

Assignment 3 – Report

Name – Harin Rishabh

UBIT Name – harinris

1.

a. The dataset was loaded into a dataframe using pandas.

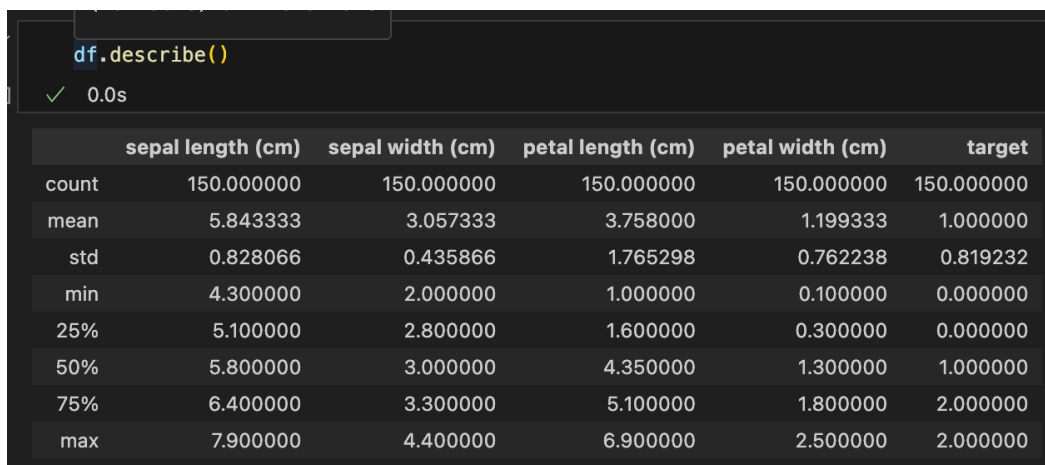
b. Target Classes: ['setosa', 'versicolour', 'virginica']

Feature Names: ['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width (cm)']

Number of Features: 4

Number of Samples: 150

The basic statistics for each column are:



```
df.describe()
```

| | sepal length (cm) | sepal width (cm) | petal length (cm) | petal width (cm) | target |
|-------|-------------------|------------------|-------------------|------------------|------------|
| count | 150.000000 | 150.000000 | 150.000000 | 150.000000 | 150.000000 |
| mean | 5.843333 | 3.057333 | 3.758000 | 1.199333 | 1.000000 |
| std | 0.828066 | 0.435866 | 1.765298 | 0.762238 | 0.819232 |
| min | 4.300000 | 2.000000 | 1.000000 | 0.100000 | 0.000000 |
| 25% | 5.100000 | 2.800000 | 1.600000 | 0.300000 | 0.000000 |
| 50% | 5.800000 | 3.000000 | 4.350000 | 1.300000 | 1.000000 |
| 75% | 6.400000 | 3.300000 | 5.100000 | 1.800000 | 2.000000 |
| max | 7.900000 | 4.400000 | 6.900000 | 2.500000 | 2.000000 |

2.

c. There are no missing values in this dataset.

d. The dataframe was scaled using Standard Scaler. We remove the target variable to avoid scaling it and then add it back to the dataframe.

3. e.

K-means Clustering:

Assumptions:

- Assumes that the data points belong to one of the predefined clusters.
- Assumes that clusters are spherical have similar shapes and sizes.
- Assumes clusters have similar density.

Advantages:

- Simple and easy to implement.
- Computationally efficient, making it suitable for large datasets.
- Works well when clusters are well-separated, and their shape is spherical.

Disadvantages:

- Requires specifying the number of clusters beforehand, which might not always be known.
- Sensitive to the initial choice of cluster centroids, which can lead to different results.
- Doesn't work well with non-linear cluster boundaries or clusters of different sizes and densities.

Hierarchical Clustering:**Assumptions:**

- Doesn't make any explicit assumptions about the shape or size of clusters.
- Builds a hierarchy of clusters by iteratively merging or splitting clusters based on distance.

Advantages:

- No need to specify the number of clusters beforehand, as it produces a dendrogram that can be cut at different levels.
- Can capture clusters of arbitrary shapes and sizes.
- Provides insights into the relationships between data points by visualizing the dendrogram.

