CS60050

Machine Learning Assignment-3

Name-Diganta Mandal
Roll-22CS30062

Part-2

Anuran Calls Dataset

Dataset Overview

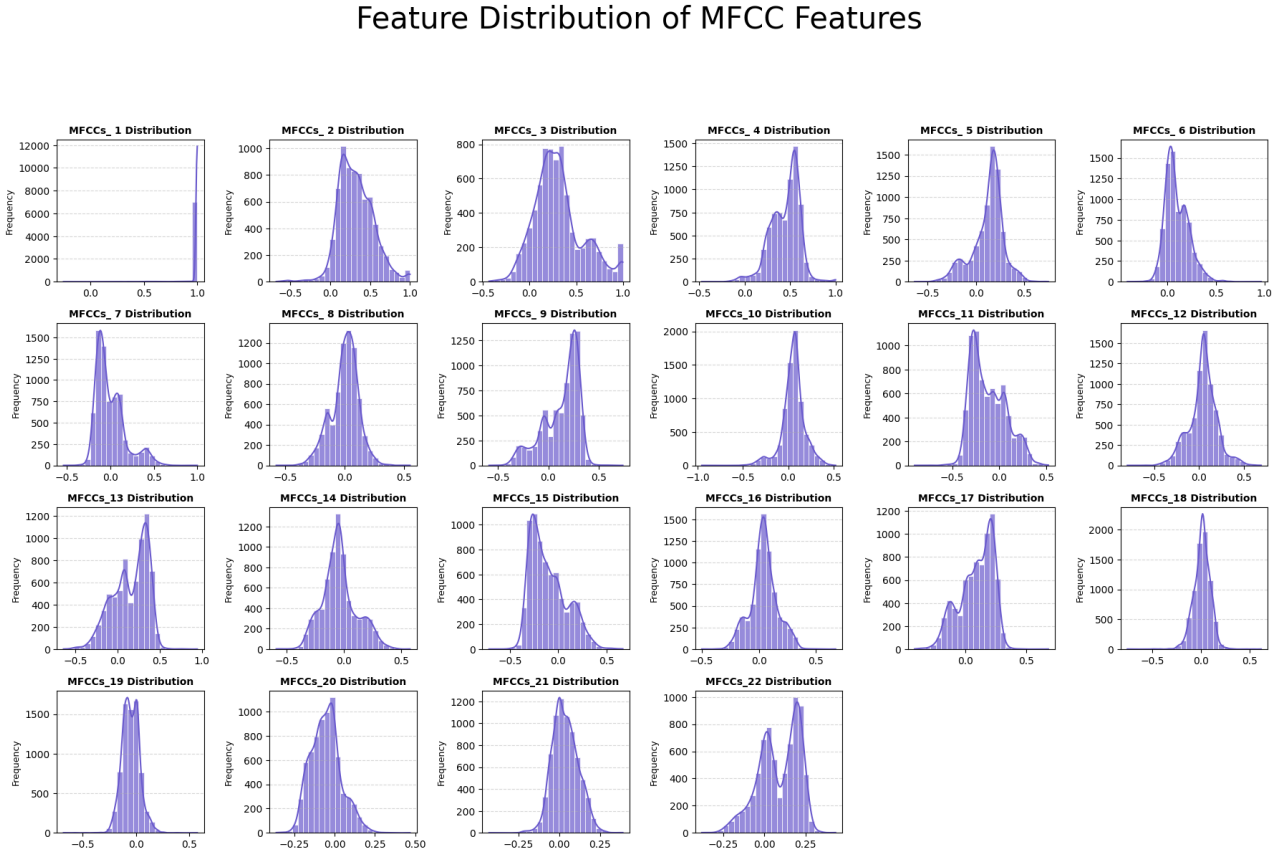
Features Overview

This dataset contains 7,195 instances of frog vocalizations extracted from 60 audio recordings, segmented into syllables. It includes labels for 4 families, 8 genera, and 10 species of frogs. Each syllable is represented by 22 Mel-Frequency Cepstral Coefficients(MFCC), normalized between -1 and 1.

Objectives

The primary objective of this analysis is to cluster the dataset using various clustering methods, with a major focus on K-means clustering and its types. Clustering aims to uncover patterns and relationships within the data, providing insights into the vocalizations of different frog species.

