INT420: Big Data Analytics

INT491: Introduction to Data Analytics and Big Data

05: **Unsupervised Learning**
Hierarchical Clustering & K-Means

Niwan Wattanakitrungroj

School of Information Technology,
King Mongkut's University of Technology Thonburi

Semester 1 - 2022

Overview

Outline

Intro. Machine Learning

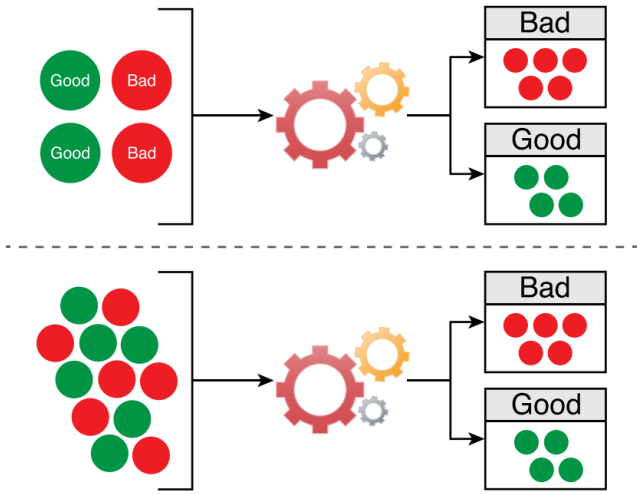
- Humans are good at spotting patterns and relationships within data. Unfortunately, we cannot process large amounts of data very quickly.
- Machines are very adept at processing large amounts of data quickly, but only if they know how.
- Machine learning and its relationship to data mining are explored through coverage of the following types of machine learning techniques:
  - Classification (Supervised Learning)
  - Clustering (Unsupervised Learning)
  - Outlier Detection
  - Filtering

Reference: Thomas Erl, Wajid Khattak, and Paul Buhler. 2016. Big Data Fundamentals: Concepts, Drivers & Techniques, Chapter8

Overview of Classification

- Classification is a supervised learning technique by which data is classified into relevant, previously learned categories. It consists of two steps:
  - The system is fed training data that is already categorized or labeled, so that it can develop an understanding of the different categories.
  - The system is fed unknown data for classification and based on the understanding it developed from the training data, the algorithm will classify the unlabeled data.

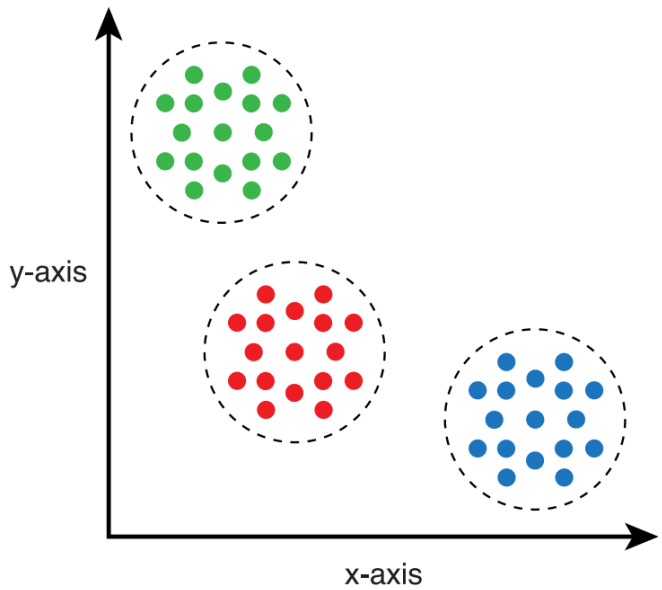
For example, a bank wants to find out which of its customers is likely to default on loan payments.



Overview of Clustering

- Clustering is an unsupervised learning technique by which data is divided into different groups so that the data in each group has similar properties.
- There is no prior learning of categories required.
- Clustering can be applied to the categorization of unknown documents and to personalized marketing campaigns by grouping together customers with similar behavior.

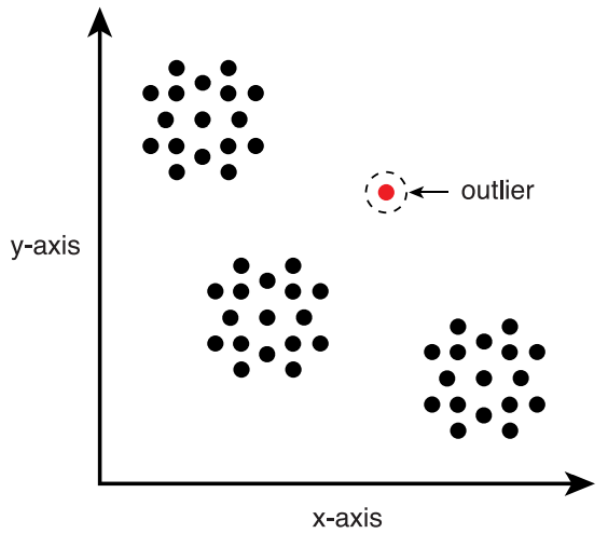
- Each group is then introduced to one or more financial products most suitable to the characteristics of the overall profile of the group.



Overview of Outlier detection

- Outlier detection is the process of finding data that is significantly different from or inconsistent with the rest of the data within a given dataset.
- Outlier detection is closely related to the concept of classification and clustering, although its algorithms focus on finding abnormal values.
- It can be based on either supervised or unsupervised learning.

- Applications for outlier detection include fraud detection, medical diagnosis, network data analysis and sensor data analysis.



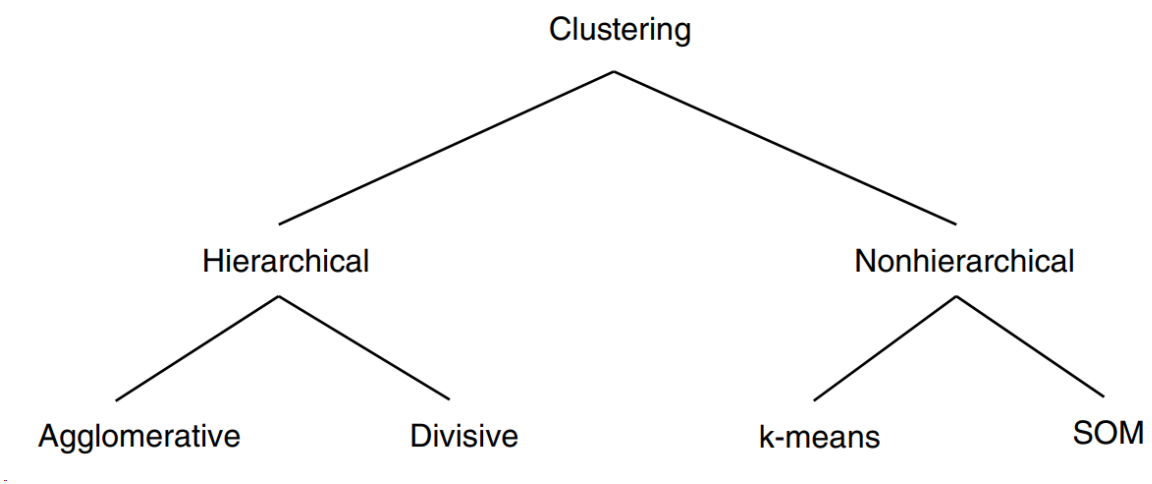
Overview of Filtering

- Items can be filtered either based on a user's own behavior or by matching the behavior of multiple users.
- Filtering is generally applied via the following two approaches:
  - collaborative filtering
  - content-based filtering
- A common medium by which filtering is implemented is via the use of a recommender system. A recommender system predicts user preferences and generates suggestions for the user accordingly.
- Sample questions can include:
  - How can only the news articles that a user is interested in be displayed ?
  - Which holiday destinations can be recommended based on the travel history of a vacationer?
  - Which other new users can be suggested as friends based on the current profile of a person?

Outline

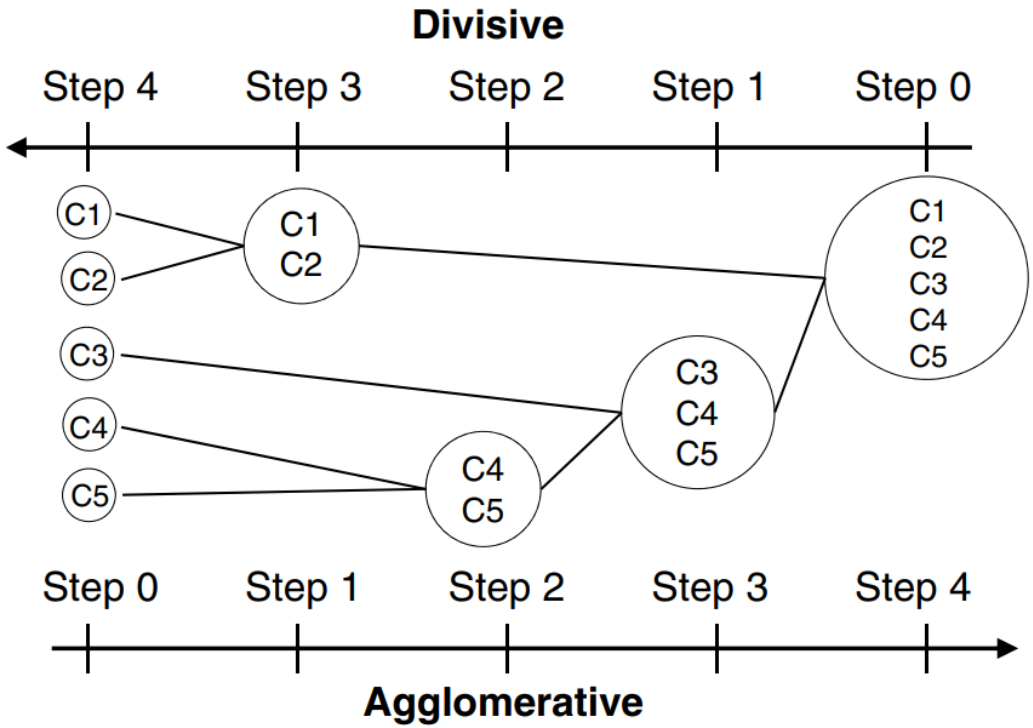
Unsupervised Learning: Clustering

Hierarchical versus Nonhierarchical Clustering Techniques



Reference: Bart Baesens. 2014. Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, Chapter 4.

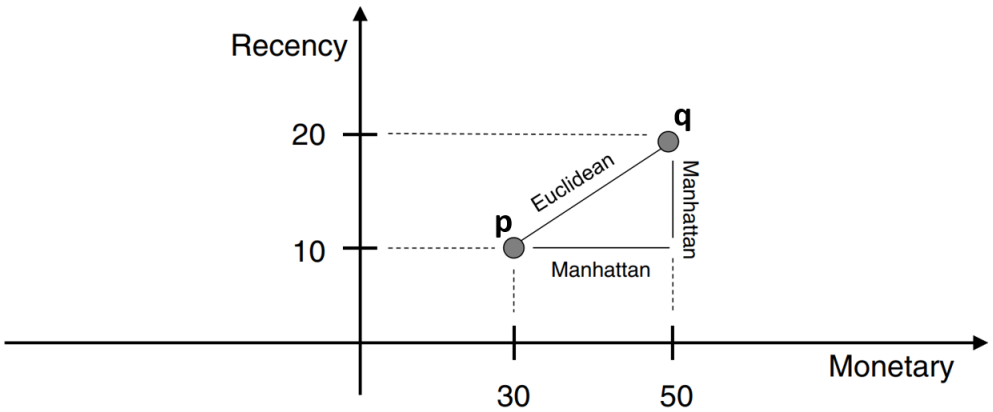
Outline



- **Divisive hierarchical clustering** starts from the whole data set in one cluster, and then breaks this up in each time smaller clusters until one observation per cluster remains
- **Agglomerative hierarchical clustering** starts from all observations in one cluster and continuing to merge the ones that are most similar until all observations make up one big cluster.

Distance or Similarity

- In order to decide on the merger or splitting, a similarity rule is needed.
- Examples of popular similarity rules are the Euclidean distance and Manhattan (city block) distance.
  - **Manhattan distance**: $dist_M(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^{m} |p_i - q_i|$
  - **Euclidean distance**: $dist_E(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{m} (p_i - q_i)^2}$
    where m is the number of dimensions/features.

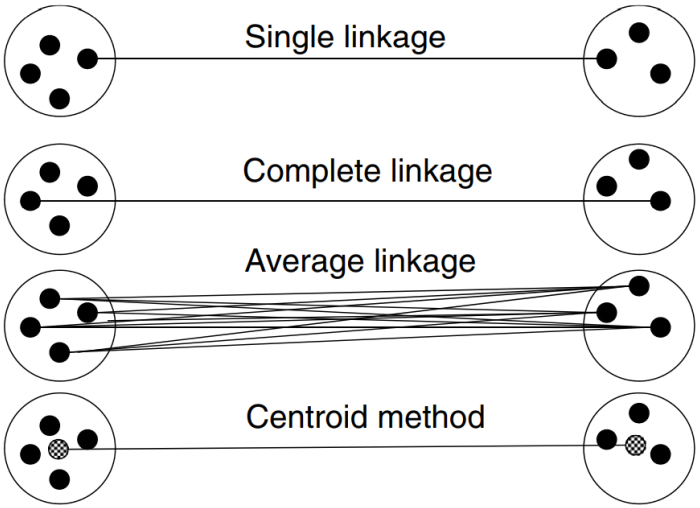


$$dist_M(\mathbf{p}, \mathbf{q}) = |53 - 30| + |20 - 10| = 30$$

$$dist_E(\mathbf{p}, \mathbf{q}) = \sqrt{(53 - 30)^2 + (20 - 10)^2} = 22$$

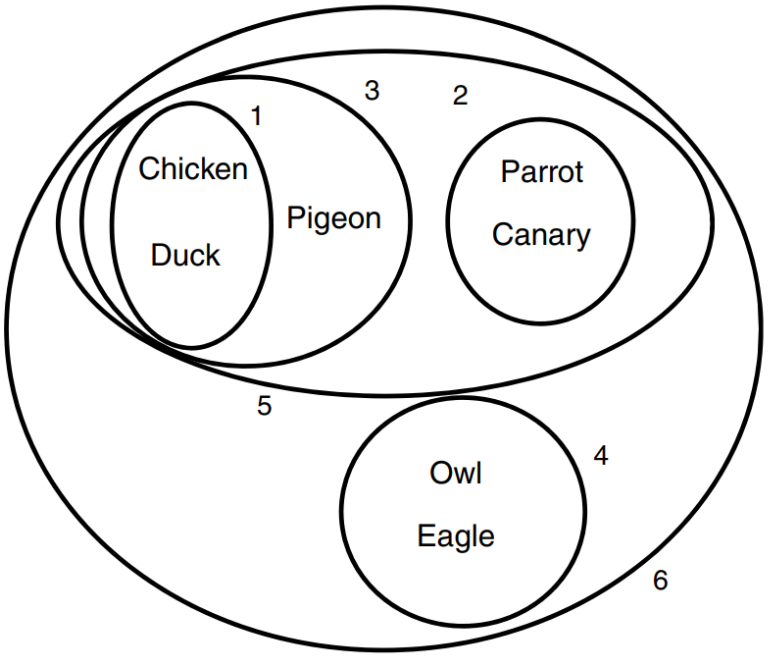
Distance Between Clusters

Various schemes can now be adopted to calculate the distance between two clusters

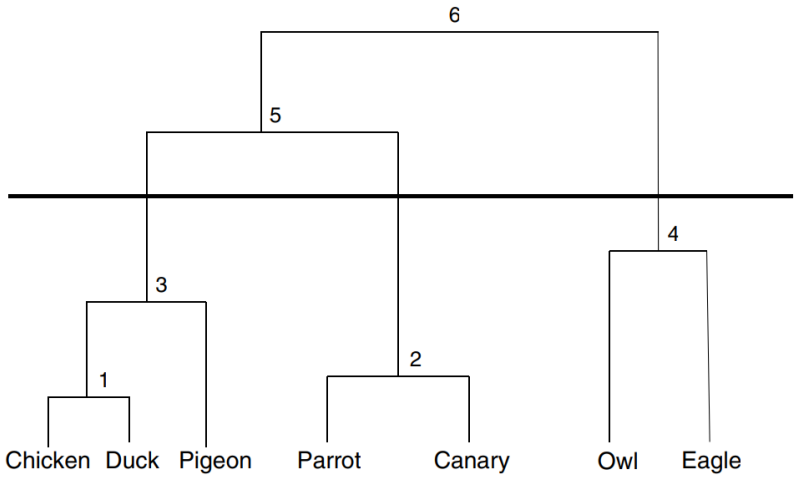


- **Single linkage method**: defines the distance between two clusters as the **shortest** possible distance, or the distance between the two most similar objects.
- **Complete linkage method**: defines the distance between two clusters as the **biggest distance**, or the distance between the two most dissimilar objects.
- **Average linkage method**: calculates the average of all possible distances
- **Centroid method**: calculates the distance between the centroids of both clusters.
- **Ward's method** merges the pair of clusters that leads to the minimum increase in total within-cluster variance after merging

Example for Clustering Birds. The numbers indicate the clustering steps.

