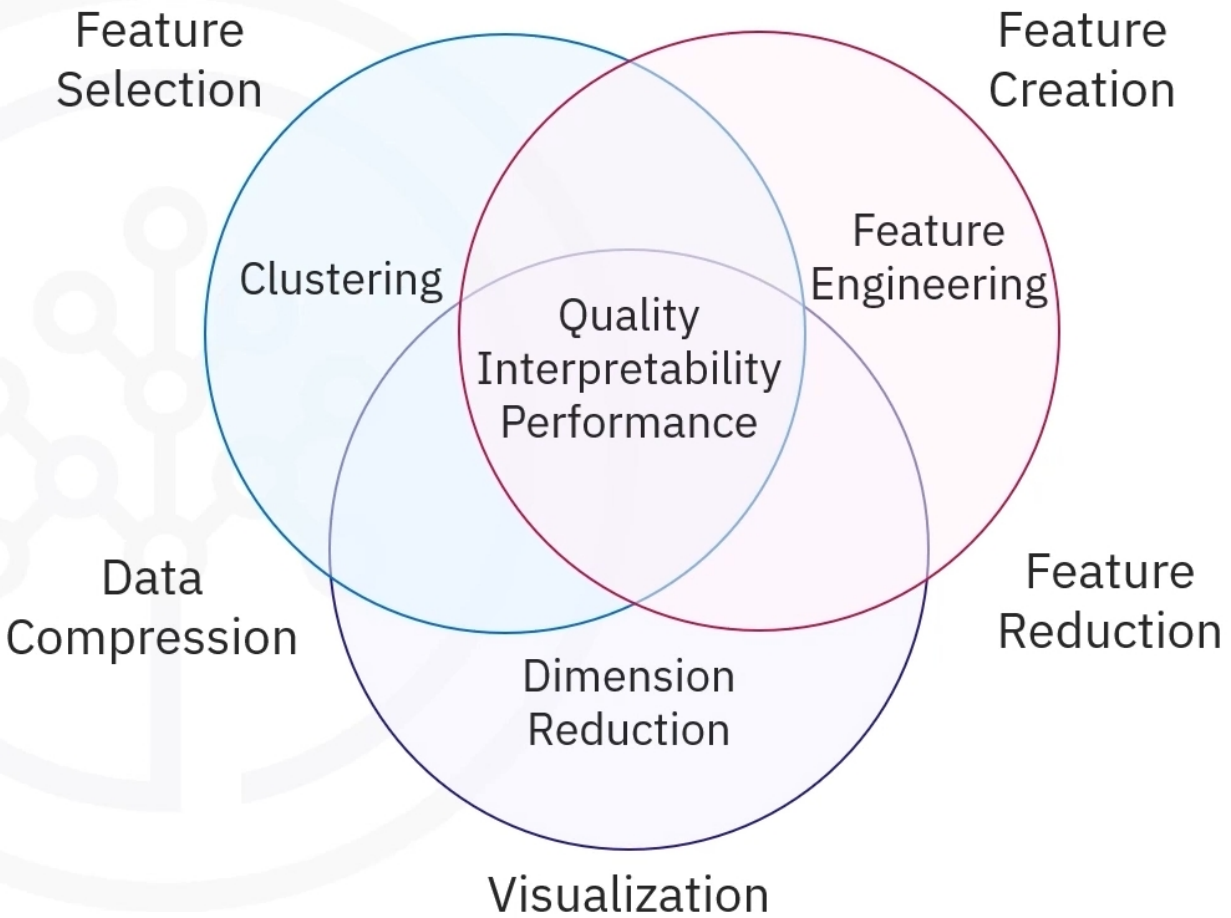
Interdependencies in data science techniques

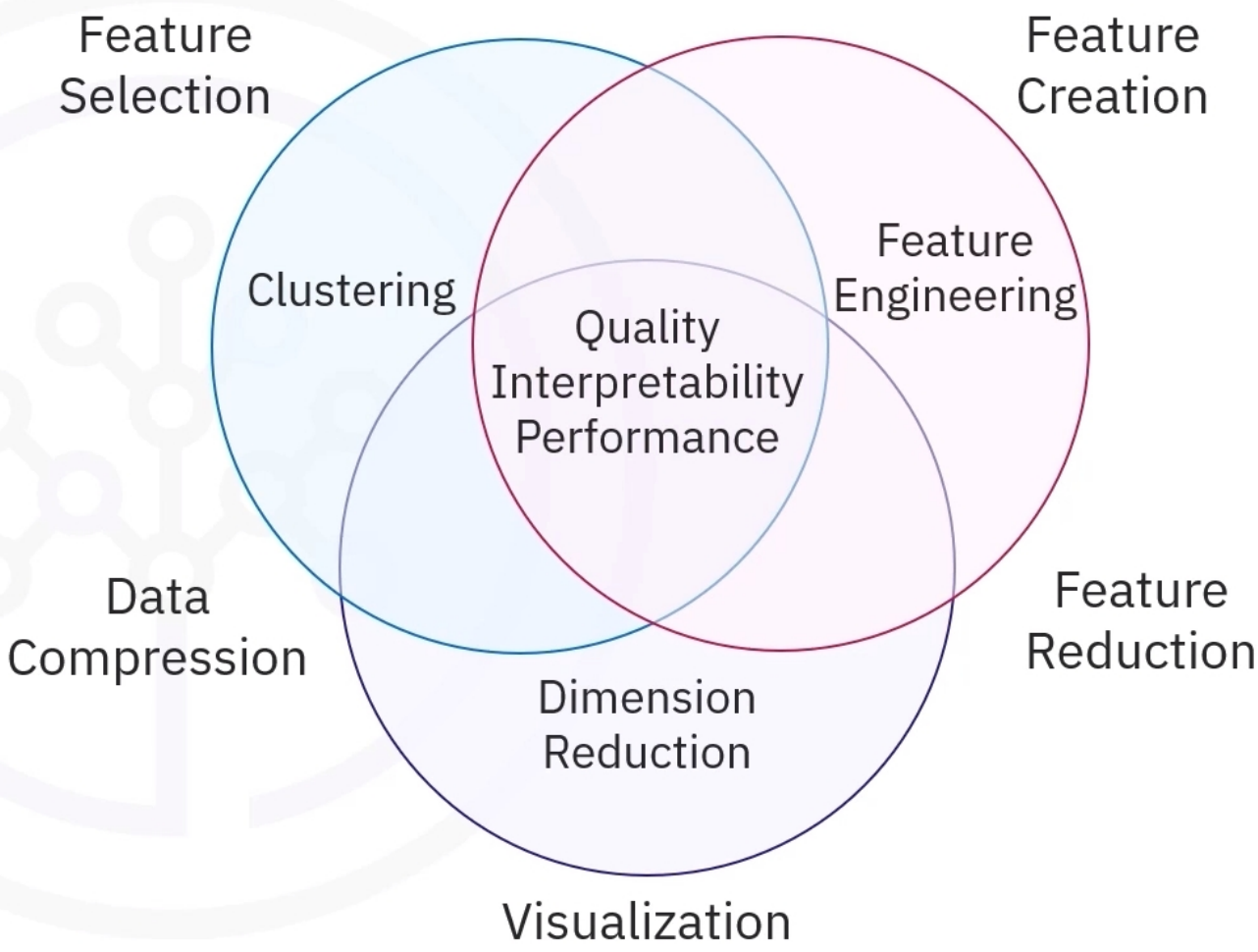
Complementary techniques in machine learning and data science