Disadvantages:

- Computationally intensive, especially for large datasets, as it requires calculating distances between all pairs of data points.
- Output can be sensitive to the choice of distance metric and linkage method.
- Interpretation of the dendrogram can be subjective, requiring manual inspection to determine the optimal number of clusters.

iii. DBSCAN (Density-Based Spatial Clustering of Applications with Noise):**Assumptions:**

- Assumes clusters are dense regions separated by sparser regions.
- Doesn't assume any specific shape or size of clusters.

Advantages

- Can identify clusters of arbitrary shapes and sizes.
- Robust to noise and outliers, as it classifies points that are not in any dense region as noise.
- Doesn't require specifying the number of clusters beforehand.

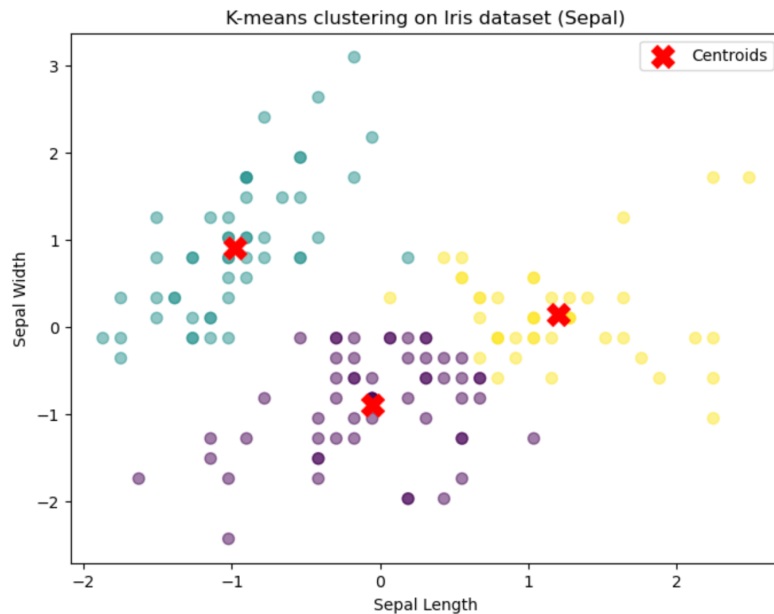
Disadvantages:

- Requires tuning of two hyperparameters (eps and min_samples), which can be challenging, especially for high-dimensional or large datasets.
- Struggles with clusters of varying densities.
- Computationally expensive, especially for large datasets, as it needs to compute the distance between each pair of data points.

Clustering Models:

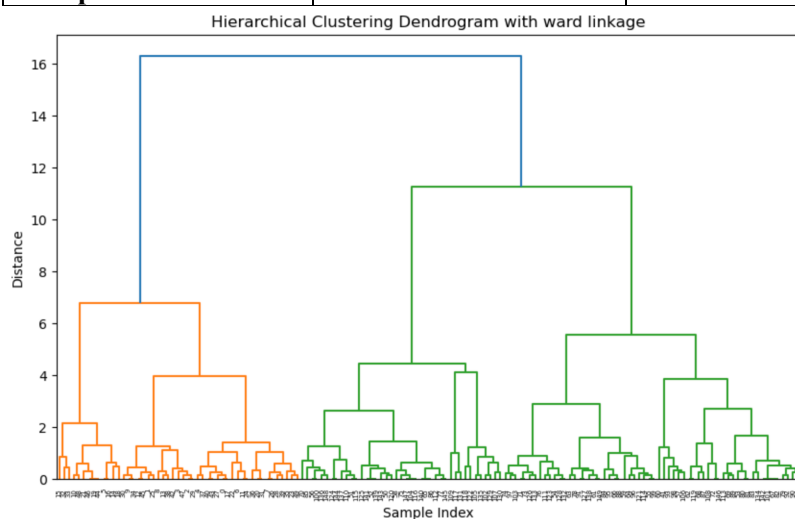
1. K-means:

| No. of clusters | Silhouette Score | Davies-Bouldin Index | Calinski-Harabasz Index |
|----------------------|------------------|----------------------|-------------------------|
| n_cluster = 2 | 0.44787 | 0.85988 | 117.94516 |
| n_cluster = 3 | 0.43887 | 0.78911 | 141.36488 |
| n_cluster = 4 | 0.41828 | 0.75648 | 134.37999 |



