**Avant UMAP** :

Historiquement, t-SNE et LargeVis ont connu une amélioration spectaculairedans la création et la préservation de la structure locale dans les données. est peut être vu qualitativement en comparant leurs intégrations à celles générées par Laplacian Cartes propres et APC

**Résultats de l’UMAP** :

Après quelques illustrations avec quelques jeux de données à savoir :

1. Pen digits [1, 10] est un ensemble d’images en niveaux de gris de 1797 de chiffres saisis à l’aide d’une tablette numérisée. Chaque image est une image 8x8 que nous traitons comme un seul 64 vecteur dimensionnel, supposé être dans l’espace vectoriel euclidien.
2. COIL 20 [43] est un ensemble de 1440 images en niveaux de gris composé de 20 objets sous 72 rotations différentes s’étendant sur 360 degrés. Chaque image est un 128x128 image que nous traitons comme un seul vecteur de dimension 16384 aux fins de calculer la distance entre les images.
3. COIL 100 [44] est un ensemble de 7200 images couleur composé de 100 objets sous 72 rotations différentes s’étendant sur 360 degrés. Chaque image se compose de 3 Matrices d’intensité 128x128 (une pour chaque couche de couleur). Nous traitons cela comme un seul vecteur de dimension 49152 aux fins du calcul de la distanceentre les images.
4. Le scRNA-seq de souris [11] est obtenu avec des données d’expression génique pour 20 921 cellules d’une souris adulte. Chaque échantillon est constitué d’un vecteur de 26 774 mesures.
5. atlog (Shuttle) [35] est un ensemble de données de la NASA composé de diverses données associées aux positions des radiateurs dans le shule spatial, y compris un horodatage.
6. MNIST [32] est un jeu de données d’images en niveaux de gris de 28x28 pixels de handwrien Chiffres. ere sont des classes de 10 chiffres (0 à 9) et 70000 images au total est est traité comme 70000 différents vecteurs de dimension 784.
7. F-MNIST [63] ou Fashion MNIST est un ensemble de données d’images en niveaux de gris de 28x28 pixels d’articles de mode (vêtements, chaussures et sacs). ere sont 10 classes et 70000 images au total. Comme avec MNIST, cela est traité comme 70000 différents Vecteurs dimensionnels 784.
8. La cytométrie en flux [51, 9] est un ensemble de données de mesures cytométriques des cellules CDT4 comprenaient 1 000 000 d’échantillons, chacun avec 17 mesures.
9. GoogleNews word vectors [41] est un jeu de données de 3 millions de mots et d’expressions dérivé d’un échantillon de documents Google Actualités et intégré dans un 300 espace dimensionnel via word2vec.

NB : Pour tous les jeux de données à l’exception de GoogleNews, nous utilisons la distance euclidienne entre les vecteurs. Pour GoogleNews, selon [41], nous utilisons la distance cosinus (ou distance angulaire en t-SNE qui prend en charge les distances non métriques, contrairement à UMAP).

Les auteurs ont conclu que l’UMAP est beaucoup plus puissante lorsque les données sont volumineuses car par exemple pour la visualisation elle arrive à bien former les différentes classes tandis que lorsque les données ne sont pas nombreuses elle a dit mal en général de bien faire la classification.

Les hyper-paramètres n-dist = distance entre les vosins et n-neigh = nombre de voisins sont mises en œuvre pour de meilleures illustrations.

Retenons :

Nous affirmons que la qualité des intégrations produit par UMAP est comparable à t-SNE lorsqu’il est réduit à deux ou trois taille. Par exemple, la figure 4 montre les intégrations UMAP et t-SNE des ensembles de données COIL20, MNIST, Fashion MNIST et Google Actualités. Alors que les intégrations précises sont différentes, UMAP distingue les mêmes comme t-SNE et LargeVis.

**Efficacité pratique**

On peut soutenir que l’UMAP a capturé plus de la structure globale et topologique des ensembles de données que t-SNE [4, 62]. Plus de boucles dans Les jeux de données COIL20 sont conservés intacts, y compris les boucles entrelacées. De la même façon les relations globales entre différents chiffres dans l’ensemble de données de chiffres MNIST sont plus clairement capturés avec 1 (rouge) et 0 (rouge foncé) aux coins les plus éloignés de l’espace d’intégration, et 4,7,9 (jaune, vert de mer et violet) et 3,5,8 (orange, chartreuse et bleu) séparés en touffes distinctes de chiffres similaires.

**Comparaison quantitative de plusieurs algorithmes**

Dans l’ensemble de données Fashion MNIST, la distinction entre les vêtements (rouge foncé, jaune, orange, vermillon) et chaussures (chartreuse, vert de mer et violet) est plus clair. Enfin, alors que t-SNE et UMAP capturent des groupes de vecteurs de mots similaires, l’intégration UMAP prouve sans doute un structure globale entre les différents groupes de mots.

L’UMAP fournit une offre largement comparable performance dans l’intégration de la qualité à t-SNE et LargeVis à l’échelle locale, mais fonctionne nettement mieux que t-SNE ou LargeVis à des échelles non locales.

**Stabilité d'enrobage**

Précision du classificateur kNN pour différentes valeurs de k sur les espaces d'intégration des ensembles de données COIL-20 et PenDigits. Des scores de précision sont donnés pour chaque pli d'une validation croisée de 10 fois pour chacun des PCA, Laplacian Eigenmaps, LargeVis, t-SNE et UMAP. Nous notons que UMAP produit des scores de précision compétitifs pour t-SNE et LargeVis dans la plupart des cas, et surpasse à la fois t-SNE et LargeVis pour des valeurs k plus élevées sur COIL-20.

