L05: PCA & t-SNE

Dimensionality Reduction for Visualization and Preprocessing

Methods and Algorithms

Spring 2026

Outline

Learning Objectives

By the end of this lecture, you will be able to:

1. **Derive** PCA from the variance maximization principle and prove the SVD–PCA equivalence
2. **Evaluate** dimensionality reduction methods (PCA vs. t-SNE vs. UMAP) for a given dataset
3. **Analyze** the effect of hyperparameters (perplexity, learning rate) on t-SNE embeddings
4. **Critique** PCA assumptions and limitations for nonlinear financial data (e.g., yield curves)

Finance Application: Portfolio risk decomposition, yield curve analysis, asset clustering

---

Bloom's Level 4–5: Analyze, Evaluate, Create

The Business Problem

Curse of Dimensionality

- Portfolio with 100+ assets: hard to visualize relationships
- Customer data with dozens of features: redundant information
- High dimensions cause sparsity and computational issues

Solutions

- **PCA**: Linear projection preserving maximum variance
- **t-SNE**: Non-linear embedding preserving local structure

Reduce dimensions while preserving important information

Key Equations

Covariance Matrix (from mean-centered data X_c):

$$C = \frac{1}{n-1} X_c^\top X_c$$

Eigendecomposition: $C\, v_k = \lambda_k\, v_k$ (principal directions & variances)
Explained Variance Ratio:

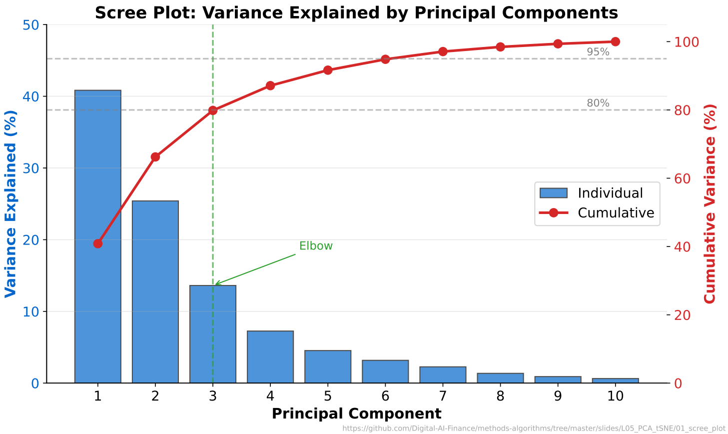
$$\mathrm{EVR}_k = \frac{\lambda_k}{\sum_{j=1}^{p} \lambda_j}$$

t-SNE High-Dimensional Similarity:

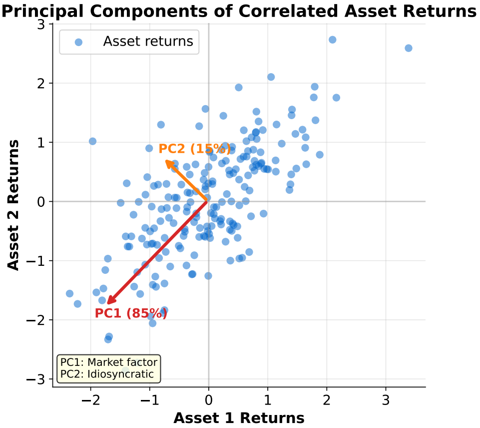
$$p_{j|i} = \frac{\exp\left(-\|x_i - x_j\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|x_i - x_k\|^2 / 2\sigma_i^2\right)}$$

PCA: linear eigen-problem; t-SNE: probabilistic neighbor embedding

Scree Plot: Choosing Components



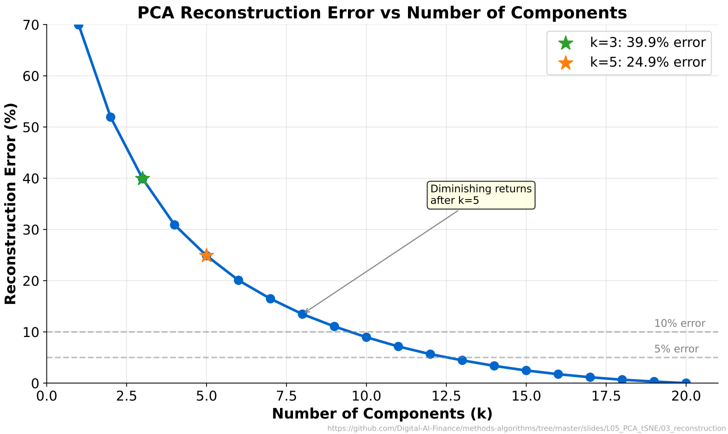
Choose k components capturing 80-95% of variance, or at the "elbow"



https://github.com/Digital-AI-Finance/methods-algorithms/tree/master/slides/L05_PCA_tSNE/02_principal_components

