Experiment 5 –Clustering Algorithms

Aim: Implementation of Clustering Algorithm Using
1. k-means
2. Hierarchical (single/complete/average)

Theory: Clustering is a fundamental unsupervised learning technique used to group data points into clusters, where points within the same cluster are more similar to each other than to those in other clusters. Two commonly used clustering algorithms are K-Means and Hierarchical clustering.

K-Means is a partitioning clustering algorithm that aims to divide data points into K clusters. It works by iteratively assigning data points to the nearest cluster centre and then recomputing the cluster centres. The process continues until convergence, typically measured by the stability of cluster assignments. K-Means is sensitive to the initial choice of cluster centres, and the algorithm may converge to local optima. Multiple runs with different initializations are often performed to mitigate this issue.

Hierarchical clustering builds a tree-like structure (dendrogram) of clusters, allowing different levels of granularity. It can be divided into two main approaches: Agglomerative (bottom-up) and Divisive (top-down).

Agglomerative hierarchical clustering starts with individual data points as clusters and merges them into larger clusters, progressing until all data points belong to a single cluster.

Divisive hierarchical clustering begins with all data points in one cluster and recursively divides them into smaller clusters.

Hierarchical clustering is less sensitive to initializations and provides a visual representation of data structure through dendrograms.

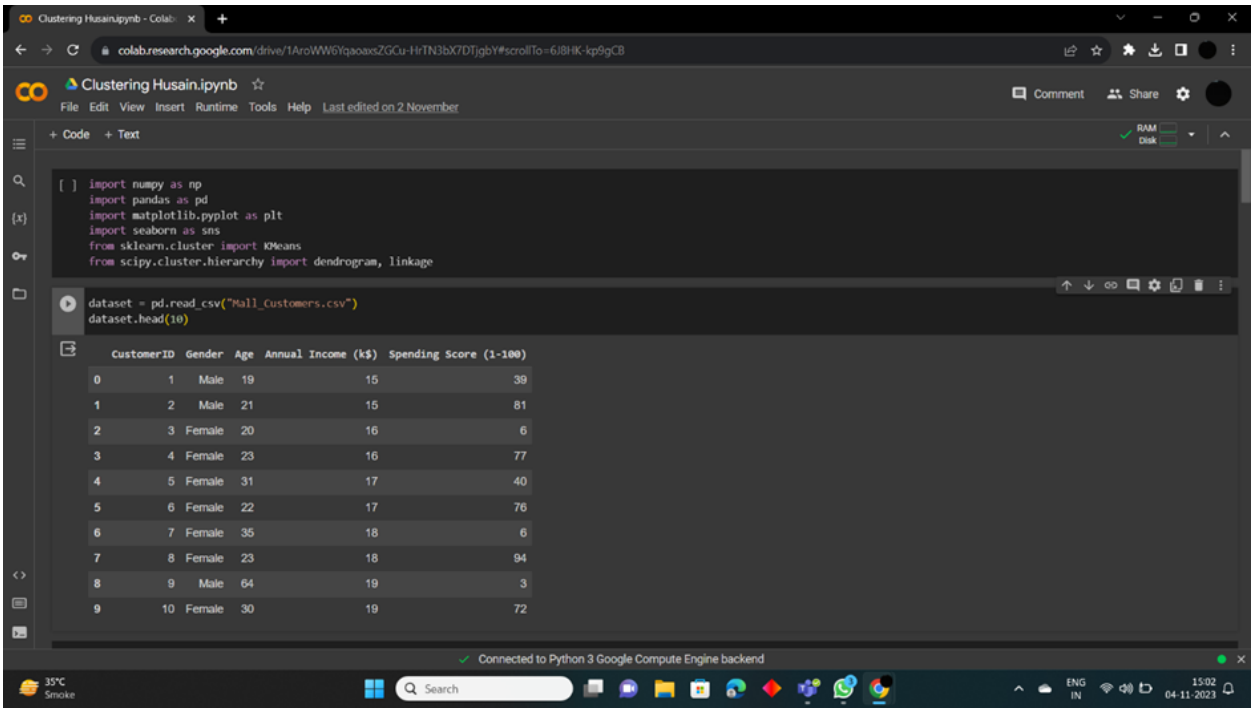
The elbow method is a technique for selecting the optimal number of clusters (K) in K-Means clustering. It involves running K-Means for a range of K values and computing the sum of squared distances (inertia) between data points and their cluster centres. As K increases, the inertia typically decreases, because clusters become smaller. However, beyond a certain point, the reduction in inertia is marginal. The "elbow point" in the plot of K versus inertia represents the optimal K value, where adding more clusters provides diminishing returns in reducing inertia.