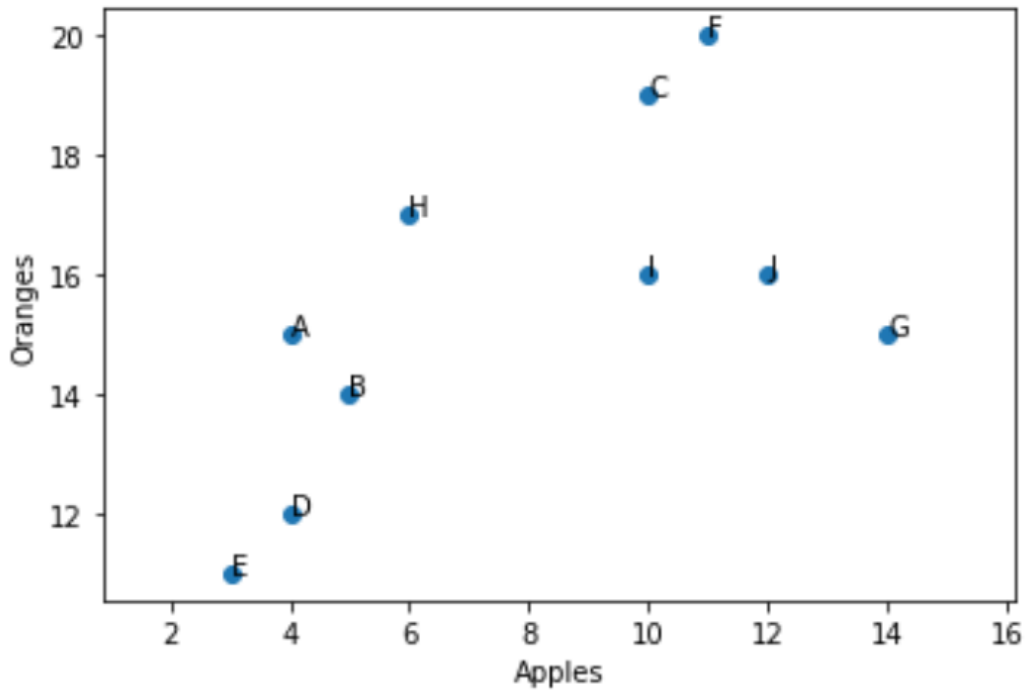


Dendrogram for Birds Example. The black line indicates the optimal clustering.



Example

Store	Apples	Oranges
A	4	15
B	5	14
C	10	19
D	4	12
E	3	11
F	11	20
G	14	15
H	6	17
I	10	16
J	12	16



Hierarchical clustering steps

Steps for Agglomerative clustering can be summarized as follows:

1. Compute the distance matrix
2. Each data point is assigned to a cluster
3. Merge the closet clusters based on a metric for the similarity between clusters
4. Update the distance matrix
5. Repeat Step 3 and Step 4 until only a single cluster remains

Example: Compute the distance matrix

Example

Store	Apples	Oranges
A	4	15
B	5	14
C	10	19
D	4	12
E	3	11
F	11	20
G	14	15
H	6	17
I	10	16
J	12	16

All pair distances

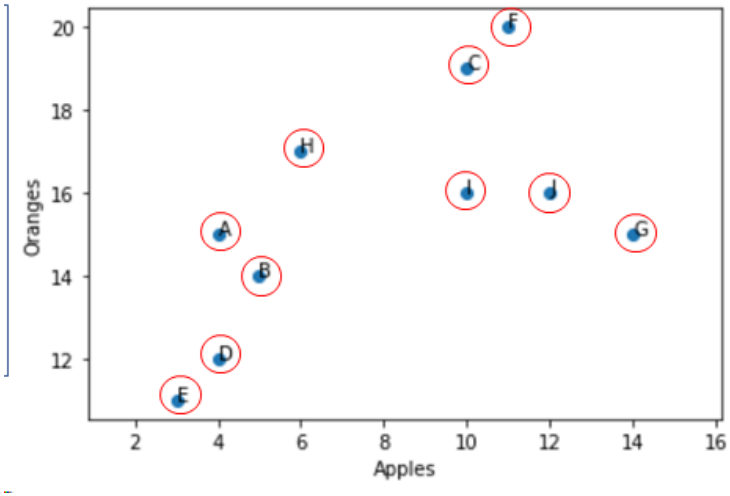
	A	B	C	D	E	F	G	H	I	J
A	0.	1.41	7.21	3.	4.12	8.6	???	2.83	6.08	8.06
B	1.41	0.	7.07	2.24	3.61	8.49	9.06	3.16	5.39	7.28
C	7.21	7.07	0.	9.22	10.63	1.41	5.66	4.47	3.	3.61
D	3.	2.24	9.22	0.	1.41	10.63	10.44	5.39	7.21	8.94
E	4.12	3.61	10.63	1.41	0.	12.04	11.7	???	8.6	10.3
F	8.6	8.49	1.41	10.63	12.04	0.	5.83	5.83	4.12	4.12
G	???	9.06	5.66	10.44	11.7	5.83	0.	8.25	4.12	2.24
H	2.83	3.16	4.47	5.39	???	5.83	8.25	0.	4.12	6.08
I	6.08	5.39	3.	7.21	8.6	4.12	4.12	4.12	0.	2.
J	8.06	7.28	3.61	8.94	10.3	4.12	2.24	6.08	2.	0.

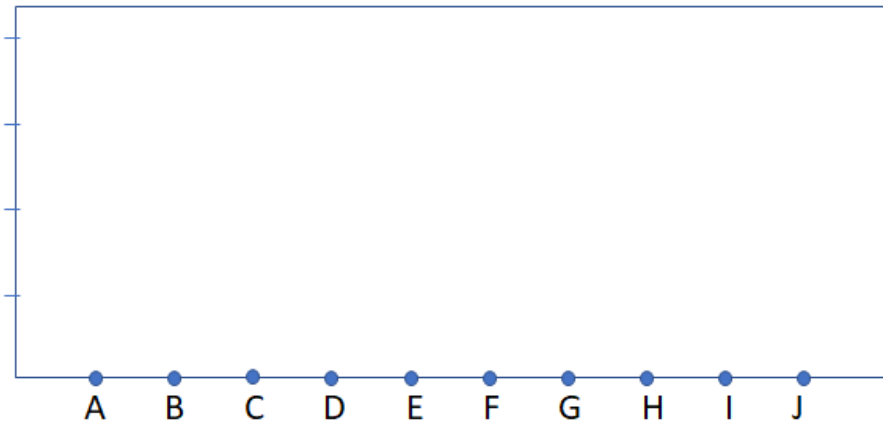
$$dist_E(B,C) = dist_E(C,B)\sqrt{(5-10)^2 + (14-19)^2} \simeq 7.07$$

Initial: Each data point is assigned to a cluster

Store	Apples	Oranges
A	4	15
B	5	14
C	10	19
D	4	12
E	3	11
F	11	20
G	14	15
H	6	17
I	10	16
J	12	16

	A	B	C	D	E	F	G	H	I	J
A	0	1.41	7.21	3	4.12	8.6	10	2.83	6.08	8.06
B	1.41	0	7.07	2.24	3.61	8.49	9.06	3.16	5.39	7.28
C	7.21	7.07	0	9.22	10.63	1.41	5.66	4.47	3	3.61
D	3	2.24	9.22	0	1.41	10.63	10.44	5.39	7.21	8.94
E	4.12	3.61	10.63	1.41	0	12.04	11.7	6.71	8.6	10.3
F	8.6	8.49	1.41	10.63	12.04	0	5.83	5.83	4.12	4.12
G	10	9.06	5.66	10.44	11.7	5.83	0	8.25	4.12	2.24
H	2.83	3.16	4.47	5.39	6.71	5.83	8.25	0	4.12	6.08
I	6.08	5.39	3	7.21	8.6	4.12	4.12	4.12	0	2
J	8.06	7.28	3.61	8.94	10.3	4.12	2.24	6.08	2	0

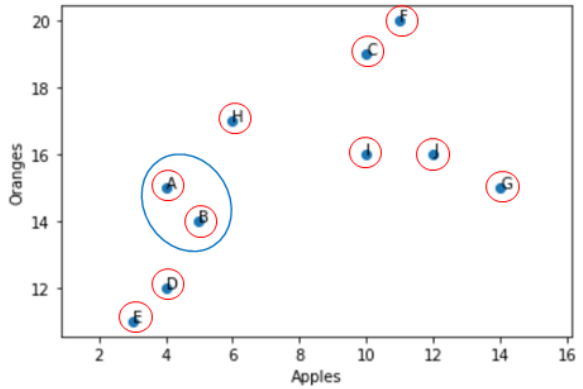


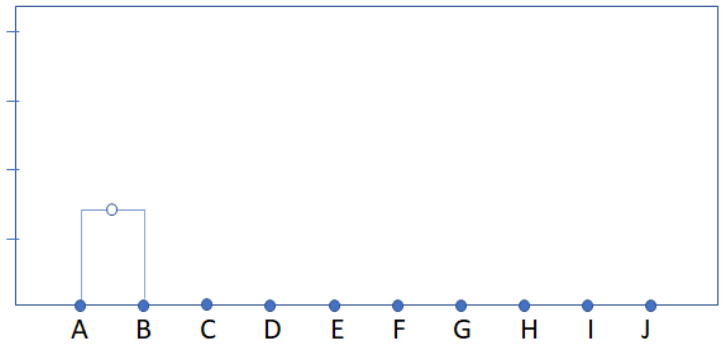


Step1: Merge the closet cluster and update distance matrix

	A	B	C	D	E	F	G	H	I	J
A	0	1.41	7.21	3	4.12	8.6	10	2.83	6.08	8.06
B	1.41	0	7.07	2.24	3.61	8.49	9.06	3.16	5.39	7.28
C	7.21	7.07	0	9.22	10.63	1.41	5.66	4.47	3	3.61
D	3	2.24	9.22	0	1.41	10.63	10.44	5.39	7.21	8.94
E	4.12	3.61	10.63	1.41	0	12.04	11.7	6.71	8.6	10.3
F	8.6	8.49	1.41	10.63	12.04	0	5.83	5.83	4.12	4.12
G	10	9.06	5.66	10.44	11.7	5.83	0	8.25	4.12	2.24
H	2.83	3.16	4.47	5.39	6.71	5.83	8.25	0	4.12	6.08
I	6.08	5.39	3	7.21	8.6	4.12	4.12	4.12	0	2
J	8.06	7.28	3.61	8.94	10.3	4.12	2.24	6.08	2	0

	AB	C	D	E	F	G	H	I	J
AB	0	7.07	2.24	4.12	8.49	9.06	2.83	5.39	7.28
C	7.07	0	9.22	10.63	1.41	5.66	4.47	3	3.61
D	2.24	9.22	0	1.41	10.63	10.44	5.39	7.21	8.94
E	4.12	10.63	1.41	0	12.04	11.7	6.71	8.6	10.3
F	8.49	1.41	10.63	12.04	0	5.83	5.83	4.12	4.12
G	9.60	5.66	10.44	11.7	5.83	0	8.25	4.12	2.24
H	2.83	4.47	5.39	6.71	5.83	8.25	0	4.12	6.08
I	5.39	3	7.21	8.6	4.12	4.12	4.12	0	2
J	7.28	3.61	8.94	10.3	4.12	2.24	6.08	2	0

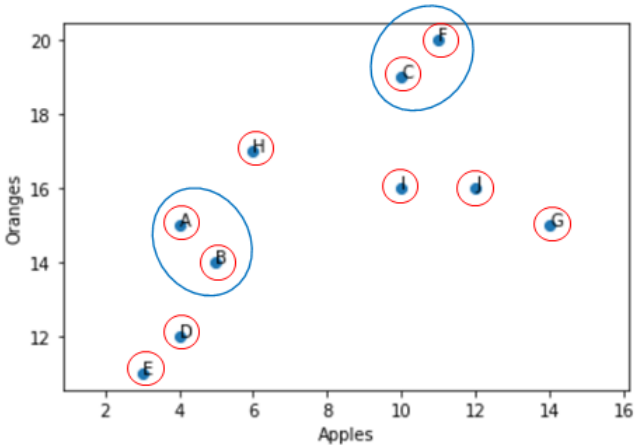


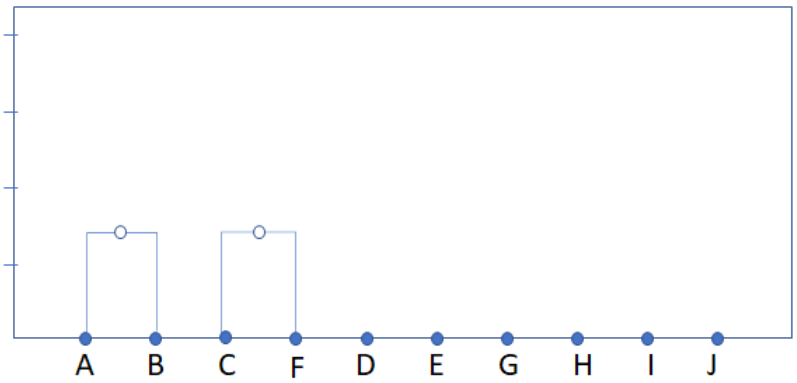


Step2: Merge the closet cluster and update distance matrix

	AB	C	D	E	F	G	H	I	J
AB	0	7.07	2.24	4.12	8.49	9.06	2.83	5.39	7.28
C	7.07	0	9.22	10.63	1.41	5.66	4.47	3	3.61
D	2.24	9.22	0	1.41	10.63	10.44	5.39	7.21	8.94
E	4.12	10.63	1.41	0	12.04	11.7	6.71	8.6	10.3
F	8.49	1.41	10.63	12.04	0	5.83	5.83	4.12	4.12
G	9.60	5.66	10.44	11.7	5.83	0	8.25	4.12	2.24
H	2.83	4.47	5.39	6.71	5.83	8.25	0	4.12	6.08
I	5.39	3	7.21	8.6	4.12	4.12	4.12	0	2
J	7.28	3.61	8.94	10.3	4.12	2.24	6.08	2	0

