HIERARCHICAL CLUSTERING

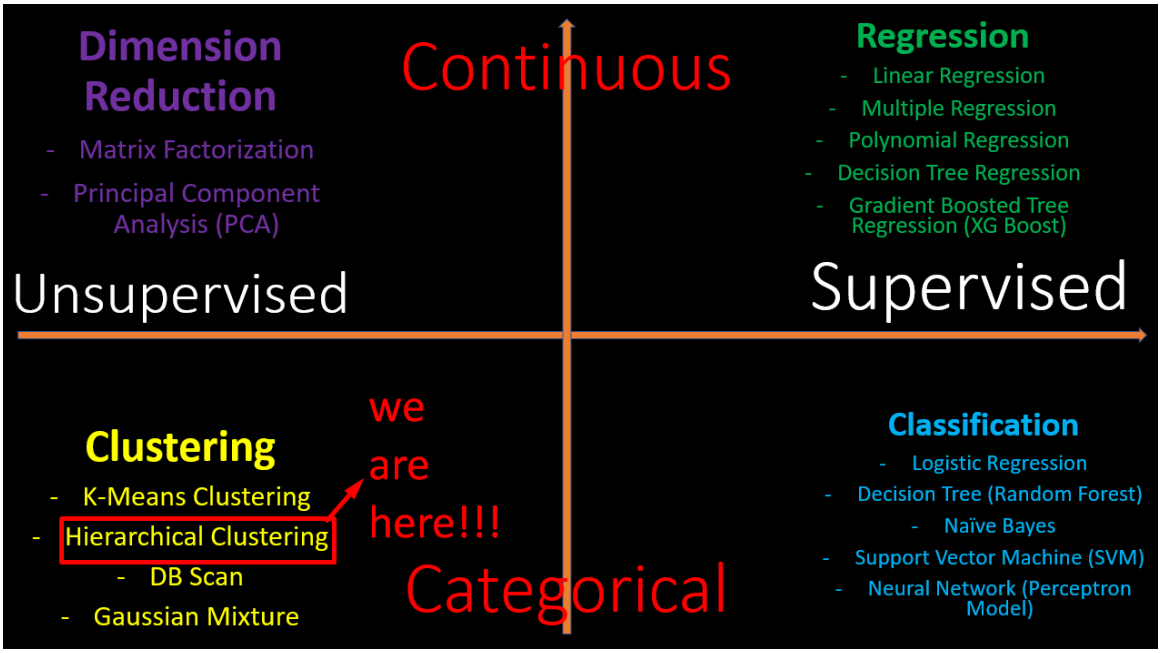
USING PYTHON
DR. ALVIN ANG

CONTENTS

I. UNDERSTANDING HIERARCHICAL CLUSTERING = UNSUPERVISED MACHINE LEARNING

Most of the stuff here are abstracted frrom:

https://www.amazon.com/Machine-Learning-PySpark-Processing-Recommender/dp/1484241304



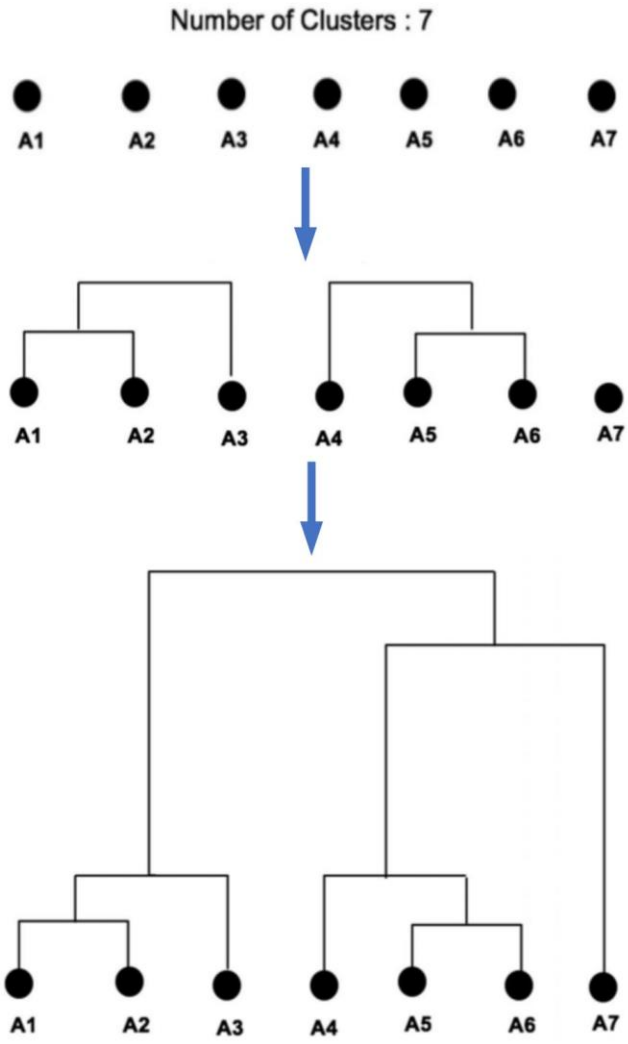
- Above is a table categorizing the different Machine Learning algorithms.
- Objective of Hierarchical Clustering is to predict a CATEGORY.

