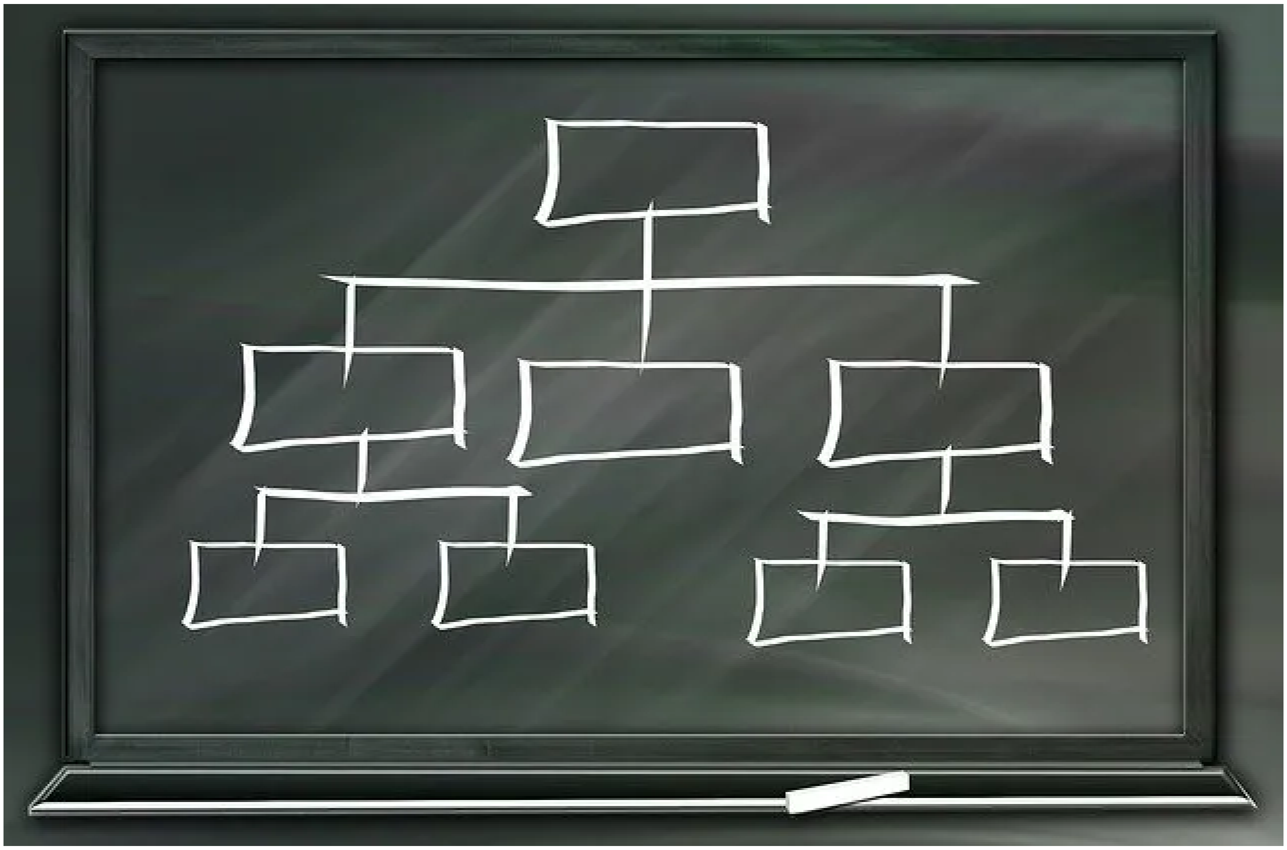


Hierarchical Clustering (Agglomerative)

Amit Ranjan · Follow

Published in Analytics Vidhya · 5 min read · Nov 30, 2020

57

In this article we will understand Agglomerative approach to Hierarchical Clustering, Steps of Algorithm and its mathematical approach.

Before deep diving into Hierarchical Clustering let's understand clustering first.