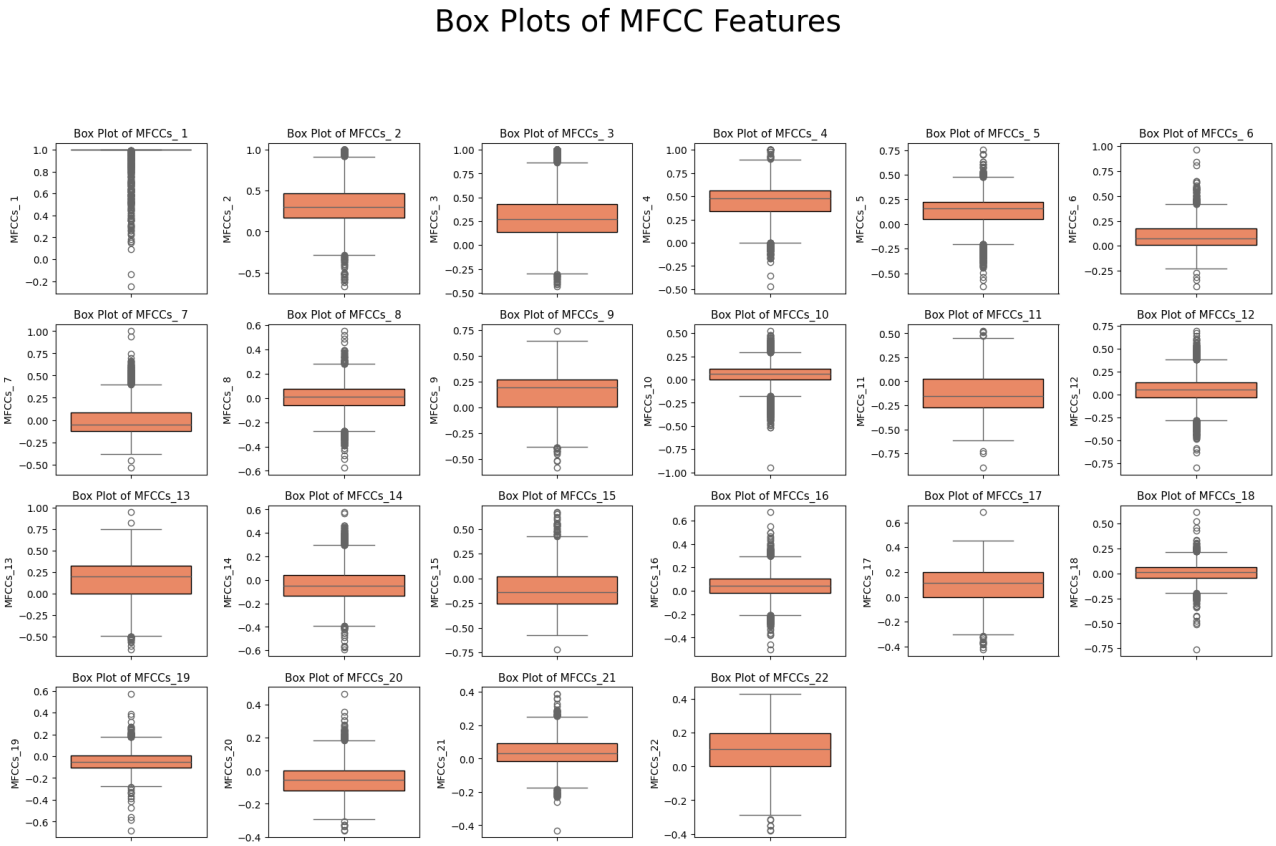
Feature Distribution



Brief analysis of the MFCC feature distributions:

1. **Skewed Distributions**: **MFCCs_1** and **MFCCs_10** show skew, possibly needing normalization.
2. **Bimodal Distributions**: **MFCCs_7** and **MFCCs_13** have bimodal shapes, potentially capturing different classes.
3. **Centered Around Zero**: Most features are centered around zero, suggesting they are mean-centered.
4. **High Variability**: **MFCCs_1** and **MFCCs_22** show wide ranges, indicating potential outliers or high variability.
5. **Compact Distributions**: **MFCCs_2** and **MFCCs_6** are more compact, with lower variability.

Outlier Detection



Brief analysis of the MFCC feature box plots:

1. **Numerous Outliers**: **MFCCs_1**, **MFCCs_10**, **MFCCs_14**, and **MFCCs_18** show a high number of outliers, indicating extreme values that may need further inspection.
2. **Minimal Outliers**: **MFCCs_2**, **MFCCs_3**, and **MFCCs_8** have fewer outliers, showing stable distributions around the median.
3. **High Variability**: **MFCCs_1** and **MFCCs_22** have wide ranges, suggesting potential scaling or normalization.
4. **Compact Ranges**: Features like **MFCCs_9** and **MFCCs_12** have tighter distributions with minimal spread.

Data preprocessing involves dealing with these factors using various transformations and scaling.

Feature Engineering

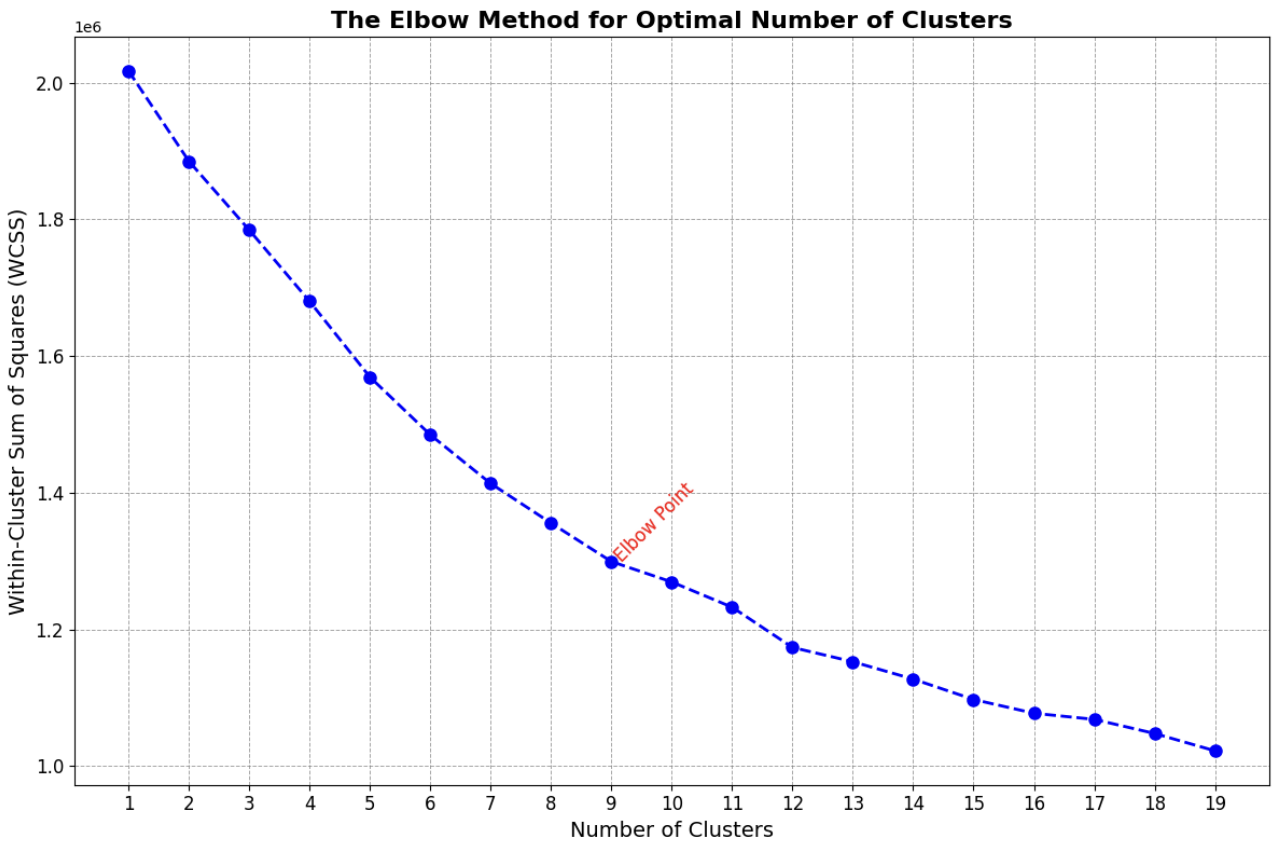
First of all polynomial features are obtained from the dataset features to train non-linearity present in data.