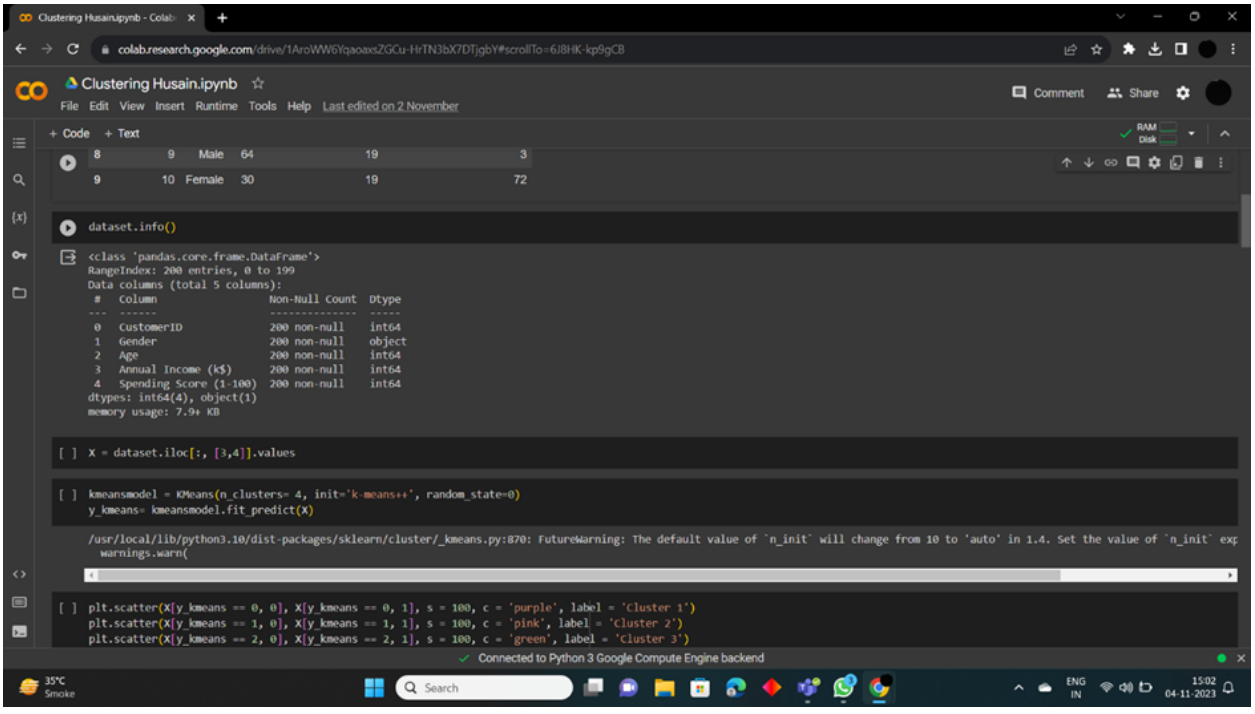
Dendrograms are hierarchical clustering visualizations that display the structure of clusters and their relationships. In a dendrogram, data points start as individual leaves at the bottom, and clusters are merged as we move upward. The height at which clusters merge on the vertical axis represents the dissimilarity between them. The longer the branch, the less similar the clusters are. Dendrograms provide insights into the hierarchical structure of data and help select the appropriate level of granularity when dividing data into clusters.