	AB	CF	D	E	G	H	I	J
AB	0	7.07	2.24	4.12	9.06	2.83	5.39	7.28
CF	7.07	0	9.22	10.63	5.66	4.47	3	3.61
D	2.24	9.22	0	1.41	10.44	5.39	7.21	8.94
E	4.12	10.63	1.41	0	11.7	6.71	8.6	10.3
G	9.60	5.66	10.44	11.7	0	8.25	4.12	2.24
H	2.83	4.47	5.39	6.71	8.25	0	4.12	6.08
I	5.39	3	7.21	8.6	4.12	4.12	0	2
J	7.28	3.61	8.94	10.3	2.24	6.08	2	0

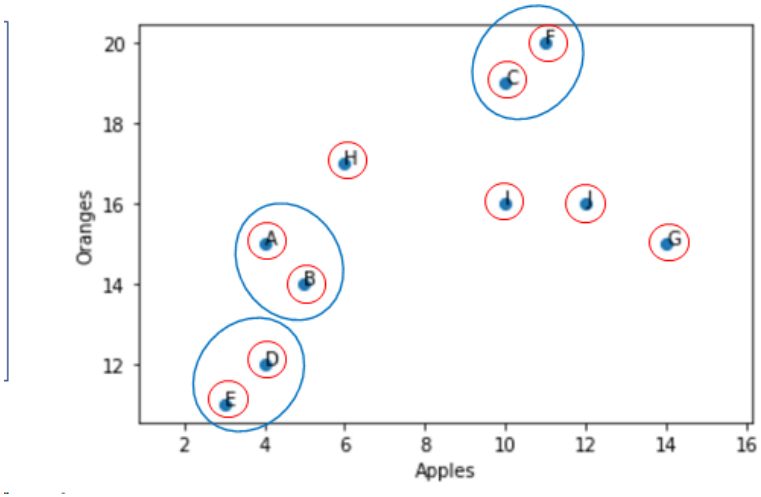


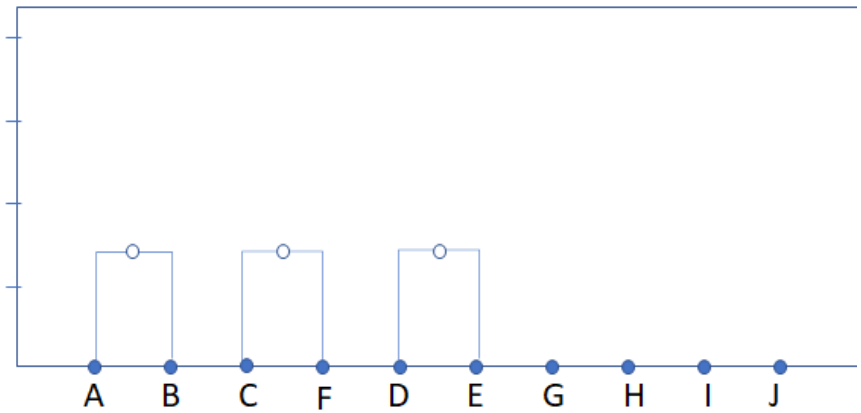


Step3: Merge the closet cluster and update distance matrix

	AB	CF	D	E	G	H	I	J
AB	0	7.07	2.24	4.12	9.06	2.83	5.39	7.28
CF	7.07	0	9.22	10.63	5.66	4.47	3	3.61
D	2.24	9.22	0	1.41	10.44	5.39	7.21	8.94
E	4.12	10.63	1.41	0	11.7	6.71	8.6	10.3
G	9.60	5.66	10.44	11.7	0	8.25	4.12	2.24
H	2.83	4.47	5.39	6.71	8.25	0	4.12	6.08
I	5.39	3	7.21	8.6	4.12	4.12	0	2
J	7.28	3.61	8.94	10.3	2.24	6.08	2	0

	AB	CF	DE	G	H	I	J
AB	0	7.07	2.24	9.06	2.83	5.39	7.28
CF	7.07	0	9.22	5.66	4.47	3	3.61
DE	2.24	9.22	0	10.44	5.39	7.21	8.94
G	9.60	5.66	10.44	0	8.25	4.12	2.24
H	2.83	4.47	5.39	8.25	0	4.12	6.08
I	5.39	3	7.21	4.12	4.12	0	2
J	7.28	3.61	8.94	2.24	6.08	2	0

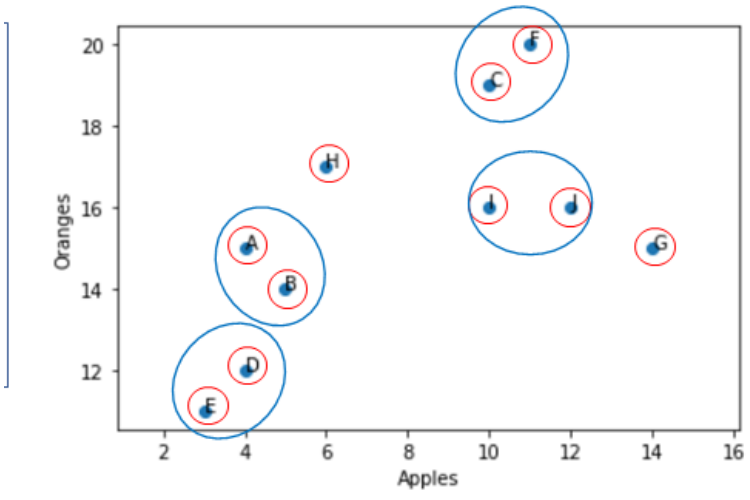


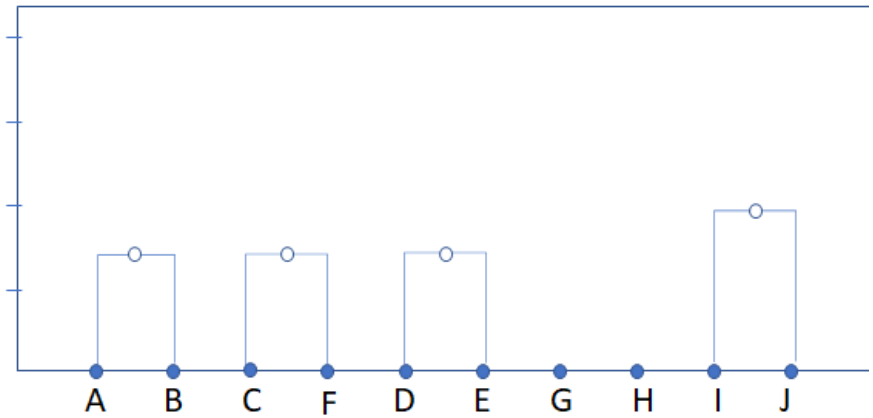


Step4: Merge the closet cluster and update distance matrix

	AB	CF	DE	G	H	I	J
AB	0	7.07	2.24	9.06	2.83	5.39	7.28
CF	7.07	0	9.22	5.66	4.47	3	3.61
DE	2.24	9.22	0	10.44	5.39	7.21	8.94
G	9.60	5.66	10.44	0	8.25	4.12	2.24
H	2.83	4.47	5.39	8.25	0	4.12	6.08
I	5.39	3	7.21	4.12	4.12	0	2
J	7.28	3.61	8.94	2.24	6.08	2	0

	AB	CF	DE	G	H	IJ
AB	0	7.07	2.24	9.06	2.83	5.39
CF	7.07	0	9.22	5.66	4.47	3
DE	2.24	9.22	0	10.44	5.39	7.21
G	9.60	5.66	10.44	0	8.25	2.24
H	2.83	4.47	5.39	8.25	0	4.12
IJ	5.39	3	7.21	2.24	4.12	0

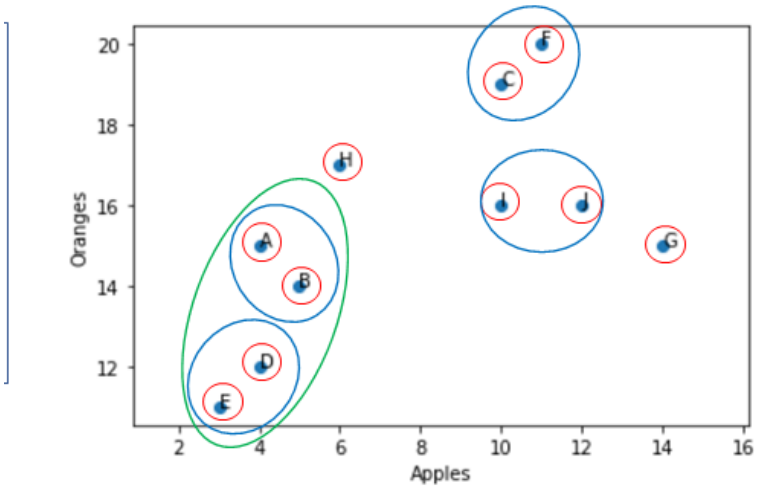


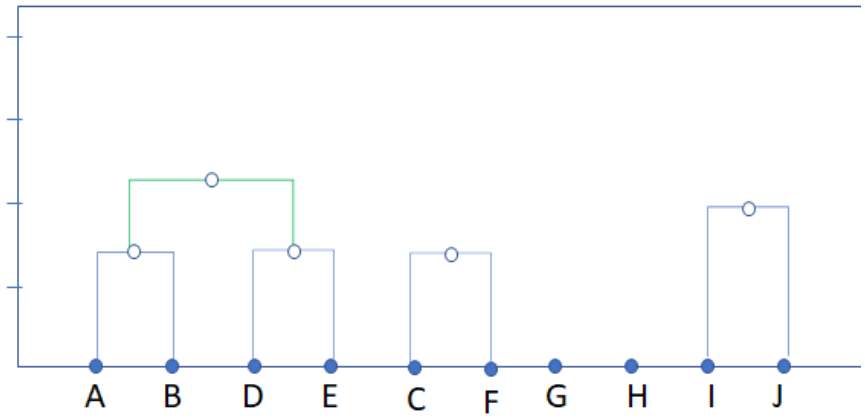


Step5: Merge the closet cluster and update distance matrix

	AB	CF	DE	G	H	IJ
AB	0	7.07	2.24	9.06	2.83	5.39
CF	7.07	0	9.22	5.66	4.47	3
DE	2.24	9.22	0	10.44	5.39	7.21
G	9.60	5.66	10.44	0	8.25	2.24
H	2.83	4.47	5.39	8.25	0	4.12
IJ	5.39	3	7.21	2.24	4.12	0

	ABDE	CF	G	H	IJ
ABDE	0	7.07	9.06	2.83	5.39
CF	7.07	0	5.66	4.47	3
G	9.06	5.66	0	8.25	2.24
H	2.83	4.47	8.25	0	4.12
IJ	5.39	3	2.24	4.12	0

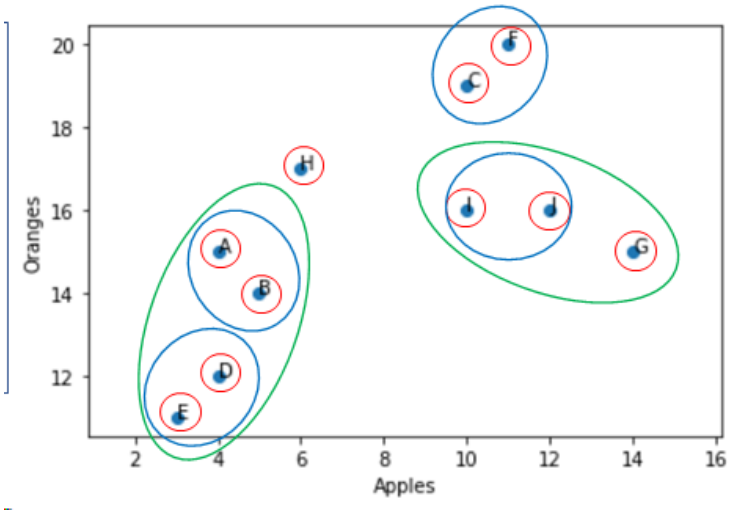


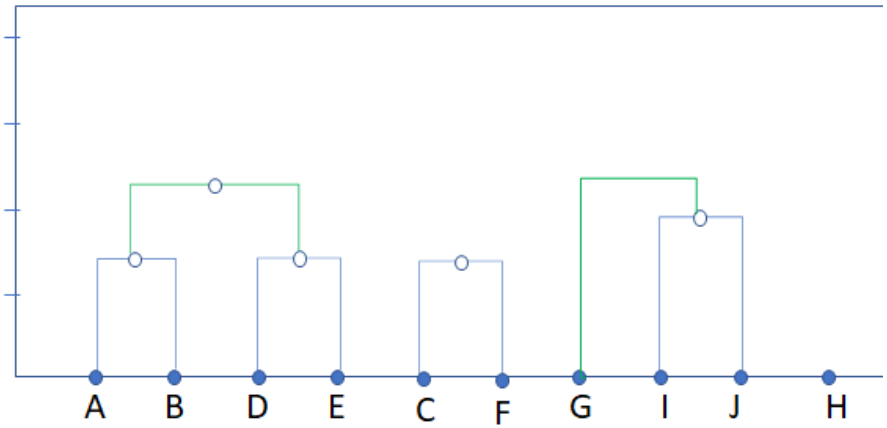


Step6: Merge the closet cluster and update distance matrix

	ABDE	CF	G	H	IJ
ABDE	0	7.07	9.06	2.83	5.39
CF	7.07	0	5.66	4.47	3
G	9.06	5.66	0	8.25	2.24
H	2.83	4.47	8.25	0	4.12
IJ	5.39	3	2.24	4.12	0

