



**YTU**

Muhammet Ali Gümüřsoy

1906A018

Homework Assignment #2

Dimensionality Reduction & Model Validation

Ertuğrul Bayraktar

1. “Advanced Reduction” tab that has user-selectable components was created under “create\_tabs” with its functions under “create\_advanced\_reduction\_tab”.

It contains:

- PCA Explained Variance
- Supervised (LDA) Reduction
- Configurable clusters (k) with Elbow Method support
- UMAP Projection
- Interactive Plotly 2D/3D

Classical ML   Deep Learning   Dimensionality Reduction   Reinforcement Learning <b>Advanced Reduction</b>	
<b>PCA Explained Variance</b> n_components: <input type="text" value="1"/> <button>Show Variance Plot</button>	<b>Supervised (LDA) + Separation</b>  <button>Compute LDA Separation</button>
<b>KMeans + Elbow Method</b> Max k for Elbow: <input type="text" value="2"/> <button>Show Elbow Plot</button>	<b>UMAP Projection</b> n_neighbors (UMAP): <input type="text" value="2"/> <button>Apply UMAP Show</button>
<b>Interactive Plotly 2D/3D</b>  <button>Show 2D Plotly</button>	  <button>Show 3D Plotly</button>

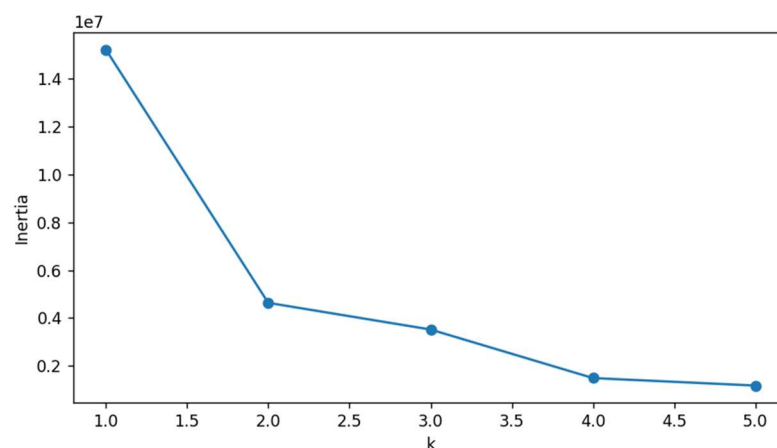


Figure 1 Elbow Method(k=5)

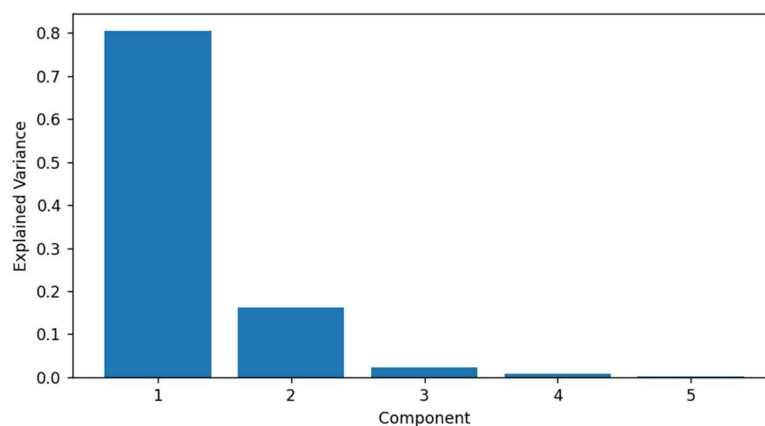


Figure 2 Explained Variance(n=5)

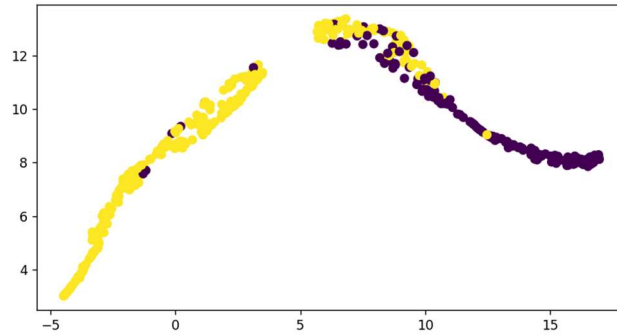


Figure 3 UMAP visualization( $n=20$ )

- I used Boston Housing(506) data for visualizations. I downloaded it separately because, it doesn't work in GUI. So, I added csv file to zip.
2. Eigenvectors were computed to project data into 1D under "project\_1d" function.
  3. k-Fold Cross-Validation ( $k=5$ ) Results:  
 Fold Sizes: [20, 20, 20, 20, 20]  
 Mean Accuracy: 0.8500  
 Std Accuracy: 0.0894
  4. Comparison of dimensionality reduction methods:

### GENERAL COMPARISONS

#### PCA

- Pros: Fast, deterministic, easy to interpret via component loadings; preserves global variance structure.
- Cons: Cannot capture nonlinear relationships; local clusters may overlap.

#### LDA

- Pros: Incorporates class labels to maximize separation; useful as preprocessing for classification.
- Cons: Limited to #classes – 1 dimensions; assumes Gaussian class distributions with equal covariance.

#### t-SNE

- Pros: Excellent at teasing out local clusters in complex, high-dimensional data; great for visualization.
- Cons: Very slow for large datasets; results can vary run-to-run; poor at reflecting global geometry.

#### UMAP

- Pros: Balances local vs. global structure better than t-SNE; faster and more scalable; often yields more meaningful global clusters.
- Cons: Still requires tuning, embedding axes are not directly interpretable.

### **PCA vs t-SNE:**

PCA is a fast, deterministic linear technique that projects data onto orthogonal axes of greatest variance, making it excellent for preserving global structure and easily interpretable via component loadings, but it cannot capture nonlinear relationships and often blurs local clusters; by contrast, t-SNE is a nonlinear, unsupervised method designed specifically to preserve local neighbourhoods revealing tight, visually distinct clusters even in complex manifolds but it is computationally intensive (quadratic in the number of samples), sensitive to parameters like perplexity and learning rate, offers no explicit mapping for new data points, and can distort overall global geometry, making it best suited for exploratory visualization of smaller datasets rather than scalable preprocessing.

In summary:

**PCA** is best when you need speed, interpretability, and global structure.

**LDA** is ideal for supervised dimensionality reduction when class labels are known.

**t-SNE** excels at visualizing local structure and clusters in 2D, especially for small datasets.

**UMAP** combines the best of t-SNE and PCA by preserving both local and global features, and it scales better with data size.