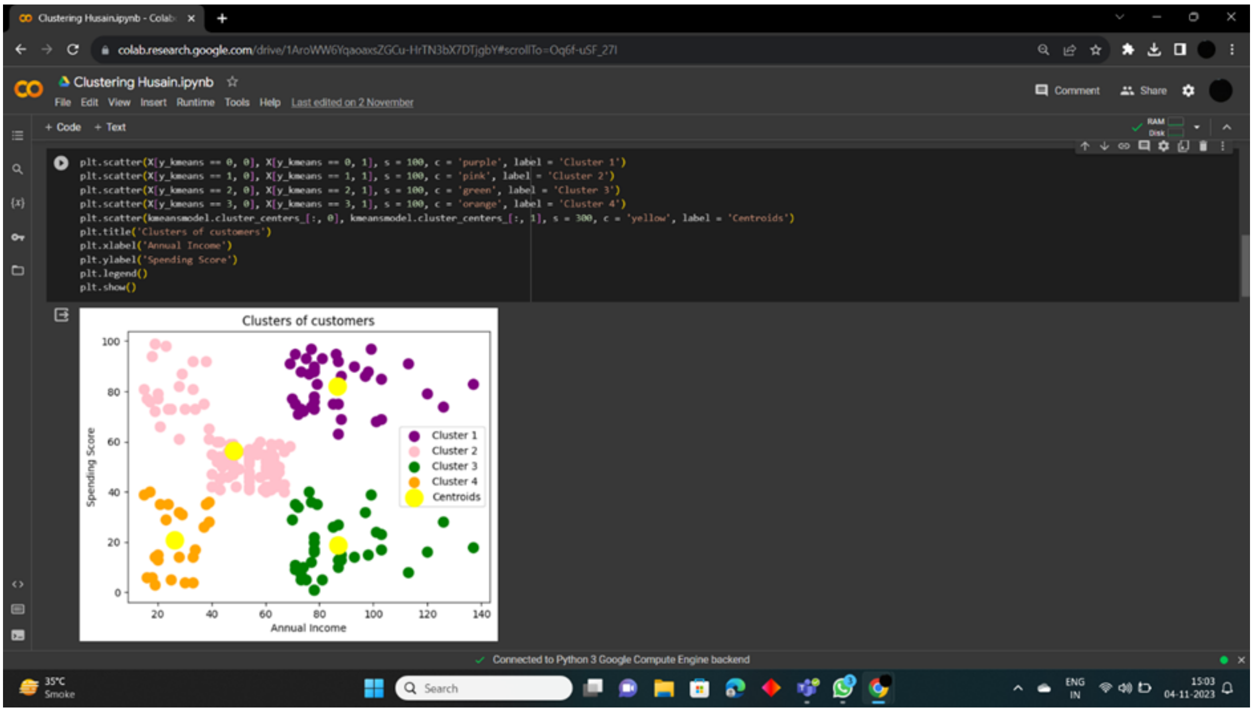
Implementation: For this experiment we were required to perform the following:

Part A:
Program using inbuilt functions.
Plot the clusters
Plot dendrogram (for hierarchical)
Part B:
Find optimum no. of cluster using elbow method.

