UMAP

Uniform Manifold Approximation and Projection

⭐ State-of-the-art for visualization and dimensionality reduction

📊 t-SNE

- Slower computation
- Local structure focus
- Limited scalability
- Mainly for visualization

🚀 UMAP

- Faster computation
- Local + global structure
- Better scalability
- Visualization + reduction

🧮 Mathematical Foundation

Riemannian geometry & topological data analysis

📈 Scalability

Handles larger datasets than t-SNE

🎯 Structure Preservation

Both local and global structure

🏷️ Supervised Mode

Supports dimensionality reduction with labels

⚙️ Key Parameters

n_neighbors

min_dist

Controls local structure preservation

Controls embedding tightness

🔬 UMAP Computation Principles

1

High-dimensional Graph Construction

Find k-nearest neighbors (k-NN) to create a weighted graph. Calculate distances from each data point to its neighbors to determine probabilistic connection strengths.

$$w(i,j) = \exp(-(d(i,j) - \rho_i) / \sigma_i)$$

2

Fuzzy Simplicial Complex

Transform the weighted graph into a fuzzy topological structure. This is a mathematical representation that preserves the topological properties of the data.

$$P(i \leftrightarrow j) = w(i,j) + w(j,i) - w(i,j) \times w(j,i)$$

3

Low-dimensional Optimization

Optimize the low-dimensional embedding using Stochastic Gradient Descent (SGD). Minimize cross-entropy to reduce the difference between high-dimensional and low-dimensional structures.

$$Loss = \Sigma\ P(i,j) \log(P(i,j)/Q(i,j)) + (1-P(i,j)) \log((1-P(i,j))/(1-Q(i,j)))$$

🔍 High-dimensional Space

- Local metric estimation
- k-NN graph construction
- Topological structure representation

📐 Low-dimensional Space

- Initial embedding generation
- Iterative optimization
- Structure preservation verification

🎮 Interactive UMAP Explorer

Try UMAP Explorer →