A. IMPORTANT POINT TO NOTE ABOUT PREDICTION USING HIERARCHICAL CLUSTERING

- You CAN'T use Hierarchical Clustering to predict new datasets.[1]
- Hierarchical clustering is not designed to predict cluster labels for new observations.
- The reason why this is happening is because it just links data points according to their distances and it is not defining "regions" for each cluster.
- If you really need to predict, use K Means Clustering.

[1] https://stackoverflow.com/questions/64589016/how-to-predict-the-cluster-label-of-a-new-observation-using-a-hierarchical-clust

B. IN A NUTSHELL….



Stage 1: Many Data Points

We want to Cluster them.

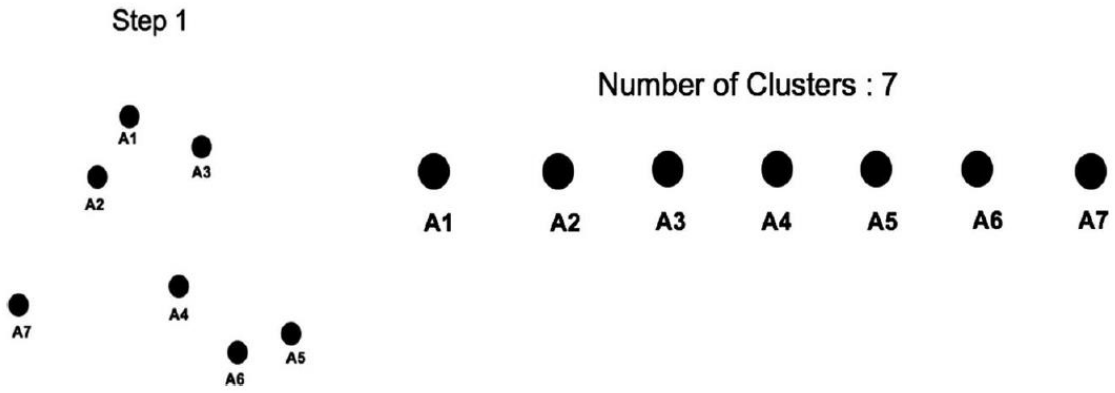
Stage 2: We Start Clustering Them

Based on their nearest distance, we cluster / group them together, point by point...

Stage 3: Completed Dendrogram

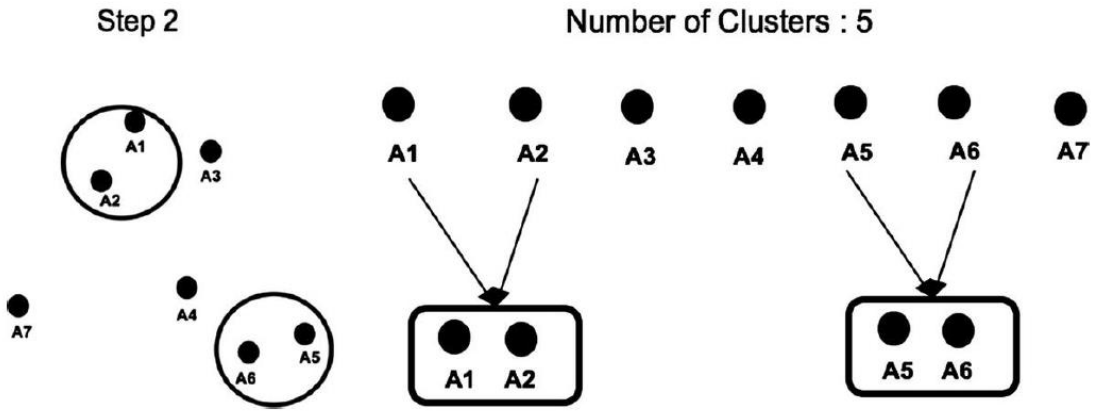
It is tempting to decide an optimal number of clusters based on the Dendrogram, but you can't. It is better to use the Elbow method to decide.