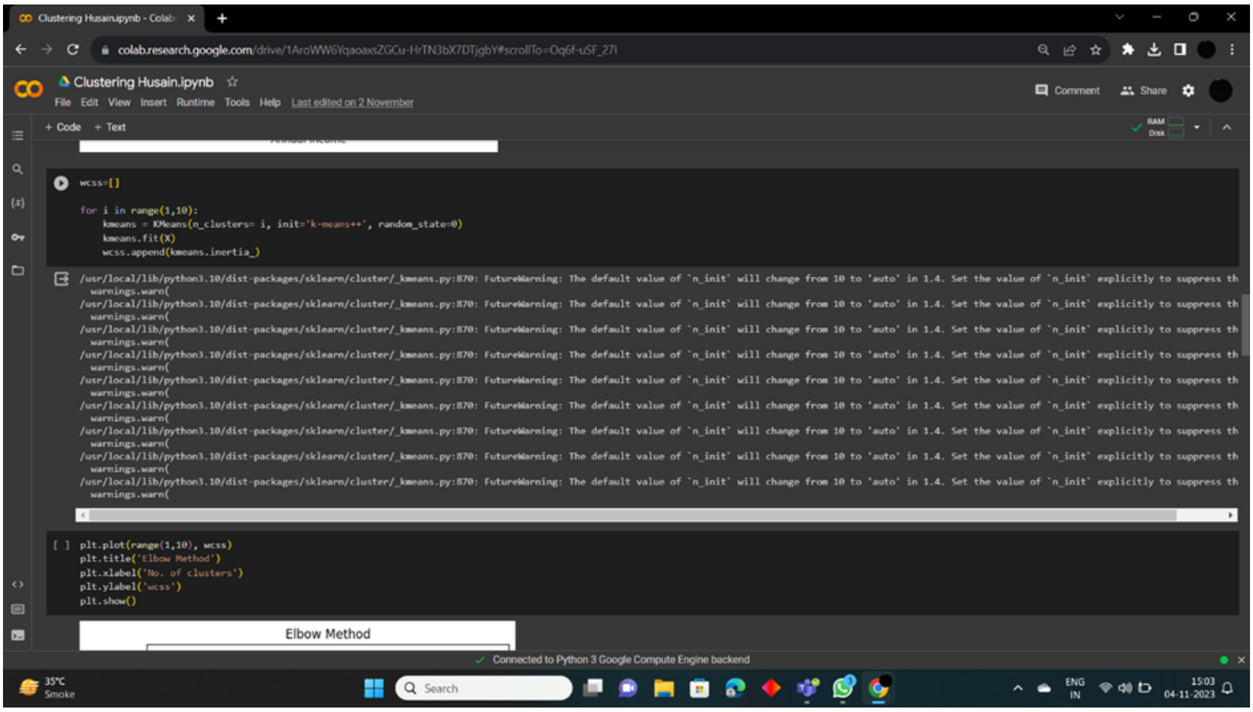
We have chosen the Mall_Customers.csv dataset for this experiment. It is a cleaned dataset having no NULL values. The attributes are
CustomerID, Gender, Age, Annual Income (k$), Spending Score (1-100)
It consists of 200 records.


Clustering Husain.ipynb - Colab
colab.research.google.com/drive/1AroWW6YqaoaxsZGCu-HrTN3bX7DTjgbY#scrollTo=6J8HK-kp9gCB
Clustering Husain.ipynb
File Edit View Insert Runtime Tools Help Last edited on 2 November
Comment
Share
+ Code + Text
RAM
Disk
[ ] import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans
from scipy.cluster.hierarchy import dendrogram, linkage
dataset = pd.read_csv("Mall_Customers.csv")
dataset.head(10)
CustomerID Gender Age Annual Income (k$) Spending Score (1-100)
0 1 Male 19 15 39
1 2 Male 21 15 81
2 3 Female 20 16 6
3 4 Female 23 16 77
4 5 Female 31 17 40
5 6 Female 22 17 76
6 7 Female 35 18 6
7 8 Female 23 18 94
8 9 Male 64 19 3
9 10 Female 30 19 72
Connected to Python 3 Google Compute Engine backend
35°C
Smoke
Search
ENG
IN
15:02
04-11-2023


Clustering Husain.ipynb - Colab
colab.research.google.com/drive/1AroWW6YqaoaxsZGCu-HrTN3bX7DTjgbY#scrollTo=6J8HK-kp9gCB
Clustering Husain.ipynb
File Edit View Insert Runtime Tools Help Last edited on 2 November
Comment
Share
+ Code + Text
RAM
Disk
8 9 Male 64 19 3
9 10 Female 30 19 72
dataset.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
Column Non-Null Count Dtype
0 CustomerID 200 non-null int64
1 Gender 200 non-null object
2 Age 200 non-null int64
3 Annual Income (k$) 200 non-null int64
4 Spending Score (1-100) 200 non-null int64
dtypes: int64(4), object(1)
memory usage: 7.9+ KB
[ ] X = dataset.iloc[:, [3,4]].values
[ ] kmeansmodel = KMeans(n_clusters= 4, init='k-means++', random_state=0)
y_kmeans= kmeansmodel.fit_predict(X)
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` exp
warnings.warn(
[ ] plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'purple', label = 'Cluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'pink', label = 'Cluster 2')
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'green', label = 'Cluster 3')
Connected to Python 3 Google Compute Engine backend
35°C
Smoke
Search
ENG
IN
15:02
04-11-2023



