

T.C

GEBZE TECHNICAL UNIVERSITY

FACULTY OF ENGINEERING

STATISTICAL DATA ANALYSIS

TERM PROJECT REPORT

|  |  |
| --- | --- |
| **Name/Surname** | Abdullah Doğan GÜVENER |
| **Number** | 244201008002 |

**TABLE OF CONTENTS**

[ABSTRACT iii](#_Toc199304724)

[1 INTRODUCTION 4](#_Toc199304725)

[1.1 Aim of the project 4](#_Toc199304726)

[2 MATERIAL AND METHOD 5](#_Toc199304727)

[1.1.1 Dataset Overview 5](#_Toc199304728)

[2.1.1 Feature Distribution Analysis (Boxplots) 6](#_Toc199304729)

[3.1.1 Z-Score Normalization and Fisher Distance Analysis 7](#_Toc199304730)

[4.1.1 Principal Component Analysis and Fisher Score Comparison 9](#_Toc199304731)

[5.1.1 Scatter Plots on PCA Projections 11](#_Toc199304732)

[6.1.1 LDA Dimensionality Reduction and Evaluation 13](#_Toc199304733)

[7.1.1 Clustering Analysis 14](#_Toc199304734)

[8.1.1 Outlier Detection and Data Cleaning 19](#_Toc199304736)

[3 CONCLUSION AND DISCUSSION 20](#_Toc199304737)

[REFERENCES 22](#_Toc199304738)

ABSTRACT

**Title**

Comprehensive Multivariate Analysis and Clustering on the Dry Bean Dataset Using Dimensionality Reduction and Outlier Detection Techniques

This study presents a detailed exploration of the Dry Bean dataset using a combination of statistical and machine learning techniques to extract patterns, evaluate feature importance, and improve data quality. The project begins with exploratory data analysis through boxplots, followed by normalization using Z-score and inter-class separability measurement via Fisher distance. Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are applied to reduce dimensionality and visualize the distribution of bean classes. Various clustering algorithms—including K-Means, DBSCAN, t-SNE, and Self-Organizing Maps (SOM)—are used to assess the natural groupings in the data. Outlier detection using a Z-score threshold enhances the overall data quality, leading to cleaner projections and better-defined clusters. Visualizations and silhouette scores are used to compare the effectiveness of PCA and LDA in capturing class separability. The results highlight the strengths of each method and suggest best practices for preprocessing and unsupervised analysis of structured agricultural datasets.

**Keywords:** Dry Bean Dataset, PCA, LDA, Fisher Distance, Z-score Normalization, K-Means, DBSCAN, t-SNE, SOM, Outlier Detection, Dimensionality Reduction, Silhouette Score, Clustering

1. INTRODUCTION
   1. Aim of the project

The increasing availability of structured datasets has enabled more sophisticated methods of data analysis and pattern recognition across various domains, including agriculture and food sciences. In this project, we examine the *Dry Bean Dataset*, which consists of 13,611 samples representing seven different bean varieties. Each sample is characterized by 16 numerical features extracted from bean images, covering geometric, shape, and density-based attributes.

The primary objective of this study is to analyze the dataset from multiple perspectives—statistical distribution, dimensionality reduction, and clustering—to uncover underlying structures and evaluate the discriminative power of the features. To achieve this, the project applies a sequence of preprocessing, transformation, and unsupervised learning techniques, including Z-score normalization, Fisher distance computation, Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and clustering algorithms such as K-Means, DBSCAN, t-SNE, and Self-Organizing Maps (SOM).

In addition, the study emphasizes the role of outlier detection in improving data quality and interpretability. The effectiveness of dimensionality reduction methods is quantitatively assessed using the silhouette score, providing an objective measure of class separability. By combining visual and numerical evaluations, this project aims to present a thorough and practical approach to understanding complex multivariate datasets.

1. MATERIAL AND METHOD

# ****Dataset Overview****

The dataset used in this project is the **Dry Bean Dataset**, publicly available from the UCI Machine Learning Repository. It consists of **13,611 samples**, each representing a single dry bean seed extracted from grayscale images using standard computer vision techniques. The primary goal of the dataset is to distinguish between **seven different bean varieties** based on their physical and geometric properties.