C. STEP 1: EACH POINT IS A CLUSTER



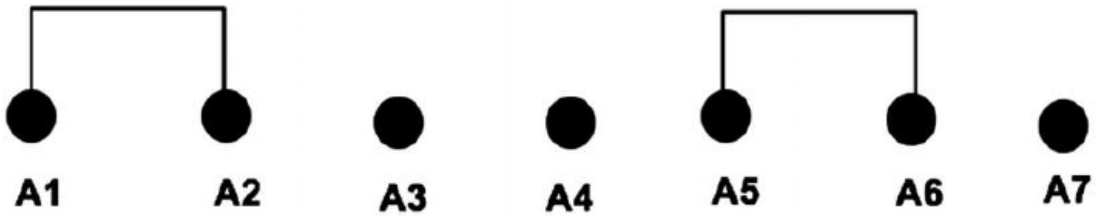
- We have 7 points → so we have 7 clusters.

D. STEP 2: START CLUSTERING



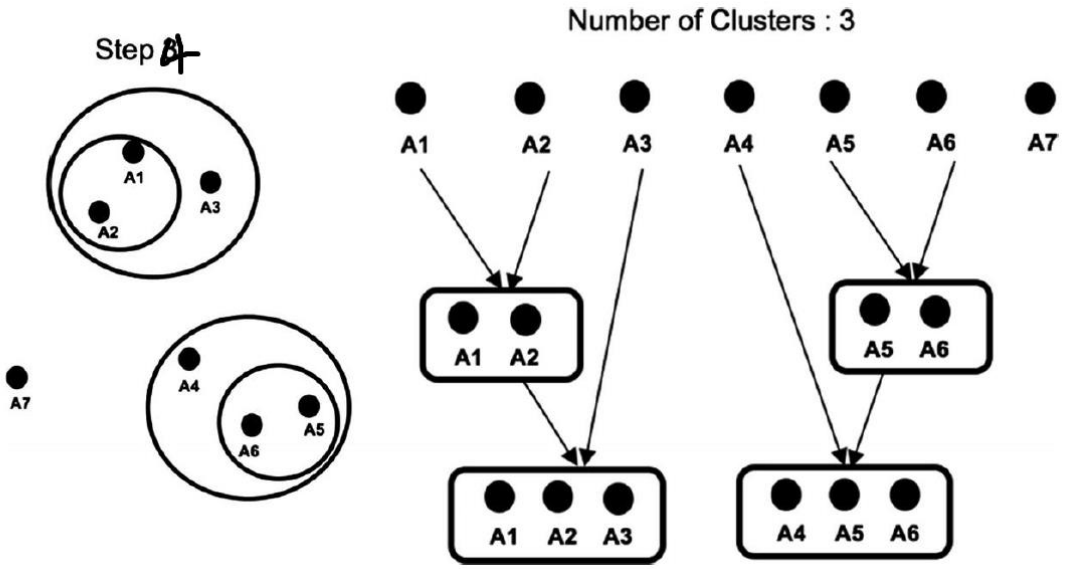
- We start clustering by nearest distance.
- A1 and A2 are nearest to each other → they form 1 cluster.
- A5 and A6 are nearest to each other → they form another cluster.

E. STEP 3: START DRAWING DENDROGRAM



- By right, the horizontal line connecting the 2 dots should represent the distance between them.
- In other words, if A1 and A2 are really really close, their horizontal line joining them should also be drawn much closer than the one joining A5 and A6.

F. STEP 4: CONTINUE CLUSTERING

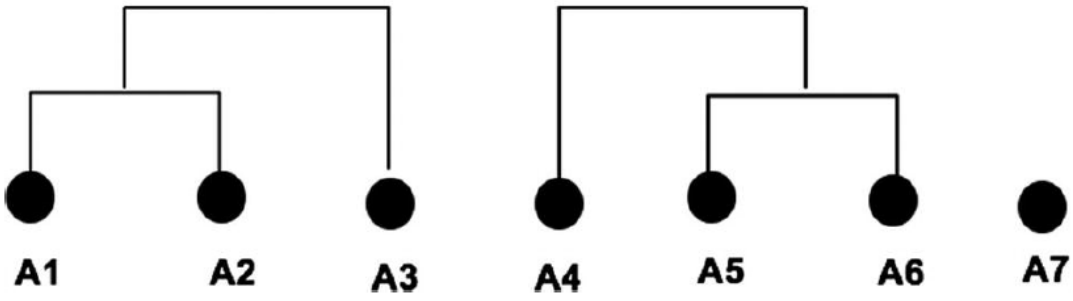


- A3 is nearest to cluster (A1, A2) → so it got sucked in.
- A4 is nearest to cluster (A5, A6) → so it got sucked in.

