Assignment No-02 (CS 6301- Machine Learning)

Title : Model Evaluation, Training & Testing

Statement :

Based on the available dataset, perform the prediction analysis. Based on the method described in the class, model the data class. Use Unsupervised Learning Algorithm (Hierarchical Clustering) for the operation. Check the performance based on the analysis of the algorithm used.

A. Identification of the Dataset : "Iris Flower Dataset"

I. Type of the Dataset : Multivariate Data

II. Data Quality and Analysis : Class:

-- Iris Setosa

-- Iris Versicolour

-- Iris Virginica

- Sepal length: the length of the sepal, which is the part of the flower that encloses the petals and forms a calyx
- Sepal width: the width of the sepal
- Petal length: the length of the petal, which is the part of the flower that is often brightly colored and attracts pollinators
- Petal width: the width of the petal

In [15]:
```
df
```

Out[15]:

	sepal_length	sepal_width	petal_length	petal_width	target
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

In [16]:
```
df.head()
```

Out[16]:

	sepal_length	sepal_width	petal_length	petal_width	target
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

In [19]:
```
df.shape
```

Out[19]: (150, 5)

```
In [20]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   sepal_length  150 non-null    float64
 1   sepal_width   150 non-null    float64
 2   petal_length  150 non-null    float64
 3   petal_width   150 non-null    float64
 4   target        150 non-null    object 
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

III. Features Pre-Processing :

```
In [18]: np.unique(df['target'])
```

```
Out[18]: array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

```
In [21]: df.isnull().sum()
```

```
Out[21]: sepal_length    0
         sepal_width     0
         petal_length    0
         petal_width     0
         target          0
         dtype: int64
```

```
In [22]: df.dropna()
```

Out[22]:

	sepal_length	sepal_width	petal_length	petal_width	target
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

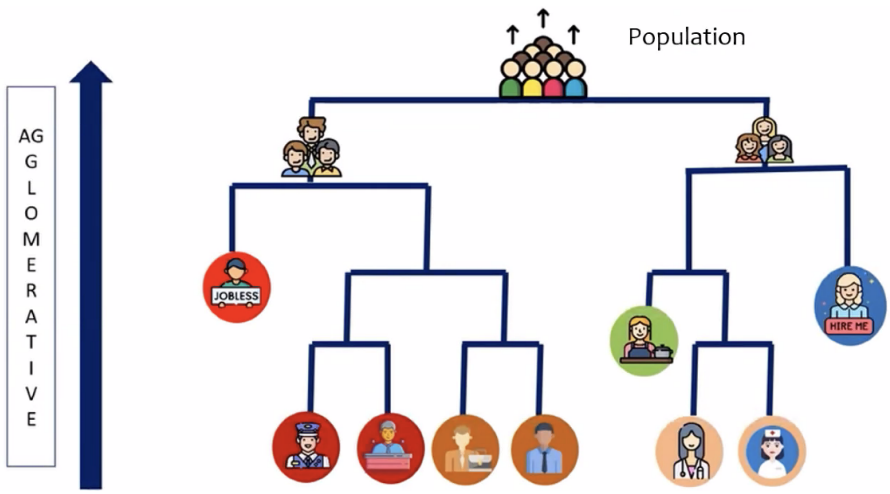
IV. Format of the Dataset : Comma Separated Value (.csv)