	ABDE	CF	GIJ	H
ABDE	0	7.07	5.39	2.83
CF	7.07	0	3	4.47
GIJ	5.39	3	0	4.12
H	2.83	4.47	4.12	0

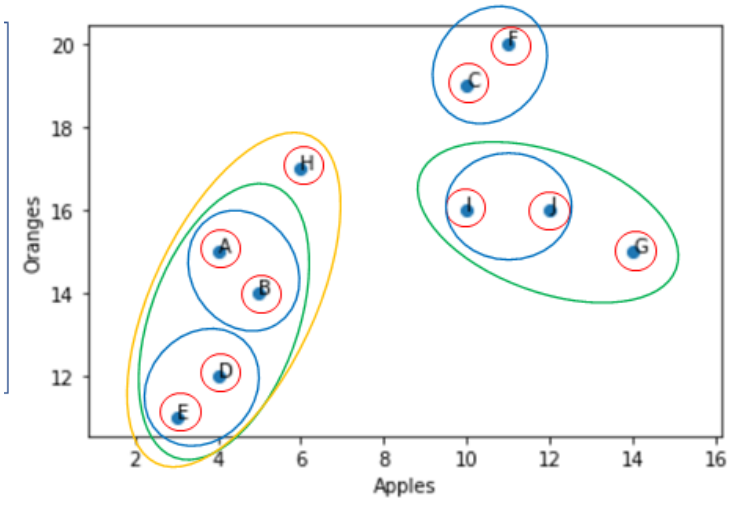


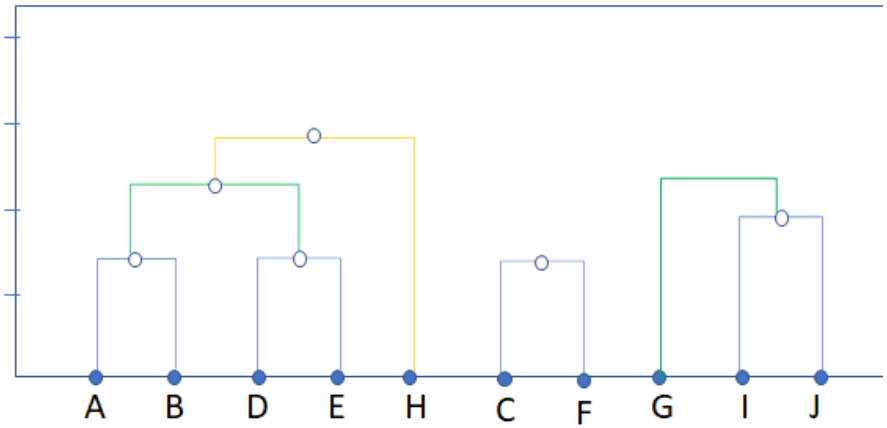


Step7: Merge the closet cluster and update distance matrix

	ABDE	CF	GIJ	H
ABDE	0	7.07	5.39	2.83
CF	7.07	0	3	4.47
GIJ	5.39	3	0	4.12
H	2.83	4.47	4.12	0

	ABDEH	CF	GIJ
ABDEH	0	4.47	4.12
CF	4.47	0	3
GIJ	4.12	3	0

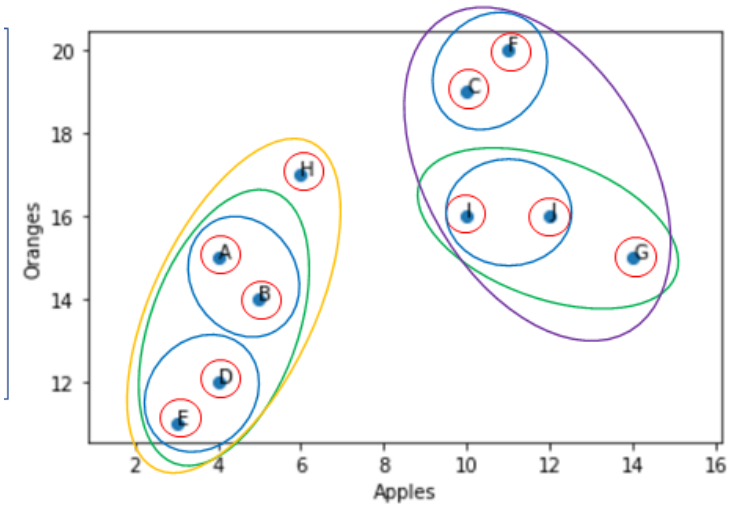


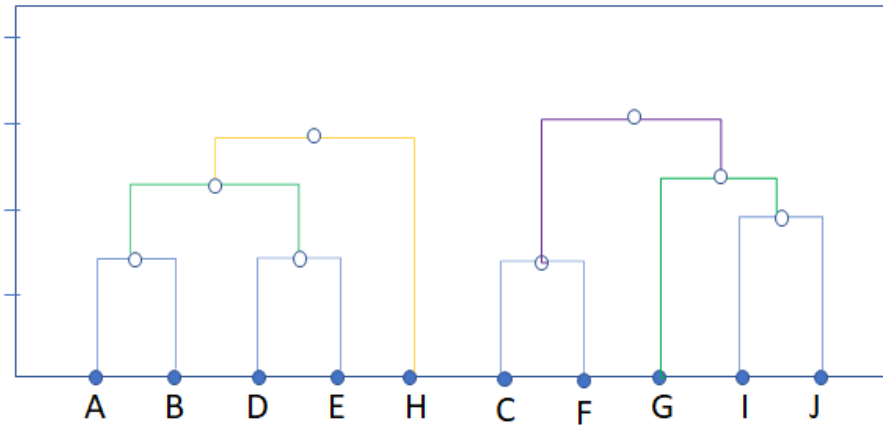


Step8: Merge the closet cluster and update distance matrix

	ABDEH	CF	GIJ
ABDEH	0	4.47	4.12
CF	4.47	0	3
GIJ	4.12	3	0

	ABDEH	CFGIJ
ABDEH	0	4.12
CFGIJ	4.12	0

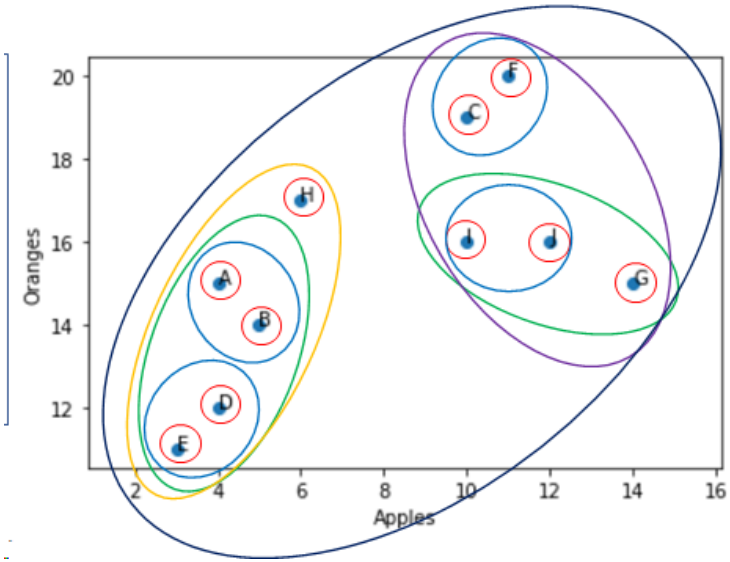


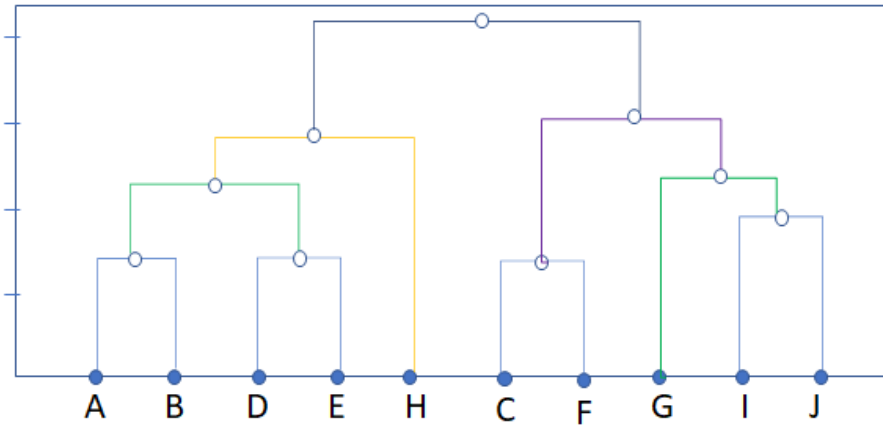


Step9: Merge the closet cluster and update distance matrix

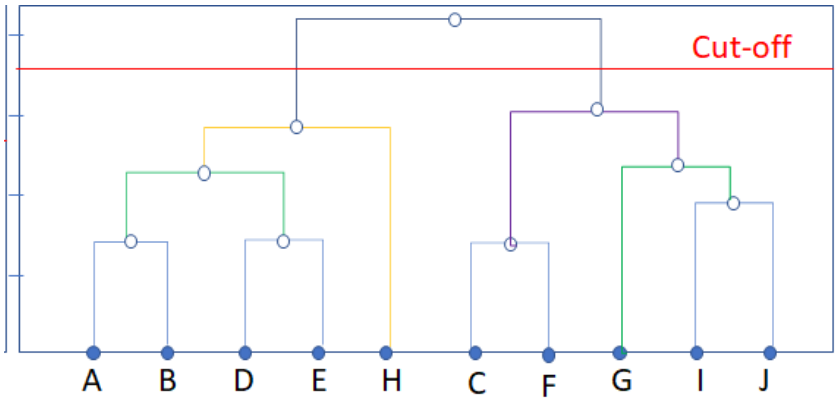
	ABDEH	CFGIJ
ABDEH	0	4.12
CFGIJ	4.12	0

	ABDEHCFGIJ
ABDEHCFGIJ	0

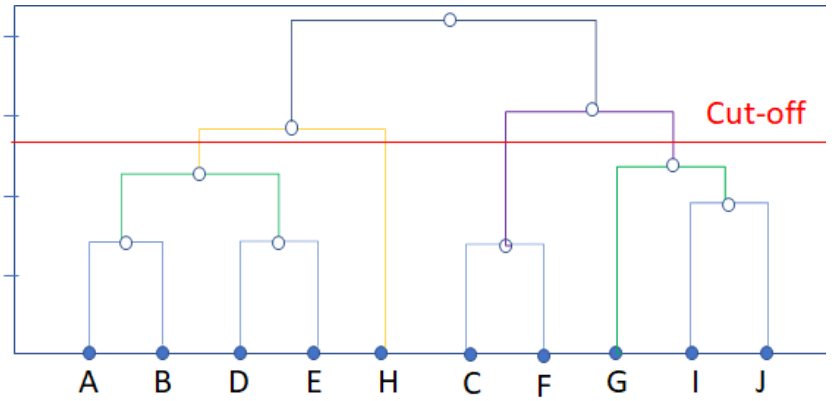




Example Results



Cluster1 = {A,B,D,E,H}
Cluster2 = {C,F,G, I, J}



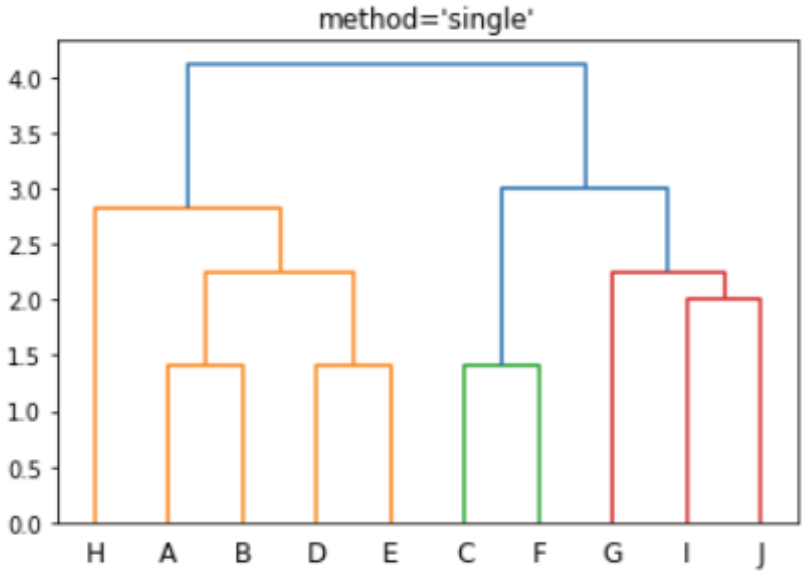
Cluster1 = {A,B,D,E}
Cluster2 = {H}
Cluster3 = {C,F}
Cluster4 = {G, I, J}

Hierarchical Clustering with Python

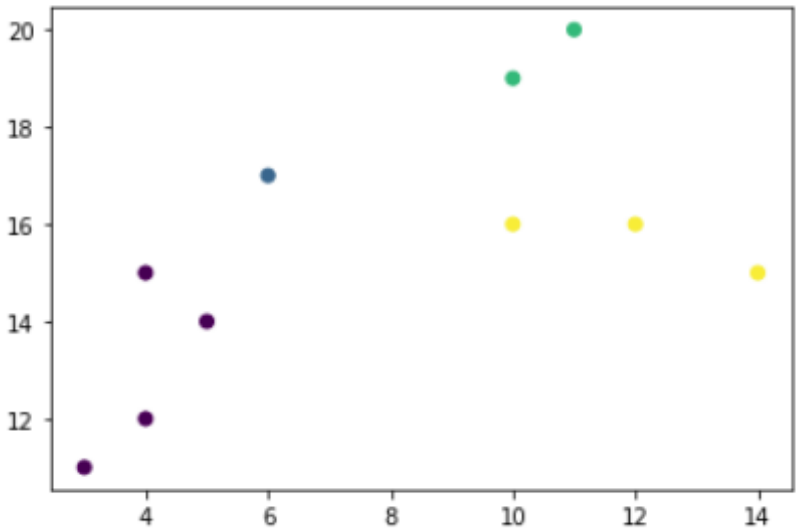
```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import dendrogram, linkage
from scipy.spatial import distance_matrix
import pandas as pd

x = [4, 5, 10, 4, 3, 11, 14 , 6, 10, 12]
y = [15, 14, 19, 12, 11, 20, 15, 17, 16, 16]
store = ['A','B','C','D','E','F','G','H','I','J']

df = pd.DataFrame()
df['Store'] = store
df['Apples'] = x
df['Oranges'] = y
data = df[['Apples','Oranges']]

linkage_data = linkage(data, method='single', metric='euclidean')
dendrogram(linkage_data,labels=df['Store'].to_list())

plt.title("method='single'")
plt.show()
```

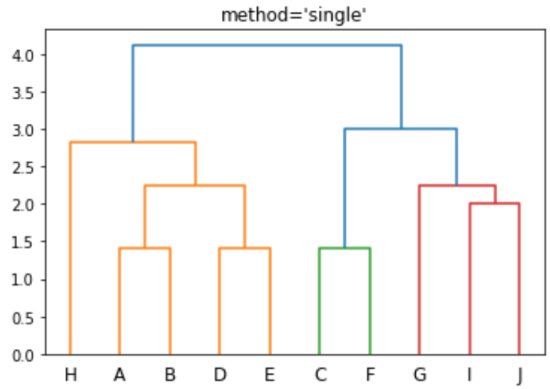
More Details: https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.linkage.html

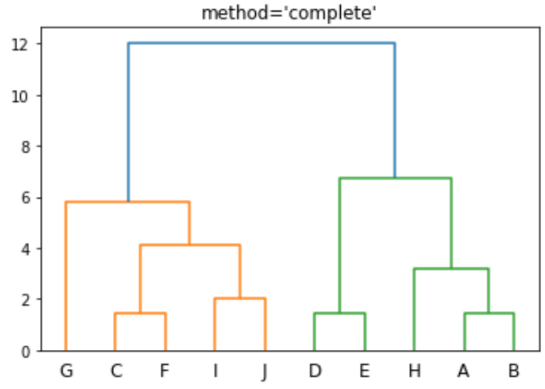
https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.fcluster.html

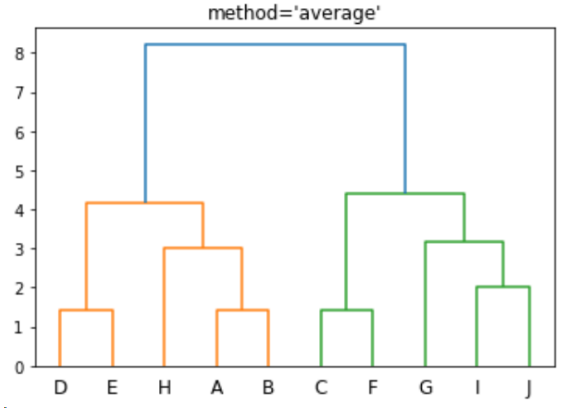


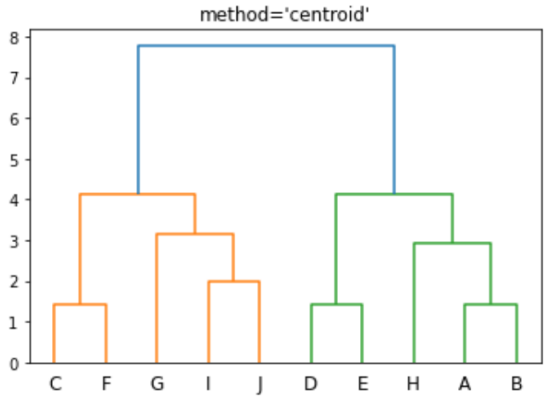

```
cluster_id = fcluster(linkage_data, t=2.5, criterion='distance')
plt.scatter(df['Apples'],df['Oranges'], c=cluster_id)
plt.show()
```

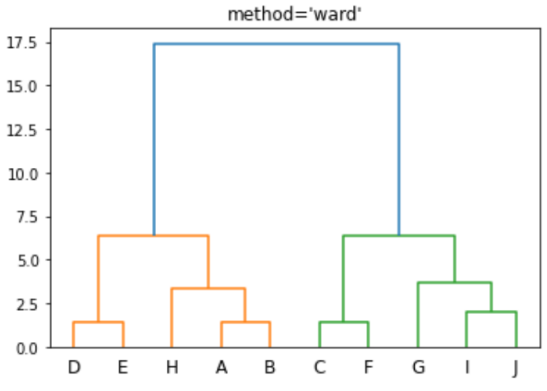



method='single'
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
H A B D E C F G I J


method='complete'
12
10
8
6
4
2
0
G C F I J D E H A B


method='average'
8
7
6
5
4
3
2
1
0
D E H A B C F G I J


method='centroid'
8
7
6
5
4
3
2
1
0
C F G I J D E H A B


method='ward'
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
D E H A B C F G I J

Outline

K-Means Clustering

K-means clustering is a nonhierarchical procedure that works along the following steps:

1. Select k observations as initial cluster centroids (seeds).
2. Assign each observation to the cluster that has the closest centroid (for example, in Euclidean sense).
3. When all observations have been assigned, recalculate the positions of the k centroids.
4. Repeat until the cluster centroids no longer change.

