UMAP Tutorial

September 17, 2025

1 Introduction

Uniform Manifold Approximation and Projection (UMAP) is a non-linear dimensionality reduction technique grounded in manifold learning and topological data analysis. UMAP builds a weighted neighbor graph in the original space, optimizes a low-dimensional embedding that preserves local connectivity, and produces layouts well-suited for exploratory visualization and clustering diagnostics.

2 Theory and Formulas

2.1 Neighbor Graph Construction

For each data point $\mathbf{x}_i$, UMAP identifies k nearest neighbors and assigns edge weights via a smooth exponential kernel:

$$\mu_{ij} = \exp\left(-\frac{\max(0, d(\mathbf{x}_i, \mathbf{x}_j) - \rho_i)}{\sigma_i}\right), \tag{1}$$

where d is the chosen distance metric, ρ_i ensures at least one neighbor at distance zero, and σ_i normalizes local connectivity. Symmetrization combines directed weights:

$$\mathbf{W} = \mu + \mu^\top - \mu \odot \mu^\top, \tag{2}$$

yielding a fuzzy topological representation of the data manifold.

2.2 Low-Dimensional Optimization

UMAP learns embeddings $\mathbf{y}_i$ by minimizing a cross-entropy between high- and low-dimensional fuzzy sets. Connection strengths in the embedding are modeled with a differentiable curve

$$\nu_{ij} = \frac{1}{1 + a\|\mathbf{y}_i - \mathbf{y}_j\|_2^{2b}}, \tag{3}$$

with parameters a and b selected from the embedding distance distribution. The loss function is

$$C = \sum_{(i,j)} \left[w_{ij} \log \frac{w_{ij}}{\nu_{ij}} + (1 - w_{ij}) \log \frac{1 - w_{ij}}{1 - \nu_{ij}} \right], \tag{4}$$

optimized via stochastic gradient descent on sampled edges.

2.3 Hyperparameters and Practical Considerations

Key settings include the number of neighbors ($_{n}eighbors)controllinglocal/globalbalance, min_{d}istinflue$

3 Applications and Tips

- **Single-cell analysis**: visualize manifold structure of gene expression profiles and identify rare cell populations.
- **Text and embeddings**: inspect sentence or document embeddings to validate semantic clustering.
- **Anomaly diagnosis**: highlight outliers or transitional states when combined with temporal or metadata overlays.

- **Best practices**: standardize features, experiment with different $_{n}eighbors/min_{d}istpairs, annotateembe$ $SNEorPCAforrobustness.$

4 Python Practice

The script `gen_t_umap_figures.py` standardizes synthetic data, fits UMAP under varying neighbor settings, and evaluates trustworthiness scores that quantify neighborhood preservation for diagnostics.

Listing 1: Excerpt from gen$_{tu}map_{f}igures.py$

```
import umap
from sklearn.manifold import trustworthiness

neighbors_list = [10, 30, 50]
embeddings = {}
trust_scores = []
for n in neighbors_list:
    reducer = umap.UMAP(n_neighbors=n, min_dist=0.1, metric="euclidean",
                        init="spectral", random_state=42)
    embedding = reducer.fit_transform(points)
    embeddings[n] = embedding
    trust_scores.append(trustworthiness(points, embedding, n_neighbors=15))
```

5 Result

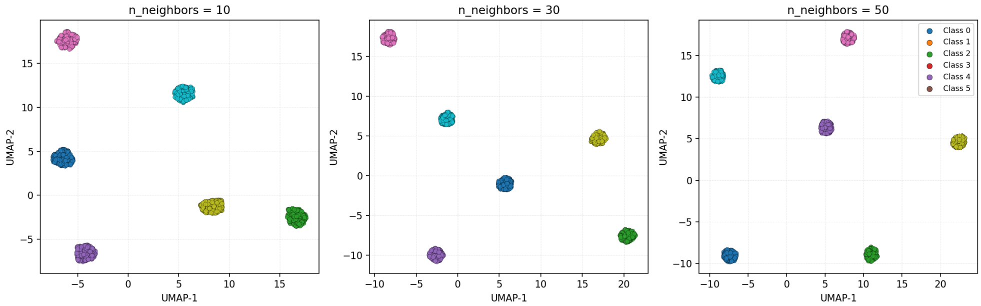

Figure 1: UMAP embeddings for multiple neighbor counts with color-coding by class

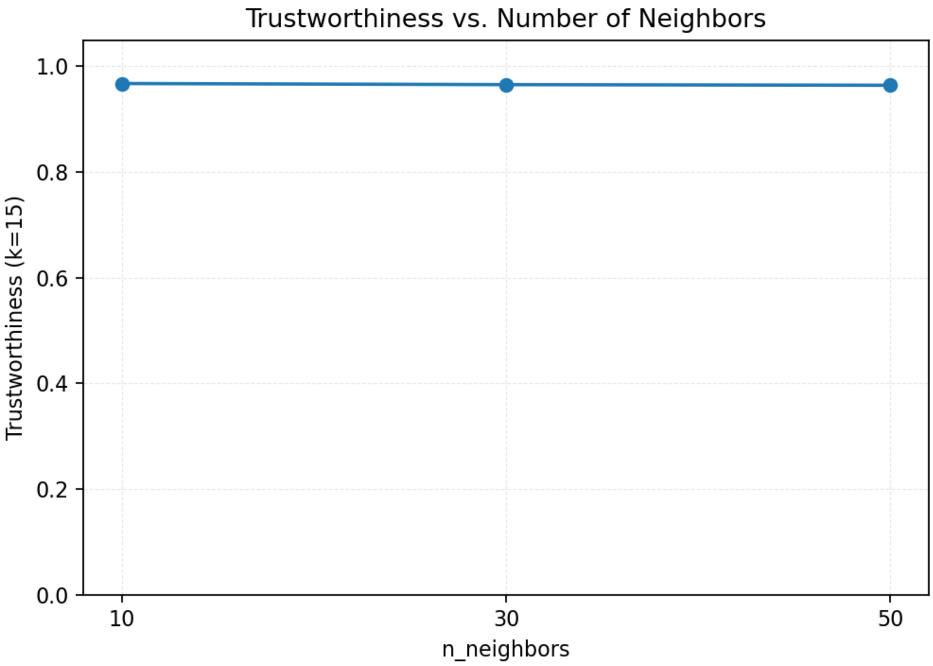

Figure 2: Trustworthiness versus number of neighbors

6 Summary

UMAP models neighborhood connectivity as fuzzy sets and optimizes a low-dimensional layout via cross-entropy minimization. Adjusting neighbor count, minimum distance, and metrics enables flexible trade-offs between local detail and global arrangement. The example demonstrates how to compare embeddings and monitor trustworthiness across parameter sweeps.