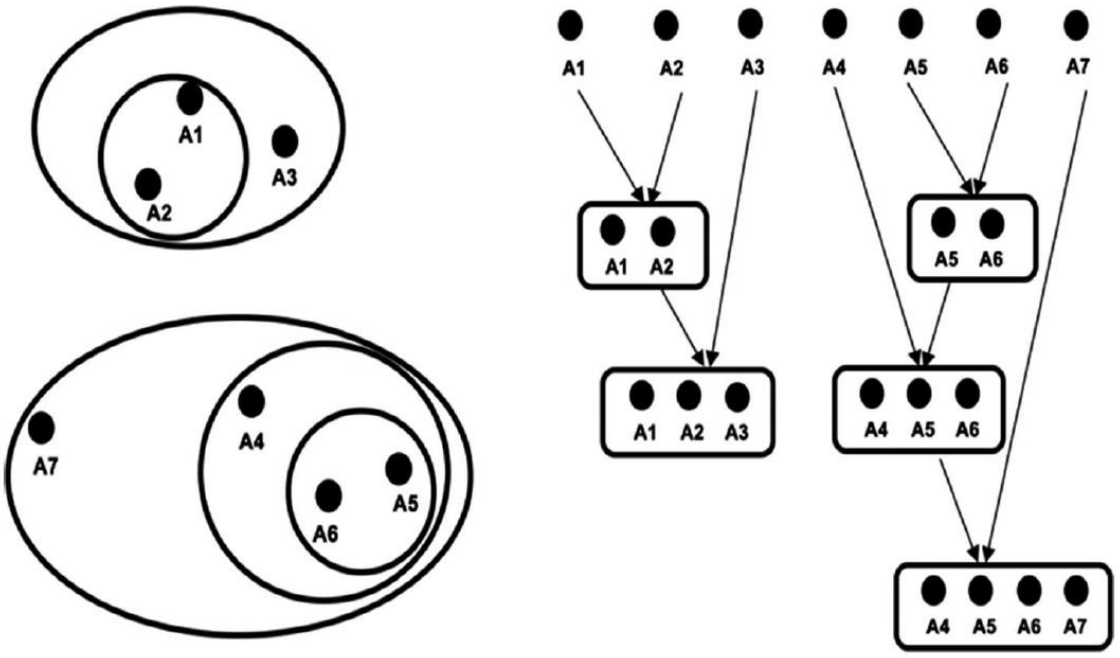
- I think it uses a Centroid of (A1, A2) & similarly (A5, A6) to calculate the distance to other points (in order to decide which point gets sucked in).

- Example: the Centroi od (A1, A2) is nearest distance to A3 (compared to other points) → thus A3 is sucked in.