B. Identification of Learning Model (Un Supervised Learning)

i. Algorithm used : Hierarchical Clustering

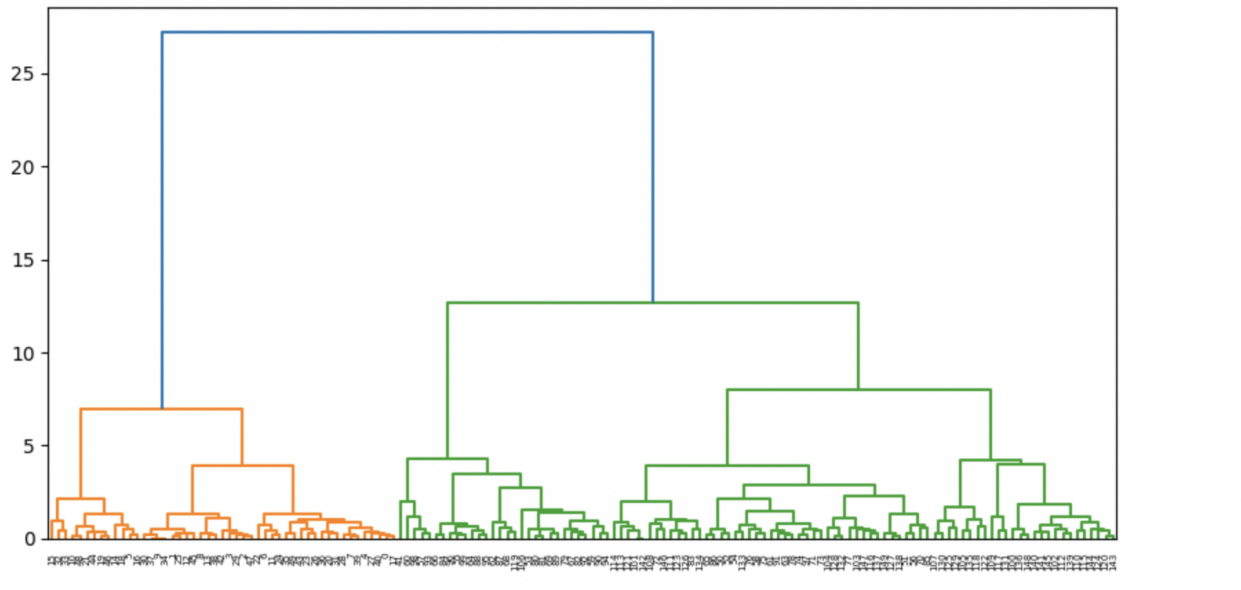
ii. Methodology used : Agglomerative Clustering

Working :We assign each observation to its own cluster, it's necessary which type of cluster we have.



iii. Model building and Testing :

```
# Import necessary libraries
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import AgglomerativeClustering
from scipy.cluster.hierarchy import linkage, dendrogram
import matplotlib.pyplot as plt

# Load the Iris dataset
df =
pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/iri
s/iris.data',
                names=['sepal_length', 'sepal_width', 'petal_length',
'petal_width', 'target'])

# Preprocess the data
X = df.drop('target', axis=1)
X = StandardScaler().fit_transform(X)

# Model the data using hierarchical clustering
model = AgglomerativeClustering(n_clusters=3)
predictions = model.fit_predict(X)

# Create a linkage matrix
Z = linkage(X, method='ward')

# Plot a dendrogram of the hierarchical clustering results
plt.figure(figsize=(10, 5))
dendrogram(Z)
plt.show()
```



iv. Model Accuracy, Prediction & Precession :

```
Classification accuracy: 1.0
```

C. Key Learning Outcomes :

After completion of this assignment I got hands- on learning and unsupervised learning algorithm, Tree based structure and is based on priority, it has predefined order.

- To conclude the analysis of the data using the hierarchical clustering algorithm, I can now evaluate the performance of the model.
- The purpose of the analysis: What was the goal of the analysis? Did the model achieve this goal? The strengths and limitations of the model: What are the strengths and limitations of the hierarchical clustering algorithm? How did these affect the results of the analysis?

Overall, the hierarchical clustering algorithm is a useful tool for grouping data into clusters based on their similarity. It is particularly useful when you do not have prior knowledge of the structure of the data or when you want to explore the data to discover patterns and relationships. However, it is important to carefully evaluate the results of the analysis using appropriate evaluation metrics and consider the limitations of the model when interpreting the results.

Submitted By:
Vidushi Yadav
Enrolment Id- AU20B1013