Next **Feature Correlation Analysis** is performed which involves identifying highly correlated pairs with correlation >=0.8.Those pairs are eliminated.

Label Encoding is performed on the non-numerical features.

Optimal Number of Clusters

Elbow Method

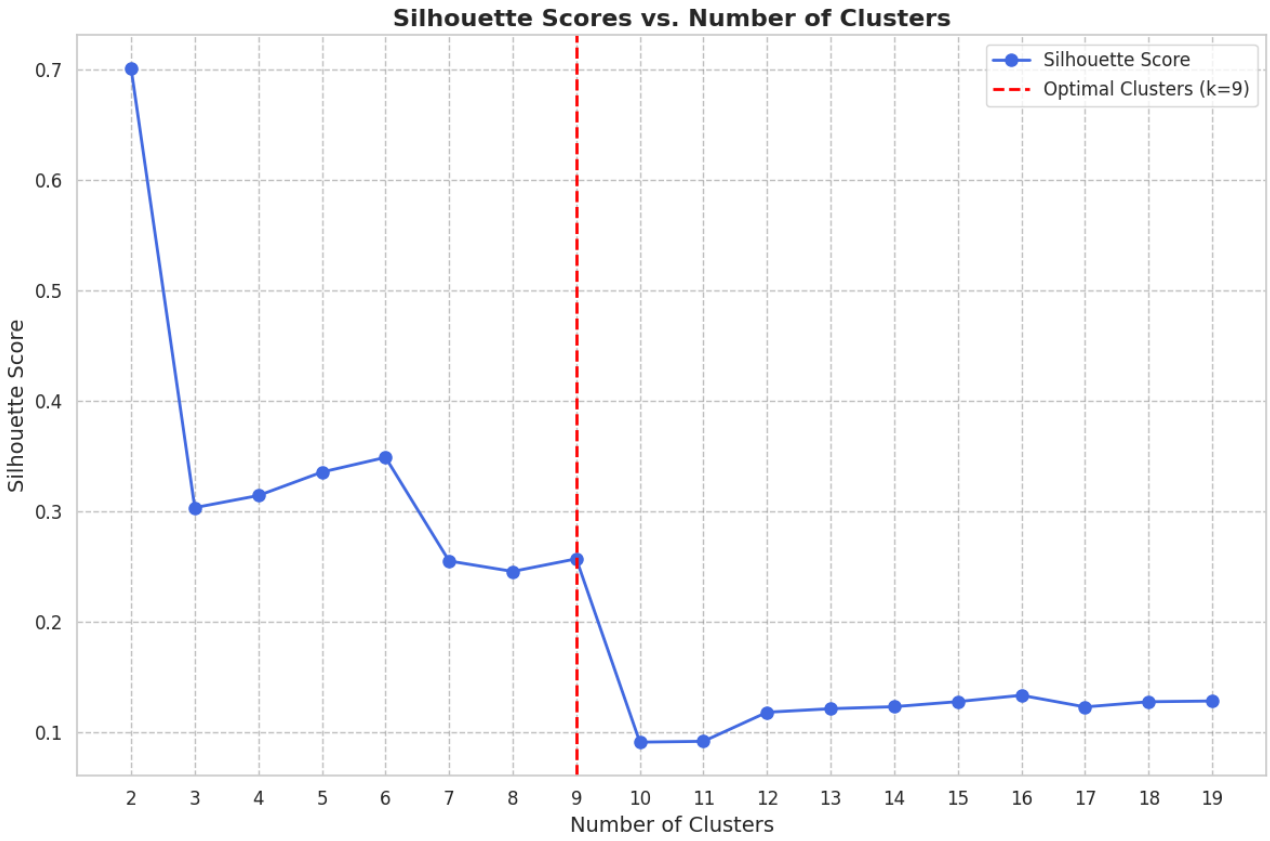


The elbow method is a technique used to determine the optimal number of clusters (k) in clustering algorithms, like K-means. It involves plotting the within-cluster sum of squares (WCSS) against the number of clusters. As the number of clusters increases, WCSS decreases, as clusters are more compact. However, after a certain point, the rate of decrease slows, forming an "elbow" in the graph.