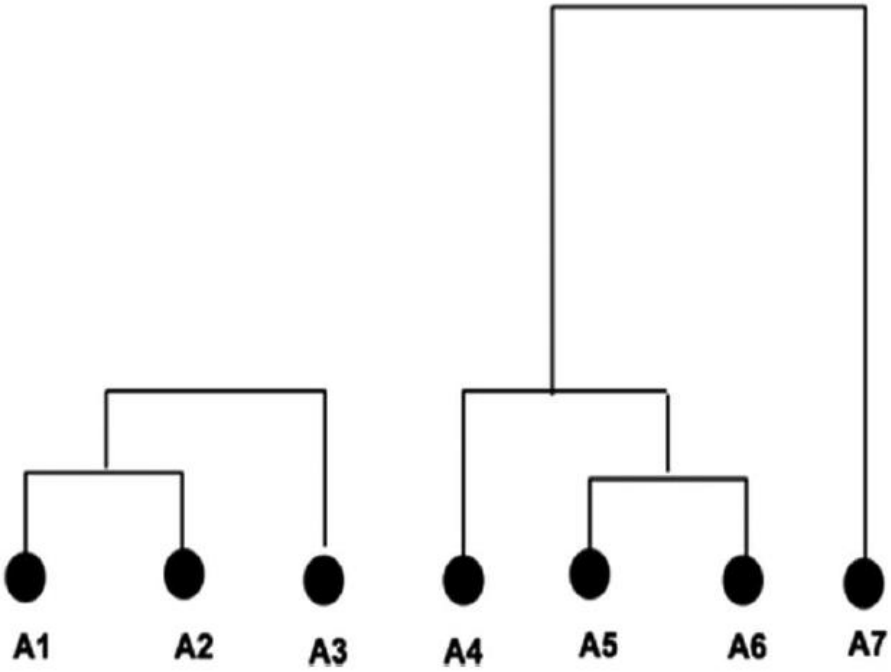
G. STEP 5: DRAW DENDROGRAM AGAIN



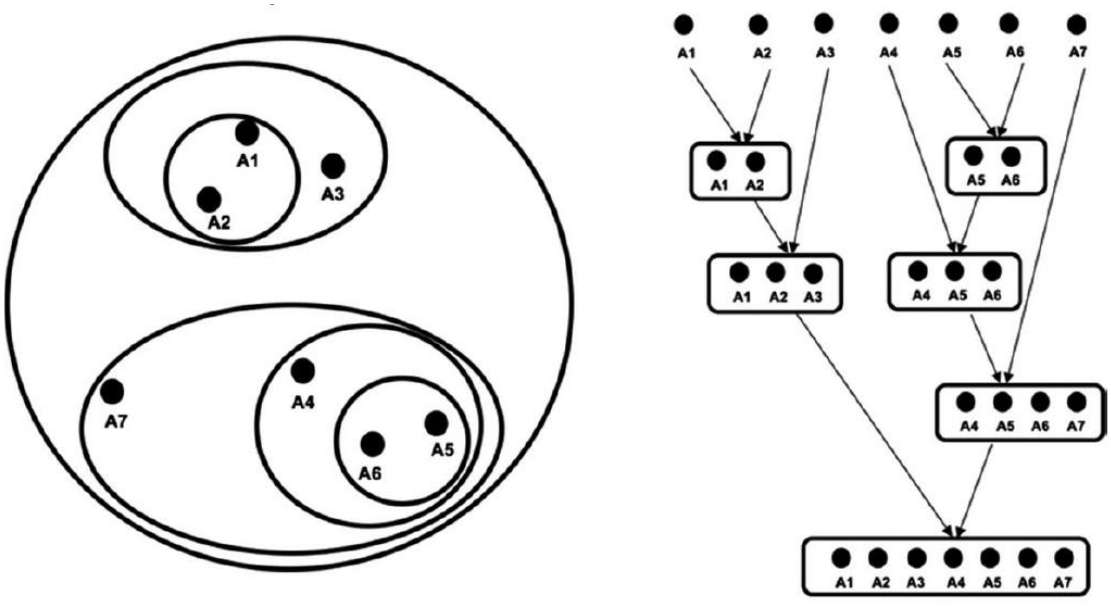
H. STEP 6: CONTINUE CLUSTERING AGAIN



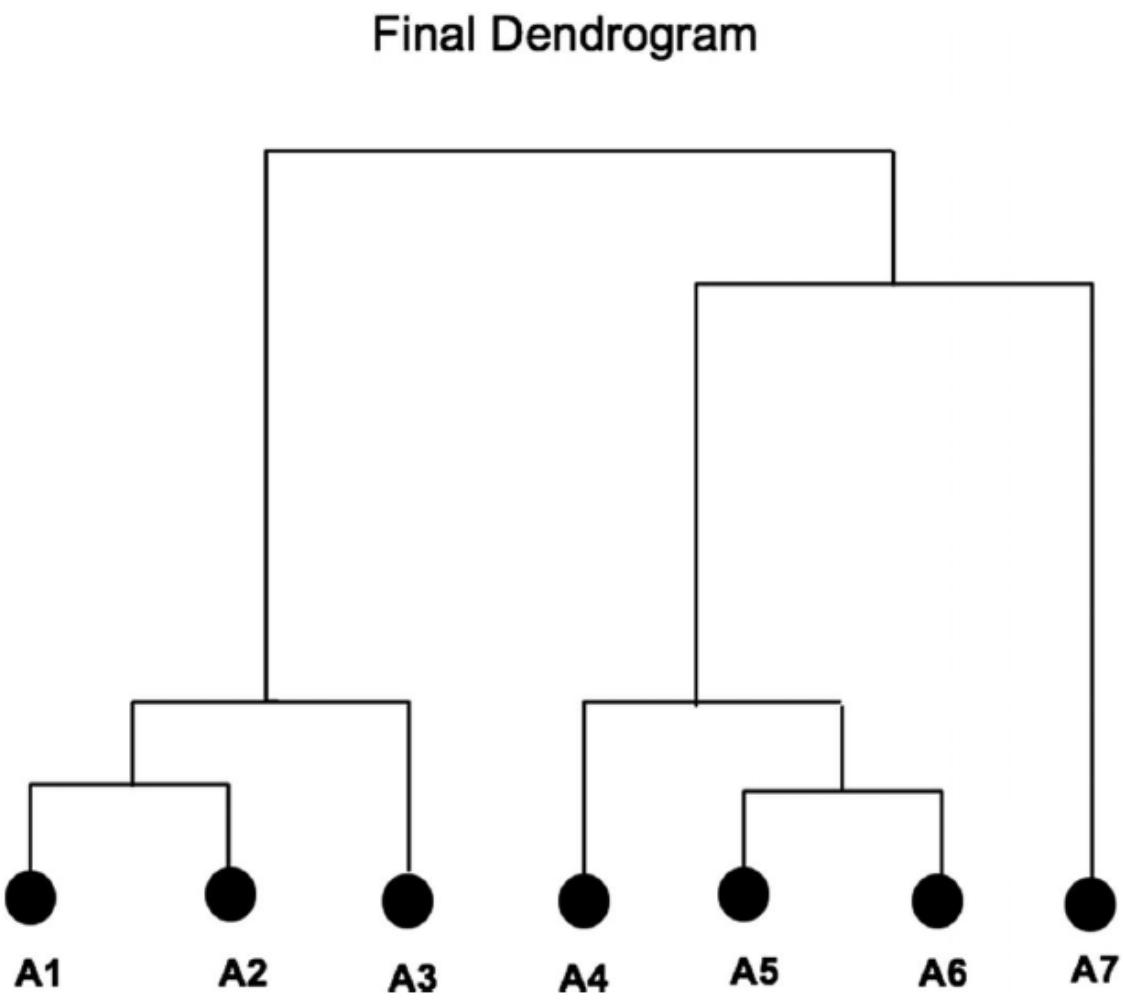
I. STEP 7: DRAW DENDROGRAM AGAIN (GOING TO COMPLETE SOON)



