Unsupervised Learning-K-means

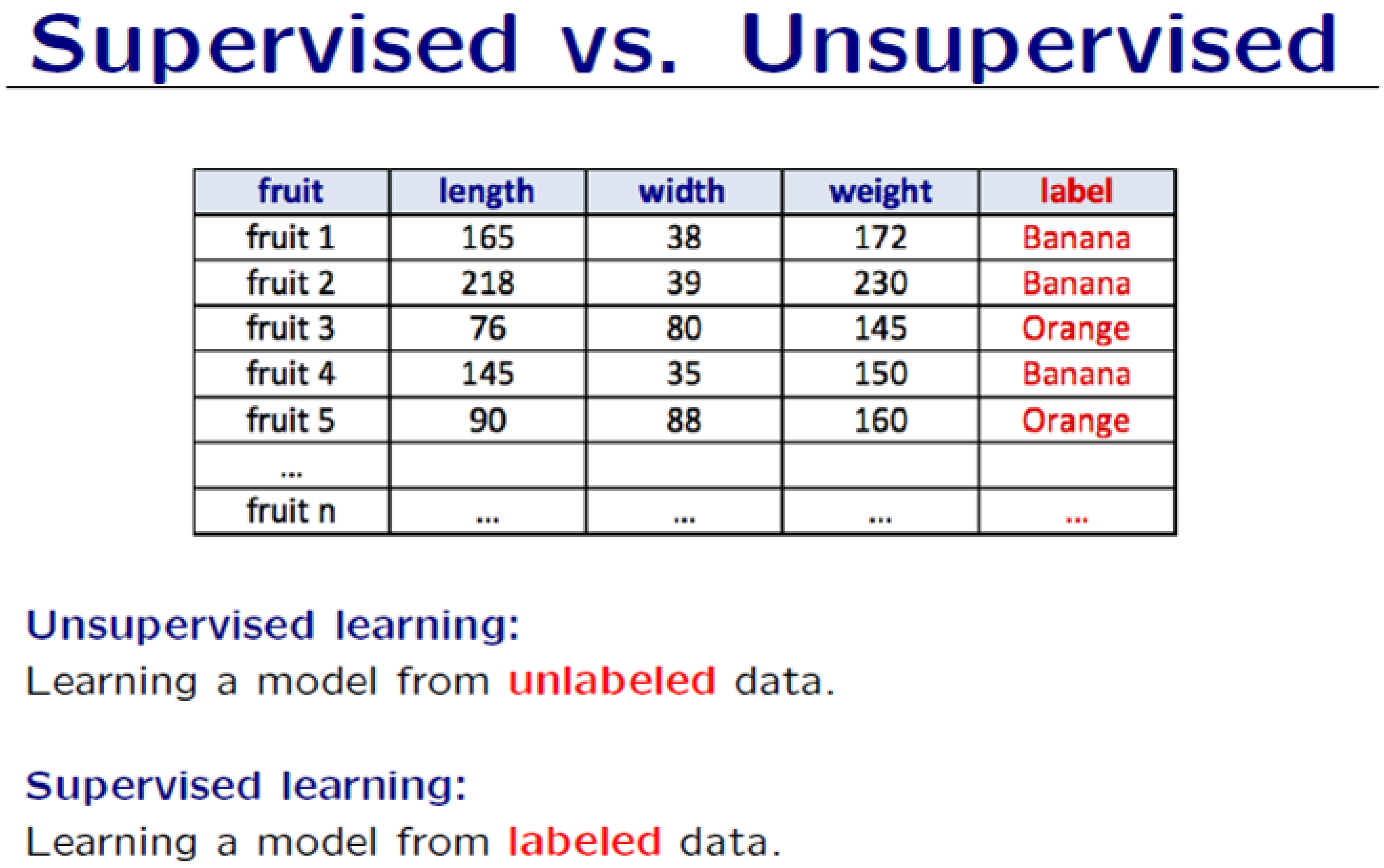
By Dr Shaik Qadeer

Professor MJCET

Learning Objectives

- Introduction to unsupervised learning
- Introduction to k-mean clustering
- Using distance measures such as Euclidean distance in clustering
- Learn to build clusters using *sklearn* library in Python.
- Finding an optimal solution for building clusters

