{max} est la valeur de k pour laquelle $Q(k) - \frac{k}{n-1}$ est maximum. Et une mesure de la préservation globale des distances par :

$$Q_{global} = \frac{1}{n - k_{max}} \sum_{k=kmax}^{n-1} Q(k)$$

4.3 Corrélation cophénétique

La corrélation cophénétique calcule la corrélation entre les triangles supérieurs et inférieurs des matrices de distance en haute et basse dimension.
Les mesures de qualité Q_{local} et Q_{global} et la corrélation cophénétique se trouvent dans le package R **dimRed**.

5 Expériences numériques sur les données artificielles

Dans cette section, nous avons effectué la calibration de différentes techniques de réduction de dimension sur les jeux de données artificielles, en utilisant une estimation de leur dimension intrinsèque, sur des hyperparamètres sélectionnés. Les techniques de réduction de dimension et les hyperparamètres sont présentés dans le tableau 1. Les performances de la réduction de dimension ont été évaluées par trois mesures de qualité : (Q_{local}, Q_{global} et la corrélation cophénétique). Notre étude présente se concentre sur 2 points principaux :

1. l'effet des mesures de qualité sur le résultat de la calibration de différentes techniques de réduction de dimension sur l'ensemble de données **Cup** ;

2. la performance de chaque technique de réduction de dimension sur tous les ensembles de données artificielles en utilisant Q_{local} comme mesure de qualité.

Tableau 1 – Réglages des hyperparamètres pour les expériences ; k : nombre de plus proches voisins, p : perplexité.

Technique	Hyperparamètre
PCA	None
Isomap	$5 \leq k \leq 100$
LLE	$5 \leq k \leq 100$
t-SNE	$1 \leq p \leq 100$

5.1 Effet des mesures de qualité sur la calibration de différentes méthodes de réduction de dimension

Nous avons fait le choix d'illustrer par Figure 3 les scores de trois mesures de qualité, à savoir Q_{local}, Q_{global}, et la corrélation cophénétique sur le jeu de donnée **Cup** en fonction de quatre techniques de réduction. On constate qu'une méthode de réduction peut être la meilleure pour un critère donné tel que la mesure Q_{local}, tandis que cette même méthode pourrait être la moins performante si on venait à considérer un autre critère. De plus les hyperparamètres optimaux pour chaque méthode de réduction diffèrent selon les mesures de qualité comme le montre Tableau 2. Par conséquent, pour obtenir des résultats souhaitables, une mesure de qualité appropriée doit être utilisée en fonction de l'application (Kraemer et *al.*, 2019). Dorénavant nous utiliserons la mesure de qualité Q_{local} pour calibrer nos modèles.

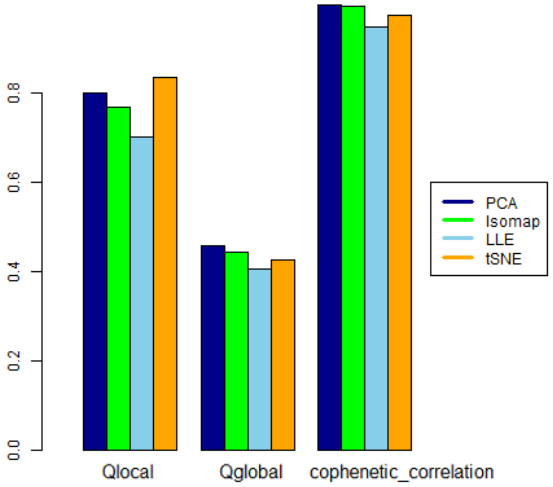

FIGURE 3 – Q_{local}, Q_{global}, corrélation cophénétique pour différentes techniques de la réduction de dimension du jeu de données **Cup**.

5.2 Comparaison de la qualité de la réduction de dimension des différents techniques

Q_{local} est la mesure de qualité selon laquelle nous calibrerons chaque méthode de réduction pour un hyperparamètre différent, à savoir le nombre de voisins k pour **Isomap** et **LLE**, et la perplexité pour t**SNE**.
Tableau 3 résume le Q_{local} en fonction de la technique de réduction de dimension sur les jeux de données artificiels. Les données en faible dimension générées par les meilleures techniques sont représentées dans Figure 4. Swissroll est un jeu de données intéressant par son aspect (rouleur, Figure 2) et la non-linéarité de sa structure. On remarquera que, la méthode **PCA** ne performe pas sur ce jeux de données, et la méthode optimale est **Isomap** qui tient compte de la distance géodésique contrairement à la distance euclidienne qui biaiserait les distances réelles entre les observations.
Au contraire, pour le dataset **Carpet**, qui

Tableau 2 – Qualité de la réduction de dimension en fonction de la mesure de qualité et de la technique de réduction de dimension sur le jeu de données **Cup**. La valeur de l'hyperparamètre correspondant est indiquée entre parenthèses. k : nombre de plus proches voisins ; p : perplexité.

