

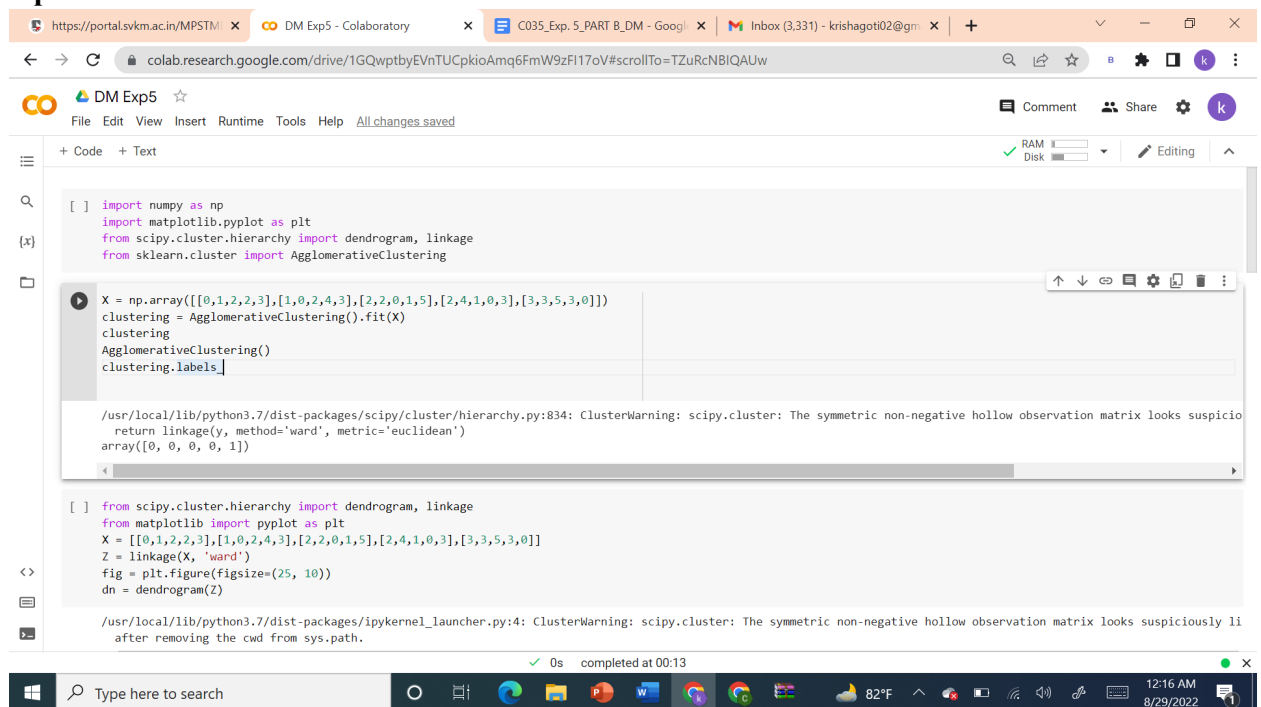
PART B

Roll No. C035	Name: KRISHA GOTI
Class : B	Batch : B1
Date of Experiment: 29/8/22	Date of Submission:29/8/22
Grade :	

B.1 Clustering Code written by student:

B.2 Input and Output:

Input Data:



The screenshot shows a Google Colab notebook interface. The browser tabs at the top include 'https://portal.svkm.ac.in/MPSTM', 'DM Exp5 - Colaboratory', 'C035_Exp_5_PART B_DM - Googl', and 'Inbox (3,331) - krishagoti02@gn'. The address bar shows the URL 'colab.research.google.com/drive/1GQwptbyEVnTUCpkioAmq6FmW9zF17oV#scrollTo=TZuRcNBiQAUw'. The notebook is titled 'DM Exp5' and has tabs for 'Code' and 'Text'. The code cell contains the following Python code:

```
[ ] import numpy as np
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import dendrogram, linkage
from sklearn.cluster import AgglomerativeClustering

X = np.array([[0,1,2,2,3],[1,0,2,4,3],[2,2,0,1,5],[2,4,1,0,3],[3,3,5,3,0]])
clustering = AgglomerativeClustering().fit(X)
clustering
AgglomerativeClustering()
clustering.labels_

/usr/local/lib/python3.7/dist-packages/scipy/cluster/hierarchy.py:834: ClusterWarning: scipy.cluster: The symmetric non-negative hollow observation matrix looks suspiciously like an unknown BLAS level-1 routine
return linkage(y, method='ward', metric='euclidean')
array([[0, 0, 0, 0, 1]])

[ ] from scipy.cluster.hierarchy import dendrogram, linkage
from matplotlib import pyplot as plt
X = [[0,1,2,2,3],[1,0,2,4,3],[2,2,0,1,5],[2,4,1,0,3],[3,3,5,3,0]]
Z = linkage(X, 'ward')
fig = plt.figure(figsize=(25, 10))
dn = dendrogram(Z)

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:4: ClusterWarning: scipy.cluster: The symmetric non-negative hollow observation matrix looks suspiciously like an unknown BLAS level-1 routine
after removing the cwd from sys.path.
```

