Ravi Rajpurohit UTA ID - 1002079916

MACHINE LEARNING
FINAL PROJECT REPORT

Introduction -

Hierarchical Clustering for Seed Categorization. Implementing an unsupervised hierarchical clustering algorithm on the UCI seed dataset to divide it into clusters. Using the derived clusters as labels for a subsequent K nearest neighbor classifier to identify the seed category.

Building multiple clusters iteratively to experiment and figure out the best number of clusters using the elbow method and KNN.

Brief about the dataset -

The UCI seed dataset has been used. It is a multivariate dataset with 210 total instances and 7 attributes. The data points are the measurements of the geometrical properties of kernels belonging to three different varieties of wheat - labeled as 1,2, and 3. According to the dataset documentation, all seven real-valued features were constructed by a soft X-ray technique, and the GRAINS package.

The dataset can be downloaded from - http://archive.ics.uci.edu/ml/datasets/seeds

How to run the code -

The best way to understand the solution is to look at the interactive Python notebook file - *main.ipynb*
You can also run Python in a terminal window and type python *main.py*, to run the .py file.
After running the code, the visual output is available in the outputs directory as a png file with the name - *best_number_of_clusters_visualization.png*

Please refer to the *readme.md* file for details of the directory structure for this project.

NOTE -
This project requires the following libraries to run successfully -

- Numpy
- Matplotlib
- Math (built-in with Python)
- Counter (built-in with Python)

A requirements.txt file has been made mentioning the external libraries used with version numbers.

Solution Approach -

Below are the detailed steps for my approach to this project's solution.

Data Preparation -

The dataset consists of 7 features and 1 target label, all of which are of float type. The dataset was loaded and organized into the necessary data structures.

- The labels and features are separated into Numpy arrays.
- The features are normalized using the standard scaler formula.
- The labels are already in integer format, hence do not require processing.

Hierarchical Clustering:

Hierarchical clustering is an unsupervised learning algorithm used to group similar data points into clusters based on their similarity. It creates a hierarchy of clusters by iteratively merging or splitting clusters, forming a tree-like structure known as a dendrogram. The algorithm starts with each data point as a separate cluster and gradually combines the most similar clusters until a stopping criterion is met. This process results in a nested structure of clusters, where each data point belongs to a specific cluster. Hierarchical clustering can be performed using two main approaches: agglomerative (bottom-up) and divisive (top-down).
Agglomerative clustering, the more commonly used approach, starts with individual data points as clusters and merges them based on similarity, while divisive clustering starts with one cluster containing all data points and recursively splits them.

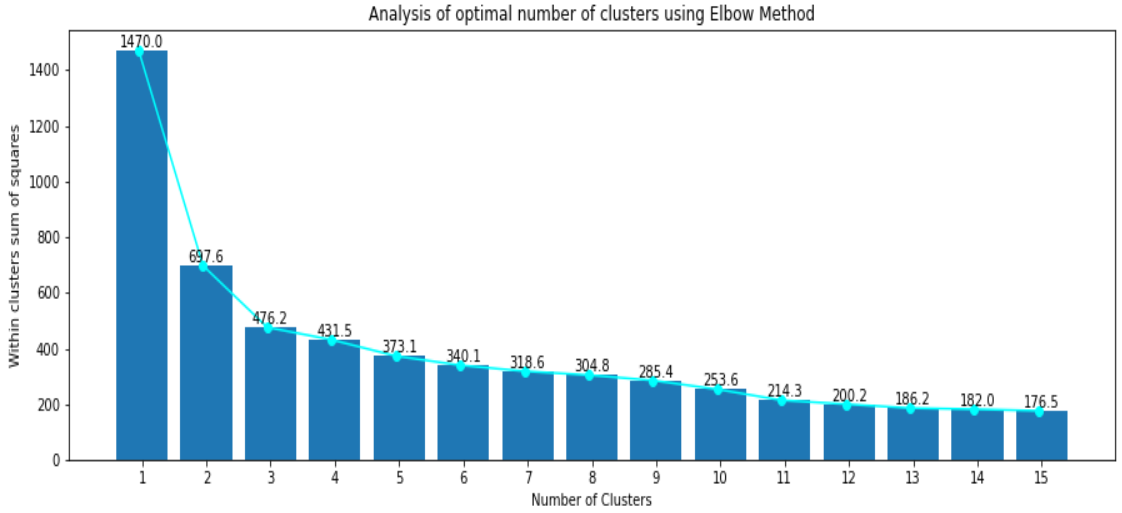
- The hierarchical clustering algorithm was implemented using the *agglomerative approach*.
- The algorithm started with each data point as a separate cluster and iteratively merged the closest clusters based on the *complete linkage criterion* - O(n2) complexity
- The *Euclidean distance* metric was used to calculate the distances between data points.
- The clustering was performed for different numbers of clusters, and the within-cluster sum of squares (WCSS) values were calculated.