Each data point is described by **16 numerical features**, most of which capture shape, area, or contour-based attributes of the bean. These features are as follows:

* **Area**: The number of pixels within the bean boundary.
* **Perimeter**: The length of the bean boundary.
* **Major Axis Length / Minor Axis Length**: The lengths of the major and minor ellipse axes.
* **Aspect Ratio**: Ratio of major to minor axis.
* **Eccentricity**: Ellipse eccentricity, indicating deviation from circularity.
* **Convex Area**: Area of the convex hull around the bean.
* **Equiv Diameter**: Diameter of a circle with the same area as the bean.
* **Extent**: Ratio of the contour area to the bounding box area.
* **Solidity**: Ratio of contour area to convex hull area.
* **Roundness, Compactness, ShapeFactor1-4**: Shape descriptors derived from geometric formulas.

The **target variable** (Class) includes the following **seven bean types**:

* **SEKER**
* **BARBUNYA**
* **BOMBAY**
* **CALI**
* **HOROZ**
* **SIRA**
* **DERMASON**

These class labels are categorical and mutually exclusive. The dataset is **balanced to a reasonable extent**, with several thousand examples per class, which enables effective statistical and machine learning analysis.

In this project, all features were treated as continuous variables. The class label was used only for evaluation and visualization purposes in clustering and dimensionality reduction tasks. For clustering algorithms, class labels were not supplied to ensure an unsupervised evaluation.

# Feature Distribution Analysis (Boxplots)

As an initial step in understanding the Dry Bean dataset, a **feature-wise exploratory data analysis** was performed using **boxplots**. This method provides a visual summary of the distribution, central tendency, spread, and presence of outliers for each of the 16 numerical attributes.

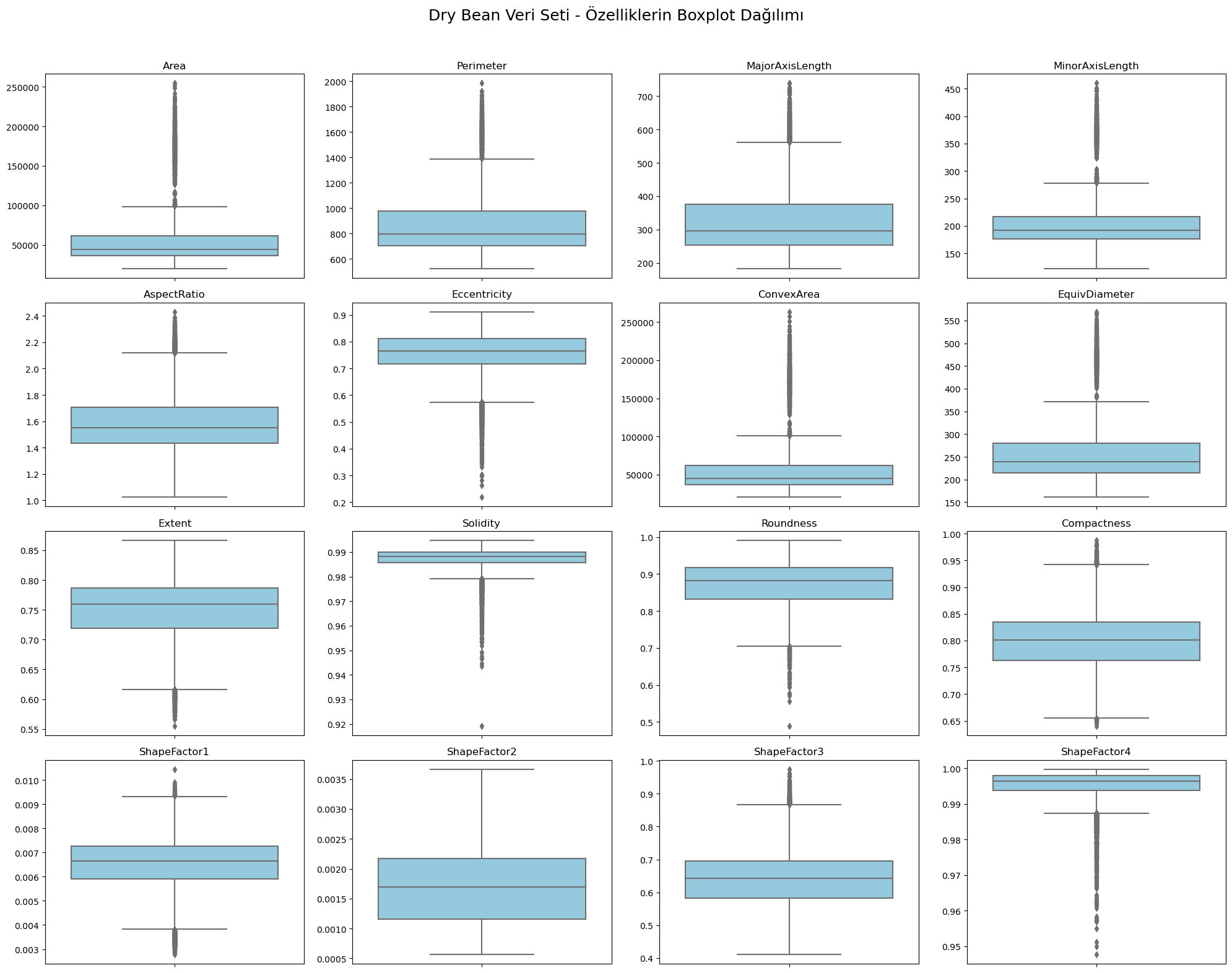
To ensure clarity, each feature was plotted individually using a **4x4 grid layout** (see **Figure 1**). This layout prevents scale-dominant features (such as Area and ConvexArea) from compressing the visual range of smaller-scale features like ShapeFactor1 or Solidity.