Reference:
Bart Baesens. 2014. Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, Chapter 4.

Example:

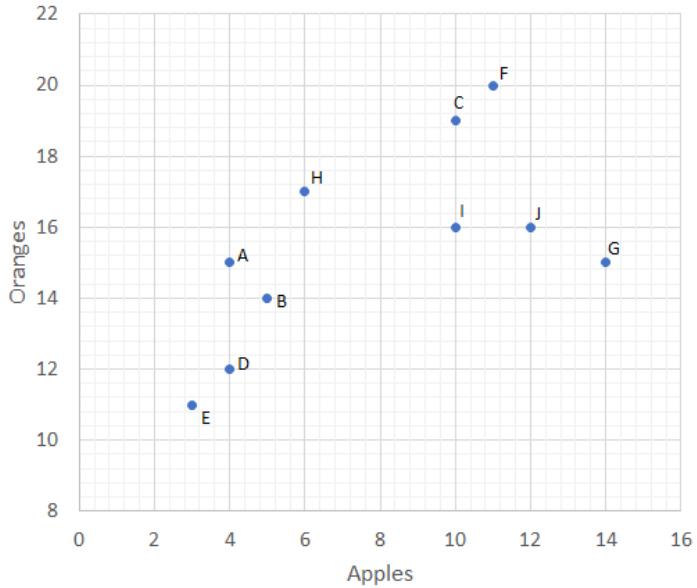
K=2

Cluster centers

C1 = C2 =

Euclidean distance: $dist_E(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{m} (p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15			
B	5	14			
C	10	19			
D	4	12			
E	3	11			
F	11	20			
G	14	15			
H	6	17			
I	10	16			
J	12	16			



Example: Initial cluster centers

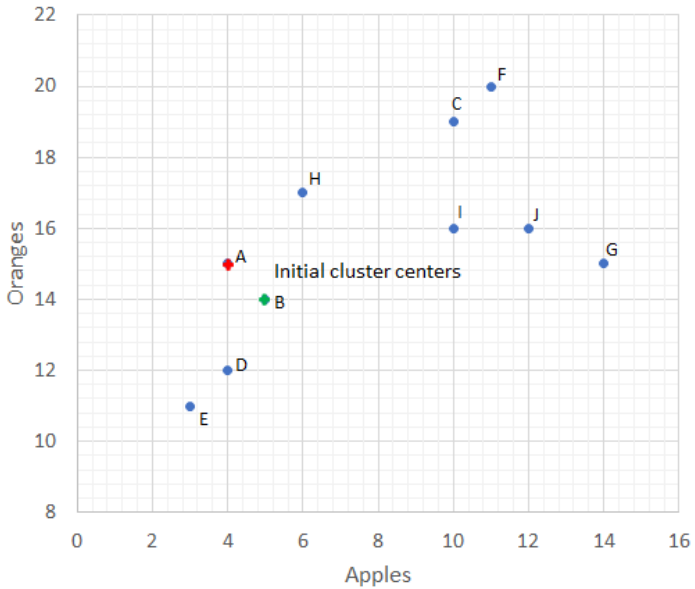
K=2

Cluster centers

C1 =(4,15) C2 =(5,14)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15			
B	5	14			
C	10	19			
D	4	12			
E	3	11			
F	11	20			
G	14	15			
H	6	17			
I	10	16			
J	12	16			



Iteration 1

Example: Find distance to centers and assign each data point to closet center.

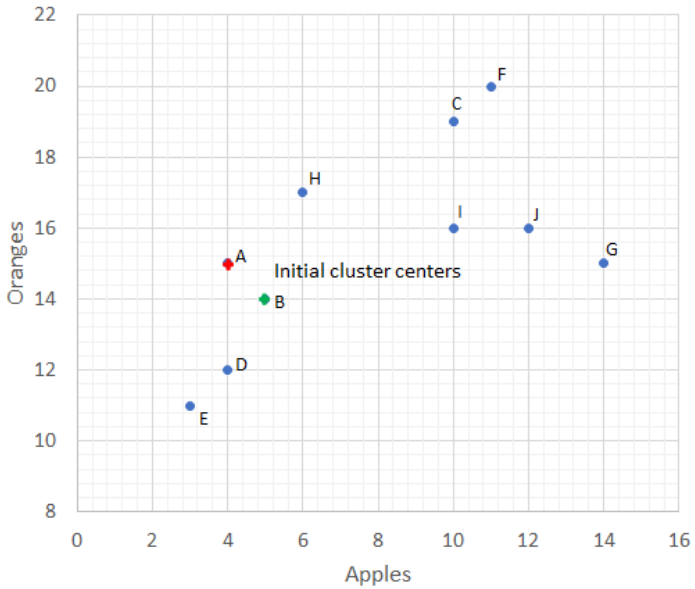
K=2

Cluster centers

C1 =(4,15) C2 =(5,14)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0+0)^0.5		
B	5	14	(1+1)^0.5		
C	10	19	(36+16)^0.5		
D	4	12	(0+9)^0.5		
E	3	11	(1+16)^0.5		
F	11	20	(49+25)^0.5		
G	14	15	(100+0)^0.5		
H	6	17	(4+4)^0.5		
I	10	16	(36+1)^0.5		
J	12	16	(64+1)^0.5		



Iteration 1

Example: Find distance to centers and assign each data point to closet center (cont.)

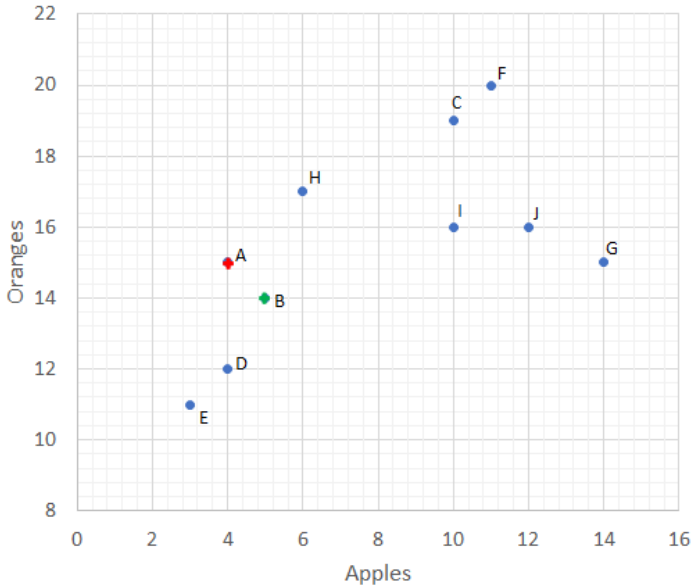
K=2

Cluster centers

C1 =(4,15) C2 =(5,14)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0+0)^0.5	(1+1)^0.5	
B	5	14	(1+1)^0.5	(0+0)^0.5	
C	10	19	(36+16)^0.5	(25+25)^0.5	
D	4	12	(0+9)^0.5	(1+4)^0.5	
E	3	11	(1+16)^0.5	(4+9)^0.5	
F	11	20	(49+25)^0.5	(36+36)^0.5	
G	14	15	(100+0)^0.5	(81+1)^0.5	
H	6	17	(4+4)^0.5	(1+9)^0.5	
I	10	16	(36+1)^0.5	(25+4)^0.5	
J	12	16	(64+1)^0.5	(49+4)^0.5	



Iteration 1

Example: Find distance to centers and assign each data point to closet center (cont.)

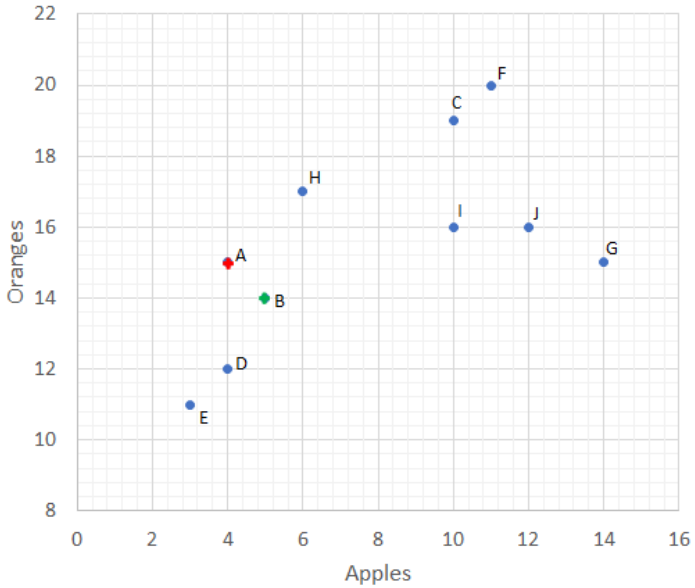
K=2

Cluster centers

C1 =(4,15) C2 =(5,14)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0+0)^0.5	(1+1)^0.5	C1
B	5	14	(1+1)^0.5	(0+0)^0.5	C2
C	10	19	(36+16)^0.5	(25+25)^0.5	C2
D	4	12	(0+9)^0.5	(1+4)^0.5	C2
E	3	11	(1+16)^0.5	(4+9)^0.5	C2
F	11	20	(49+25)^0.5	(36+36)^0.5	C2
G	14	15	(100+0)^0.5	(81+1)^0.5	C2
H	6	17	(4+4)^0.5	(1+9)^0.5	C1
I	10	16	(36+1)^0.5	(25+4)^0.5	C2
J	12	16	(64+1)^0.5	(49+4)^0.5	C2



Iteration 1

Example: Find distance to centers and assign each data point to closet center (cont.)

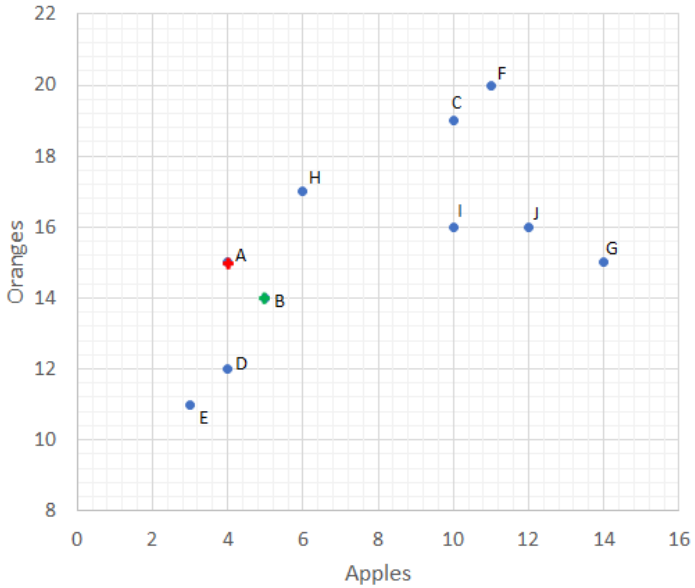
K=2

Cluster centers

C1 =(4,15) C2 =(5,14)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0+0)^0.5	(1+1)^0.5	C1
B	5	14	(1+1)^0.5	(0+0)^0.5	C2
C	10	19	(36+16)^0.5	(25+25)^0.5	C2
D	4	12	(0+9)^0.5	(1+4)^0.5	C2
E	3	11	(1+16)^0.5	(4+9)^0.5	C2
F	11	20	(49+25)^0.5	(36+36)^0.5	C2
G	14	15	(100+0)^0.5	(81+1)^0.5	C2
H	6	17	(4+4)^0.5	(1+9)^0.5	C1
I	10	16	(36+1)^0.5	(25+4)^0.5	C2
J	12	16	(64+1)^0.5	(49+4)^0.5	C2



Iteration 1

Example: Update cluster centers

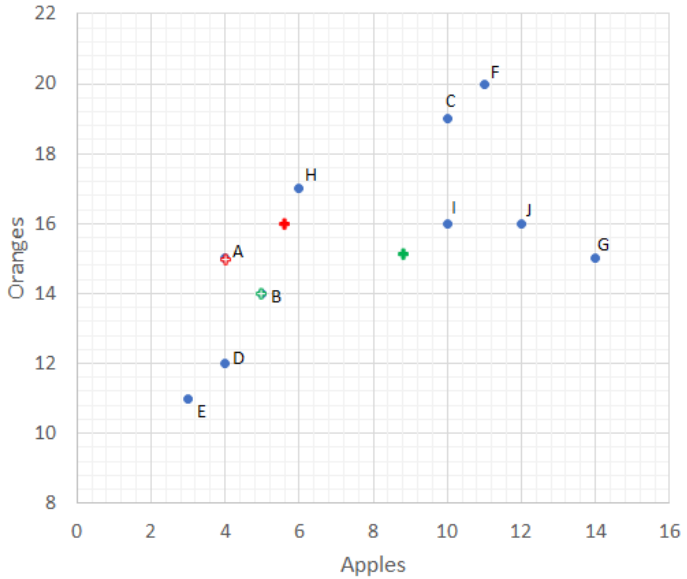
K=2

Cluster centers

C1 =(4,15) C2 =(5,14)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0+0)^0.5	(1+1)^0.5	C1
B	5	14	(1+1)^0.5	(0+0)^0.5	C2
C	10	19	(36+16)^0.5	(25+25)^0.5	C2
D	4	12	(0+9)^0.5	(1+4)^0.5	C2
E	3	11	(1+16)^0.5	(4+9)^0.5	C2
F	11	20	(49+25)^0.5	(36+36)^0.5	C2
G	14	15	(100+0)^0.5	(81+1)^0.5	C2
H	6	17	(4+4)^0.5	(1+9)^0.5	C1
I	10	16	(36+1)^0.5	(25+4)^0.5	C2
J	12	16	(64+1)^0.5	(49+4)^0.5	C2



Update Centers

C1 = ((4+6)/2,(15+17)/2) = (5,16)

C2 = ((5+10+4+3+11+14+10+12)/8,(14+19+12+11+20+15+16+16)/8)
= (8.625,15.375) ~ (8.6,15.4)

Iteration 1

Example: Repeat until centers do not change

K=2

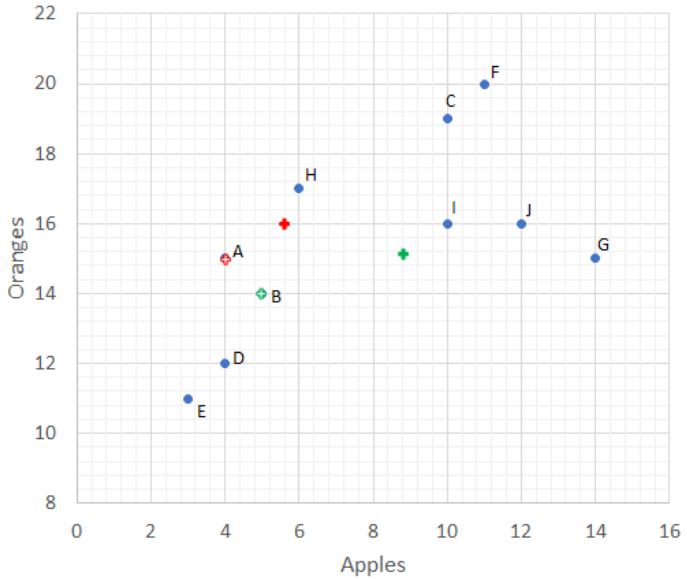
Cluster centers

C1 =(4,15) C2 =(5,14)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0+0)^0.5	(1+1)^0.5	C1
B	5	14	(1+1)^0.5	(0+0)^0.5	C2
C	10	19	(36+16)^0.5	(25+25)^0.5	C2
D	4	12	(0+9)^0.5	(1+4)^0.5	C2
E	3	11	(1+16)^0.5	(4+9)^0.5	C2
F	11	20	(49+25)^0.5	(36+36)^0.5	C2
G	14	15	(100+0)^0.5	(81+1)^0.5	C2
H	6	17	(4+4)^0.5	(1+9)^0.5	C1
I	10	16	(36+1)^0.5	(25+4)^0.5	C2
J	12	16	(64+1)^0.5	(49+4)^0.5	C2

Centers change.
Continue Iteration 2.