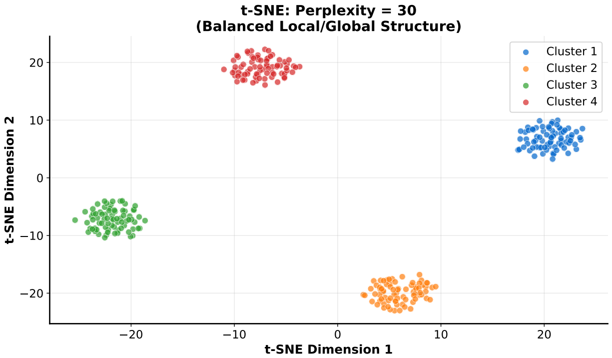
Principal components are orthogonal directions of maximum variance

Reconstruction Error



More components = lower error, but diminishing returns after elbow

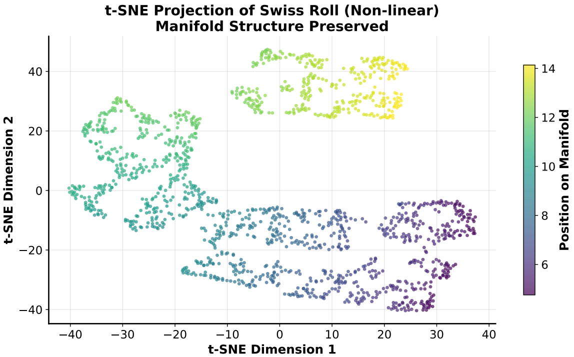
t-SNE: Perplexity Effect



https://github.com/Digital-AI-Finance/methods-algorithms/tree/master/slides/L05_PCA_tSNE/04b_tsne_perplexity_30

Perplexity controls local vs global structure preservation (try 5-50)

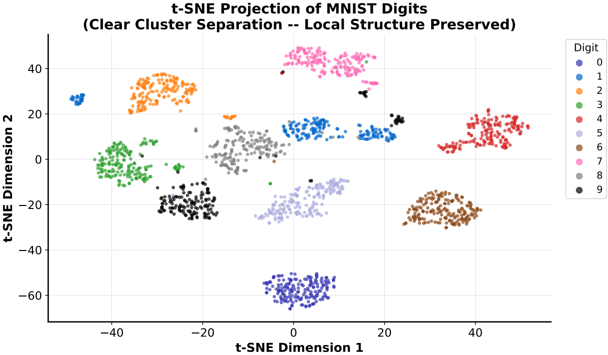
PCA vs t-SNE: Swiss Roll



https://github.com/Digital-AI-Finance/methods-algorithms/tree/master/slides/L05_PCA_tSNE/05b_tsne_swiss_roll

t-SNE unrolls non-linear manifolds that PCA cannot handle

Cluster Preservation



https://github.com/Digital-AI-Finance/methods-algorithms/tree/master/slides/L05_PCA_tSNE/06c_tsne_cluster_projection

t-SNE on MNIST digits: clear digit clusters vs. overlapping PCA projection

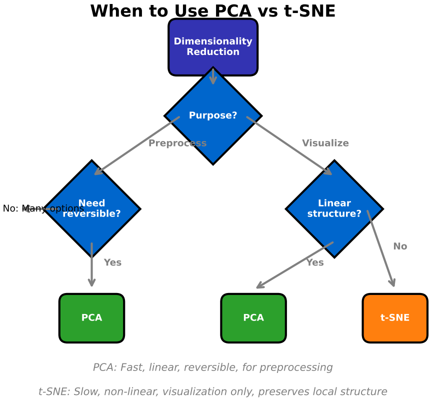
Hands-on Exercise

Open the Colab Notebook

- Exercise 1: Apply PCA to high-dimensional finance data
- Exercise 2: Visualize clusters with t-SNE
- Exercise 3: Compare PCA vs t-SNE for different datasets

Link: `https://colab.research.google.com/` See course materials

Decision Framework



https://github.com/Digital-AI-Finance/methods-algorithms/tree/master/slides/L05_PCA_tSNE/07_decision_flowchart

PCA for preprocessing/speed, t-SNE for visualization only

References

- Jolliffe, I.T. (2002). *Principal Component Analysis*. Springer.
- van der Maaten, L. & Hinton, G. (2008). *Visualizing Data using t-SNE*. JMLR.
- James et al. (2021). *Introduction to Statistical Learning*. `https://www.statlearning.com/`