In your plot, the elbow appears around **k = 9**, where the reduction in WCSS starts to level off. This suggests that **9 clusters** is an optimal choice, as adding more clusters beyond this point offers diminishing returns in reducing WCSS.

Now ,we verify our optimal choice by plotting the Silhouette Score for different values of clusters.

Silhouette Score



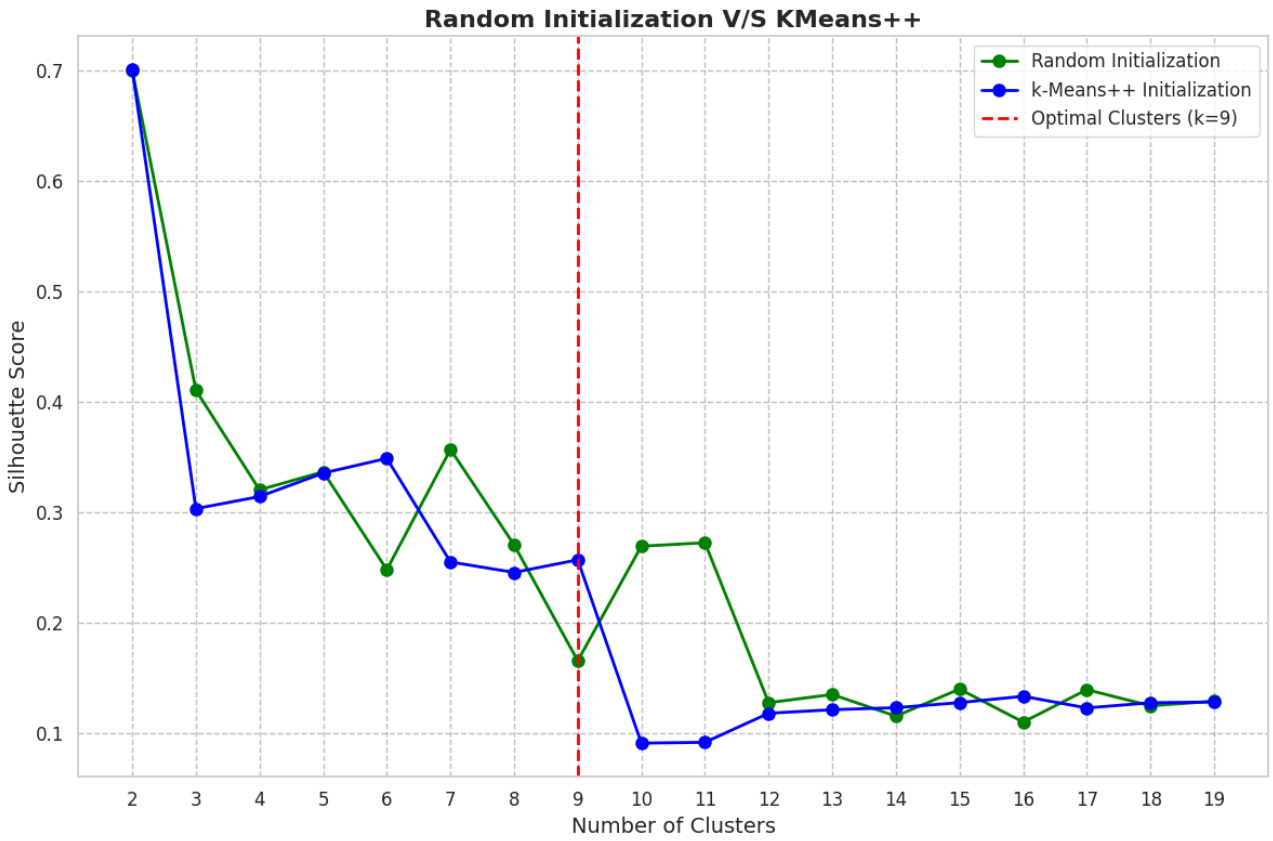
As the graph suggests it does reach a local maxima at 9 suggesting that k=9 is a good choice for number of clusters.
Very high value of Silhouette score for small number of clusters is because of the proximity of data points which lead to a higher score but will eventually end up underfitting the data

---

KMeans++ V/S Random Initialisation

Random Initialization: Centroids are chosen randomly from the data points. This can lead to poor clustering results or slow convergence if initial centroids are poorly placed.

K-means++: This is an improved initialization method where the first centroid is chosen randomly, and subsequent centroids are selected based on a probability proportional to their distance from the nearest existing centroid. This helps spread out the initial centroids, leading to faster convergence and better clustering results compared to random initialization