J. STEP 8: CLUSTER UNTIL YOU CAN'T CLUSTER ANYMORE…



K. STEP 9: VERY LAST DENDROGRAM



- Once again, note that the above Dendrogram is not drawn to scale.
- If it were, you will be able to see the distance between points for better gauge.
- Then, although it isn't right to use the Dendrogram to decide upon the optimal number of clusters, you may still use it (as a guide)[2].

[2] https://www.displayr.com/what-is-dendrogram/

I. HIERARCHICAL CLUSTERING USING PYTHON (SCIKIT LEARN)

Most of the code is taken from:

https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/

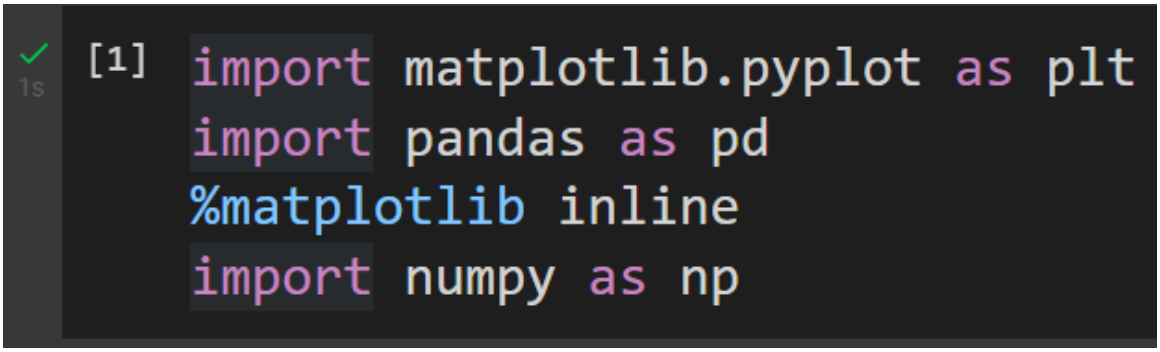
Ipynb file here:

https://www.alvinang.sg/s/Hierarchical_Clustering_using_Python.ipynb

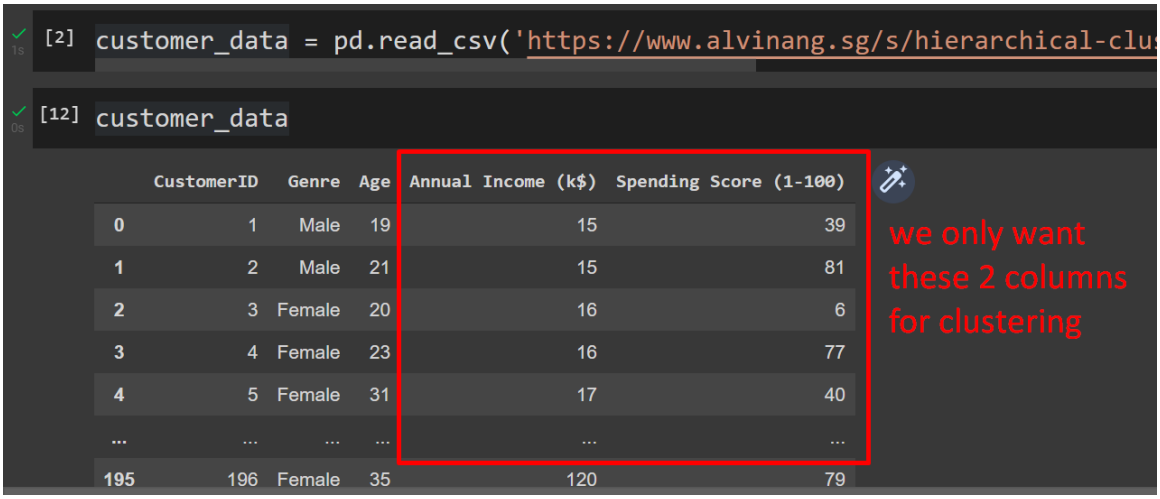
Datafile here:

https://www.alvinang.sg/s/hierarchical-clustering-with-python-and-scikit-learn-shopping-data.csv

