TP1 – Suite
Méthodes de réduction de dimensionnalité
t-SNE, UMAP

Acquis d'apprentissage
- Comprendre les principes fondamentaux de plusieurs méthodes de réduction de dimension,
- Visualiser leurs résultats sur des jeux de données synthétiques,
- Comparer leurs performances, avantages et limites selon le type de données,
- Acquérir une intuition sur les cas d'usage adaptés à chaque méthode.

Vous utiliserez **Python** et les bibliothèques `scikit-learn`, `matplotlib`, `umap-learn` et/ou `seaborn`. Ce TP s'appuie sur des jeux de données classiques (Swiss Roll, Moons, Données linéaires) pour évaluer les méthodes suivantes :

1. PCA (Principal Component Analysis)
2. t-SNE (t-distributed Stochastic Neighbor Embedding)
3. UMAP (Uniform Manifold Approximation and Projection)

Partie 1 – Génération et visualisation des données

1. Générez les jeux de données suivants :
   - Données Swiss Roll (3D, non linéaires)
   - Données Moons (2D, non linéaires)
   - Données multivariées simulées (gaussiennes ou lignes corrélées, 2D ou 3D)
2. Visualisez les jeux de données en utilisant matplotlib (en 2D ou 3D selon le cas).

Exemple code

```
from sklearn.datasets import make_swiss_roll, make_moons, make_classification
import matplotlib.pyplot as plt

# Swiss Roll
X_swiss, color_swiss = make_swiss_roll(n_samples=1000, noise=0.05)
# Moons
X_moons, y_moons = make_moons(n_samples=500, noise=0.1)
# Données linéaires
X_linear, y_linear = make_classification(n_samples=500, n_features=5, n_informative=2, n_redundant=0, n_clusters_per_class=1)

# Affichage Swiss Roll 3D
fig = plt.figure(figsize=(6, 5))
ax = fig.add_subplot(111, projection='3d')
ax.scatter(X_swiss[:, 0], X_swiss[:, 1], X_swiss[:, 2], c=color_swiss, cmap='Spectral')
plt.title('Swiss Roll')
plt.show()
```

Ce code utilise `make_swiss_roll` (scikit-learn >= 1.1) pour le Swiss Roll.

Partie 2 – Application des méthodes

Pour chaque jeu de données, appliquez successivement les trois méthodes suivantes :

1.PCA

- Affichez la projection sur les deux premières composantes principales.
- Calculez la variance expliquée cumulée.
- Commentez la capacité de PCA à préserver la structure des données.

2. t-SNE

- Appliquez t-SNE avec une perplexité de 30, puis comparez les résultats avec d'autres valeurs (5, 10, 50).
- Discutez de l'effet du paramètre perplexity.
- Observez si les clusters sont bien séparés.

3. UMAP

- Appliquez UMAP avec n_neighbors=15 et min_dist=0.1, puis testez d'autres paramètres.
- Comparez les résultats avec t-SNE (temps de calcul, séparation visuelle, stabilité).
- Commentez la forme et la fidélité de la représentation.

Partie 3 – Questions de compréhension

1. Quelle méthode vous semble la plus adaptée pour la visualisation **?**
2. Laquelle privilégieriez-vous pour réduire la dimension avant un modèle supervisé **?**
3. Comment choisir entre t-SNE et UMAP dans une tâche réelle ?
4. En quoi la linéarité d'une méthode influence-t-elle ses résultats ?