Mesures de qualité	PCA	Isomap	LLE	t-SNE
Q_{local}	0.804	0.782 (k=100)	0.653 (k=60)	**0.834** (p=60)
Q_{global}	**0.464**	0.443 (k=30)	0.372 (k=100)	0.441 (p=100)
Corrélation cophénétique	**0.998**	0.993 (k=100)	0.874 (k=100)	0.986 (p=80)

Tableau 3 – Q_{local} en fonction de la technique de réduction de dimension sur les jeux de données artificiels.

Méthodes	PCA	Isomap	LLE	t-SNE
Carpet	0.540	0.545 (k=10)	0.548 (k=20)	**0.682** (p=40)
Swissroll	0.493	**0.940** (k=20)	0.737 (k=10)	0.854 (p=50)
Cup	0.804	0.782 (k=100)	0.653 (k=60)	**0.834** (p=60)
Heart	**0.937**	0.925 (k=100)	0.787 (k=10)	0.856 (p=20)

a une dimension intrinsèque égale 1, **PCA** montre une performance comparable avec **Isomap** et **LLE** dont la paramètre k optimale relativement élevée. Comme la donnée n'est pas complètement linéaire, la qualité de ces techniques reste relativement faible. Par ailleurs, la technique non-linéaire t-**SNE** permet d'améliorer la performance de la réduction de dimension.
Concernant Cup, bien que le dataset soit non-linéaire, PCA résulte d'un score Q_{local} plus élevé que Isomap et LLE qui préservent les distances locales. Ce résultat peut être expliqué par le fait que **Isomap** et **LLE** échouent lorsqu'ils sont confrontés à des manifolds contenant des trous (Laurens et *al.*, 2009). En revanche, t-**SNE**, qui se base sur une interprétation probabiliste des proximités, s'agit du meilleur technique. Sur la figure en 2D (Figure 4), on constate effectivement que t-**SNE** préserve bien la distance dans l'espace de petite dimension.
La dataset **Heart** est original au sens où, malgré sa structure non linéaire (Figure 2), il obtient le meilleur score avec la méthode **PCA**. Seul **Isomap**, avec un grand nombre de voisins (100) rivalise avec la méthode **PCA** alors que les autres techniques de réduction non linéaires échouent.

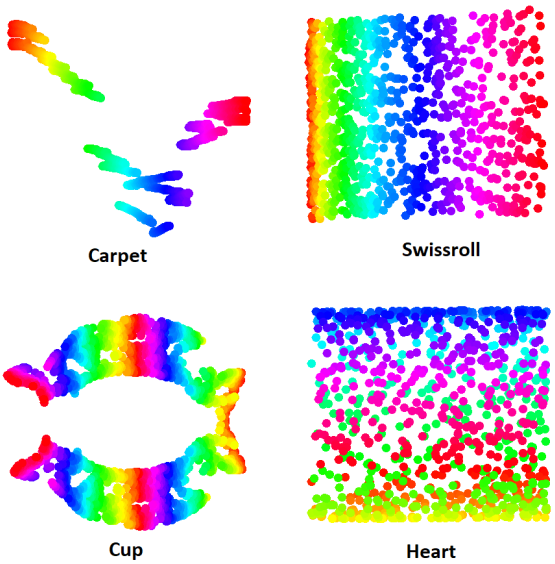

FIGURE 4 – Donnée artificielles, représentées en dimension 2, obtenues par la meilleure technique de réduction de dimension.

6 Expériences numériques sur des données réelles

Dans cette section, nous avons appliqué d'abord la réduction de dimension avec calibration sur les quatre premières colonnes du jeu de donnée **Iris**, comme fait précédemment pour les jeux de données artificielles. Puis nous avons effectué la classification sur le jeux de données **Iris** ainsi que sur les ensembles des données obtenus en faible dimension pour évaluer comment la réduction de dimension avec des techniques différentes (**PCA**, **Isomap**, **LLE**, *t*-**SNE**) peut affecter la performance de la classification par la régression logistique.

La dimension intrinsèque de l'ensemble de données **Iris** s'est avérée être égale à 2. Le meilleur score Q_{local} correspondant à chaque technique de réduction de dimension a été résumé dans le tableau 4. On peut voir que la méthode *t*-**SNE** a donné un meilleur score par rapport aux autres techniques. **PCA** et **Isomap** ont donné un résultat comparable tandis que **LLE** a obtenu le score le plus bas.

Les graphiques montrant les jeux de données en faible dimension résultants des quatre techniques sont représentés sur 5. Les points ont des couleurs basées sur les espèces correspondantes. On constate qu'il existe une bonne séparation entre les classes pour toutes les techniques. Cependant, dans les données réduites avec **LLE**, les variances intra-classe semblent plus faibles par rapport à celles des données réduites par les autres techniques, ce qui concorde avec un score de la mesure Q_{local} plus faible de cette technique par rapport aux autres. Ensuit, le jeu de données **Iris** ainsi que ceux en faible dimension obtenus par les quatre techniques ont été utilisés pour la classification par régression logistique. Chaque ensemble de données a été séparé en un de jeu de données d'entraînement et un jeu de test de la même manière, ce qui garantit que les observations correspondantes sont les mêmes. Après avoir entraîné le modèle avec le jeu de données d'entraînement, les erreurs de classification ont été calculées en appliquant le modèle entraîné sur le jeu de données de test. Le résultat est résumé dans le tableau 1. On peut noter que le **tSNE** qui a donné la meilleure performance de réduction de dimension est la seule technique ayant abouti à l'amélioration de la performance de la classification. En revanche, **LLE**, qui présentait la valeur *Qlocal* la plus basse, a donné les mêmes performances de classification que l'ensemble de données d'origine, mais mieux que celles d'**Isomap** et de **LLE**. Ces résultats suggèrent qu'il n'y aurait pas de corrélation directe entre la performance des techniques de réduction de dimension et la performance de la classification.