> **What is Clustering?**
> *Clustering is the method of dividing the objects (or data points) into clusters which are similar between them and dissimilar to the objects (or data points) belonging to another clusters.*

Clustering can be divided into two types:
1. Hierarchical Clustering
2. Partial Clustering

> **Hierarchical Clustering**
> *Hierarchical Clustering is separating the data into different groups from the hierarchy of clusters based on some measure of similarity.*

Hierarchical Clustering is of two types:
1. Agglomerative
2. Divisive

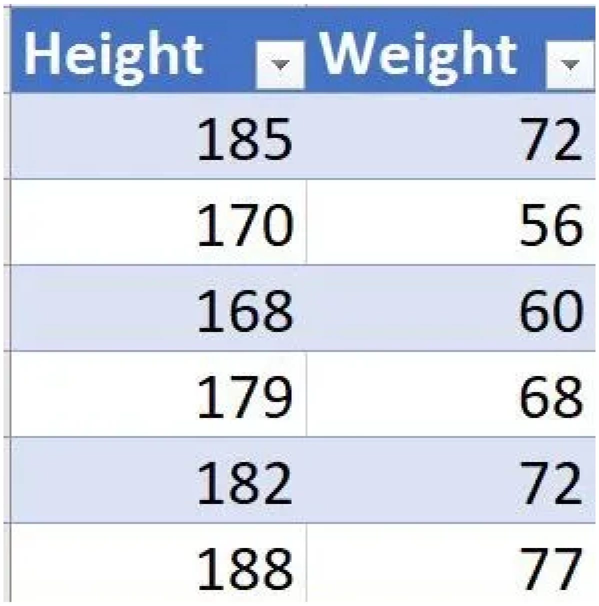
> **Agglomerative Clustering**
> *Agglomerative Clustering is also known as bottom-up approach.*
> *In this approach we take all data points as clusters and start merging it based on the distance between clusters. This will be done until we form one big cluster.*
>
> **Divisive Clustering**
> *Divisive Clustering is known as top-down approach.*

> *In this approach we take on huge cluster and starts breaking it up into smaller clusters until it reaches individual data points (or single point clusters).*

Till now we have seen about Clustering, Hierarchical clustering, Agglomerative clustering and Divisive clustering. Now let’s understand one of the techniques, Agglomerative Clustering in detail.

Algorithm of Agglomerative Clustering
1. Make each data point as a single-point cluster.
2. Take the two closest distance clusters by single linkage method and make them one clusters.
3. Repeat step 2 until there is only one cluster.
4. Create a Dendrogram to visualize the history of groupings.
5. Find optimal number of clusters from Dendrogram.

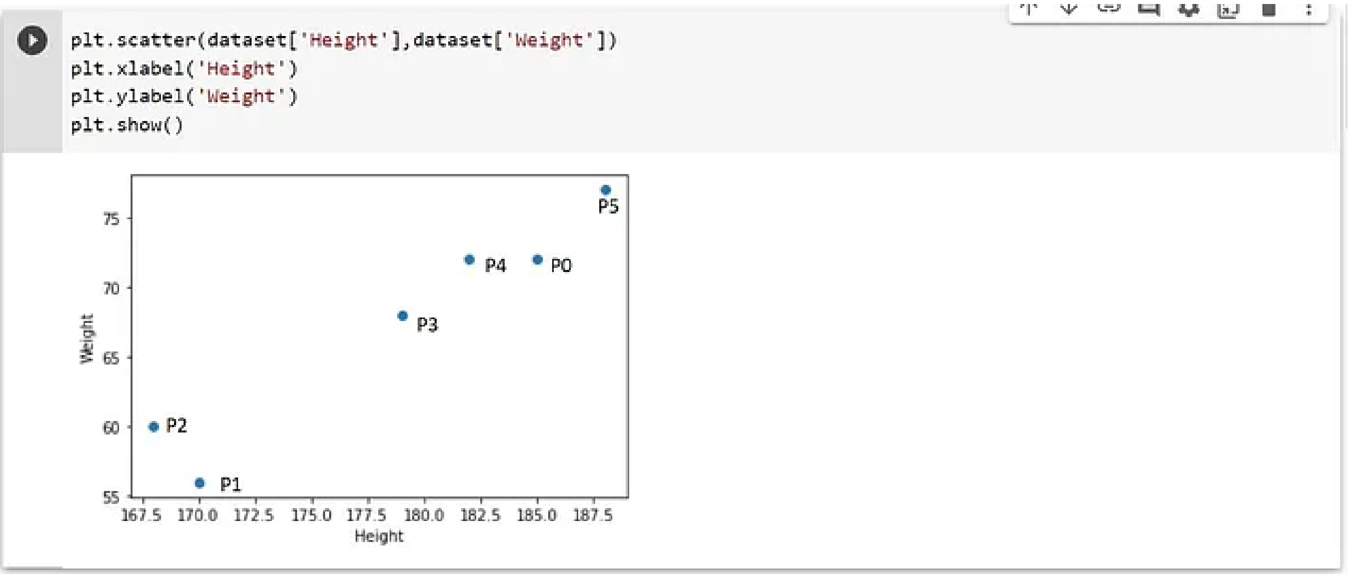
Mathematical Approach to Agglomerative Clustering
Let’s take dataset containing Height and Weight of a customer. For simplicity I am taking only 6 rows.