Update Centers

C1 = ((4+6)/2,(15+17)/2) = (5,16)

C2 = ((5+10+4+3+11+14+10+12)/8,(14+19+12+11+20+15+16+16)/8)
= (8.625,15.375) ~ (8.6,15.4)

Iteration 1

Example: Find distance to centers and assign each data point to closet center

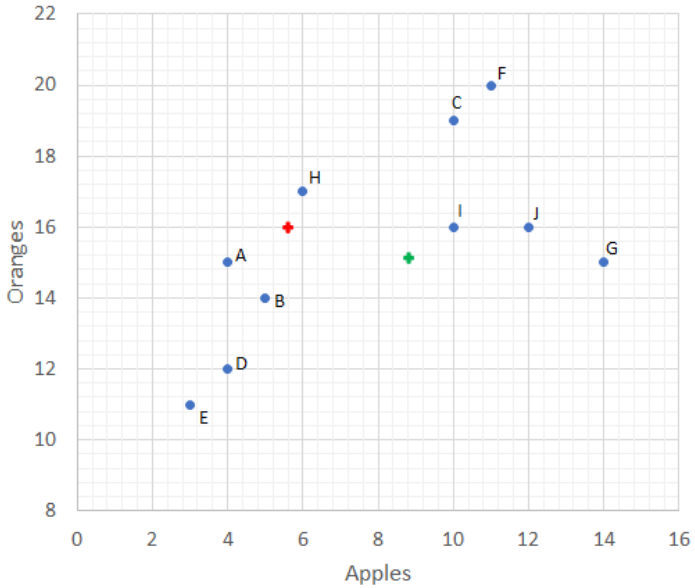
K=2

Cluster centers

C1 =(5,16) C2 =(8.6,15.4)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15			
B	5	14			
C	10	19			
D	4	12			
E	3	11			
F	11	20			
G	14	15			
H	6	17			
I	10	16			
J	12	16			



Update Centers

C1 =

C2 =

Iteration 2

Example: Find distance to centers and assign each data point to closet center (cont.)

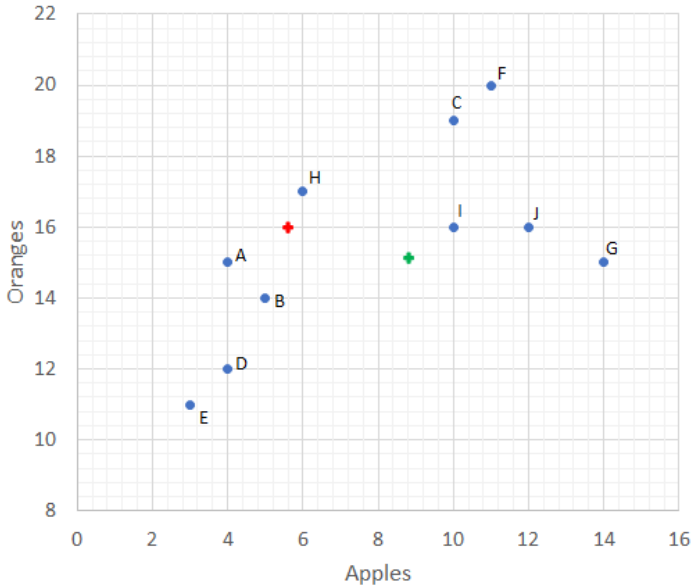
K=2

Cluster centers

C1 =(5,16) C2 =(8.6,15.4)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(1^2+1^2)^0.5		
B	5	14	(0+2^2)^0.5		
C	10	19	(5^2+3^2)^0.5		
D	4	12	(1^2+4^2)^0.5		
E	3	11	(2^2+5^2)^0.5		
F	11	20	(6^2+4^2)^0.5		
G	14	15	(9^2+1^2)^0.5		
H	6	17	(1^2+1^2)^0.5		
I	10	16	(5^2+0)^0.5		
J	12	16	(8^2+0)^0.5		



Update Centers

C1 =

C2 =

Iteration 2

Example: Find distance to centers and assign each data point to closet center (cont.)

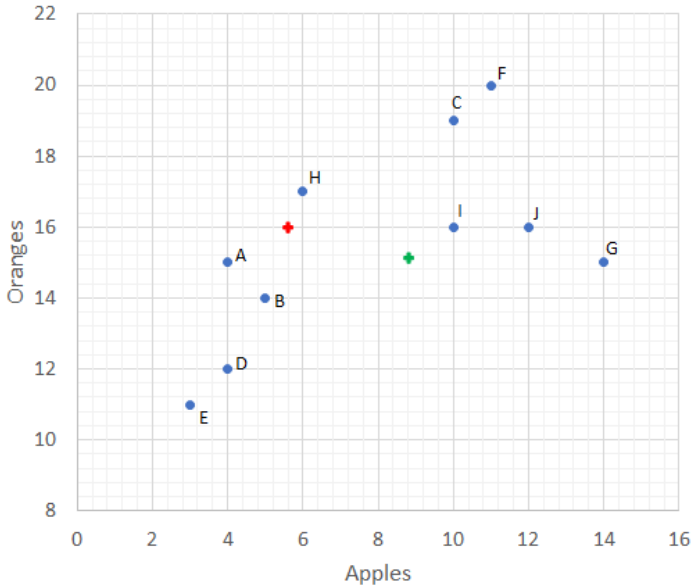
K=2

Cluster centers

C1 =(5,16) C2 =(8.6,15.4)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(1^2+1^2)^0.5	(4.6^2+0.4^2)^0.5	
B	5	14	(0+2^2)^0.5	(3.6^2+1.4^2)^0.5	
C	10	19	(5^2+3^2)^0.5	(1.4^2+3.6^2)^0.5	
D	4	12	(1^2+4^2)^0.5	(4.6^2+3.4^2)^0.5	
E	3	11	(2^2+5^2)^0.5	(5.6^2+4.4^2)^0.5	
F	11	20	(6^2+4^2)^0.5	(2.4^2+4.6^2)^0.5	
G	14	15	(9^2+1^2)^0.5	(5.4^2+0.4^2)^0.5	
H	6	17	(1^2+1^2)^0.5	(2.6^2+1.6^2)^0.5	
I	10	16	(5^2+0)^0.5	(1.4^2+0.6^2)^0.5	
J	12	16	(8^2+0)^0.5	(3.4^2+0.6^2)^0.5	



Update Centers

C1 =

C2 =

Iteration 2

Example: Find distance to centers and assign each data point to closet center (cont.)

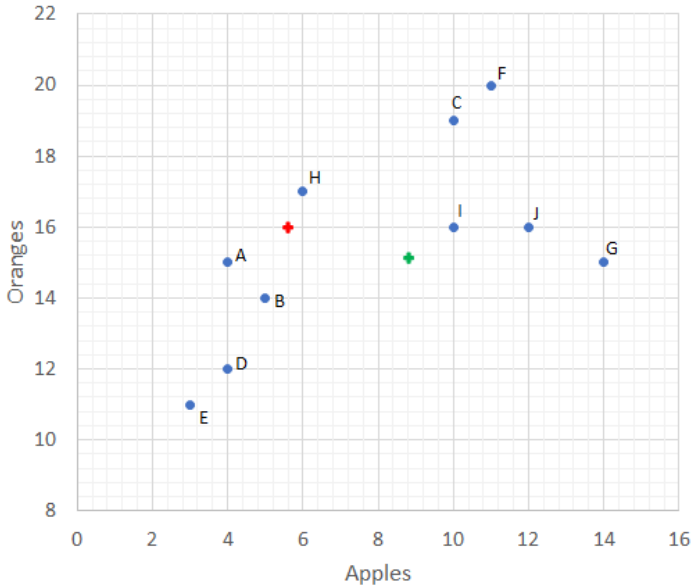
K=2

Cluster centers

C1 =(5,16) C2 =(8.6,15.4)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(1^2+1^2)^0.5	(4.6^2+0.4^2)^0.5	C1
B	5	14	(0+2^2)^0.5	(3.6^2+1.4^2)^0.5	C1
C	10	19	(5^2+3^2)^0.5 =34^0.5	(1.4^2+3.6^2)^0.5 =14.92^05	C2
D	4	12	(1^2+4^2)^0.5	(4.6^2+3.4^2)^0.5	C1
E	3	11	(2^2+5^2)^0.5	(5.6^2+4.4^2)^0.5	C1
F	11	20	(6^2+4^2)^0.5	(2.4^2+4.6^2)^0.5	C2
G	14	15	(9^2+1^2)^0.5	(5.4^2+0.4^2)^0.5	C2
H	6	17	(1^2+1^2)^0.5	(2.6^2+1.6^2)^0.5	C1
I	10	16	(5^2+0)^0.5	(1.4^2+0.6^2)^0.5	C2
J	12	16	(8^2+0)^0.5	(3.4^2+0.6^2)^0.5	C2



Update Centers

C1 =

C2 =

Iteration 2

Example: Update cluster centers

K=2

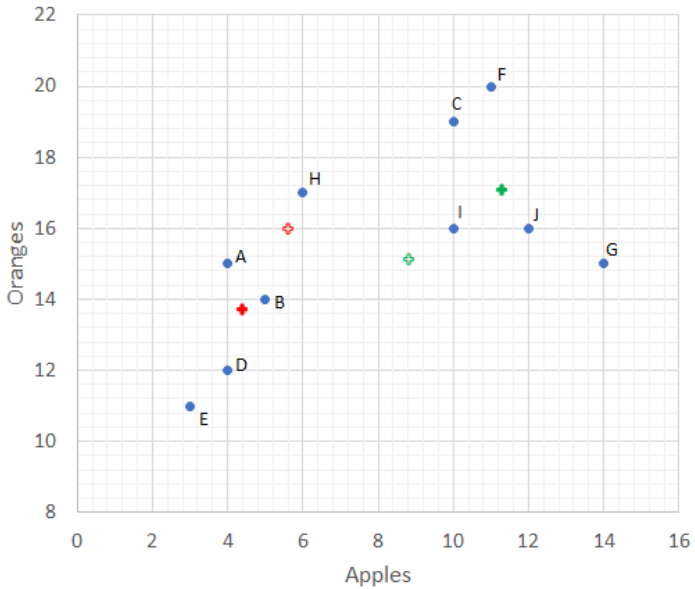
Cluster centers

C1 =(5,16) C2 =(8.6,15.4)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(1^2+1^2)^0.5	(4.6^2+0.4^2)^0.5	C1
B	5	14	(0+2^2)^0.5	(3.6^2+1.4^2)^0.5	C1
C	10	19	(5^2+3^2)^0.5 =34^0.5	(1.4^2+3.6^2)^0.5 =14.92^05	C2
D	4	12	(1^2+4^2)^0.5	(4.6^2+3.4^2)^0.5	C1
E	3	11	(2^2+5^2)^0.5	(5.6^2+4.4^2)^0.5	C1
F	11	20	(6^2+4^2)^0.5	(2.4^2+4.6^2)^0.5	C2
G	14	15	(9^2+1^2)^0.5	(5.4^2+0.4^2)^0.5	C2
H	6	17	(1^2+1^2)^0.5	(2.6^2+1.6^2)^0.5	C1
I	10	16	(5^2+0)^0.5	(1.4^2+0.6^2)^0.5	C2
J	12	16	(8^2+0)^0.5	(3.4^2+0.6^2)^0.5	C2

Centers change.
Continue Iteration 3



Update Centers

C1 = ((4+5+4+3+6)/5,(15+14+12+11+17)/5) = (4.4,13.8)

C2 = ((10+11+14+10+12)/5, (19+20+15+16+16)/5) = (11.4,17.2)

Iteration 2

Example: Find distance to centers and assign each data point to closet center

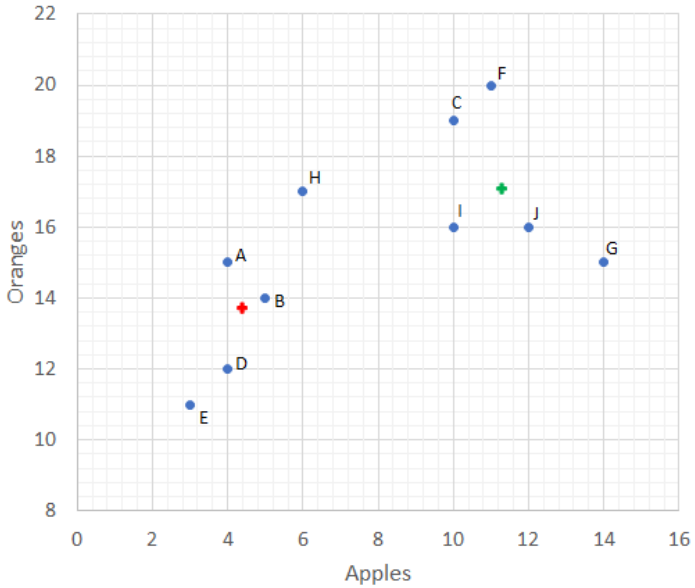
K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15			
B	5	14			
C	10	19			
D	4	12			
E	3	11			
F	11	20			
G	14	15			
H	6	17			
I	10	16			
J	12	16			



Update Centers

C1 =

C2 =

Iteration 3

Example: Find distance to centers and assign each data point to closet center (cont.)