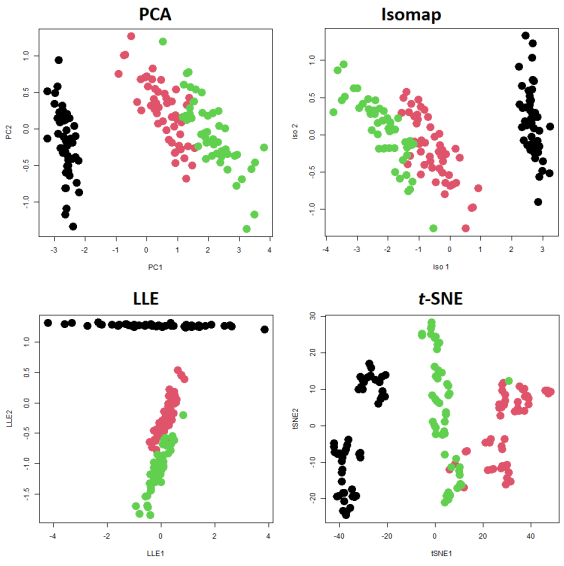

FIGURE 5 – Tracé du jeu de données **Iris** après réduction de la dimension par de différentes techniques avec calibration.

Tableau 4 – Q_{local} et erreurs de la classification par régression logistique sur le jeu de données **Iris** en fonction de la technique de réduction de dimension.

Méthode	Sans réduction	PCA	Isomap	LLE	t-SNE
Q_{local}	-	0.653	0.652	0.593	**0.853**
Taux d'erreur	0.04	0.06	0.06	0.04	**0.02**

7 Conclusion

Dans cette étude, nous avons appliqué différentes techniques de réduction de dimension (**PCA**, **Isomap**, **LLE**, ***t*-SNE**) sur quatre jeu de données simumlées et un jeu de données réelles (**Iris**) en utilisant la dimension intrinsèque et des hyperparamètres sélectionnés. La dimension intrinsèque a été estimée sur la base de la dimension fractale. L'optimisation des hyperparamètres a été réalisée sur la base d'une mesure de qualité (Q_{local}, Q_{global} ou la corrélation cophénétique).
Les résultats de l'étude sur les ensembles de données artificielles ont montré que les meilleurs hyperparamètres et la meilleure technique pour un ensemble de données en particulier varient selon la mesure de qualité utilisée. De plus, on a constaté que la qualité de réduction de la dimension d'une technique varie entre des ensembles de données ayant des caractéristiques différentes (ex. forme, dimension intrinsèque). Par conséquent, une réduction de la dimension réussie d'un ensemble de données nécessite de trouver une technique appropriée dont les hyperparamètres sont calibrés à l'aide d'une mesure de qualité adéquate qui dépend de l'application.
Enfin, nous avons appliqué notre approche de calibration pour les techniques de réduction de dimension sur l'ensemble de données bien connu **Iris**, puis nous avons effectué la classification par la régression logistique sur des ensembles de données de haute et de faible dimension. Le ***t*-SNE**, qui a donné la meilleure qualité de réduction de dimension a aussi obtenu le meilleur score en classification. Cependant, en général, il ne semble pas y avoir de corrélation entre la performance de réduction de dimension d'une technique et le taux d'erreur en classification pour l'ensemble de données de faible dimension qui en résulte.

8 Références

1. Laurens van der Maaten, Eric Postma, Jaap van den Herik. Dimension Reduction : A Comparative Review. Tilburg University.
2. Demartines,P. (1994). Analyse de données par réseaux de neurones auto-organisés.
3. Kraemer, G., Reichstein, M., et Mahecha, M. D. (2018). dimRed and coRanking—Unifying dimensionality reduction in R. R Journal, 10(1), 342-358.
4. Lawrence K. Saul. An Introduction to Locally Linear Embedding. ATetT Labs – Research.
5. Roweis, S. and Saul, L. (2000). Nonlinear dimensionality reduction by locally linear embedding. Science, 290(5500) :2323–2326.
6. Saul, L. and Roweis, S. (2002). Think globally, fit locally : unsupervised learning of low dimensional manifolds. Journal of Machine Learning Research, 4 :119–155.
7. Scholkopf, B., Smola, A., and M ¨ uller, K.-R. (1998). Nonlinear component analysis as a kerneleigenvalue problem. Neural Computation, 10 :1299–1319