Height	Weight
185	72
170	56
168	60
179	68
182	72
188	77

Dataset

Let’s plot it on graph and visualize better.



```
plt.scatter(dataset['Height'],dataset['Weight'])
plt.xlabel('Height')
plt.ylabel('Weight')
plt.show()
```

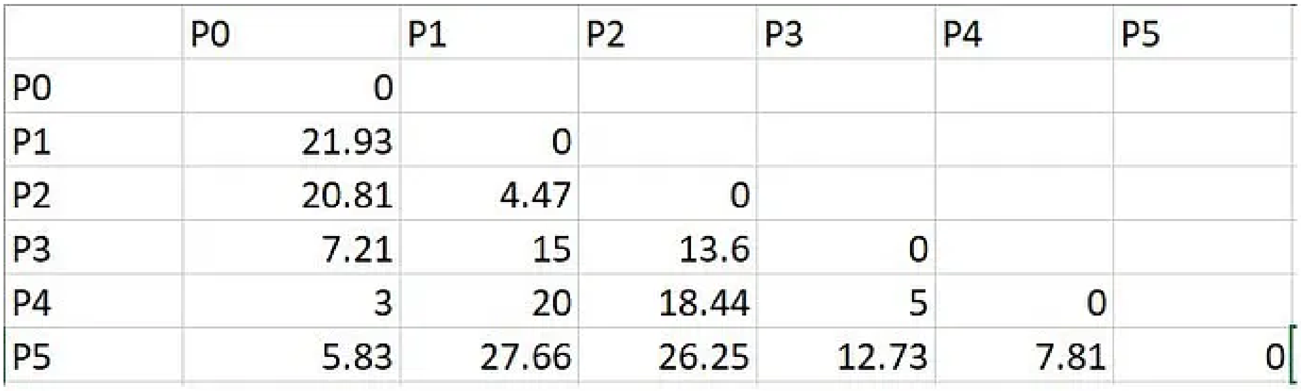
Step 2: Take the two closest distance clusters by single linkage method and make them one clusters.

Before using single linkage method on each clusters we must know the distance between clusters.
Let’s visualize the distance between each clusters with the help of distance matrix. Here, I am taking Euclidean distance between two points.

P00 = 0, P11 = 0, P22 = 0, P33 = 0, P44 = 0
(this is because distance between self is 0)

Distance between two points P12
= sqrt((P1.X — P2.X)^2 + (P1.Y — P2.Y)^2)
= sqrt((170–168)^2 + (56–60)^2)
= sqrt(4 + 16) = sqrt(20) = 4.47