1. **Key Observations:**

* Features such as **Area**, **ConvexArea**, and **EquivDiameter** show **significant positive skew** and a large number of **outliers**, particularly in the upper range. These attributes are strongly influenced by bean size, and their variability is expected due to natural differences across bean types.
* Features like **Solidity** and **ShapeFactor4** exhibit **extremely tight distributions**, with most values clustering near 1.0. These features may offer limited discriminative value due to their lack of variation.
* **AspectRatio**, **Roundness**, and **Compactness** display more **symmetric distributions** but still contain visible outliers.
* The **ShapeFactor** features, especially ShapeFactor1 and ShapeFactor2, operate on a much smaller numerical scale, and their distributions are concentrated near zero with occasional deviations.

1. **Interpretation:**

The boxplot analysis suggests that while some features capture high-variance and potentially informative geometric traits, others show minimal variation and may be less useful for class discrimination. Additionally, the presence of extreme values across several features supports the need for **normalization and potential outlier treatment**, both of which are addressed in the following sections.



**Figure 1:** *Boxplots of all 16 features in the Dry Bean dataset, arranged in a 4x4 grid layout.*

# Z-Score Normalization and Fisher Distance Analysis

To enable fair comparison between features with varying numerical scales, **Z-score normalization** was applied to all 16 numerical attributes. This transformation ensures that each feature has a mean of 0 and a standard deviation of 1, which is essential for distance-based analysis and clustering algorithms.

Following normalization, **Fisher Distance** was computed for each feature to evaluate its **class-separating capability**. The Fisher Score measures the ratio of **between-class variance to within-class variance**; higher scores indicate that a feature is more effective at distinguishing between classes.

**Methodology:**

* The normalized dataset was used to compute the Fisher Distance across **seven bean classes**.
* For each feature, the class-wise means and variances were calculated and aggregated to compute the Fisher Score.
* A multi-class adaptation of the Fisher formula was used to handle the 7-way class structure.
* The results were sorted and presented in **descending order** to highlight the most informative features.



**Table 1:** Bar chart of Fisher Distance scores for all 16 features (sorted in descending order).

The features with the highest Fisher scores—such as Area, ConvexArea, and EquivDiameter—are related to size and shape, confirming their strong relevance in classifying different bean types.

On the other hand, features like Solidity, Extent, and ShapeFactor4 exhibit very low Fisher scores, suggesting limited variation across classes and minimal contribution to class separability.

This analysis supports the idea that certain geometric and contour-based features are far more informative than compactness or solidity descriptors, especially for distinguishing among the seven bean categories.

# Principal Component Analysis and Fisher Score Comparison

To further understand the internal structure of the dataset and reduce dimensionality for visualization and clustering tasks, **Principal Component Analysis (PCA)** was applied to the Z-score normalized data. PCA transforms the original 16-dimensional space into a new set of orthogonal axes (principal components), ranked by the amount of variance they capture from the original data.