A. IMPORT LIBRARIES

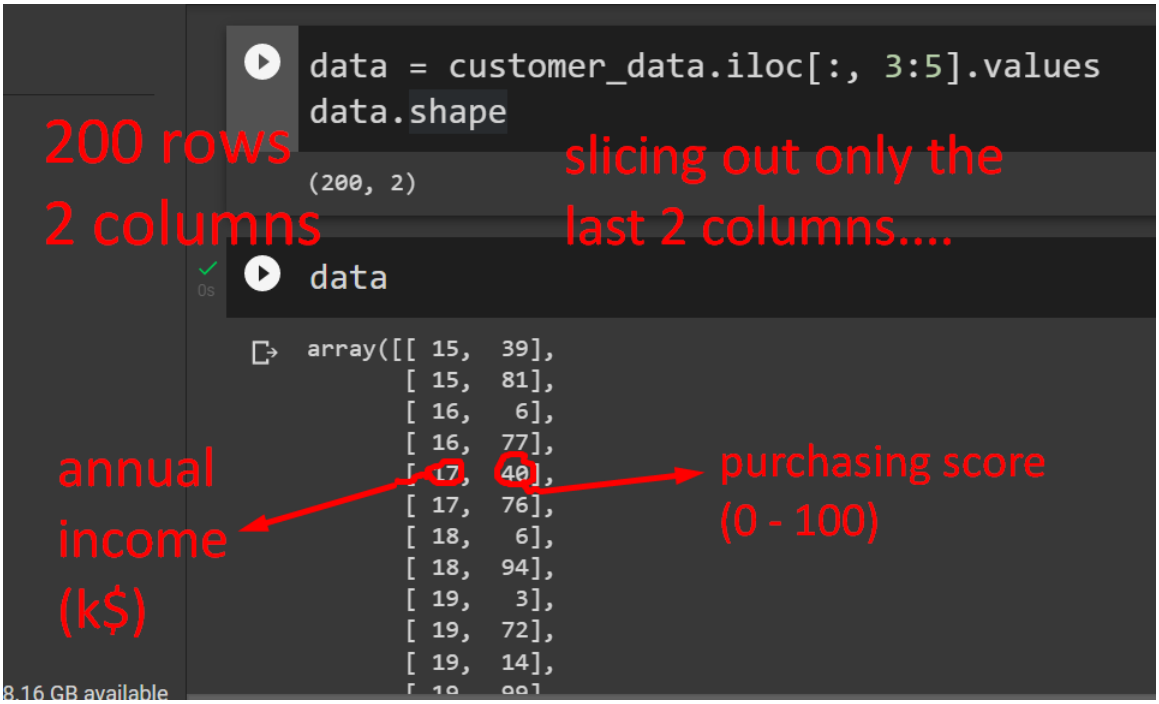


B. IMPORTING AND PREVIEWING THE DATA



	CustomerID	Genre	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
...	...	...	...	...	...
195	196	Female	35	120	79