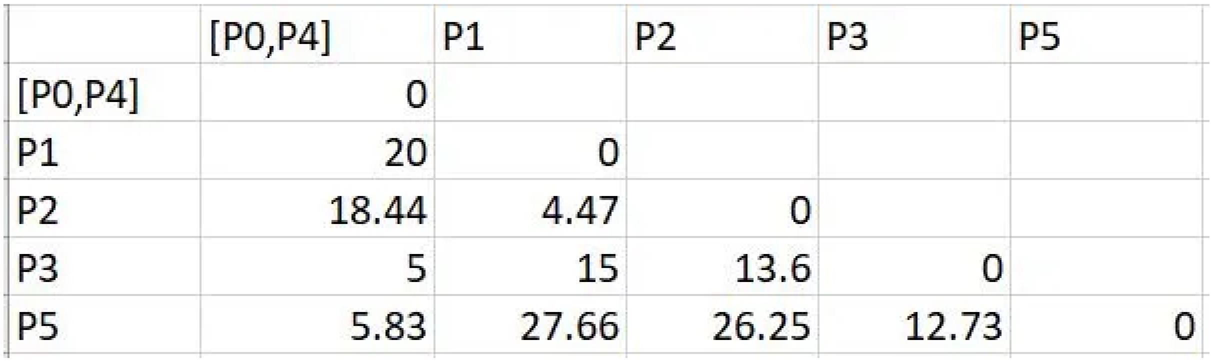
Similarly, we have to calculate the distance between all the clusters and make a distance matrix.



	P0	P1	P2	P3	P4	P5
P0	0					
P1	21.93	0				
P2	20.81	4.47	0			
P3	7.21	15	13.6	0		
P4	3	20	18.44	5	0	
P5	5.83	27.66	26.25	12.73	7.81	0

Distance Matrix of all Points

Now, we have to see which two cluster has minimum distance. Yes, you are right! Its the distance between P0 and P4 which is 3. So, we have to group these two clusters together. Now with the help of single linkage method we will merge two clusters.



	[P0,P4]	P1	P2	P3	P5
[P0,P4]	0				
P1	20	0			
P2	18.44	4.47	0		
P3	5	15	13.6	0	
P5	5.83	27.66	26.25	12.73	0

After merging P0 and P4

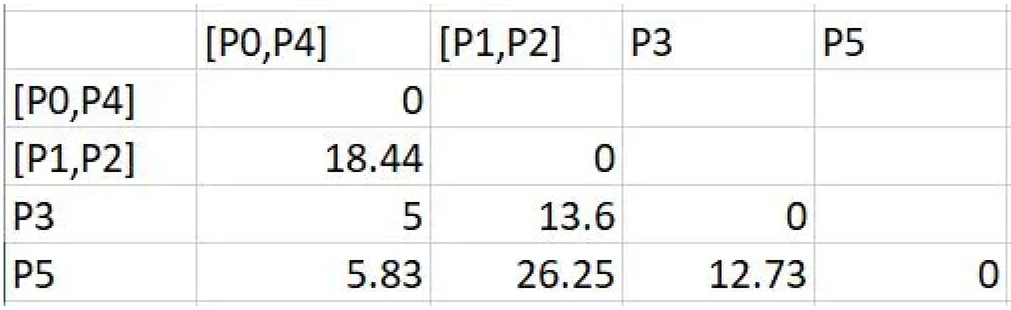
I know you are thinking wait a minute, How we have arrive the value of P1-[P0,P4] , P2-[P0,P4],P3-[P0,P4],P5-[P0,P4]. We have got these values with the help of single linkage method.

It says that, Distance of P1-[P0,P4] = d(P1,[P0-P4])
= min(d(P1,P0),d(P1,P4)) = min(21.93, 20) = 20

Distance of P2-[P0,P4] = d(P2,[P0,P4])
= min(d(P2,P0),d(P2,P4)) = min(20.81, 18.44) = 18.44

Similarly we have calculated all the distances.