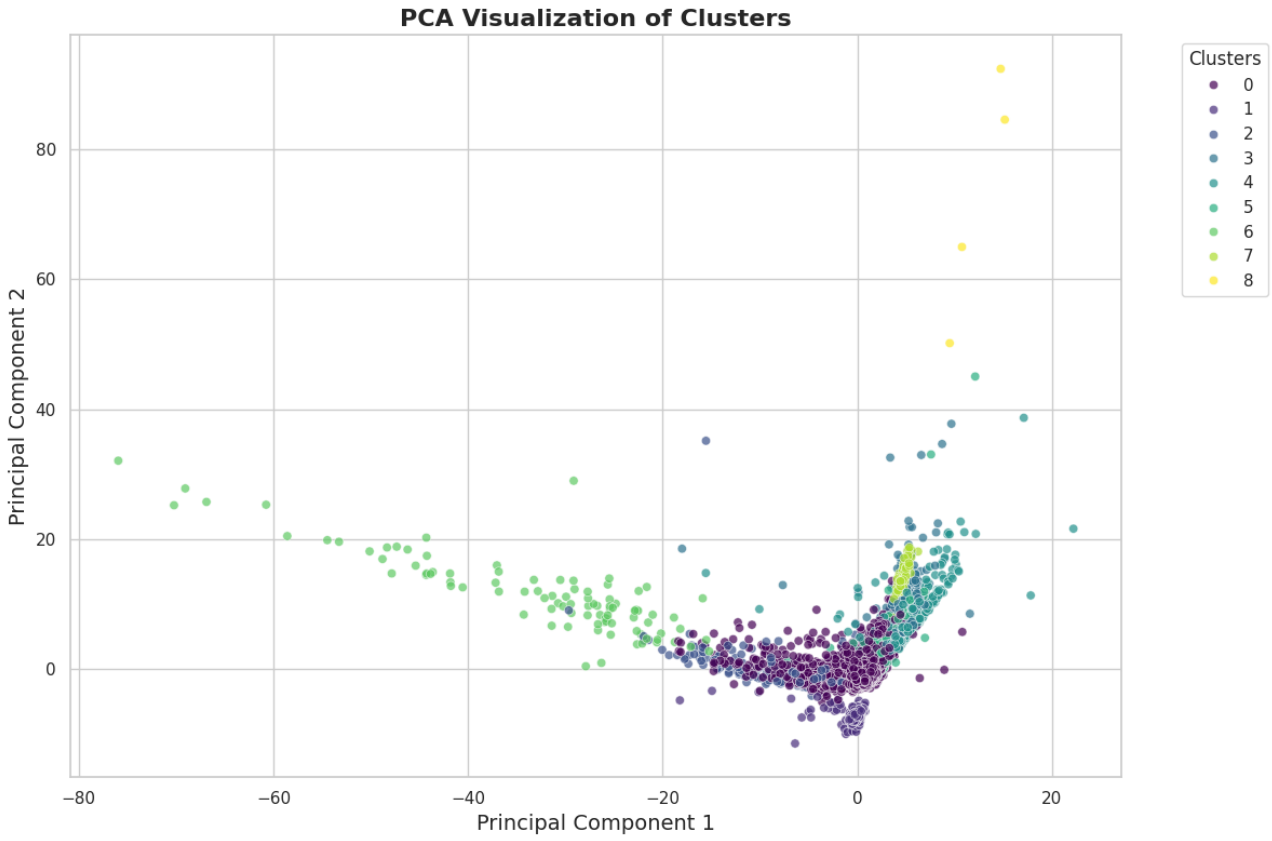


KMeans++ lead to much stable Silhouette score since it doesn't randomly choose centre point of clusters. Thus KMeans++ is useful in this case for analysing optimal value of number of clusters.

Cluster Visualisation

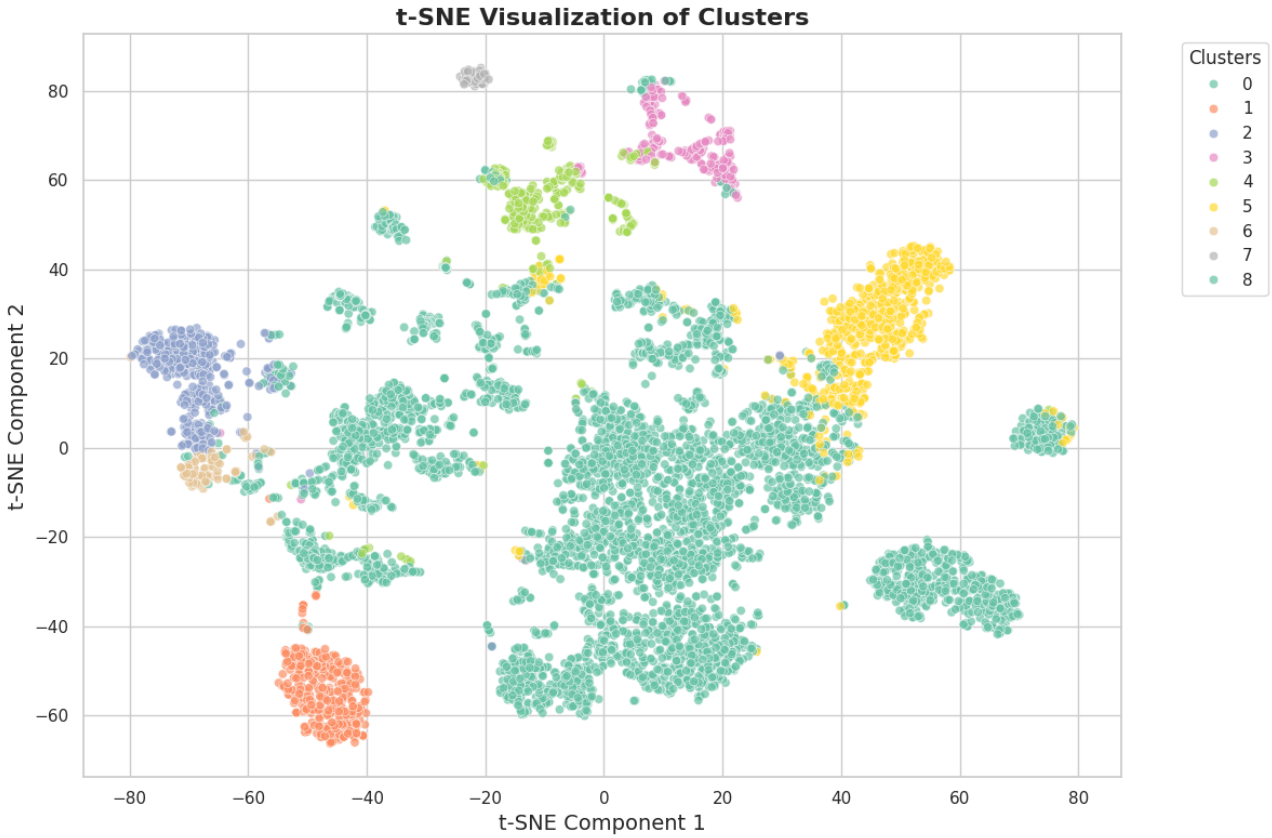
Dimensionality reduction is performed to visualise the clusters using two different methods:

PCA



- The clusters appear to be moderately separated, with some overlap between them. This indicates that the clustering algorithm has identified distinct groups within the data.
- The clusters exhibit varying densities. Some clusters are tightly packed, suggesting that the data points within those clusters are highly similar. Other clusters are more dispersed, indicating greater variability among the data points.
- There are a few isolated points that do not seem to belong to any specific cluster. These could be potential outliers or noise in the data.

t-SNE

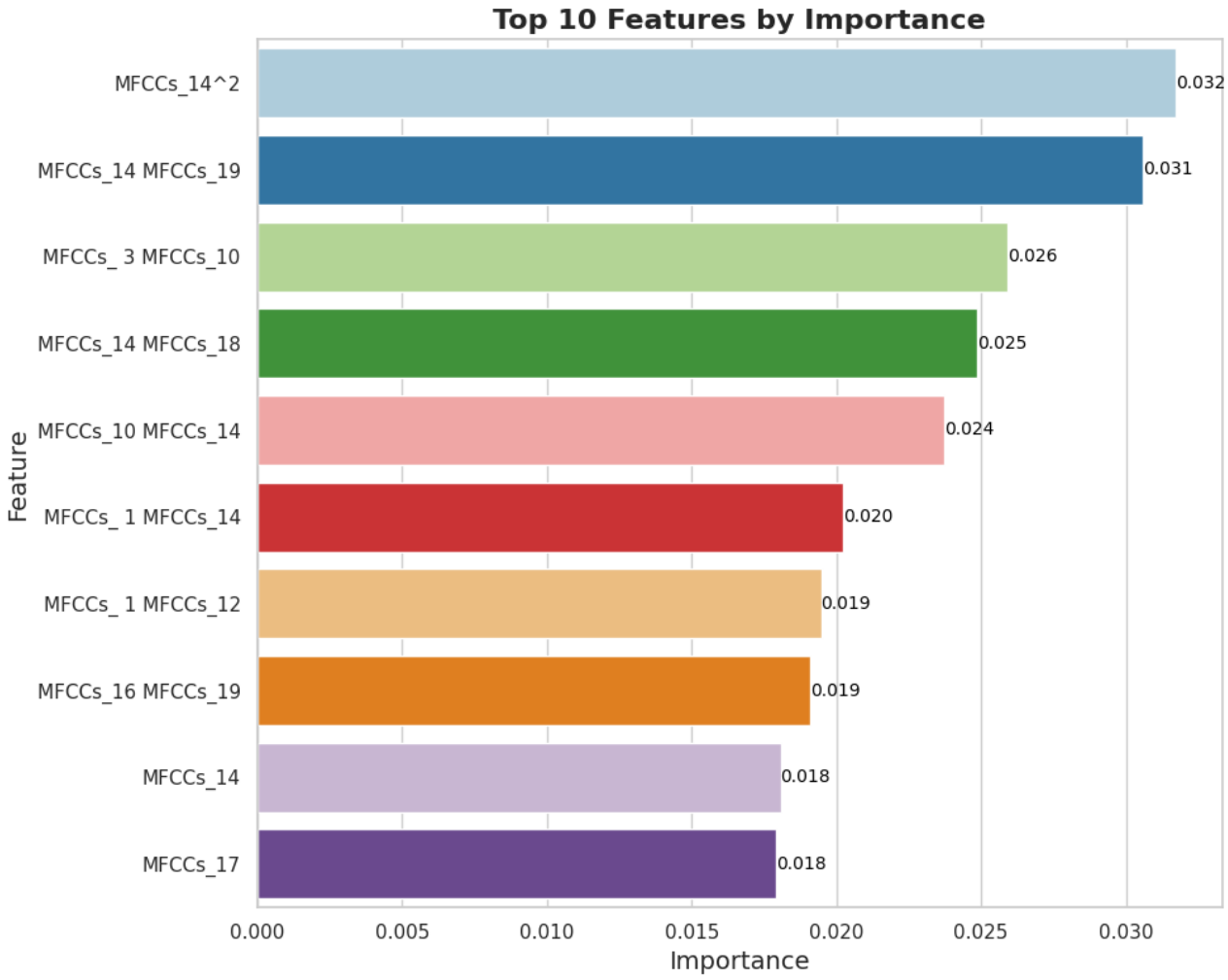


- The t-SNE visualization reveals the complex structure of the data and the effectiveness of the clustering algorithm in identifying meaningful groups.
- The separation of clusters suggests that the algorithm has captured distinct patterns or characteristics within the data.
- The varying densities and shapes of the clusters indicate that the data is not uniformly distributed and may contain substructures or subgroups.
- The presence of outliers highlights the need for further investigation to determine their significance.