Improve model performance, quality, and interpretability



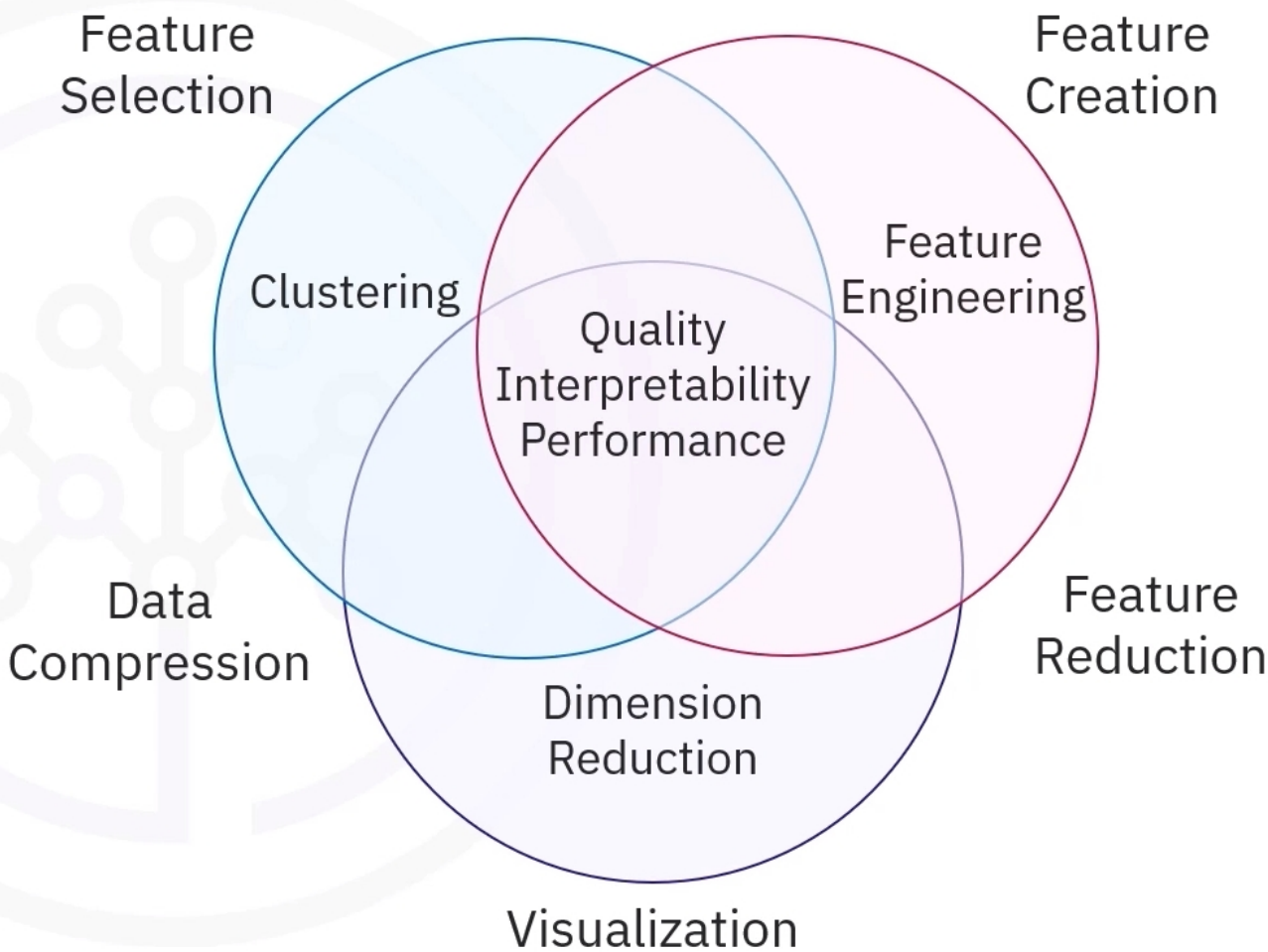
Clustering

- Helps with feature selection and creation
- Supports dimension reduction
- Enhances computational efficiency and scalability



Dimension reduction

- Simplifies visualization of high-dimensional clustering
- Aids in feature engineering and improves model quality
- Reduces the number of features required



Dimension reduction before clustering

Used as a preprocessing step for clustering

Simplifying data structure and improving outcomes

High-dimensional data:

- Poses challenges for distance-based clustering algorithms
- Causes data points to become sparse
- Leads to smaller clusters

PCA, t-SNE, and UMAP:

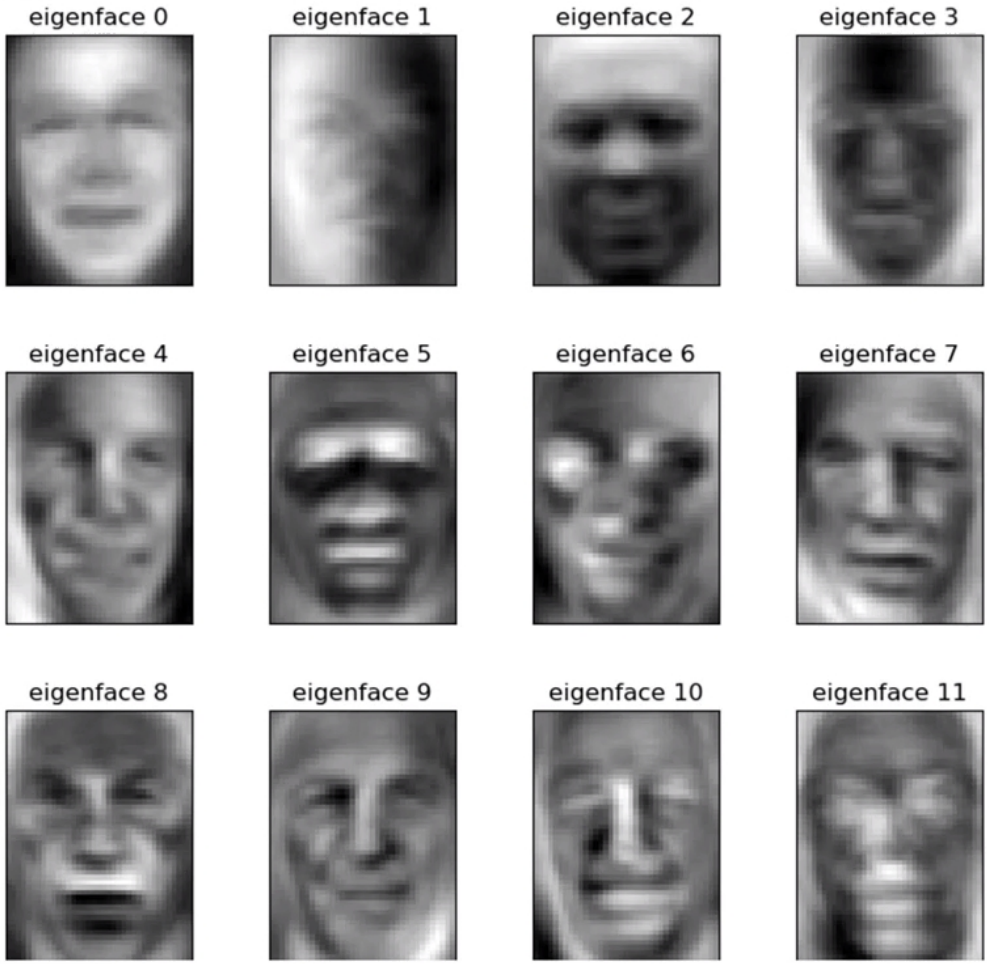
- Reduce dimensions
- Enhance efficiency