Determining the Optimal Number of Clusters:

Elbow Method - It is a technique used to determine the optimal number of clusters in a dataset for clustering algorithms. It involves plotting the within-cluster sum of squares (WCSS) against the number of clusters and identifying the "elbow" point, which is the point of inflection on the plot. The elbow point represents a trade-off between the WCSS and the number of clusters.

The optimal number of clusters is often chosen as the point where adding more clusters does not significantly reduce the WCSS. The elbow method helps in finding a balance between model complexity (more clusters) and clustering quality (lower WCSS), providing insights into the appropriate number of clusters to use for further analysis.

- The within-cluster sum of squares (WCSS) values were examined for different numbers of clusters.
- The number of clusters that yielded the lowest WCSS value was selected as the optimal number of clusters.
- Once the WCSS value reaches a saturation point, according to the elbow method, the subsequent values even if slightly lower are not optimal.
- In this project, the best WCSS value was obtained when *using 3 clusters*.



KNN Classification:

KNN (K-Nearest Neighbors) classification is a supervised learning algorithm used for classifying data points based on their distance from neighboring data points. It operates on the principle that similar instances tend to belong to the same class. Given a test instance, KNN finds the K nearest neighbors in the training data based on the Euclidean distance metric and assigns the majority class label among those neighbors as the predicted label for the test instance.

KNN does not require training or building a model but instead uses the entire training dataset for classification. It is a simple yet effective algorithm, suitable for both binary and multiclass classification problems.

- The KNN classifier was employed to predict the target labels using the hierarchical clustering results.

- The cluster IDs obtained from hierarchical clustering were used as labels for the KNN classifier.
- The *Euclidean distance* metric was utilized to measure the distances between the test instance and the training data points.
- The K nearest neighbors (where K is a chosen parameter) were considered for classification.
- The most common label among the nearest neighbors was assigned as the predicted label for the test instance.

Evaluation:

- The accuracy of the KNN classifier was assessed by comparing the predicted labels with the true labels.
- The accuracy was calculated for the *best WCSS value obtained with 3 clusters*.
- In this project, the KNN classifier achieved a *test accuracy of 95%*.

This approach involved the combination of unsupervised hierarchical clustering for grouping the data and supervised KNN classification for label prediction, showcasing the potential of these techniques for similar machine-learning tasks.

Conclusion -

Based on the analysis of the hierarchical clustering and KNN classification results, we can conclude that -

- The within-cluster sum of squares (WCSS) values were calculated for different numbers of clusters (1 to 12).
- Among the tested numbers of clusters, the best WCSS value was obtained when using 3 clusters.
- This indicates that the dataset can be effectively divided into three distinct groups based on the given features.
- The KNN classifier was applied using the hierarchical clustering results.
- The KNN classifier achieved an accuracy of 95% when using the best WCSS value obtained with 3 clusters.
- This high accuracy suggests that the features used in the dataset are informative and sufficient for predicting the target labels.

The combination of hierarchical clustering and KNN classification yielded promising results for this project. The hierarchical clustering analysis identified an optimal number of clusters, which helped in grouping similar data points together. The KNN classifier effectively utilized the cluster IDs obtained from hierarchical clustering as labels for predicting the target labels with a high test accuracy of 95%. This suggests that the

clustering results provided meaningful information for classification, and the chosen features were informative in distinguishing the target labels.

Based on these findings, it can be concluded that the hierarchical clustering technique, followed by KNN classification, is a suitable approach for analyzing and predicting the target labels in this dataset.

References -

Standard scaler -
https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

Hierarchical Clustering -
https://www.analyticsvidhya.com/blog/2019/05/beginners-guide-hierarchical-clustering/
Pulkit Sharma — Published On May 27, 2019 and Last Modified On April 20th, 2023

KNN Algorithm -
https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/
Tavish Srivastava — Published On March 26, 2018 and Last Modified On April 26th, 2023

Elbow Method
https://towardsdatascience.com/cheat-sheet-to-implementing-7-methods-for-selecting-optimal-number-of-clusters-in-python-898241e1d6ad
Indraneel Dutta Baruah - Published on October 25, 2020 - Towards Data Science
https://www.analyticsvidhya.com/blog/2021/01/in-depth-intuition-of-k-means-clustering-algorithm-in-machine-learning/
Basil Saji — Published On January 20, 2021 and Last Modified On April 26th, 2023