Step 3: Repeat step 2
Again the minimum distance is P1-P2. So, the next distance matrix will be:

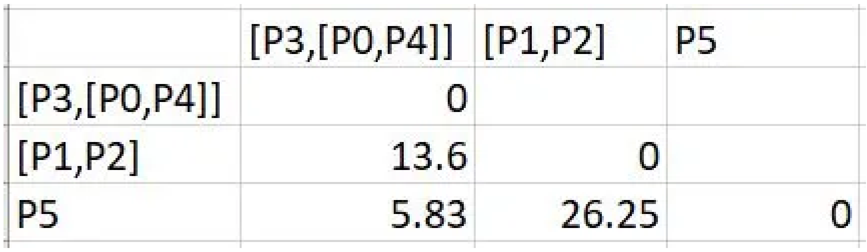


	[P0,P4]	[P1,P2]	P3	P5
[P0,P4]	0			
[P1,P2]	18.44	0		
P3	5	13.6	0	
P5	5.83	26.25	12.73	0

Merging P1 and P2

Step 3: Repeat step 2

Now minimum distance is P3-[P0,P4] which is 5. So, the next distance matrix will be:

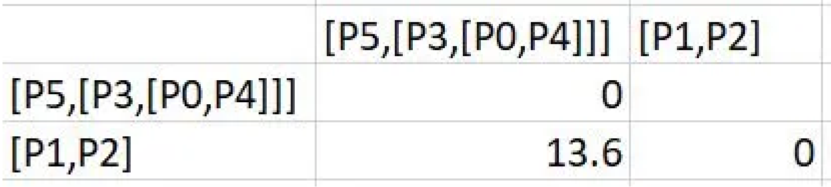


	[P3,[P0,P4]]	[P1,P2]	P5
[P3,[P0,P4]]	0		
[P1,P2]	13.6	0	
P5	5.83	26.25	0

Merging of P3 and [P0,P4]

Step 3: Repeat step 2

Now minimum distance is P5-[P3,[P0,P4]] which is 5.83. So, the next distance matrix will be:

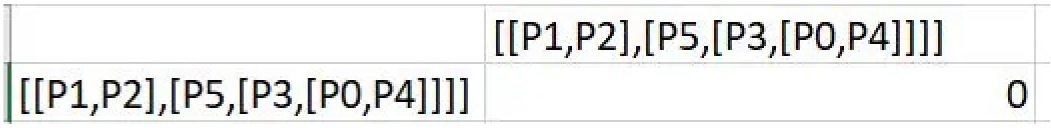


	[P5,[P3,[P0,P4]]]	[P1,P2]
[P5,[P3,[P0,P4]]]	0	
[P1,P2]	13.6	0

Merging of P5 and [P3,[P0,P4]]

Step 3: Repeat step 2

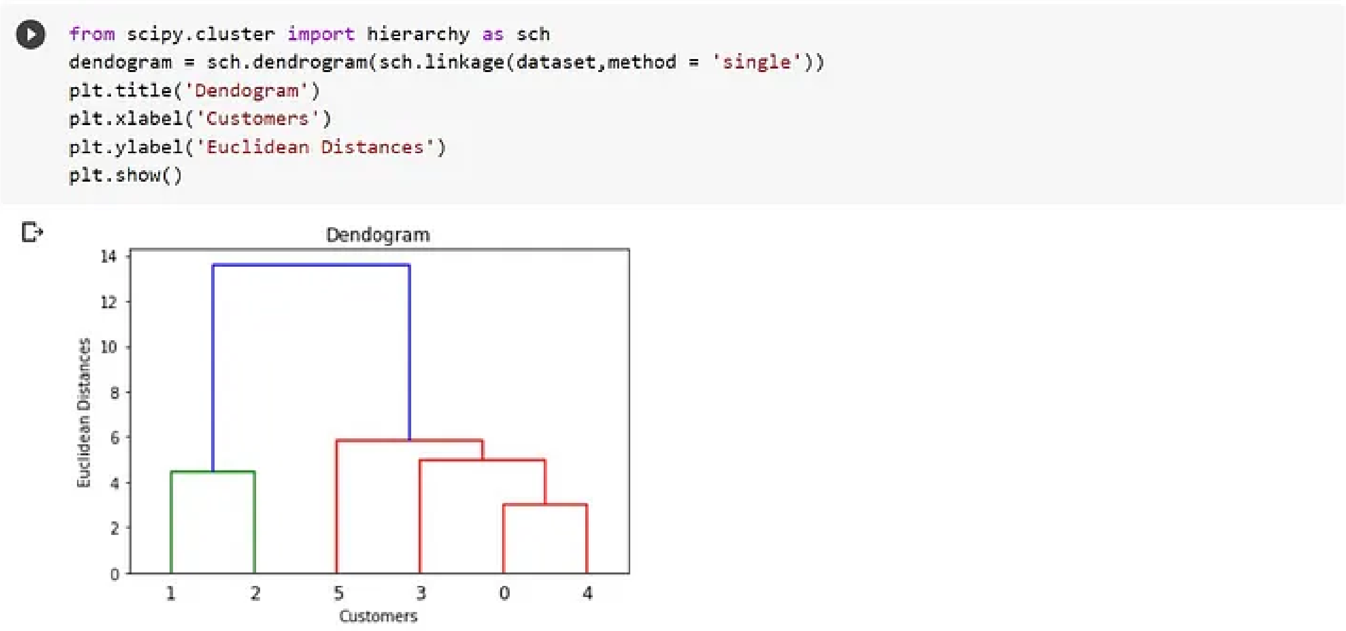
Now there are only two clusters whose distance is 13.6. So, the final distance matrix will be:



	[[P1,P2],[P5,[P3,[P0,P4]]]]
[[P1,P2],[P5,[P3,[P0,P4]]]]	0

Merging of [P1,P2] and [P5,[P3,[P0,P4]]]

Step 4: Create a Dendrogram to visualize the history of groupings.

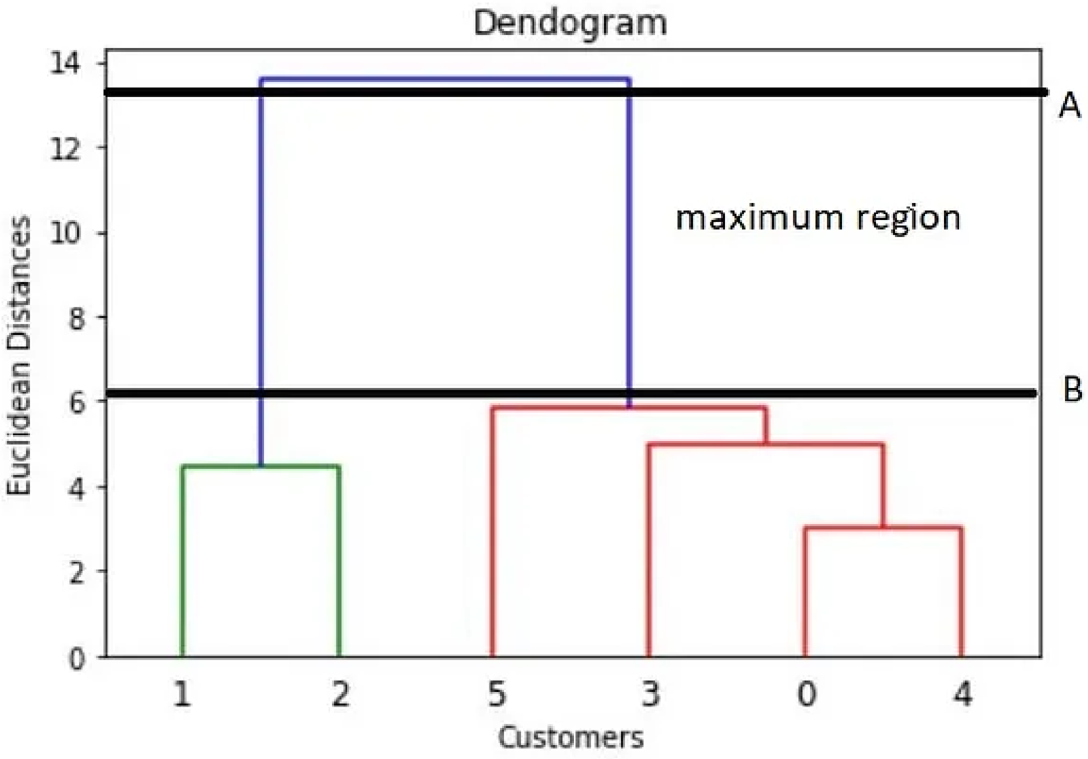


```
from scipy.cluster import hierarchy as sch
dendogram = sch.dendrogram(sch.linkage(dataset,method = 'single'))
plt.title('Dendogram')
plt.xlabel('Customers')
plt.ylabel('Euclidean Distances')
plt.show()
```