2. Hierarchical Clustering:

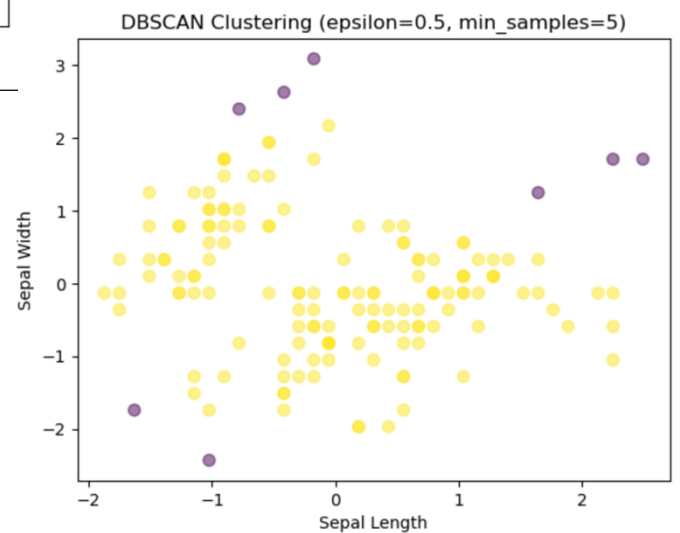
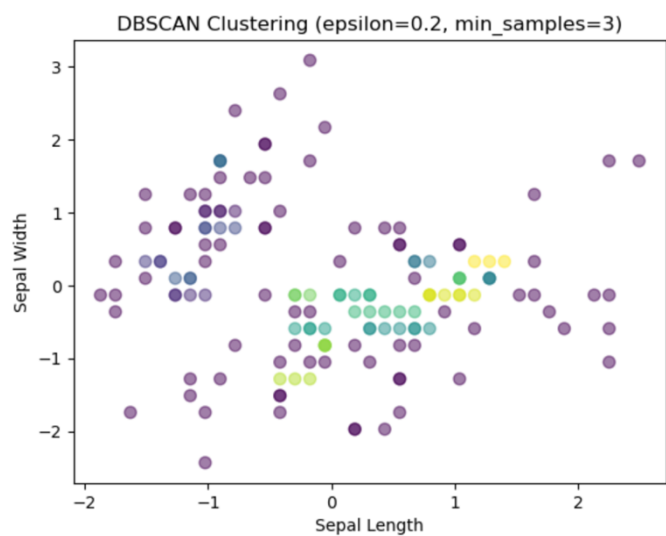
| Linkage Type | Silhouette Score | Davies-Bouldin Index | Calinski-Harabasz Index |
|-----------------|------------------|----------------------|-------------------------|
| Ward | 0.43859 | 0.79832 | 139.01140 |
| Single | 0.22617 | 0.54830 | 5.75259 |
| Complete | 0.38127 | 0.79832 | 110.79681 |





3. DBSCAN:

| Linkage Type | Silhouette Score | Davies-Bouldin Index | Calinski-Harabasz Index |
|--------------|------------------|----------------------|-------------------------|
| eps=0.5; n=5 | 0.39195 | 2.96150 | 5.38576 |
| eps=0.2; n=3 | -0.10205 | 2.38576 | 2.66504 |
| eps=1; n=10 | 0.41797 | 0.46596 | 9.06706 |



Analysis:

1. For K-means clustering we can clearly say that $k=3$ makes the most sense as we know the number of labels. However, $K=2$ gives better metrics. This could be due to the distribution of the data.
2. For Hierarchical Clustering, using Ward linkage seems to give the best results across all three metrics.
3. Varying values of DBSCAN gives us a variety of results and clusters do not seem to be very clear. DBSCAN might not be a great model for this dataset.

Results:

Since, we know the number of labels here, K-means gives us the best results. However, if we did not know the number of labels, Hierarchical Clustering with Ward linkage gives fairly decent results.