Using dimension reduction for face recognition

Uses eigenfaces as input features for recognition

Performs PCA on an unlabeled face data set

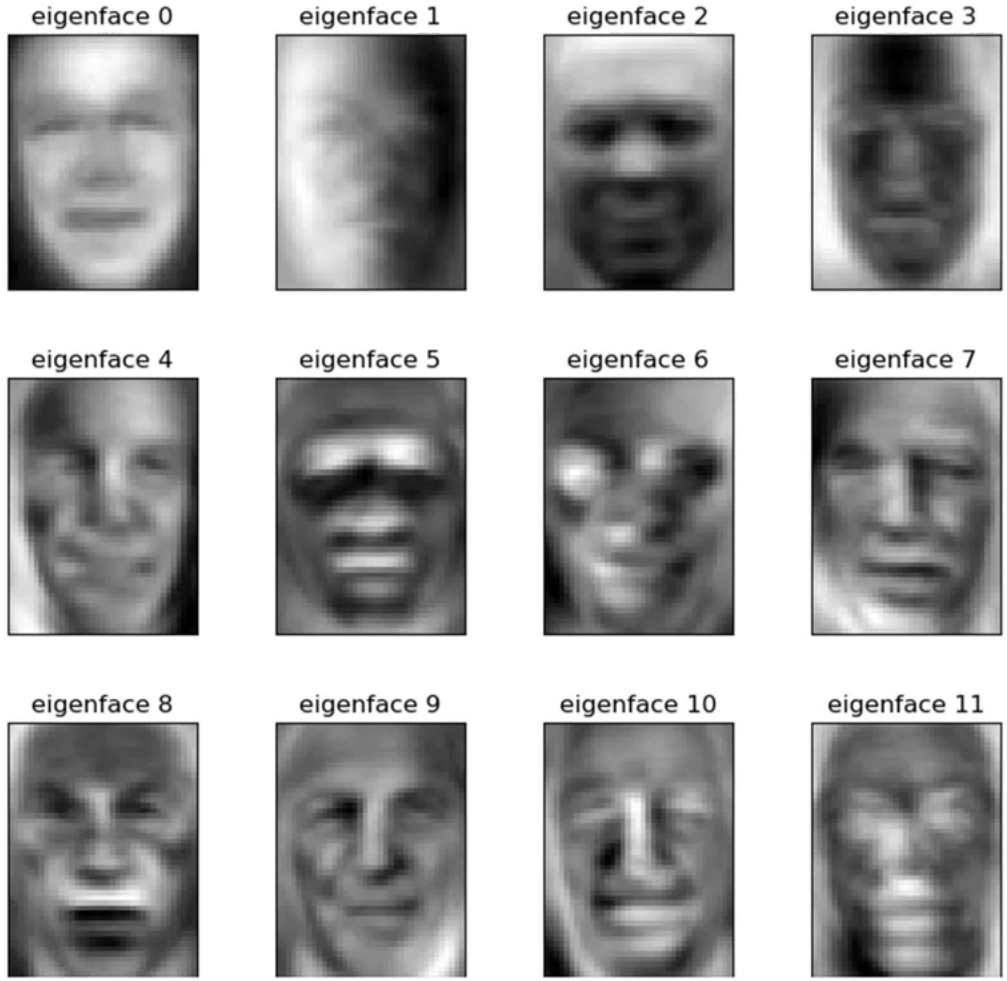
Extracts top 150 eigenfaces from 966 faces



Forms an orthonormal basis for feature space

Projects input data onto the eigenface basis

Trains an SVM to predict faces

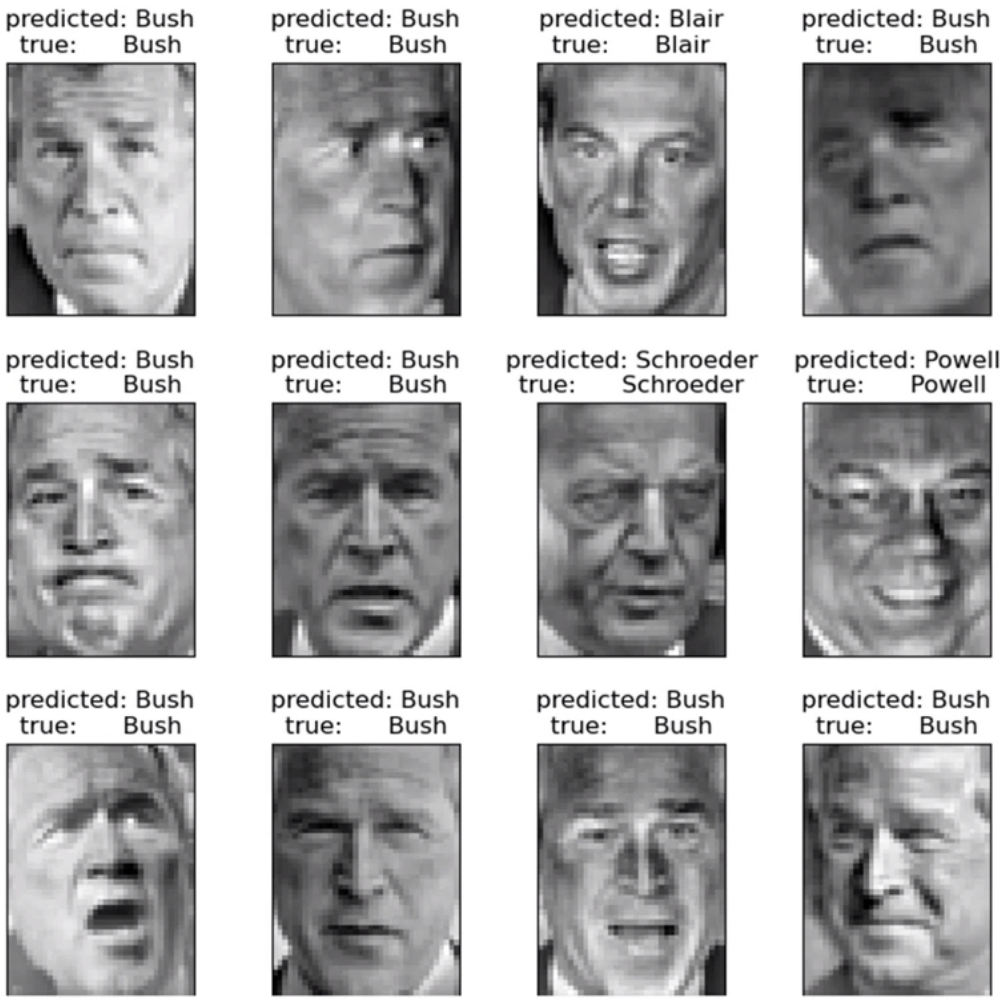


Face recognition result

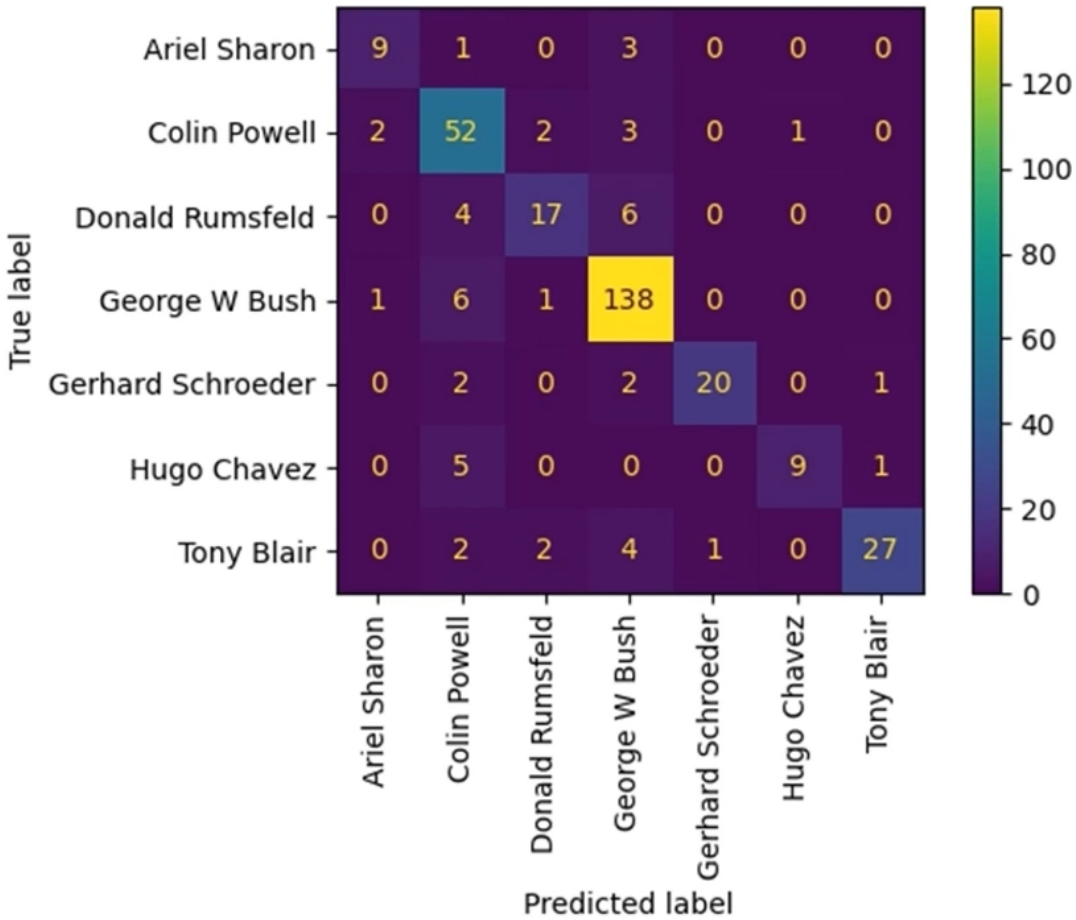
Preserves key features for face identification

