Assignment: Identifying Groups of Similar Wines

Project Submitted to: Anubavam

Submitted By:

Monisha R

Ph No: +91 8610831486

Mail: monishavenkatesh7@gmail.com

Batch: MDTM25

GUVI

Assignment: Identifying Groups of Similar Wines

1. Introduction

In this project, we aim to group wines with similar attributes using clustering techniques. The wine dataset contains features such as alcohol content, color intensity, and more. The objective is to implement custom matrix operations for clustering, including calculating Euclidean and Weighted Euclidean distances, and to identify distinct groups based on these attributes.

2. Methodology

We developed a clustering model using custom matrix operations to perform the following tasks:

- **Data Standardization:** Each feature in the dataset is standardized.
- **Distance Calculation:** Both Euclidean and weighted Euclidean distances between data points are calculated.
- **Centroid Calculation:** For each cluster, centroids are computed.
- **Cluster Assignment:** Data points are assigned to the nearest centroid using weighted distances.
- **Weight Update:** Feature weights are iteratively updated based on within-cluster and between-cluster separations.

The process is repeated until convergence, i.e., when cluster assignments no longer change.

3.Code Implementation

The core of this project revolves around implementing clustering techniques using matrix operations, distance metrics, and weighted distance calculations. Below, we will delve into each function and method, explaining the strategies and techniques applied.

3.1. Matrix Class

The matrix class encapsulates a 2D matrix, which is central to various operations such as standardization, distance calculation, and clustering. It contains methods to load data from a CSV, standardize the matrix, and compute Euclidean and weighted Euclidean distances.

3.1.1. init and load_from_csv Methods

Strategy: The constructor initializes a matrix object, optionally loading a dataset from a CSV file. If a filename is provided, the load_from_csv method reads the CSV file into a NumPy array, allowing easy manipulation of the matrix.

3.1.2. Standardization Method

Technique: The data is standardized using the formula:

$$D'_{ij} = \frac{D_{ij} - \overline{D_j}}{max(D_j) - min(D_j)}$$

Each element is normalized by subtracting the mean and dividing by the range (max-min) of the respective column. This ensures all data features are on the same scale, improving clustering performance.

3.1.3. Euclidean Distance Calculation

Strategy: The Euclidean distance between a row from the current matrix and all rows in other_matrix is calculated. This is done by:

1. Finding the difference between corresponding elements,
2. Squaring the differences,
3. Summing the squared differences,
4. Taking the square root of the sum.

The result is a vector of distances between the selected row and all rows in the other matrix.

3.1.4. Weighted Euclidean Distance

Technique: Weighted Euclidean distance incorporates feature-specific weights during distance calculation, which adjusts the contribution of each feature to the total distance. This method follows the same process as Euclidean distance but multiplies the squared differences by the respective weights before summing them.

3.2. Clustering Functions

Clustering is achieved by iteratively updating the cluster centroids and adjusting weights based on the separation within and between clusters.

Assignment: Identifying Groups of Similar Wines

3.2.1. Initial Weights Calculation

Strategy: This function generates random weights between 0 and 1 for each feature and normalizes them so their sum equals 1. The weights help control the influence of each feature on the clustering process.

3.2.2. Centroid Calculation

Technique: The centroids are recalculated based on the current cluster assignments. For each cluster k, the mean of all points assigned to that cluster is computed for every feature. If no points are assigned to a cluster, a random point is chosen as the new centroid.

3.2.3. Separation Within and Between Clusters

Strategy: The separation within clusters measures the sum of squared distances between data points and their corresponding centroids. The result is used to evaluate the compactness of clusters. Similarly, get_separation_between measures the separation between cluster centroids and the global mean of the data.

3.2.4. Cluster Assignment and Weight Update

Technique: This function iteratively assigns each data point to the nearest centroid using weighted distances. The centroids and weights are updated until the cluster assignments converge (i.e., no changes occur). The weight update is based on the separation between and within clusters, ensuring that features contributing more to between-cluster separation receive higher weights.

4. Results

After running the code with varying cluster sizes (from 2 to 10 clusters) and multiple iterations, the following cluster frequency distributions were observed:

Cluster 2: {Cluster 1: X instances, Cluster 2: Y instances}

Cluster 3: {Cluster 1: A instances, Cluster 2: B instances, Cluster 3: C instances}

...

5. Conclusion

The custom matrix-based clustering algorithm successfully grouped wines into distinct clusters based on their attributes. The iterative process of updating feature weights improved cluster assignments, providing more accurate separation between the groups. This method can be extended to other datasets for similar clustering tasks.

References

Guvi Classes

Anubhavam Project discussion

https://youtube.com/playlist?list=PLeo1K3hjS3utXiAr1FqrssqNU1Q0ai84x&si=PodXx05t0HByNUjC

https://youtube.com/playlist?list=PLeo1K3hjS3uu_n_a__MI_KktGTLYopZ12&si=wupSrJFmUwDYPaq9