Dendrogram

As we can see in the dendrogram firstly P0 and P4 are merged, then P1 and P2 are merged, then P3 and [P0,P4] merged, then P5 and [P3,[P0,P4]] and finally [P1,P2] and [P5,[P3,[P0,P4]]].

Step 5: Find optimal number of clusters from Dendrogram.



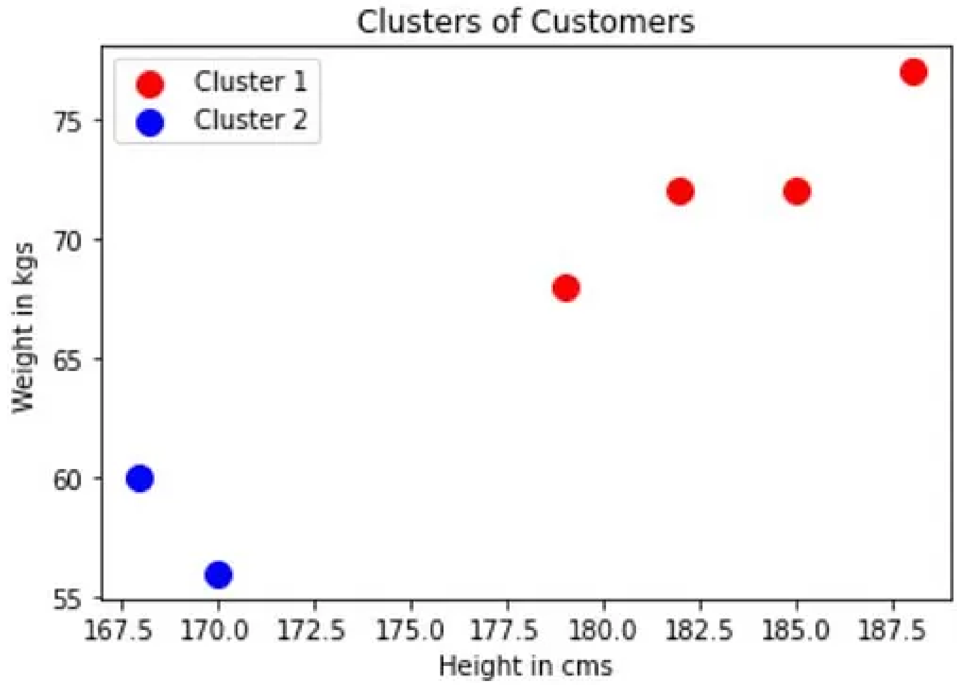
Dendrogram

For finding the optimal number of clusters we need to :
1. Determine the largest vertical distance that doesn't intersect any other cluster.
2. Draw two horizontal lines at both extremes like A and B in above figure.
3. The optimal number of cluster = number of vertical lines going through the horizontal lines.

Here, from above Dendrogram we can clearly see that there are 2 vertical lines going through horizontal lines.

Therefore, Optimal number of clusters = 2.

Visualizing the final clusters



Final Clusters

This is how Agglomerative Clustering works. I hope this helped you understanding one of the way to use Hierarchical Clustering.

Note: We have used single linkage method to determine the distance between two clustering but you can use other linkage method to compute the distance.

References:

1. https://www.youtube.com/watch?v=EFhcDnw7RGY
2. https://towardsdatascience.com/machine-learning-algorithms-part-12-hierarchical-agglomerative-clustering-example-in-python-1e18e0075019
3. https://www.youtube.com/watch?v=9U4h6pZw6f8&t=908s

Hierarchical Clustering Clustering Machine Learning Data Science