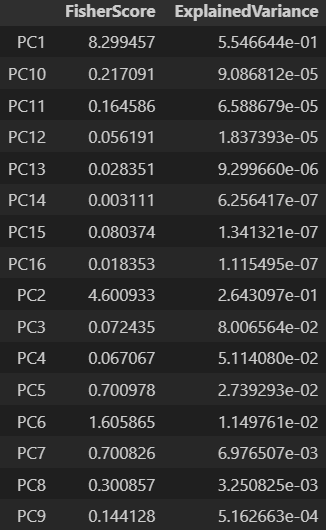
Each principal component (PC) is a linear combination of the original features. The components that explain the highest proportion of variance are typically the most informative in representing the underlying structure of the data.

#### ****Explained Variance:****

* The first principal component (PC1) alone accounts for a significant portion of the variance in the dataset.
* The cumulative explained variance of the top few components drops off quickly, indicating that a small number of components retain most of the data’s informative structure.

#### ****Fisher Distance on Principal Components:****

To assess the **class-separating power** of each principal component, **Fisher Distance** was again computed—this time on the transformed PCA space. Each PC was treated as a single feature, and its ability to discriminate between bean classes was measured using the same multi-class Fisher formula as in Section 4.



**Table 2:** Line chart comparing Fisher Scores and Explained Variance for each principal component (PC1 to PC16).

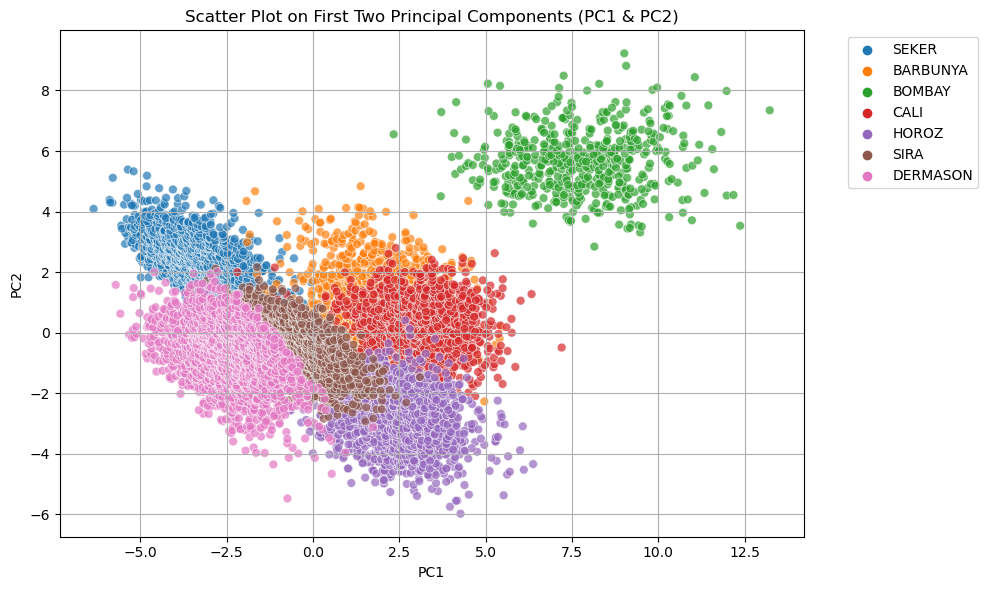
* **PC1** had both the **highest explained variance** and the **highest Fisher Score**, indicating that it effectively captures both general structure and class-relevant information.
* Interestingly, **PC2** and **PC6** displayed relatively high Fisher Scores despite capturing lower variance. This suggests that **lower-variance components can still carry class-discriminative information**.
* Conversely, some components (e.g., PC9 to PC15) contributed little to both variance and class separability, making them suitable candidates for dimensionality reduction.

While PCA is an unsupervised method focused solely on preserving variance, this analysis confirms that its leading components (especially PC1) also carry significant class-separating power. However, the results also reveal that **variance alone is not a perfect proxy** for discriminative utility. Combining PCA with supervised metrics like Fisher Distance offers a more complete picture of which components are most valuable for classification and clustering.