Minimizes computational load

Accurately predicts 12 faces



Counts of correct face recognition



Illustrates quantitative evaluation of the model's quality

Dimension reduction after clustering

Clustering results are impacted beyond three dimensions

Advanced dimension reduction techniques:

- Project outcomes into two or three dimensions
- Improve visual interpretation

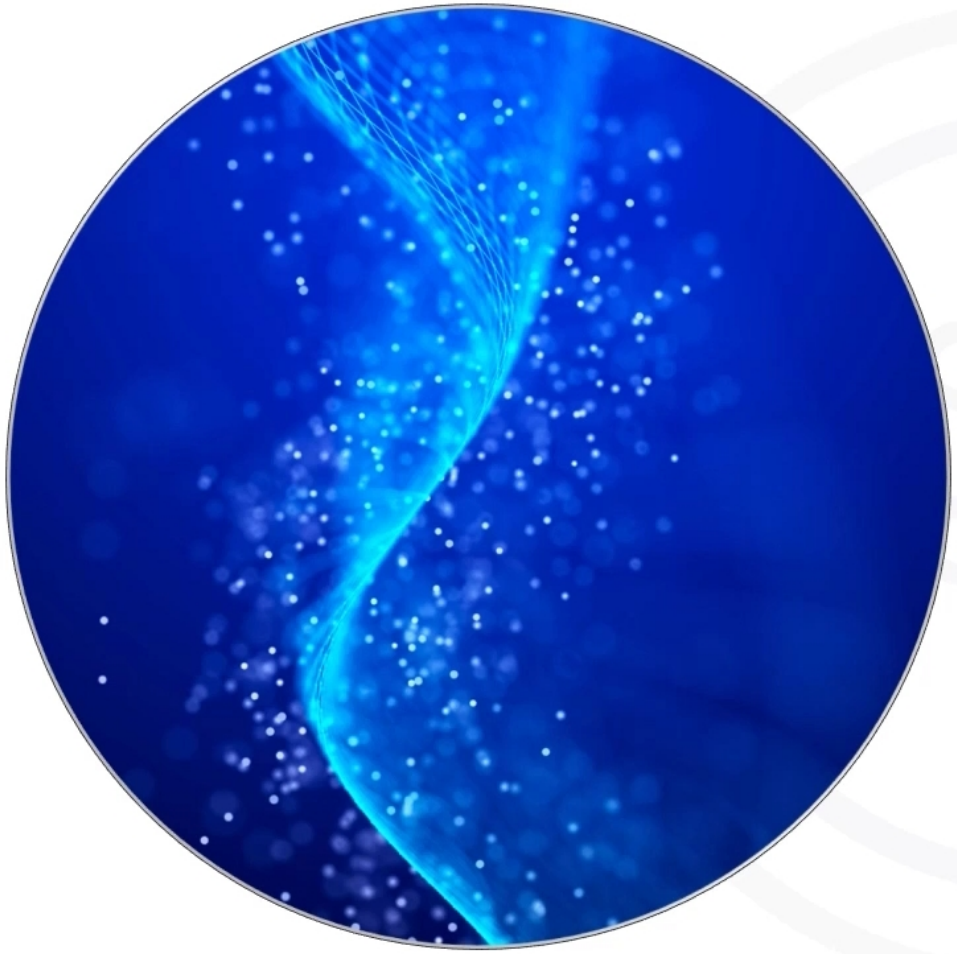
PCA, t-SNE, and UMAP:

- Allow meaningful projections of higher-dimensional clusters
- Facilitates visualization of clustering quality

Reducing the dimensionality of data:

- Enhances cluster interpretability
- Identifies key patterns obscured in higher dimensions

Clustering for feature selection



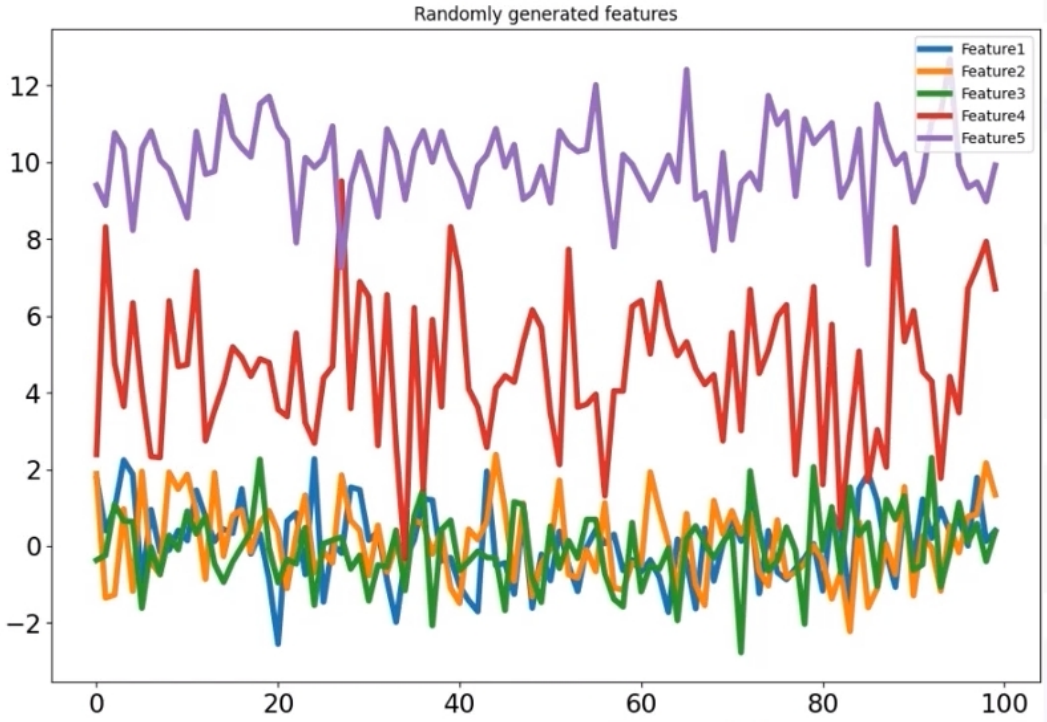
Identifying redundant features
Apply clustering to both observations and features
Cluster similar or correlated features
Identify and remove redundant information
Choose a representative feature from each cluster
Reduce feature count while preserving information

