Decoding the Kernel Code: A Clustering Tale of Patterns and Projections

2025-4-20

1. Preparing the Dataset: Laying the Groundwork

Before diving into clustering, it was essential to clean and standardize the dataset. The preprocessing steps I followed are outlined below, with normalized sample results summarized in Table 1.

1. **Loading the Data:** Imported the dataset from a TSV file into a pandas DataFrame.
2. **Labeling Columns:** Assigned intuitive column names for each feature and the class label to improve readability.
3. **Separating Features from Labels:** The target (class) column was isolated, ensuring that only the feature set was standardized.
4. **Feature Normalization:** Applied `StandardScaler` from `sklearn.preprocessing` [1] to transform feature values to a standard normal distribution (mean = 0, std = 1). This step is crucial for distance-based algorithms like k-means to work effectively.

Area A	Perimeter P	Compactness C	Length of Kernel	Width of Kernel	Asymmetry Coefficient	Length of Kernel Groove	Class
0.011840	0.009234	0.427494	-0.166808	0.197647	-1.792787	-0.921971	1
-0.190940	-0.358353	1.438945	-0.760533	0.208238	-0.672161	-1.188607	1
-0.345602	-0.473224	1.036904	-0.686035	0.319438	-0.965484	-1.229314	1
0.444896	0.330872	1.371233	0.067974	0.803955	-1.568128	-0.476222	1
-0.160007	-0.266457	1.019976	-0.546070	0.142047	-0.830155	-0.921971	1

Table 1: Sample of Normalized Kernel Characteristics

2. Finding the Sweet Spot: Choosing the Right Number of Clusters

K-Means with Varying k

To decide the optimal number of clusters, I used the Elbow Method by computing inertia (sum of squared distances to the nearest centroid) for k ranging from 1 to 10.

Inertia Explained

- Inertia measures how tightly data points are grouped within a cluster.
- Lower values of inertia indicate more compact clusters.

Clustering Process

- Implemented k-means clustering with the following parameters:
  - `init='k-means++'`: For smart centroid initialization.

– `n_init=10`: The model is run 10 times with different seeds.

- Feature columns were used exclusively (class labels were excluded).

Elbow Curve Insights

Figure 1 presents the inertia vs. number of clusters. The elbow appears around $k = 3$, indicating that three clusters strike a good balance between model complexity and fit.

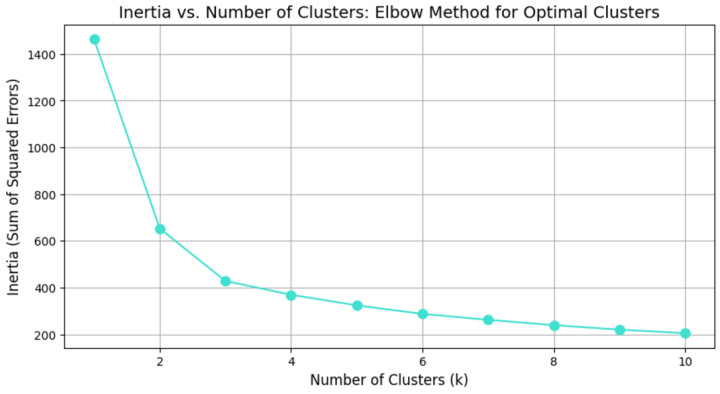

Figure 1: Elbow Curve for Optimal k in K-Means Clustering

3. Uncovering Hidden Patterns: Feature Visualization

a.) Relationships Between Features

By creating scatter plots between feature pairs (Figure 2), I explored how the data is structured across different features. The plot between **Length of Kernel** and **Width of Kernel** (Figure 3) was particularly telling, revealing clear class distinctions.