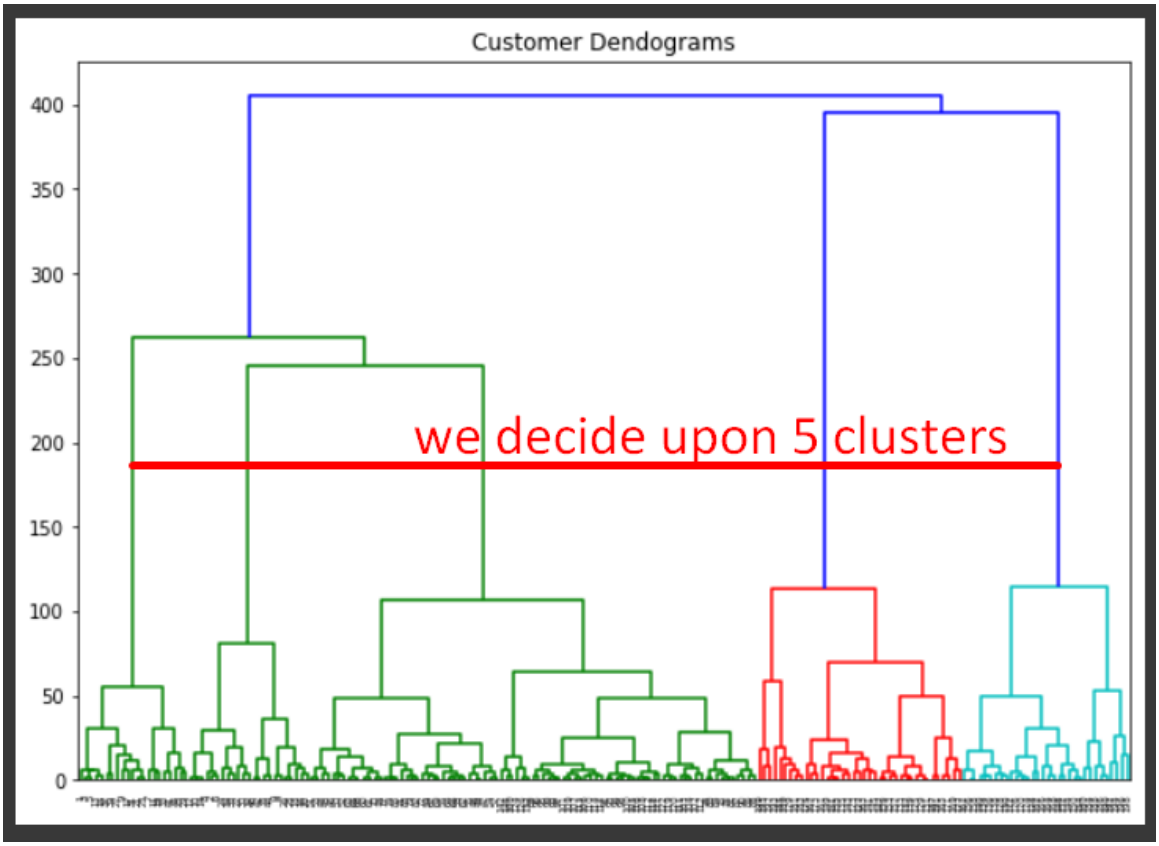
C. SLICING THE DATA



D. DRAWING DENDROGRAM

```
import scipy.cluster.hierarchy as shc

plt.figure(figsize=(10, 7))
plt.title("Customer Dendograms")
dend = shc.dendrogram(shc.linkage(data, method='ward'))
```



- The x axis refers to each individual point of the 200 rows i.e. 1 row = 1 point.
    - E.g. Point 1 = [15, 30] ; Point 2 = [15, 81] etc….
- The heights (y axis) reflect the distance between the clusters (it has no units… it ISN'T the annual income NOR the purchasing score).
- The taller the vertical line, the further the distance between clusters.

E. CLUSTERING THE 200 ROWS

```
from sklearn.cluster import AgglomerativeClustering

cluster = AgglomerativeClustering(n_clusters=5, affinity='euclidean', \
                                  linkage='ward')
a = cluster.fit_predict(data)
a
```

```
array([4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3,
       4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 1,
       4, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 2, 1, 2, 0, 2, 0, 2,
       1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 1, 2, 0, 2, 0, 2, 0, 2,
       0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2])
```

clustering for the 200 rows of data

0 to 4
5 clusters

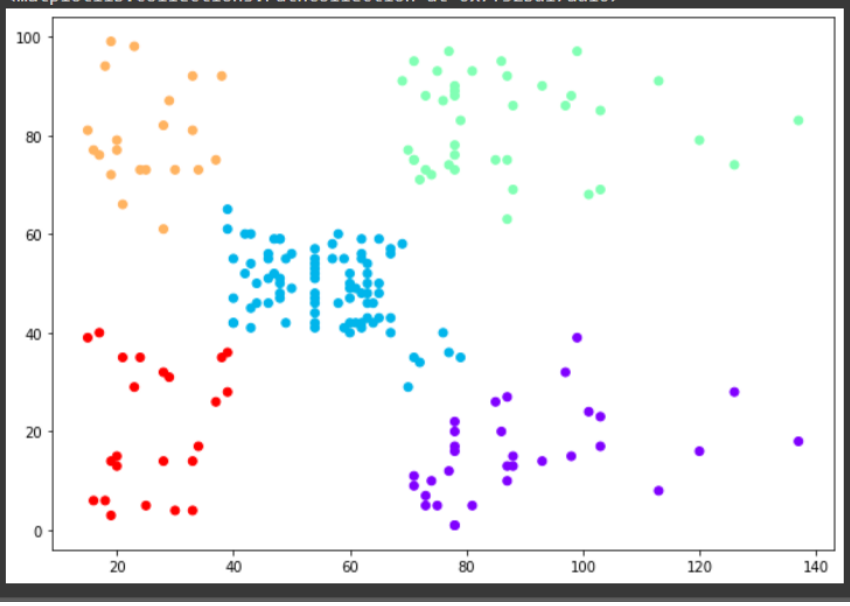
```
[11] a.shape
```

```
(200,)
```

F. PLOTTING

```
plt.figure(figsize=(10, 7))
plt.scatter(data[:,0], data[:,1], c=cluster.labels_, cmap='rainbow')
```

```
<matplotlib.collections.PathCollection at 0x7f32bd17ad10>
```



- We can see 5 clusters.

II. ABOUT DR. ALVIN ANG

Dr. Alvin Ang earned his Ph.D., Masters and Bachelor degrees from NTU, Singapore. He is a scientist, entrepreneur, as well as a personal/business advisor. More about him at www.AlvinAng.sg.