Informing feature engineering

Guides feature engineering decisions

Reveals subgroups for feature interaction insights

Feature selection with k-means



- Five features generated with random normal distribution
- Features have means of 1, 5, and 10
- All variances are 1, except feature #4 with 2

Cluster 1:

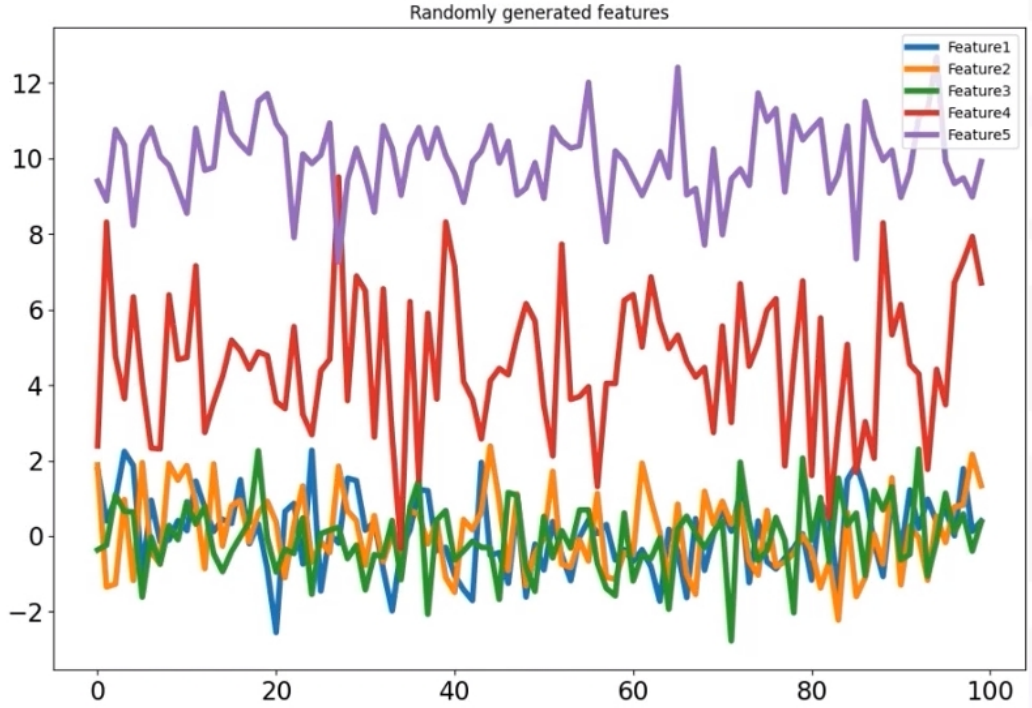
- **Feature 1**
- **Feature 2**
- **Feature 3**

Cluster 2:

- **Feature 5**

Cluster 3:

- **Feature 4**



- Features 1-3 are statistically similar
- Features 4 and 5 stand out
- K-means (K=3) correctly clusters features

Cluster 1:

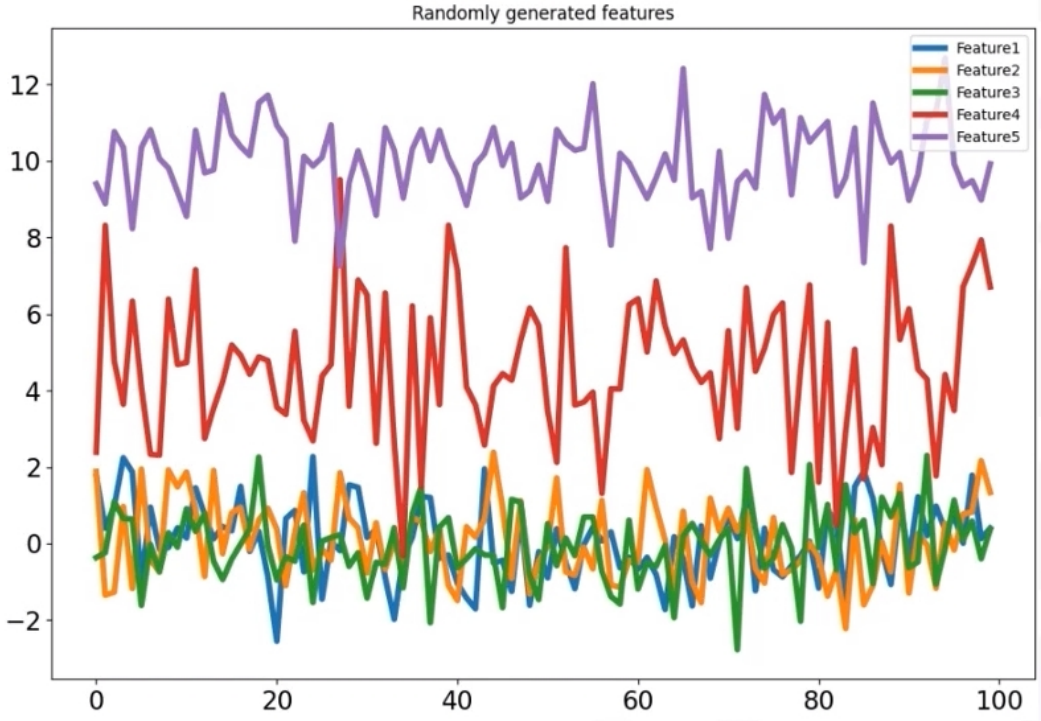
- **Feature 1**
- **Feature 2**
- **Feature 3**

Cluster 2:

- **Feature 5**

Cluster 3:

- **Feature 4**



- Cluster one contains redundant features
- Select only one feature for modeling

Cluster 1:

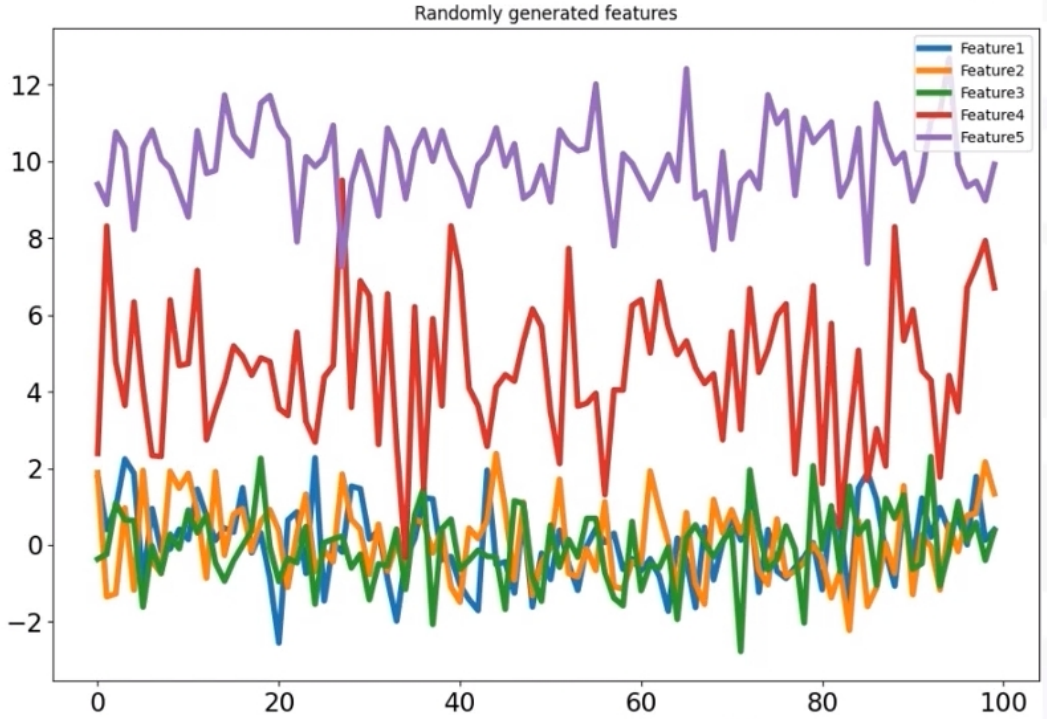
- **Feature 1**
- **Feature 2**
- **Feature 3**