# Scatter Plots on PCA Projections

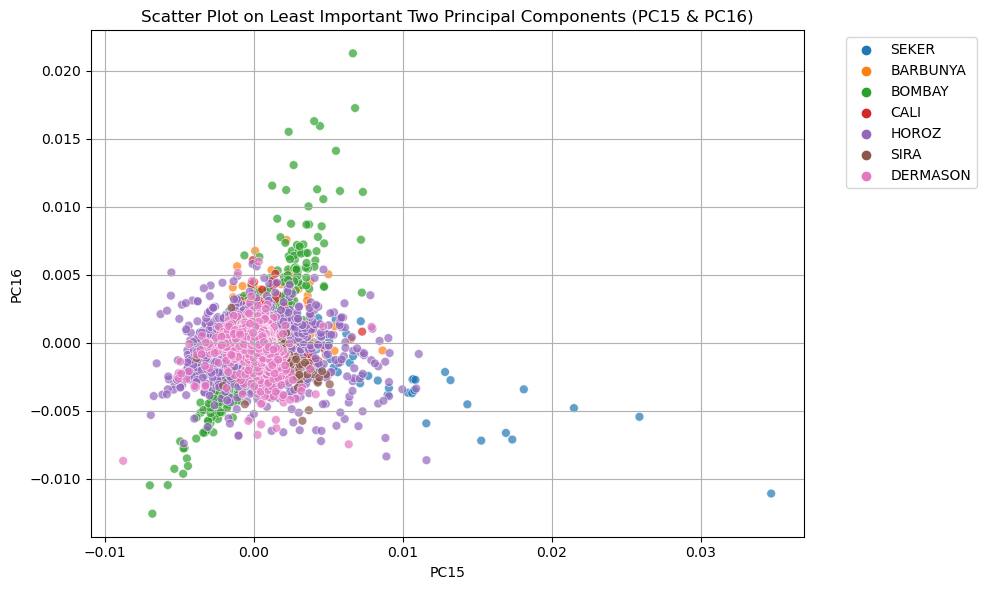
To visualize the distribution of bean classes in reduced dimensional space, the dataset was projected onto selected pairs of principal components. These 2D scatter plots provide insights into how well PCA captures **class structure** and highlights which components are the most useful for separating classes.

Two distinct projections were analyzed:



**Figure 2:** Scatter plot of the dataset on the first two principal components (PC1 vs PC2), color-coded by true class labels.

* This projection provides a **clear view of the dataset’s global structure**.
* Some classes such as **BOMBAY**, **DERMASON**, and **SEKER** appear well-separated and form **distinct clusters**.
* Other classes, particularly **CALI**, **SIRA**, and **BARBUNYA**, show **moderate overlap**, suggesting shared feature characteristics or projection distortion.
* Overall, the PC1–PC2 plot is effective for visualizing most class boundaries, justifying their use in clustering and further analysis.



**Figure 3:** Scatter plot of the dataset on the least informative components (PC15 vs PC16), color-coded by true class labels.

* This projection reveals **no significant class structure**.
* Samples from different classes are heavily **overlapping**, and the data points form an **unstructured cloud**.
* These components explain minimal variance and show weak Fisher scores, confirming their **low relevance** to both feature representation and class separation.

The contrast between PC1–PC2 and PC15–PC16 plots illustrates the importance of selecting the right components when visualizing high-dimensional data. While the leading principal components effectively preserve variance and class structure, the lower-ranked ones are mostly noise. This validates the use of PCA not only for dimensionality reduction but also for meaningful 2D class visualization.

# LDA Dimensionality Reduction and Evaluation

In addition to PCA, **Linear Discriminant Analysis (LDA)** was applied to reduce the dimensionality of the dataset while explicitly maximizing **class separability**. Unlike PCA, which is unsupervised and focuses on preserving overall variance, LDA is a **supervised technique** that constructs new axes to **maximize the distance between class means** and **minimize within-class variance**.

Since there are **7 distinct classes** in the Dry Bean dataset, LDA can generate up to **6 discriminant components**. For visualization purposes, the projection was limited to the **first two linear discriminants (LD1 and LD2)**.

metin, ekran görüntüsü, öykü gelişim çizgisi; kumpas; grafiğini çıkarma, diyagram içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**Figure 4:** Scatter plot of samples projected onto the first two LDA components (LD1 vs LD2), with color-coded class labels.