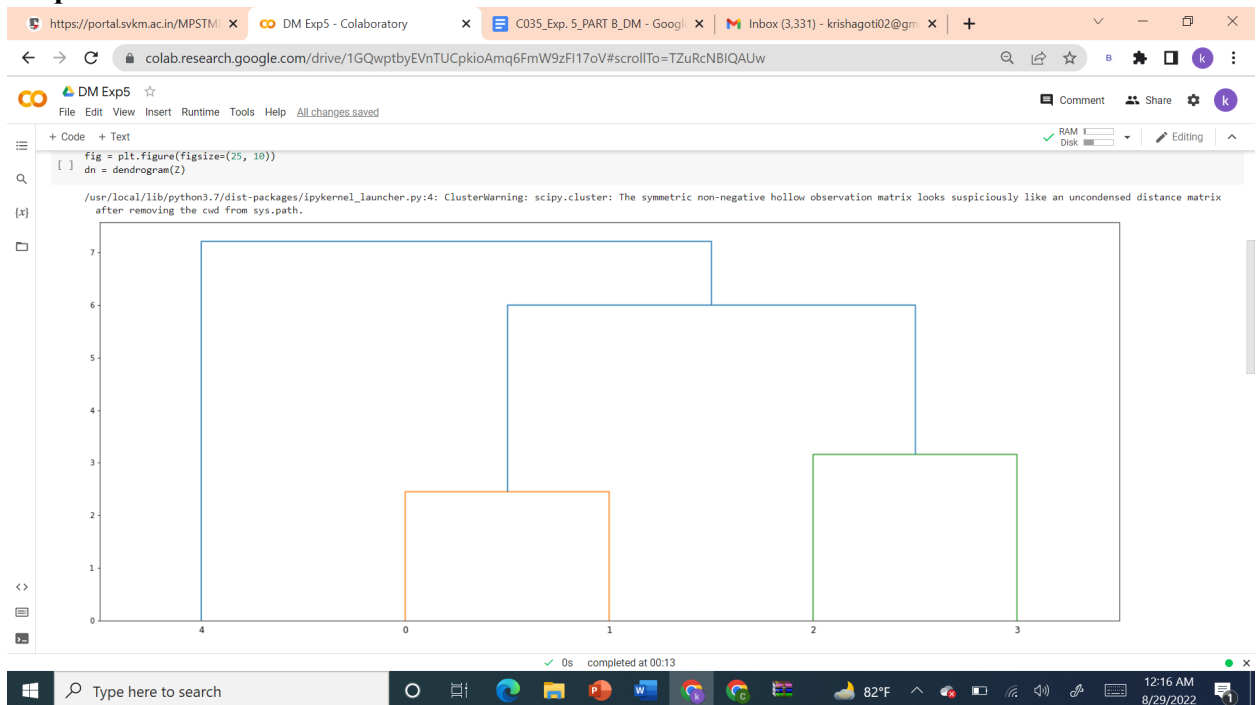
The output shows a warning message from scipy.cluster: "ClusterWarning: scipy.cluster: The symmetric non-negative hollow observation matrix looks suspiciously like an unknown BLAS level-1 routine". The output also shows the result of the clustering: "array([[0, 0, 0, 0, 1]])". The notebook status bar at the bottom indicates "0s completed at 00:13". The system tray at the bottom shows the date and time: "12:16 AM 8/29/2022".

```

def print_dc(connection):
    temp=[]
    for node in connection:
        if len(temp)==0:
            temp.append(connection[node])
        elif connection[node] not in temp:
            temp.append(connection[node])
    return temp
values = [
    [0,1,2,2,3],
    [1,0,2,4,3],
    [2,0,0,1,5],
    [2,4,1,0,3],
    [3,3,5,3,0]
]
connection = {
    "A":set(),
    "B":set(),
    "C":set(),
    "D":set(),
    "E":set()
}
d=0
k=len(values)
difference=5
while k!=1:
    for value in values:
        for distance in value:
            if distance == d:
                x1 = chr(values.index(value)+difference)
                x2 = chr(value.index(distance)+difference)
                connection[x1].add(x2)
                connection[x2].add(x1)
        for var in connection:
            for node in connection:
                if var in connection[node]:
                    connection[var] = connection[node].union(connection[var])
    clusters = print_dc(connection)
    k = len(clusters)
    print("d", d, {k}, {clusters})
    d+=1

```

Output Clusters:



```
DM Exp5
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
k=len(values)
difference=65
while k!=1:
    for value in values:
        for distance in value:
            if distance == d:
                x1 = chr(values.index(value)+difference)
                x2 = chr(value.index(distance)+difference)

                connection[x1].add(x2)
                connection[x2].add(x1)

    for var in connection:
        for node in connection:
            if var in connection[node]:
                connection[var] = connection[node].union(connection[var])
    clusters = print_dc(connection)
    k = len(clusters)
    print(f"<d>, {k}, {clusters}>")
    d+=1

<0, 5, [{'A'}, {'B'}, {'C'}, {'D'}, {'E'}]>
<1, 3, [{'B', 'A'}, {'D', 'C'}, {'E'}]>
<2, 2, [{'B', 'D', 'A', 'C'}, {'E'}]>
<3, 1, [{'B', 'D', 'E', 'A', 'C'}]>
```

0s completed at 00:13

B.3 Observations and learning:

I learnt about Agglomerative Clustering Algorithm which basically groups the datasets into clusters, it follows the bottom-up approach.

B.4 Conclusion:

After successfully completing this experiment, I have understood:-

1. concept of Data Mining by implementing some of its algorithms.
2. various clustering technique by mining
3. clustering using agglomerative algorithms.