Clustering Husain.ipynb - Colab
colab.research.google.com/drive/1AroWW6YqaoaxsZGCu-HrTN3bX7DTjgbY#scrollTo=Oq6f-uSF_27I
Clustering Husain.ipynb
File Edit View Insert Runtime Tools Help Last edited on 2 November
Comment Share
+ Code + Text
plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'purple', label = 'Cluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'pink', label = 'Cluster 2')
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'green', label = 'Cluster 3')
plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'orange', label = 'Cluster 4')
plt.scatter(kmeansmodel.cluster_centers_[:, 0], kmeansmodel.cluster_centers_[:, 1], s = 300, c = 'yellow', label = 'Centroids')
plt.title('Clusters of customers')
plt.xlabel('Annual Income')
plt.ylabel('Spending Score')
plt.legend()
plt.show()
Clusters of customers
Spending Score
Annual Income
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Centroids
Connected to Python 3 Google Compute Engine backend
35°C Smoke
Search
ENG IN
15:03 04-11-2023



Clustering Husain.ipynb - Colab
colab.research.google.com/drive/1AroWW6YqaoaxsZGCu-HrTN3bX7DTjgbY#scrollTo=Oq6f-uSF_27I
Clustering Husain.ipynb
File Edit View Insert Runtime Tools Help Last edited on 2 November
Comment Share
+ Code + Text
wcss=[]
for i in range(1,10):
kmeans = KMeans(n_clusters= i, init='k-means++', random_state=0)
kmeans.fit(X)
wcss.append(kmeans.inertia_)
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress th
warnings.warn(
plt.plot(range(1,10), wcss)
plt.title('Elbow Method')
plt.xlabel('No. of clusters')
plt.ylabel('wcss')
plt.show()
Elbow Method
Connected to Python 3 Google Compute Engine backend
35°C Smoke
Search
ENG IN
15:03 04-11-2023

Clustering Husain.ipynb - Colab

colab.research.google.com/drive/1AroWW6YqaoaxsZGCu-HrTN3bX7DTjgbY#scrollTo=Oq6f-uSF_27I

Clustering Husain.ipynb

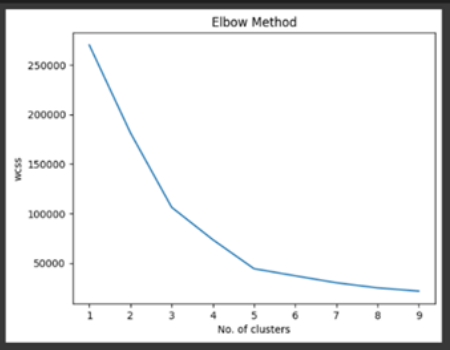
File Edit View Insert Runtime Tools Help Last edited on 2 November

Comment Share

+ Code + Text

RAM Disk

```
[ ] plt.plot(range(1,10), wcss)
    plt.title('Elbow Method')
    plt.xlabel('No. of clusters')
    plt.ylabel('wcss')
    plt.show()
```




```
[ ] Z = linkage(X, method='complete', metric='euclidean')
```

```
[ ] labelList = list(dataset['Annual Income (k$)'])
```

Connected to Python 3 Google Compute Engine backend

35°C Smoke

Search

ENG IN 15:03 04-11-2023

Clustering Husain.ipynb - Colab

colab.research.google.com/drive/1AroWW6YqaoaxsZGCu-HrTN3bX7DTjgbY#scrollTo=Oq6f-uSF_27I