Cluster 2:

- **Feature 5**

Cluster 3:

- **Feature 4**



- Example of feature selection implementation
- Part of feature engineering
- Also viewed as dimension reduction

Cluster 1:

- **Feature 1**
- **Feature 2**
- **Feature 3**

Cluster 2:

- **Feature 5**

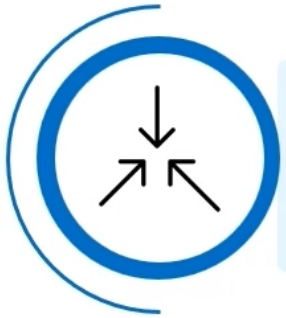
Cluster 3:

- **Feature 4**

Recap

- Explain clustering, dimension reduction, and feature engineering
- Explain dimension reduction and how it is used as a preprocessing step
- Analyze how dimension reduction is used for face recognition
- Analyze how clustering can be used for feature selection
- Analyze feature selection using k-means

Dimension reduction algorithms



Reduce data set features without sacrificing critical information

Simplify data set for machine learning models

Types:

- Principle Component Analysis (PCA)
- t-distributed stochastic neighbor embedding (t-SNE)
- Uniform Manifold Approximation and Projection (UMAP)

Transform original dimensions to create new features

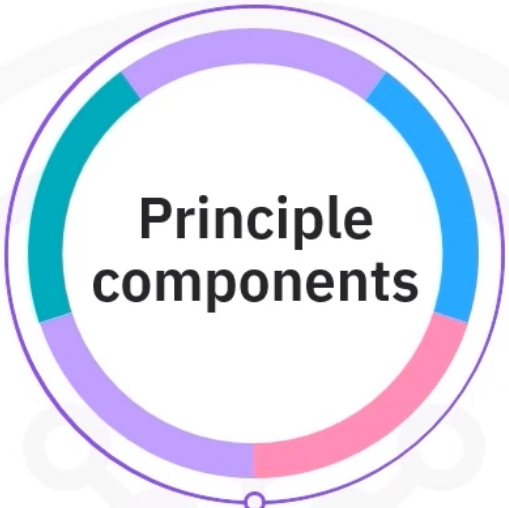
Principle Component Analysis (PCA)

Assumes data set features are linearly correlated

Simplifies data, reduces dimensionality and reduces noise and minimize information loss

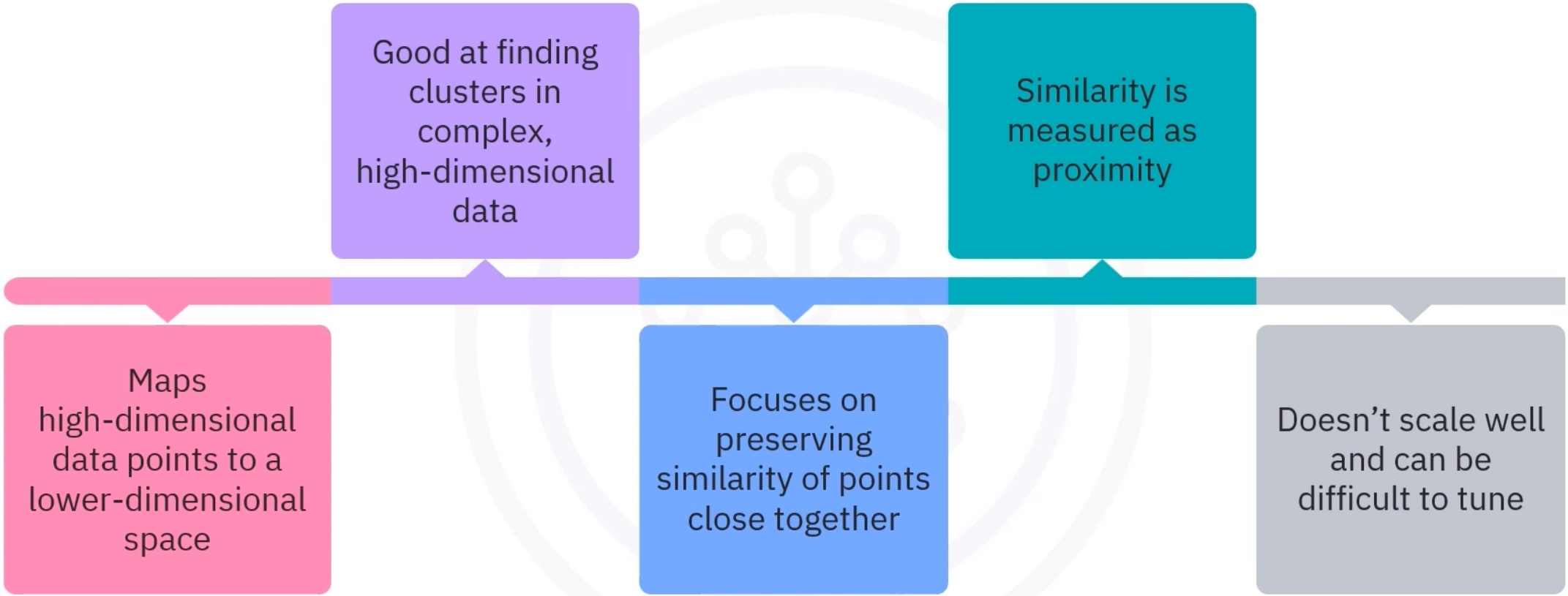
Can transform features into principle components and retain variance

Principle Component Analysis (PCA)



- Orthogonal to each other
- Define a new coordinate system
- Organized in decreasing order of importance or how much feature space variance they explain
- First few components contain most of the information

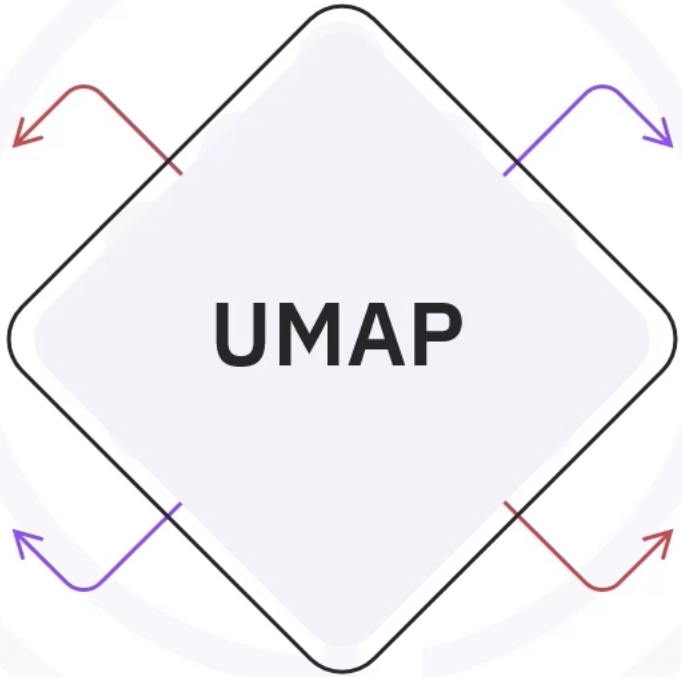
T-distributed stochastic neighbor embedding (t-SNE)



Uniform Manifold Approximation and Projection (UMAP)

Constructs a high-dimensional graph representation of the data based on manifold theory