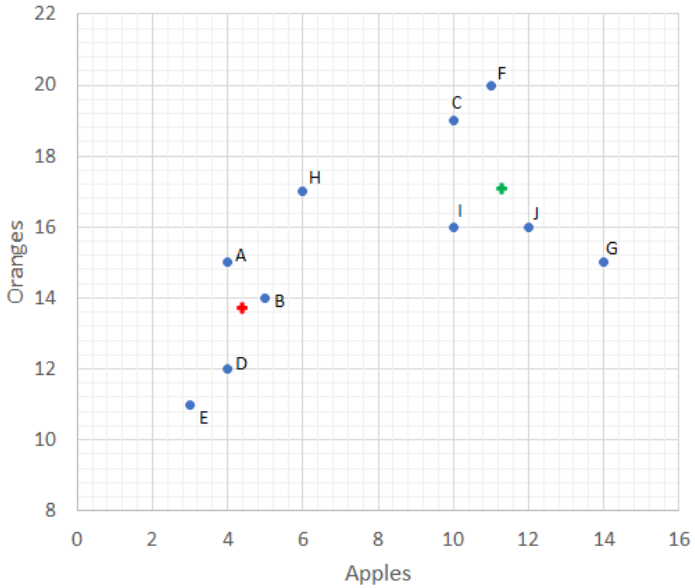
K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Euclidean distance: $dist_E(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{m} (p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0.4^2+1.2^2)^0.5		
B	5	14	(0.6^2+0.2^2)^0.5		
C	10	19	(3.6^2+5.2^2)^0.5		
D	4	12	(0.4^2+0.2^2)^0.5		
E	3	11	(1.4^2+2.8^2)^0.5		
F	11	20	(5.6^2+6.2^2)^0.5		
G	14	15	(9.6^2+1.2^2)^0.5		
H	6	17	(1.6^2+3.2^2)^0.5		
I	10	16	(5.6^2+2.2^2)^0.5		
J	12	16	(7.6^2+2.2^2)^0.5		



Update Centers

C1 =

C2 =

Iteration 3

Example: Find distance to centers and assign each data point to closet center (cont.)

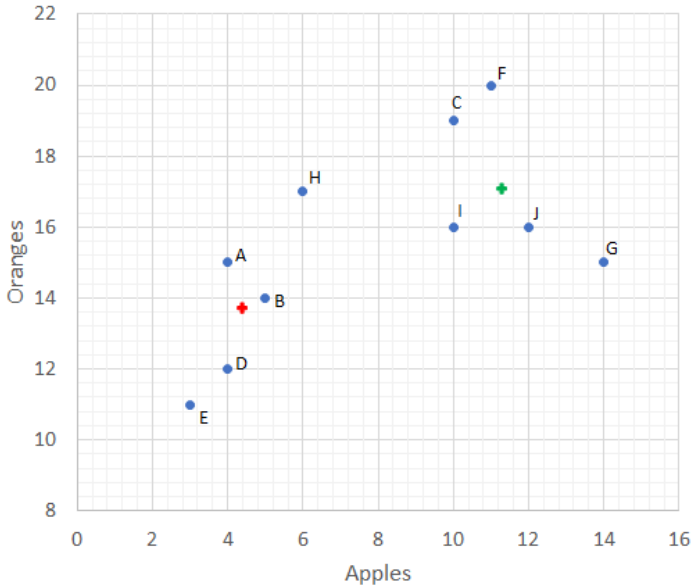
K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Euclidean distance: $dist_E(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{m} (p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0.4^2+1.2^2)^0.5	(7.4^2+2.2^2)^0.5	
B	5	14	(0.6^2+0.2^2)^0.5	(6.4^2+3.2^2)^0.5	
C	10	19	(3.6^2+5.2^2)^0.5	(1.4^2+1.8^2)^0.5	
D	4	12	(0.4^2+0.2^2)^0.5	(7.4^2+5.2^2)^0.5	
E	3	11	(1.4^2+2.8^2)^0.5	(8.4^2+6.2^2)^0.5	
F	11	20	(5.6^2+6.2^2)^0.5	(0.4^2+2.8^2)^0.5	
G	14	15	(9.6^2+1.2^2)^0.5	(2.6^2+2.2^2)^0.5	
H	6	17	(1.6^2+3.2^2)^0.5	(5.4^2+0.2^2)^0.5	
I	10	16	(5.6^2+2.2^2)^0.5	(1.4^2+1.2^2)^0.5	
J	12	16	(7.6^2+2.2^2)^0.5	(0.6^2+1.2^2)^0.5	



Update Centers

C1 =

C2 =

Iteration 3

Example: Find distance to centers and assign each data point to closet center (cont.)

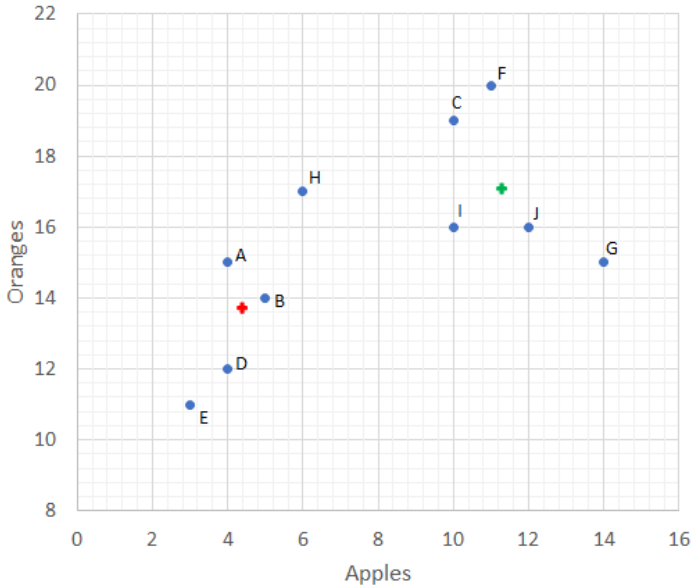
K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Euclidean distance: $dist_E(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{m}(p_i - q_i)^2}$

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0.4^2+1.2^2)^0.5	(7.4^2+2.2^2)^0.5	C1
B	5	14	(0.6^2+0.2^2)^0.5	(6.4^2+3.2^2)^0.5	C1
C	10	19	(3.6^2+5.2^2)^0.5	(1.4^2+1.8^2)^0.5	C2
D	4	12	(0.4^2+0.2^2)^0.5	(7.4^2+5.2^2)^0.5	C1
E	3	11	(1.4^2+2.8^2)^0.5	(8.4^2+6.2^2)^0.5	C1
F	11	20	(5.6^2+6.2^2)^0.5	(0.4^2+2.8^2)^0.5	C2
G	14	15	(9.6^2+1.2^2)^0.5	(2.6^2+2.2^2)^0.5	C2
H	6	17	(1.6^2+3.2^2)^0.5	(5.4^2+0.2^2)^0.5	C1
I	10	16	(5.6^2+2.2^2)^0.5	(1.4^2+1.2^2)^0.5	C2
J	12	16	(7.6^2+2.2^2)^0.5	(0.6^2+1.2^2)^0.5	C2



Update Centers

C1 =

C2 =

Iteration 3

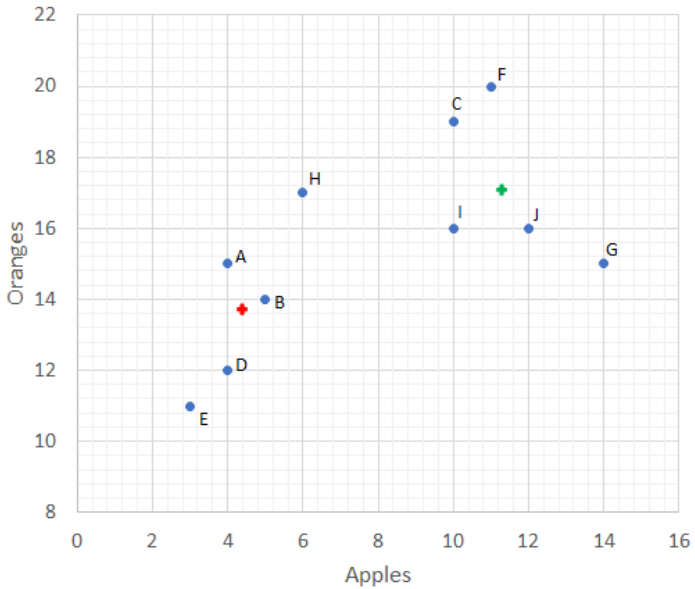
Example: Update cluster centers

K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0.4^2+1.2^2)^0.5	(7.4^2+2.2^2)^0.5	C1
B	5	14	(0.6^2+0.2^2)^0.5	(6.4^2+3.2^2)^0.5	C1
C	10	19	(3.6^2+5.2^2)^0.5	(1.4^2+1.8^2)^0.5	C2
D	4	12	(0.4^2+0.2^2)^0.5	(7.4^2+5.2^2)^0.5	C1
E	3	11	(1.4^2+2.8^2)^0.5	(8.4^2+6.2^2)^0.5	C1
F	11	20	(5.6^2+6.2^2)^0.5	(0.4^2+2.8^2)^0.5	C2
G	14	15	(9.6^2+1.2^2)^0.5	(2.6^2+2.2^2)^0.5	C2
H	6	17	(1.6^2+3.2^2)^0.5	(5.4^2+0.2^2)^0.5	C1
I	10	16	(5.6^2+2.2^2)^0.5	(1.4^2+1.2^2)^0.5	C2
J	12	16	(7.6^2+2.2^2)^0.5	(0.6^2+1.2^2)^0.5	C2



Update Centers

C1 = ((4+5+4+3+6)/5,(15+14+12+11+17)/5) = (4.4,13.8)

C2 = ((10+11+14+10+12)/5, (19+20+15+16+16)/5) = (11.4,17.2)

Iteration 3

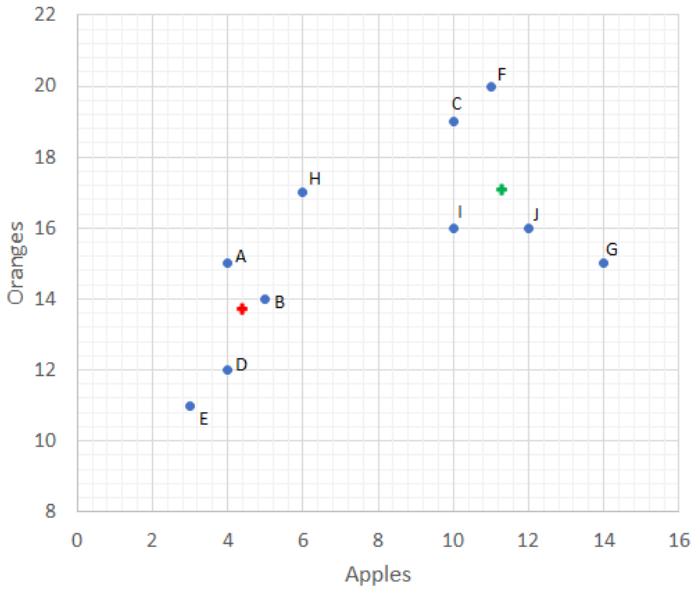
Example: Centers do not change. STOP

K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0.4^2+1.2^2)^0.5	(7.4^2+2.2^2)^0.5	C1
B	5	14	(0.6^2+0.2^2)^0.5	(6.4^2+3.2^2)^0.5	C1
C	10	19	(3.6^2+5.2^2)^0.5	(1.4^2+1.8^2)^0.5	C2
D	4	12	(0.4^2+0.2^2)^0.5	(7.4^2+5.2^2)^0.5	C1
E	3	11	(1.4^2+2.8^2)^0.5	(8.4^2+6.2^2)^0.5	C1
F	11	20	(5.6^2+6.2^2)^0.5	(0.4^2+2.8^2)^0.5	C2
G	14	15	(9.6^2+1.2^2)^0.5	(2.6^2+2.2^2)^0.5	C2
H	6	17	(1.6^2+3.2^2)^0.5	(5.4^2+0.2^2)^0.5	C1
I	10	16	(5.6^2+2.2^2)^0.5	(1.4^2+1.2^2)^0.5	C2
J	12	16	(7.6^2+2.2^2)^0.5	(0.6^2+1.2^2)^0.5	C2



Centers do not change.

Result:

Cluster1_members = {A,B,D,E,H}

Cluster2_members = {C,F,G,I,J}

Cluster1_center = (5,16)

Cluster2_center = (11.4,17.2)

Update Centers

C1 = ((4+5+4+3+6)/5,(15+14+12+11+17)/5) = (4.4,13.8)

C2 = ((10+11+14+10+12)/5, (19+20+15+16+16)/5) = (11.4,17.2)

Iteration 3

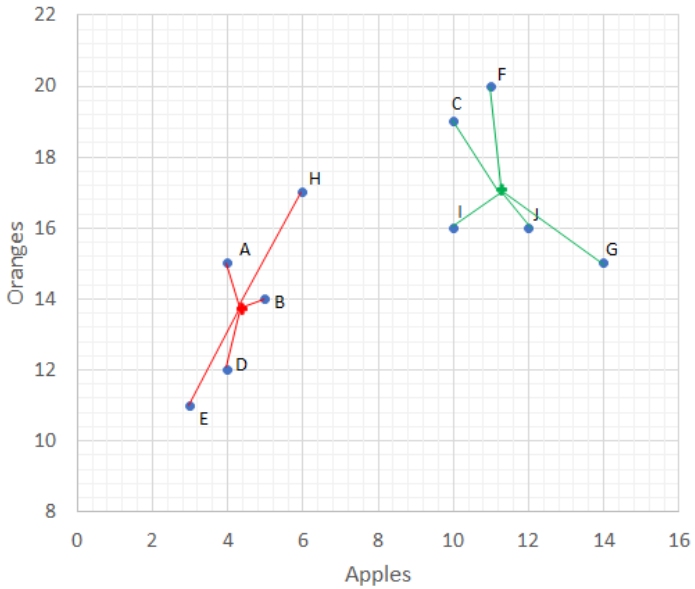
Example: Clustering Result

K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0.4^2+1.2^2)^0.5	(7.4^2+2.2^2)^0.5	C1
B	5	14	(0.6^2+0.2^2)^0.5	(6.4^2+3.2^2)^0.5	C1
C	10	19	(3.6^2+5.2^2)^0.5	(1.4^2+1.8^2)^0.5	C2
D	4	12	(0.4^2+0.2^2)^0.5	(7.4^2+5.2^2)^0.5	C1
E	3	11	(1.4^2+2.8^2)^0.5	(8.4^2+6.2^2)^0.5	C1
F	11	20	(5.6^2+6.2^2)^0.5	(0.4^2+2.8^2)^0.5	C2
G	14	15	(9.6^2+1.2^2)^0.5	(2.6^2+2.2^2)^0.5	C2
H	6	17	(1.6^2+3.2^2)^0.5	(5.4^2+0.2^2)^0.5	C1
I	10	16	(5.6^2+2.2^2)^0.5	(1.4^2+1.2^2)^0.5	C2
J	12	16	(7.6^2+2.2^2)^0.5	(0.6^2+1.2^2)^0.5	C2



Centers do not change.

Result:

Cluster1_members = {A,B,D,E,H}

Cluster2_members = {C,F,G,I,J}

Cluster1_center = (5,16)

Cluster2_center = (11.4,17.2)

Iteration 3

Update Centers

C1 = ((4+5+4+3+6)/5,(15+14+12+11+17)/5) = (4.4,13.8)

C2 = ((10+11+14+10+12)/5, (19+20+15+16+16)/5) = (11.4,17.2)

Perpendicular bisector

K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0.4^2+1.2^2)^0.5	(7.4^2+2.2^2)^0.5	C1
B	5	14	(0.6^2+0.2^2)^0.5	(6.4^2+3.2^2)^0.5	C1
C	10	19	(3.6^2+5.2^2)^0.5	(1.4^2+1.8^2)^0.5	C2
D	4	12	(0.4^2+0.2^2)^0.5	(7.4^2+5.2^2)^0.5	C1
E	3	11	(1.4^2+2.8^2)^0.5	(8.4^2+6.2^2)^0.5	C1
F	11	20	(5.6^2+6.2^2)^0.5	(0.4^2+2.8^2)^0.5	C2
G	14	15	(9.6^2+1.2^2)^0.5	(2.6^2+2.2^2)^0.5	C2
H	6	17	(1.6^2+3.2^2)^0.5	(5.4^2+0.2^2)^0.5	C1
I	10	16	(5.6^2+2.2^2)^0.5	(1.4^2+1.2^2)^0.5	C2
J	12	16	(7.6^2+2.2^2)^0.5	(0.6^2+1.2^2)^0.5	C2

Centers do not change.