Clustering Husain.ipynb

File Edit View Insert Runtime Tools Help Last edited on 2 November

Comment Share

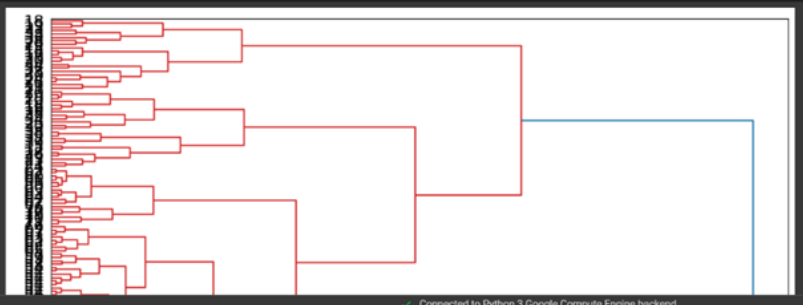
+ Code + Text

RAM Disk

```
[ ] Z = linkage(X, method='complete', metric='euclidean')
```

```
[ ] labelList = list(dataset['Annual Income (k$)'])

    plt.figure(figsize=(13, 12))
    dendrogram(
        Z,
        orientation='right',
        labels=labelList,
        distance_sort='descending',
        show_leaf_counts=False,
        leaf_font_size=16
    )
    plt.show()
```

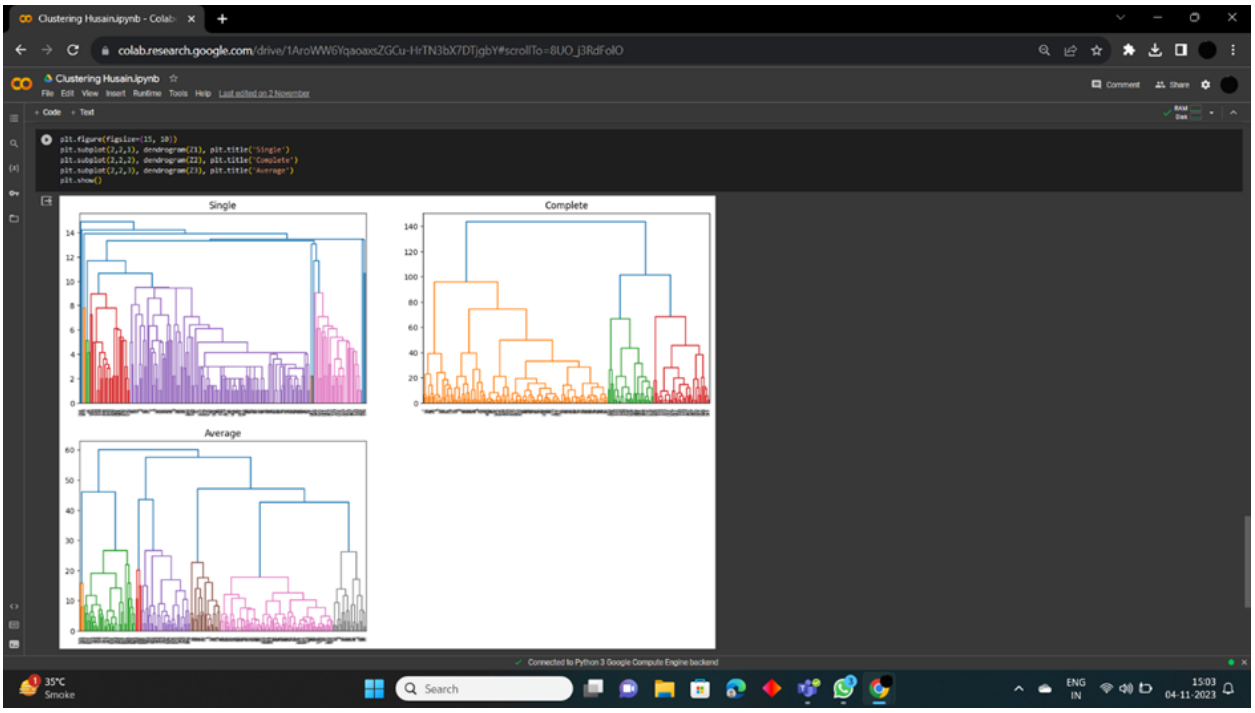


Connected to Python 3 Google Compute Engine backend

35°C Smoke

Search

ENG IN 15:03 04-11-2023



Conclusion: K-Means and Hierarchical clustering are fundamental methods for unsupervised data grouping. The elbow method aids in determining the optimal number of clusters in K-Means, while dendrograms offer a visual representation of hierarchical clustering, facilitating the interpretation of the cluster structure. These techniques play a crucial role in data exploration, pattern recognition, and decision-making in various applications.