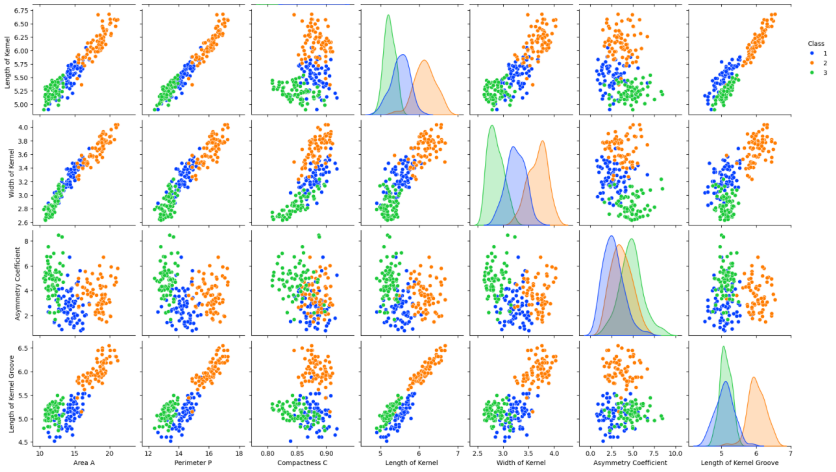

Figure 2: Pairwise Feature Relationships with Class Coloring

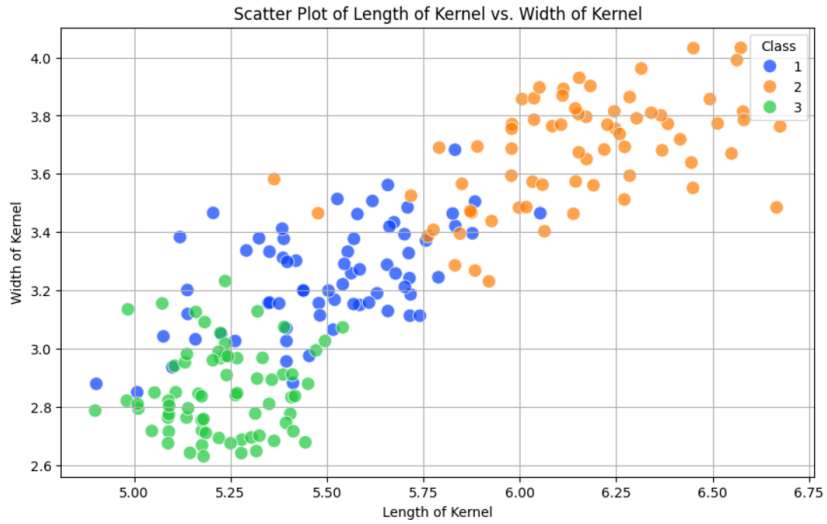

Figure 3: Length vs. Width of Kernel by Class

b.) Dimensionality Reduction with Gaussian Random Projection

To visualize high-dimensional data in 2D, I applied Gaussian Random Projection [2]. The result (Figure 4) shows:

- **Good separation** for Class 2.
- **Overlap** between Classes 1 and 3.
- **Noise sensitivity:** Each run slightly varies due to randomness.

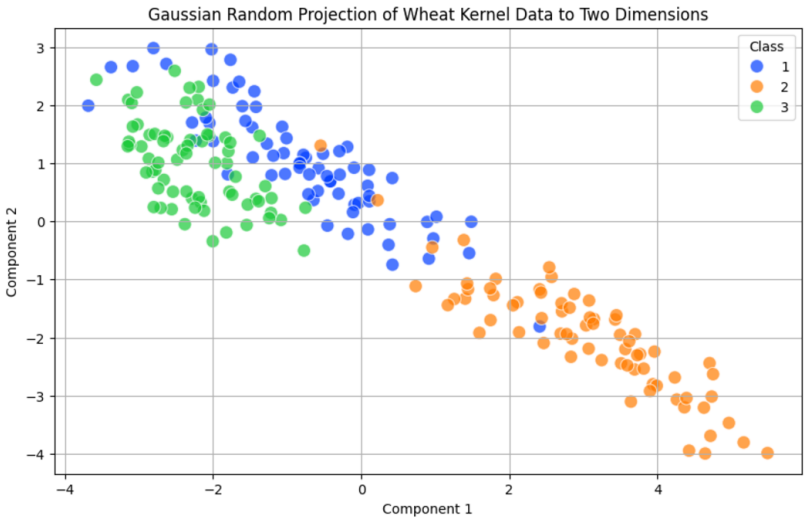

Figure 4: 2D Projection via Gaussian Random Projection

c.) UMAP: A Non-linear View

Using UMAP [3] for dimensionality reduction revealed well-separated groups in 2D space (Figure 5).

- Class 3 forms a distinct cluster.
- Classes 1 and 2 show some overlap but still retain structure.

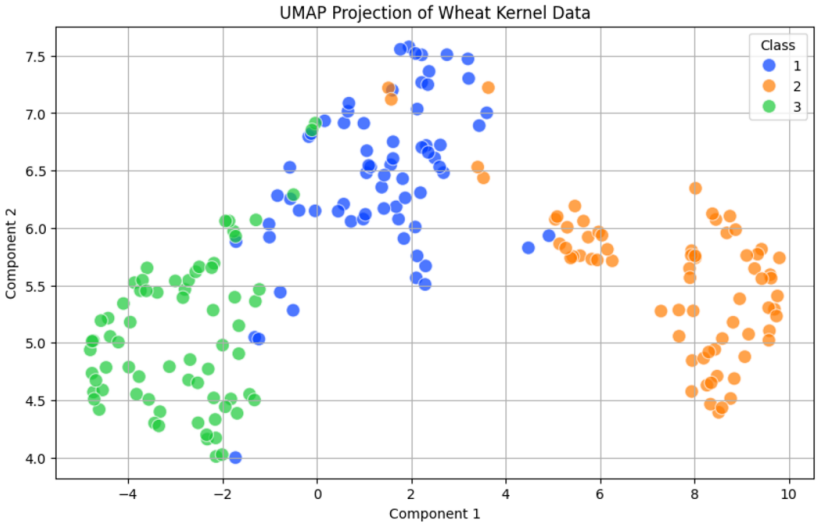

Figure 5: UMAP Projection of Kernel Data

d.) Observations on Linearity

Linearly Separable? No. The visualizations reveal overlapping classes, particularly between Classes 1 and 3, suggesting non-linear boundaries.

Clustering Implications: Linear clustering methods like k-means may struggle. Non-linear methods or kernel-based approaches could better capture the structure.

4. Clustering Performance: How Good Was the Model?

With $k = 3$ (from the Elbow Method), I applied k-means clustering and evaluated it against true class labels using the Rand Index.

Rand Index Evaluation

To fairly evaluate clustering accuracy, I implemented a permutation matching function to map predicted labels to actual classes. Final results are summarized in Table **??**.

Conclusion

This project demonstrates how thoughtful preprocessing, cluster validation, dimensionality reduction, and interpretability tools can reveal valuable patterns in data. While k-means offered a strong baseline, advanced clustering techniques could be explored for better handling of overlapping and non-linear structures.

references