Result:

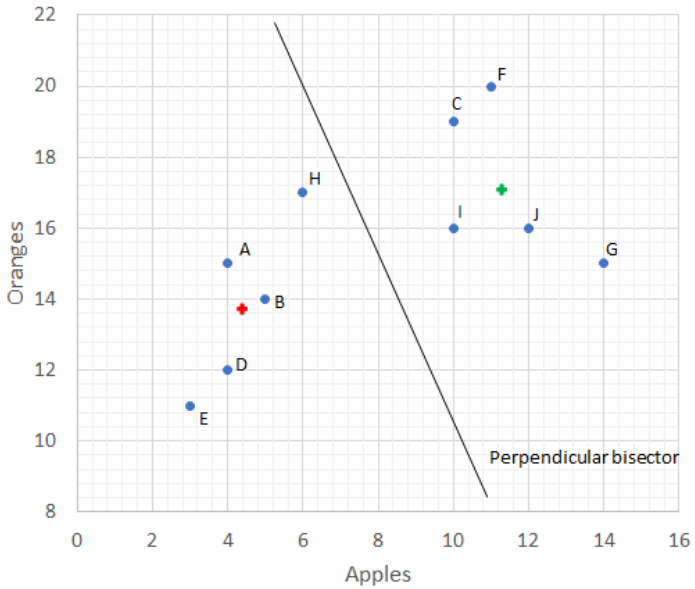
Cluster1_members = {A,B,D,E,H}

Cluster2_members = {C,F,G,I,J}

Cluster1_center = (5,16)

Cluster2_center = (11.4,17.2)

Iteration 3



Boundary of clusters

K=2

Cluster centers

C1 = (4.4,13.8) C2 = (11.4,17.2)

Store (p)	Apples	Oranges	distE(p,c1)	distE(p,c2)	Cluster_ID
A	4	15	(0.4^2+1.2^2)^0.5	(7.4^2+2.2^2)^0.5	C1
B	5	14	(0.6^2+0.2^2)^0.5	(6.4^2+3.2^2)^0.5	C1
C	10	19	(3.6^2+5.2^2)^0.5	(1.4^2+1.8^2)^0.5	C2
D	4	12	(0.4^2+0.2^2)^0.5	(7.4^2+5.2^2)^0.5	C1
E	3	11	(1.4^2+2.8^2)^0.5	(8.4^2+6.2^2)^0.5	C1
F	11	20	(5.6^2+6.2^2)^0.5	(0.4^2+2.8^2)^0.5	C2
G	14	15	(9.6^2+1.2^2)^0.5	(2.6^2+2.2^2)^0.5	C2
H	6	17	(1.6^2+3.2^2)^0.5	(5.4^2+0.2^2)^0.5	C1
I	10	16	(5.6^2+2.2^2)^0.5	(1.4^2+1.2^2)^0.5	C2
J	12	16	(7.6^2+2.2^2)^0.5	(0.6^2+1.2^2)^0.5	C2

Centers do not change.

Result:

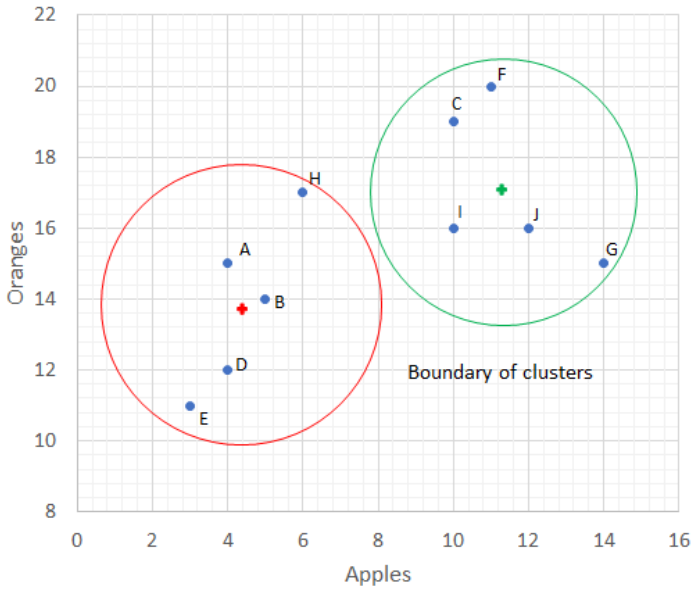
Cluster1_members = {A,B,D,E,H}

Cluster2_members = {C,F,G,I,J}

Cluster1_center = (5,16)

Cluster2_center = (11.4,17.2)

Iteration 3



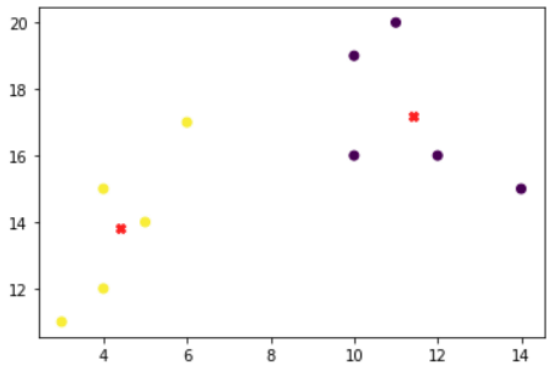
Boundary of clusters can be applied for

- Outlier Identifying
- Anomaly Detection
- Etc.

K-Means with Python

```python
import numpy as np
import matplotlib.pyplot as plt
from sklearn import cluster
import pandas as pd

x = [4, 5, 10, 4, 3, 11, 14 , 6, 10, 12]
y = [15, 14, 19, 12, 11, 20, 15, 17, 16, 16]
store = ['A','B','C','D','E','F','G','H','I','J']

df = pd.DataFrame()
df['Store'] = store
df['Apples'] = x
df['Oranges'] = y
data = df[['Apples','Oranges']]

model_kmeans = cluster.KMeans(n_clusters=2, max_iter=50, random_state=1)
model_kmeans.fit(data)

df['cluster_id'] = model_kmeans.labels_

centroids = model_kmeans.cluster_centers_
print(centroids)

plt.scatter(df['Apples'],df['Oranges'], c=df['cluster_id'])
plt.scatter(centroids[:,0],centroids[:,1],marker='X', c='r')
plt.show()
```

```
[[11.4 17.2]
 [ 4.4 13.8]]
```



More Tutorial: http://www.cse.msu.edu/ ptan/dmbook/tutorials/tutorial8/tutorial8.html
https://matplotlib.org/stable/plot_types/index.html

Outline

Example Application: Image segmentation

```
from PIL import Image
import matplotlib.pyplot as plt
from numpy import asarray
from sklearn import cluster
import numpy as np

im = Image.open("img-bird.jpg")
#display(im)
```

```
data = asarray(im)
imshape = data.shape
plt.imshow(data)
```

```
fig = plt.figure()
ax = fig.add_subplot(projection='3d')
ax.scatter(X[:,0],X[:,1],X[:,2], c = X/255.0, marker='.')
plt.show()
```

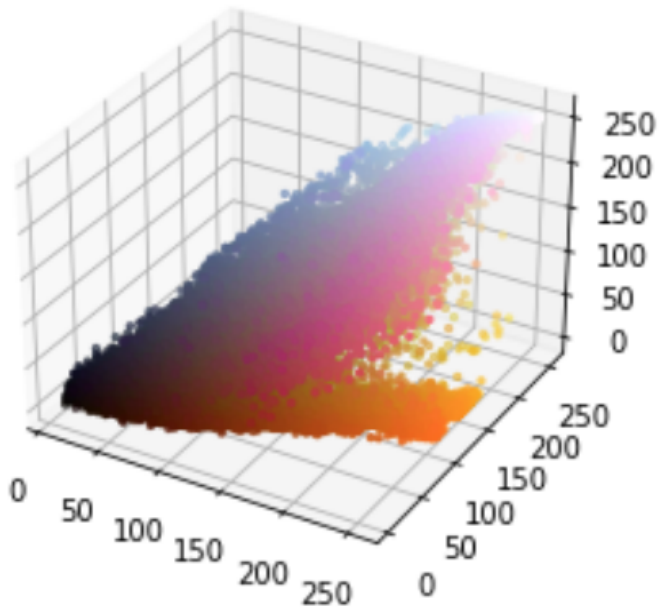
```
#K-Means
k = 5
model_kmeans = cluster.KMeans(n_clusters=k, max_iter=50, random_state=1)
model_kmeans.fit(X)
cluster_id = model_kmeans.labels_
centroids = model_kmeans.cluster_centers_
print(centroids)
```

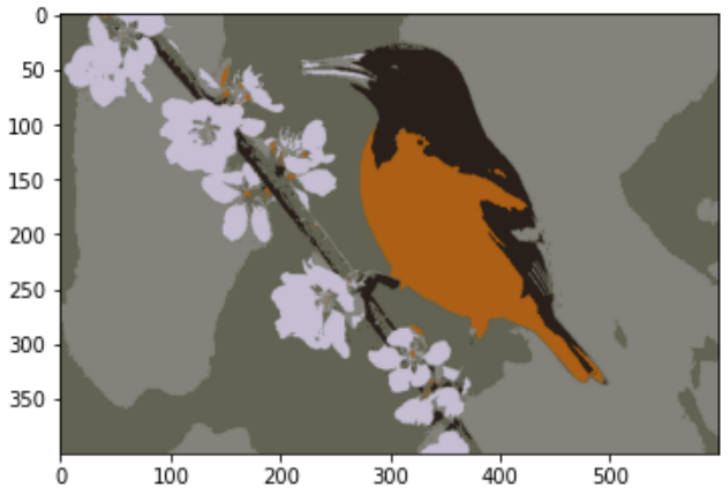
```
Xnew = X.copy()
for i in range(k):
    Xnew[cluster_id==i,:] = centroids[i,:]
```

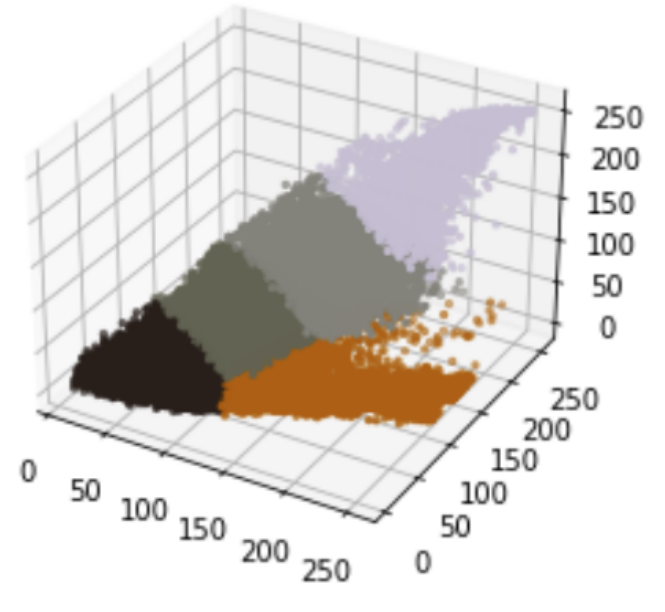
```
fig = plt.figure()
ax = fig.add_subplot(projection='3d')
ax.scatter(X[:,0],X[:,1],X[:,2], c = Xnew/255.0, marker='.')
plt.show()
```

```
Xnew = Xnew.reshape(imshape)
plt.imshow(Xnew)
```


0
50
100
150
200
250
300
350
0
100
200
300
400
500


250
200
150
100
50
0
0
50
100
150
200
250


0
50
100
150
200
250
300
350
0
100
200
300
400
500


250
200
150
100
50
0
0
50
100
150
200
250

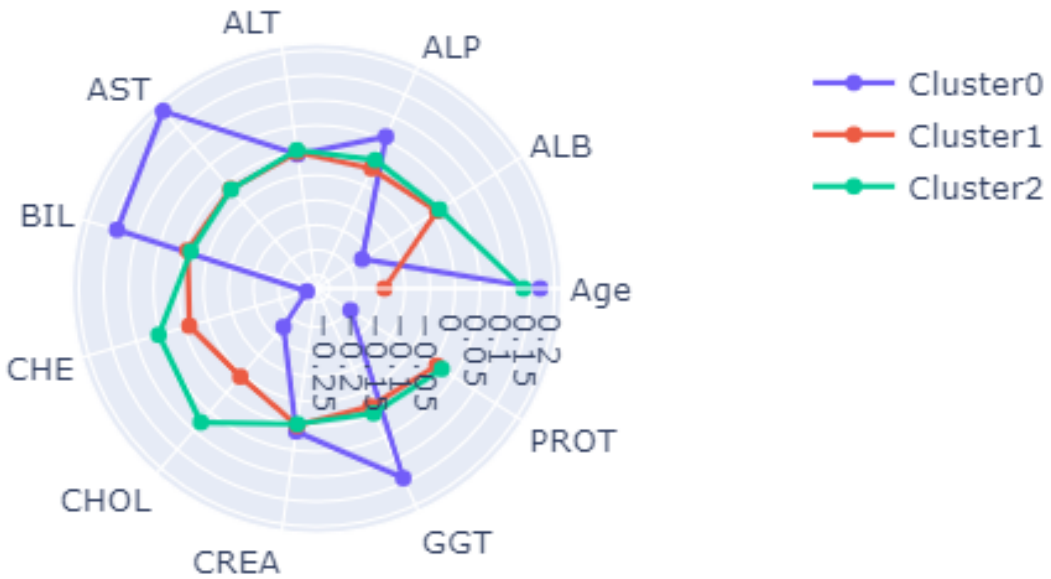
Example: Data Analytics

```
import pandas as pd
from sklearn import cluster
from sklearn import preprocessing

df_org = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/00571/hcvdat0.csv')

df = df_org[['Age', 'ALB', 'ALP', 'ALT', 'AST',
       'BIL', 'CHE', 'CHOL', 'CREA', 'GGT', 'PROT']]

df = df.dropna()
df = (df - df.mean()) / (df.max() - df.min())
```

```
k=3
model_kmeans = cluster.KMeans(n_clusters=k, max_iter=50, random_state=1)
model_kmeans.fit(df)
```

https://archive.ics.uci.edu/ml/datasets/HCV+data
https://plotly.com/python/radar-chart/
https://plotly.com/python/subplots/

```
import plotly.graph_objects as go

features = ['Age', 'ALB', 'ALP', 'ALT', 'AST', 'BIL', 'CHE', 'CHOL', 'CREA', 'GGT', 'PROT']

fig = go.Figure()
for i in range(k):
    fig.add_trace(go.Scatterpolar(r= model_kmeans.cluster_centers_[i,:],
          theta=features, name='Cluster'+str(i) ))

fig.update_layout(polar=dict(radialaxis=dict(visible=True)), showlegend=True)
fig.show()
```



```
#https://plotly.com/python/subplots/
from plotly.subplots import make_subplots
import plotly.graph_objects as go

features = ['Age', 'ALB', 'ALP', 'ALT', 'AST', 'BIL', 'CHE', 'CHOL', 'CREA',
       'GGT', 'PROT']

fig = make_subplots( rows=1, cols=3, subplot_titles=("Cluster 0", "Cluster 1", "Cluster 2") )

fig.add_trace(go.Bar(x=features, y=model_kmeans.cluster_centers_[0,:], ), 1,1)
fig.add_trace(go.Bar(x=features, y=model_kmeans.cluster_centers_[1,:], ), 1,2)
fig.add_trace(go.Bar(x=features, y=model_kmeans.cluster_centers_[2,:], ), 1,3)

fig.update_layout(coloraxis=dict(colorscale='Bluered_r'), showlegend=False)
fig.show()
```