* The LDA projection shows **tight and compact clusters** for most classes.
* **BOMBAY** is notably **well-isolated**, while classes like **SEKER**, **DERMASON**, and **HOROZ** remain fairly distinct.
* Some classes still show partial overlap—particularly **CALI**, **SIRA**, and **BARBUNYA**—but the clusters are more **compact** than in PCA projections.

**Quantitative Evaluation using Silhouette Score**

To provide an objective comparison between PCA and LDA projections, the **Silhouette Score** was computed for both 2D representations. This score evaluates how similar a sample is to its own cluster compared to other clusters, ranging from -1 to 1.

**Silhouette Score (PCA PC1&PC2): 0.3347**

**Silhouette Score (LDA LD1&LD2): 0.2105**

**Note:** Silhouette score was calculated using ground truth class labels on the 2D projection coordinates.

* While **LDA provides visually compact and interpretable clusters**, its silhouette score is lower than that of PCA.
* This may be due to **inter-cluster proximity** in LDA space: classes are well-separated but located close to each other in some regions.
* **PCA**, although unsupervised, yields a better silhouette score in 2D, suggesting **better global spread** of clusters in the PC1–PC2 space.

LDA successfully emphasizes local class structure, making it well-suited for visualization and classification. However, PCA’s ability to preserve both variance and broad class boundaries in low dimensions may offer better overall separation in some cases. A combined evaluation using both visual and metric-based assessments provides a more complete understanding of projection quality.

# Clustering Analysis

To evaluate the latent structure of the Dry Bean dataset without using class labels, **unsupervised clustering** techniques were applied. These methods aim to reveal natural groupings among the samples based on their geometric and statistical characteristics. Four different algorithms were tested:

* **K-Means** and **DBSCAN** were applied to the PCA-reduced 2D space (PC1 & PC2),
* **t-SNE** and **Self-Organizing Maps (SOM)** were applied to the full normalized dataset.

#### ****K-Means Clustering (on PCA space)****

# metin, diyagram, ekran görüntüsü, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

**Figure 5:** K-Means clustering result (7 clusters) plotted on PC1 vs PC2, with samples colored by true class labels.

 K-Means was able to form **well-defined clusters**, especially for **BOMBAY**, **DERMASON**, and **SEKER**.

 Classes like **CALI**, **SIRA**, and **BARBUNYA** exhibited **some overlap**, due to their proximity in PCA space.

 The result aligns reasonably well with the ground truth, making K-Means a good baseline for this dataset.

**DBSCAN Clustering (on PCA space)**

metin, diyagram, ekran görüntüsü, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**Figure 6:** DBSCAN clustering result with ε=0.8 and min\_samples=10.

* DBSCAN identified only **three major groups** and labeled many samples as **noise**.
* Although **BOMBAY** was mostly captured within a single cluster, other classes were **merged or ignored**.
* The algorithm’s performance was hindered by **density variation** in the PCA-reduced space.

**t-SNE Visualization (on full normalized data)**

metin, harita, diyagram, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**Figure 7:** t-SNE 2D projection with samples color-coded by class label.

 t-SNE revealed **highly separated and compact clusters** across all seven bean types.

 Even classes that previously overlapped—such as **SIRA**, **BARBUNYA**, and **CALI**—are now distinguishable.

 This demonstrates t-SNE’s strength in **preserving local relationships** and mapping them into a visually meaningful space.

 Though not a clustering algorithm, t-SNE’s projections validate that the classes are inherently **well-separated**.

**Self-Organizing Maps (SOM)**

metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**Figure 10:** SOM 10x10 neuron grid, where each data sample is assigned to a best matching neuron and colored by class.

* SOM provided a **structured 2D representation** of the dataset based on topological similarity.
* Some classes (e.g., **SEKER**, **DERMASON**) formed **localized clusters**, while others were more dispersed.
* SOM is particularly useful for exploring **neighborhood relationships** and high-dimensional data manifolds, although its separation was less crisp compared to t-SNE.