Une illustration de cette comparaison est faite sur le jeu de données Wine est disponible sur Github.

\subsection{T-Distributed Stochastic Neighbor Embedding (T-SNE)}

( Intégration de voisin stochastique)

T-SNE met l'accent sur la modélisation de points de données dissemblables au moyen de

grandes distances par paires, et la modélisation de points de données similaires au moyen de petites distances par paires. De plus, en raison de ces caractéristiques de la fonction de coût t-SNE (et en raison de l'invariance d'échelle approximative de la distribution t de Student), l'optimisation de la fonction de coût t-SNE est beaucoup plus facile que l'optimisation de les fonctions de coût de SNE et UNI-SNE. Plus précisément, t-SNE introduit des forces à longue portée dans la carte à faible dimension qui peuvent rapprocher deux (groupes de) points similaires qui sont séparés au début de l'optimisation. Nous commençons par présenter une procédure de descente de gradient relativement simple pour optimiser lafonction de coût t-SNE.

La première astuce, que nous appelons "compression précoce", consiste à forcer les points de la carte à rester proches les uns des autres au début de l'optimisation. La compression précoce est mise en oeuvre en ajoutant une pénalité L2 supplémentaire à la fonction de coût qui est proportionnelle à la somme des carrés distances des points de la carte à partir de l'origine. L'amplitude de ce terme de pénalité et l'itération à laquelle il est supprimé sont définies manuellement,mais le comportement est assez robuste à travers les variations de ces deux paramètres d'optimisation supplémentaires.\\

Une manière moins évidente d'améliorer l'optimisation, que nous appelons « exagération précoce », consiste à multiplier tous les pi j par, par exemple, 4, dans les premières étapes de l'optimisation. Cela signifie que presque tous les $q\_{ij}$, qui totalisent encore 1, sont beaucoup trop petits pour modéliser leurs $p\_{ij}$ correspondants(voir annex pour les formules )\\

Pour évaluer le t-SNE, nous présentons des expériences dans lesquelles le t-SNE est comparé à sept autres techniques non paramétriques de réduction de la dimensionnalité. En raison des contraintes d'espace, dans l'article, nous comparons uniquement t-SNE avec: (1) la cartographie de Sammon, (2) l'Isomap et (3) la LLE. Dans le matériel de support, nous comparons également t-SNE avec : (4) CCA, (5) SNE, (6) MVU et (7) Laplacian Eigenmap

Dans toutes nos expériences, nous commençons par utiliser l'ACP pour réduire la dimensionnalité des données à 30.

Cela accélère le calcul des distances par paires entre les points de données et supprime un peu de bruit sans déformer

gravement les distances entre les points.\\

Nous utilisons ensuite chacune des techniques de réduction de dimensionnalité pour convertir la représentation à 30 dimensions en une carte à deux dimensions et nous montrons la carte résultante sous

forme de nuage de points. Pour tous les ensembles de données, il existe des informations sur la classe de chaque point de données, mais les informations de classe ne sont utilisées que pour sélectionner une couleur et/ou un symbole pour les points de la carte.\\

\section{Annexes}

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

$q\_{ij} = \frac{(1 + || y\_{i} - y\_{j}||^2)^{-1}}{\sum \limits\_{\underset{k \neq l}}^n(|| y\_{k} - y\_{l}||^2)^{-1} }$

1.Les performances du t-SNE sur les tâches générales de réduction de la dimensionnalité ne sont pas claires:Il n'est pas évident de savoir comment t-SNE se comportera sur la tâche plus générale de réduction de la dimensionnalité (c'est-à-dire lorsque la dimensionnalité des données n'est pas réduite à deux ou trois, mais à d > 3 dimensions).\\

2.La nature relativement locale du t-SNE le rend sensible à la malédiction de la dimensionnalité intrinsèque des données: Le t-SNE réduit la dimensionnalité des données principalement en fonction des propriétés locales des données, ce qui rend le t-SNE sensible à la malédiction de la dimensionnalité intrinsèque des données (Bengio, 2007). Dans les ensembles de données avec une dimensionnalité intrinsèque élevée et une variété sous-jacente qui varie fortement, l'hypothèse de linéarité locale sur la variété que t-SNE fait implicitement (en utilisant des distances euclidiennes entre voisins proches) peut être violée. En conséquence, t-SNE pourrait avoir moins de succès s'il est appliqué sur des ensembles de données avec une dimensionnalité intrinsèque très élevée (par exemple, une étude récente de Meytlis et Sirovich (2007) estime l'espace des images de visages à constituer de environ 100 dimensions).\\

3.La t-SNE n'est pas assuré de converger vers un optimum global de sa fonction de coût :\\

Une propriété intéressante de la plupart des techniques de réduction de dimensionnalité de pointe (telles que la mise à l'échelle classique, Isomap, LLE et les cartes de diffusion) est la convexité de leurs fonctions de coût. Une faiblesse majeure de t-SNE est que la fonction de coût n'est pas convexe, ce qui nécessite de choisir plusieurs paramètres d'optimisation. Les solutions construites dépendent de ces choix de paramètres d'optimisation et peuvent être différentes chaque fois que t-SNE est exécuté à partir d'une configuration aléatoire initiale de points de carte.\\