Top Features using Random Forest

I used random forest classifier to get insight into the most important features during the clustering process

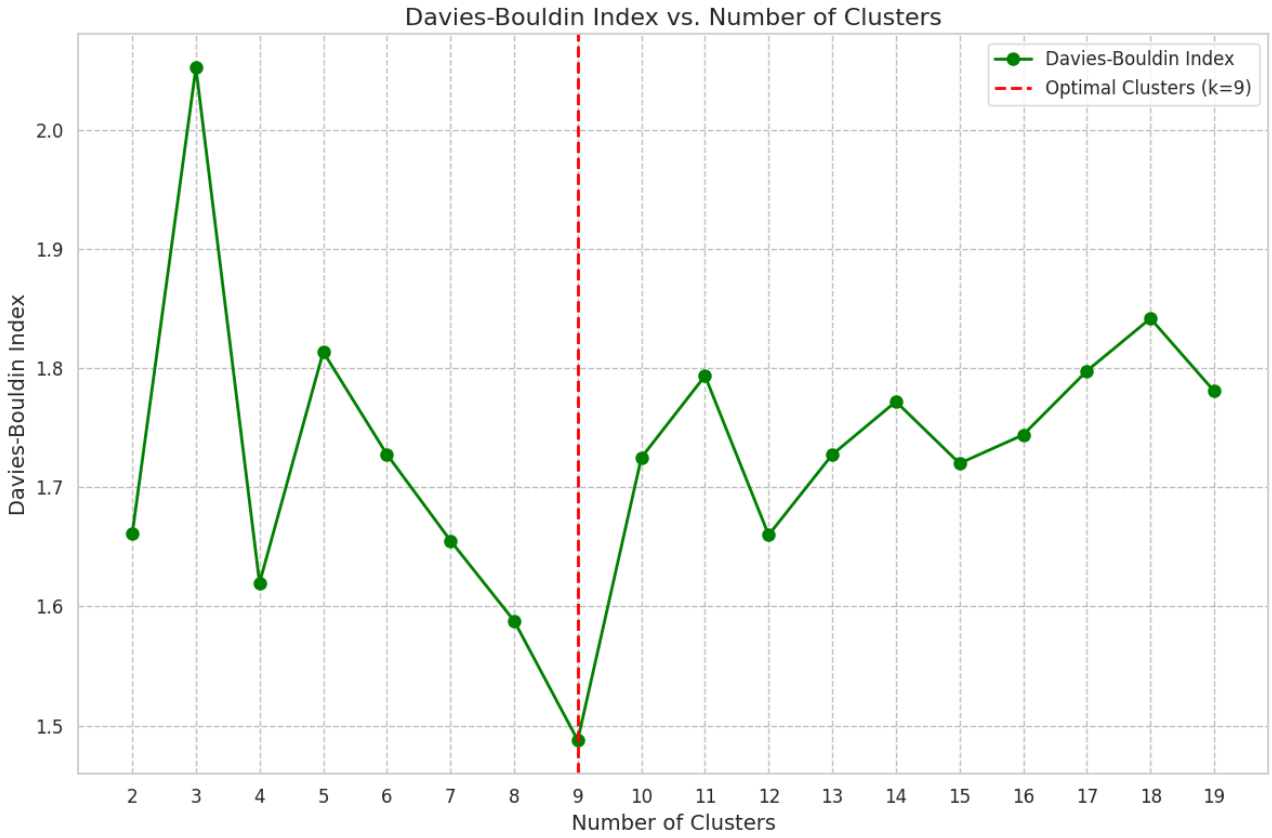
Random Forest is a powerful machine learning algorithm that's particularly well-suited for feature importance analysis due to its inherent nature. It's an ensemble method, meaning it combines multiple decision trees to make a more accurate prediction. Thus, it can capture complex interactions between features, making it suitable for identifying top features.



Initial 5 6 features are having very high importance but after that many features are present with almost similar accuracies.