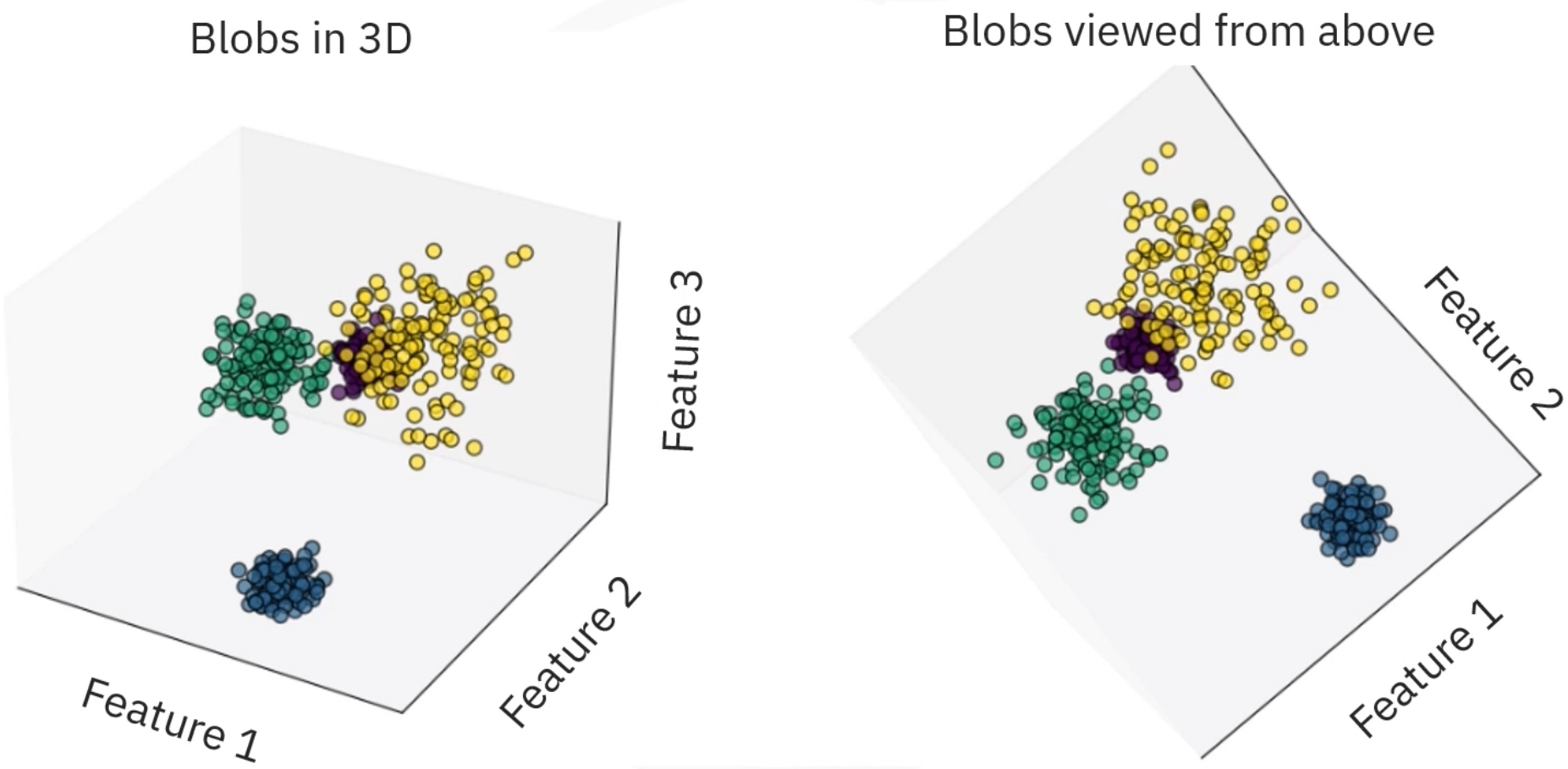
Optimizes a low-dimensional graph structure that best preserves relationships between points