**Summary of Clustering Observations:**

|  |  |  |
| --- | --- | --- |
| **Algorithm** | **Strengths** | **Limitations** |
| **K-Means** | Clear and distinct clusters for many classes | Moderate overlap for nearby classes |
| **DBSCAN** | Handles noise and non-linear clusters | Underperformed due to density imbalance |
| **t-SNE** | Best visual class separation | Non-deterministic, unsuitable for prediction |
| **SOM** | Preserves neighborhood structure | Less distinct cluster boundaries |

Among all methods, **t-SNE provided the clearest class separation**, while **K-Means** served as a simple but effective clustering baseline. **DBSCAN** struggled due to density differences in 2D PCA space, and **SOM** offered valuable structural insights. These results show that **clustering performance heavily depends on both data representation and algorithm selection**, reinforcing the need to combine dimensionality reduction with appropriate clustering techniques.

# Outlier Detection and Data Cleaning

Outliers can distort the statistical structure of a dataset, reducing model performance and obscuring cluster patterns. To improve the overall quality of the Dry Bean dataset, an **outlier detection and cleaning step** was performed using the **Z-score method**.

**Outlier Detection Method:**

Each feature in the normalized dataset was analyzed using its **Z-score**. Any sample with **at least one feature exceeding ±3 standard deviations** was classified as an outlier.

* A total of **1,124 outliers** were identified out of **13,611 samples**, accounting for approximately **8.26%** of the dataset.

**Data Cleaning Process:**

All identified outliers were removed, and the dataset was re-indexed. The remaining samples were then re-evaluated using PCA to assess the visual effect of cleaning.



**Figure 9:** Side-by-side PCA scatter plots before and after outlier removal.

* **Before cleaning**: The PCA projection displayed elongated clusters with **visible outliers**, especially at the edges.
* **After cleaning**: Clusters became more **compact and well-defined**, particularly for **DERMASON**, **SEKER**, and **BOMBAY**.
* The central region, which previously showed **dense class overlap**, became **less crowded**, improving visual separation between similar bean types like **SIRA**, **CALI**, and **BARBUNYA**.

The Z-score method provided a simple yet effective strategy for identifying extreme samples in this dataset. Removing only ~8% of the samples yielded **noticeable improvements** in visual clustering and dimensionality reduction outcomes. This demonstrates the value of targeted outlier detection as a **preprocessing step for pattern recognition, classification, and visualization**.

1. CONCLUSION AND DISCUSSION

This project conducted a comprehensive multivariate analysis of the Dry Bean dataset using a structured pipeline involving normalization, dimensionality reduction, clustering, and outlier detection. The results demonstrated the effectiveness of combining statistical methods with machine learning techniques to uncover patterns and evaluate feature utility.

Key findings include:

* **Boxplot analysis** revealed that some features, such as Area, ConvexArea, and Perimeter, displayed wide variation and potential outliers, while others like Solidity and ShapeFactor4 were nearly constant across classes.
* **Z-score normalization** allowed for meaningful comparison across features, and **Fisher Distance** identified the most discriminative attributes—primarily those related to bean size and geometry.
* **PCA** successfully reduced dimensionality while preserving class structure. Its first few components held both high variance and class-separating power, as validated through Fisher scores.
* **LDA** produced compact class clusters and was especially effective at isolating certain classes like BOMBAY. However, its silhouette score was lower than that of PCA, suggesting better global separation in PCA space.
* **Clustering analysis** highlighted the strengths and weaknesses of various unsupervised methods. **t-SNE** delivered the best visual separation, while **K-Means** offered practical clustering performance. **SOM** preserved topological relationships, and **DBSCAN** struggled with non-uniform densities.
* **Outlier removal** based on Z-score thresholds significantly enhanced cluster compactness and class distinction, improving the dataset’s quality for downstream tasks.

Overall, the project demonstrated how an integrated approach—using both supervised and unsupervised learning techniques—can provide deep insight into the structure and behavior of complex datasets. These findings underscore the importance of **feature analysis, dimensionality reduction, and data quality** as key components in successful data science workflows.

REFERENCES

The Implementation of the project and codes are available at github.

[1] <https://github.com/adguvener/x-ray-line-detection.git>

The dataset:

[2] ucimlrepo. (2023). *ucimlrepo: A lightweight interface to fetch UCI datasets* [Python package]. Retrieved from <https://github.com/uci-ml-repo/ucimlrepo>