Cluster Evaluation

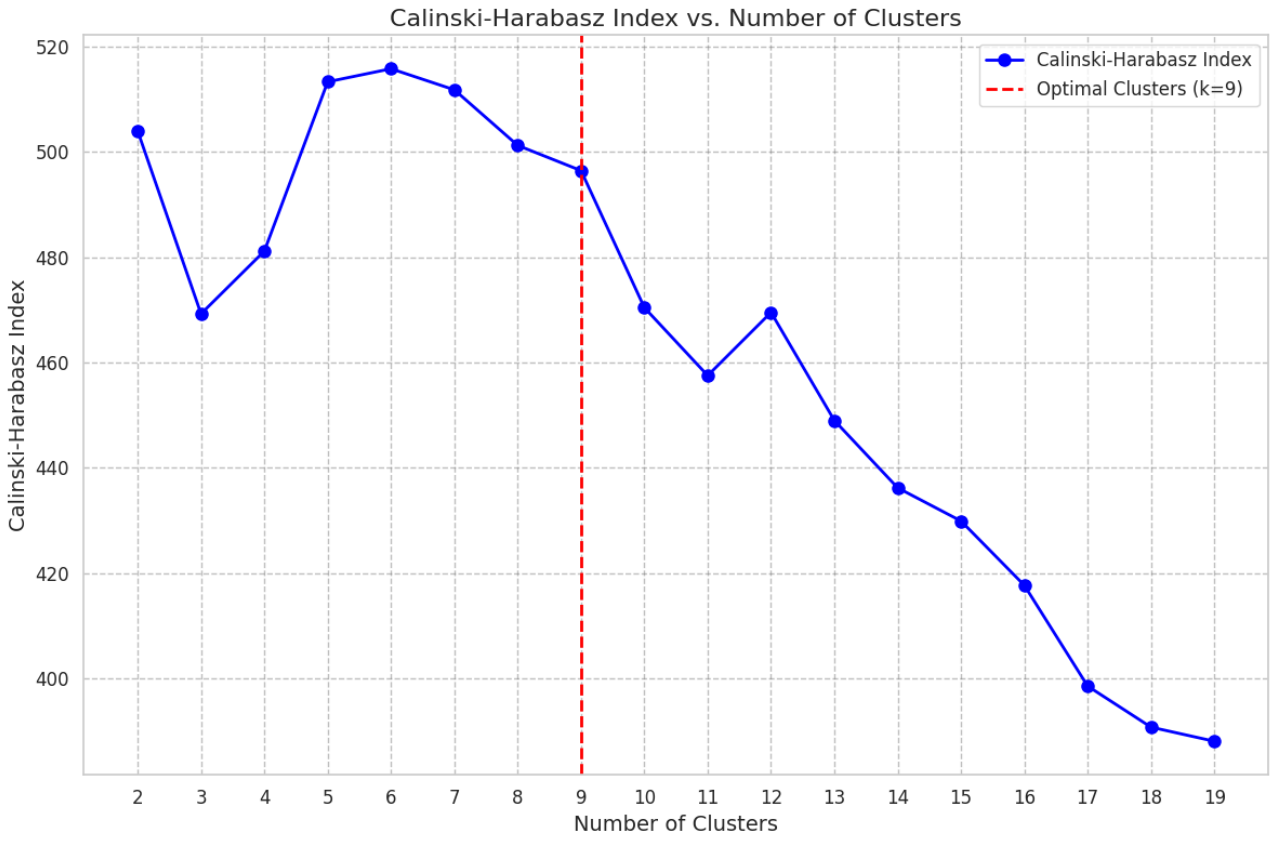
Davies-Boudin Index



Lower value of DBI suggests good clustering.Here, it reaches the minima for k=9.Thus asserting the statement that k=9 was indeed the optimal number of clusters. Also it tends to gradually increase with very high values of k suggesting bad clustering.

Calinski-Harabasz Index

Higher value of CHI is good for an optimal cluster.The index effectively balances the distances between cluster centroids (separation) and the spread of points within each cluster (compactness).



k=9 has a CHI of nearly 500 which suggests good clustering has taken place.
With increase in k CHI decreases significantly signalling very bad clustering.

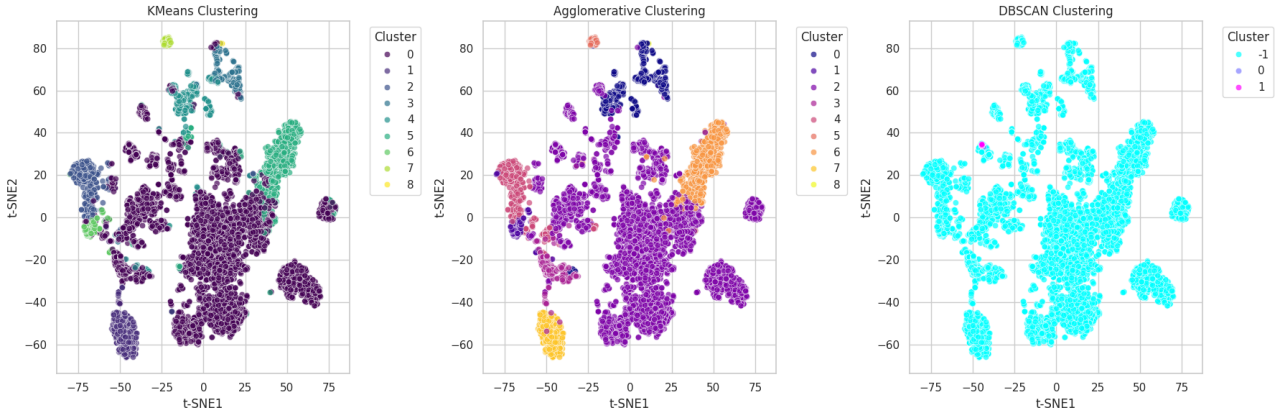
Comparisons with Other Clustering Algorithms

Besides **KMeans**, **Agglomerative Hierarchical Clustering** and **DBSCAN** is also used to train the model and comparison is made with each other.

Agglomerative Hierarchical Clustering (AHC) is a bottom-up clustering technique. It starts by treating each data point as an individual cluster. At each step, the two closest clusters are merged into a single cluster. This process continues until all data points belong to a single cluster. The result is a hierarchical structure represented as a dendrogram.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm. It groups together points that are closely packed (high-density regions) and marks as outliers points that lie alone in low-density regions. DBSCAN is effective for discovering clusters of arbitrary shape and handling noise in the data.

	Algorithm	Silhouette Score	Davies-Bouldin Index	Calinski-Harabasz Index
0	K-Means	0.256890	1.487729	496.421582
1	Agglomerative	0.203912	1.599611	452.844611
2	DBSCAN	-0.319160	2.167155	1.517146



Both KMeans and Hierarchical clustering perform substantially good on this dataset with Means having a relatively better performance

On the other hand,DBSCAN performed extremely poor due to the fact that DBSCAN is sensitive to the choice of density parameters.Since our data has clusters of varying density, which overlaps on each other DBSCAN performed extremely poor in this case.

Conclusion

After a comprehensive analysis of various clustering techniques, K-Means clustering with K-Means++ initialization emerged as the most suitable method for our dataset. While Hierarchical Clustering offer valuable insights, K-Means demonstrated superior performance in terms of computational efficiency and cluster quality.

K-Means++'s intelligent initialization strategy significantly improves the convergence speed and quality of the resulting clusters. Its ability to handle large datasets efficiently and produce well-defined clusters makes it a robust choice for our specific use case.

By carefully selecting the optimal number of clusters (K) and tuning the hyperparameters, I was able to achieve a high-quality clustering solution that effectively groups similar data points together.