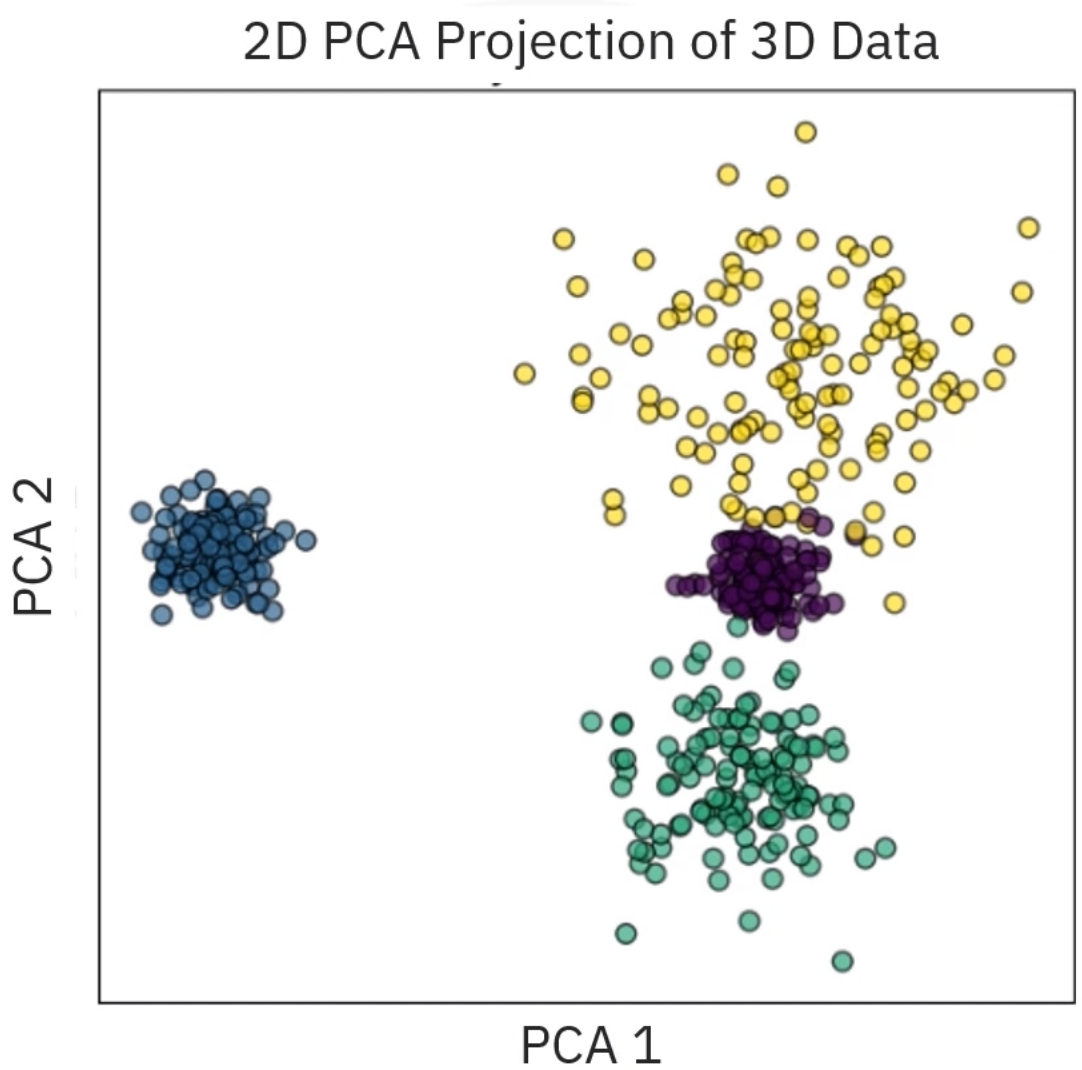
Scales better than t-SNE

Preserves the global structure of data

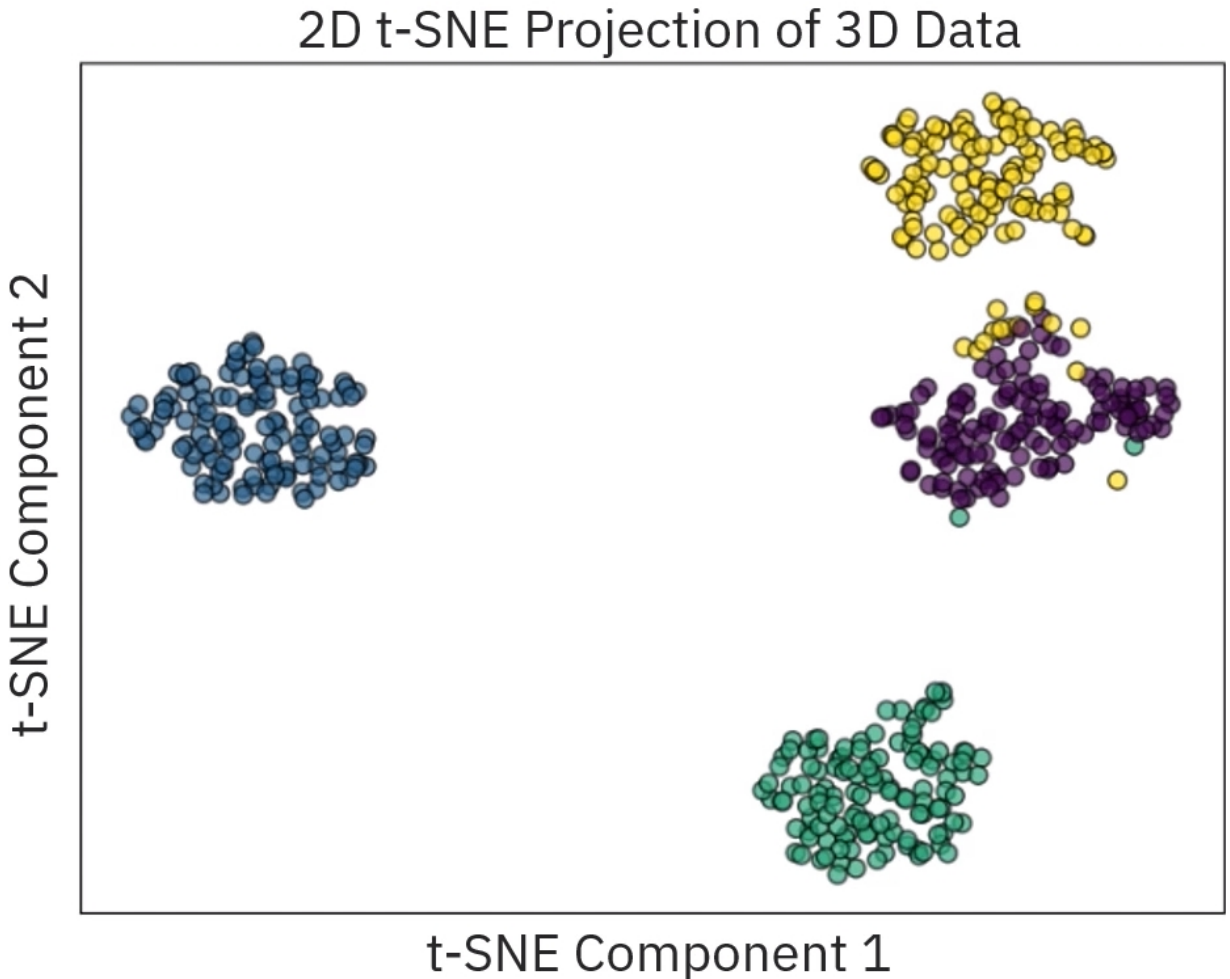
Dimension reduction scenario



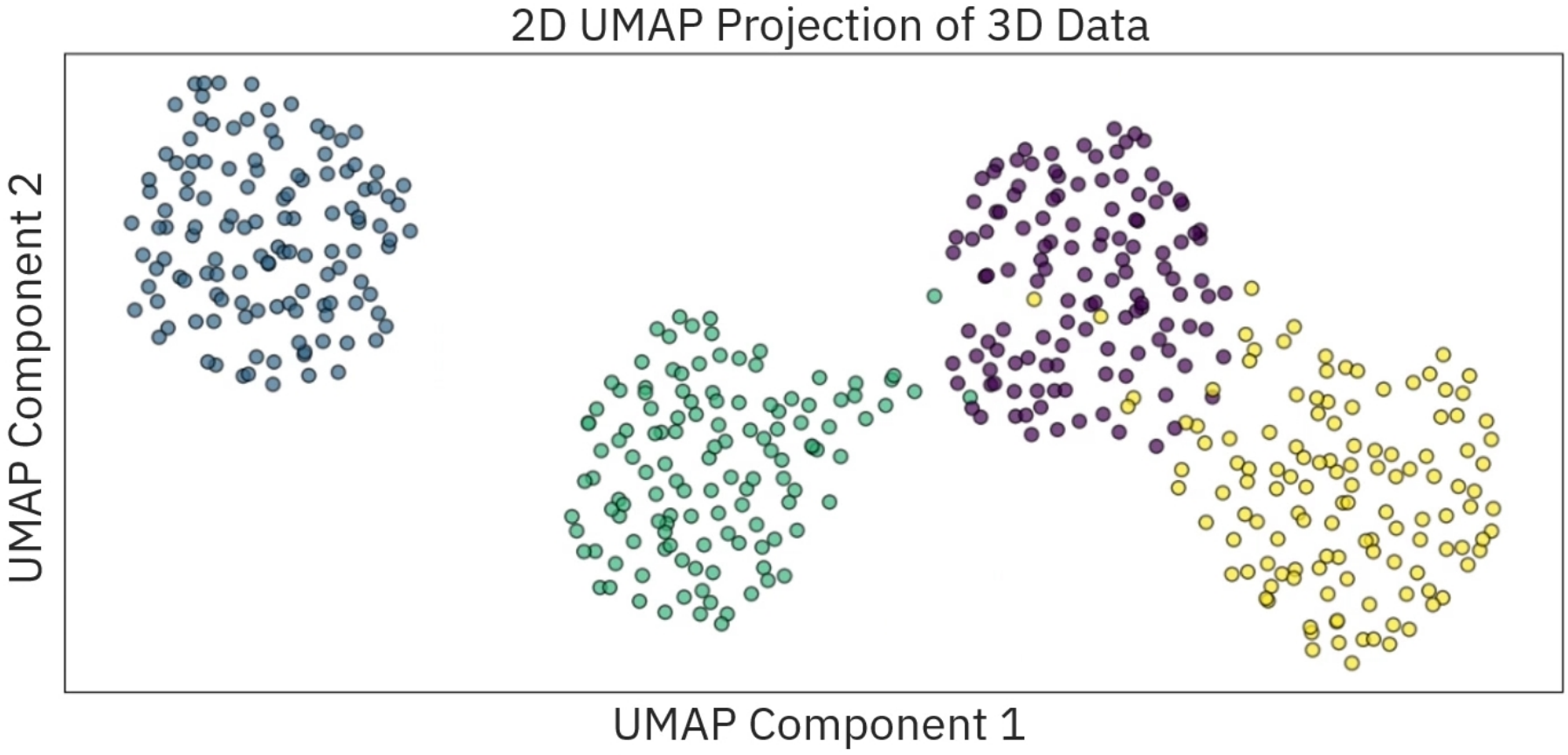
PCA result



t-SNE result


2D t-SNE Projection of 3D Data
t-SNE Component 2
t-SNE Component 1

UMAP result


2D UMAP Projection of 3D Data
UMAP Component 2
UMAP Component 1

Recap

- Dimensionality reduction algorithms reduce data set features without sacrificing critical information
- Types of dimensionality reduction algorithms: PCA, t-SNE, and UMAP
- PCA simplifies data, reduces dimensionality, and reduces noise
- t-SNE maps high-dimensional data points to a lower-dimensional space
- UMAP creates a low-dimensional representation of data