Introduction unsupervised Learning ML



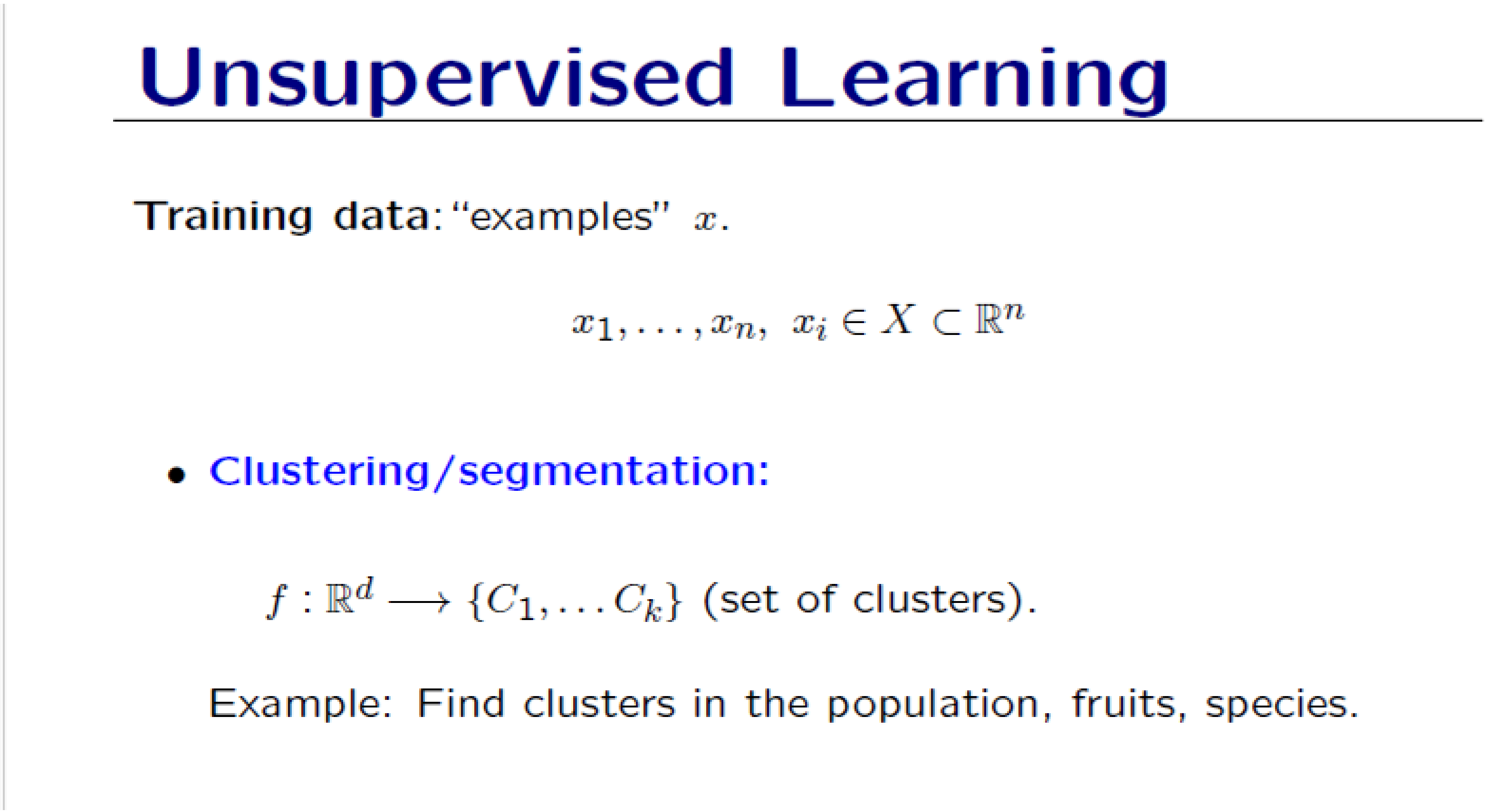
Supervised vs. Unsupervised

fruit	length	width	weight	label
fruit 1	165	38	172	Banana
fruit 2	218	39	230	Banana
fruit 3	76	80	145	Orange
fruit 4	145	35	150	Banana
fruit 5	90	88	160	Orange
...				
fruit n	...	...	...	...

Unsupervised learning:
Learning a model from unlabeled data.

Supervised learning:
Learning a model from labeled data.

Introduction unsupervised Learning ML



Unsupervised Learning

Training data: "examples" x.

$$x_1, \ldots, x_n,\ x_i \in X \subset \mathbb{R}^n$$

- Clustering/segmentation:

$f : \mathbb{R}^d \longrightarrow \{C_1, \ldots C_k\}$ (set of clusters).

Example: Find clusters in the population, fruits, species.

Introduction to Clustering

- Clustering is a divide-and-conquer strategy which divides the dataset into homogenous groups which can be further used to prescribe the right strategy for different groups.
- In clustering, **the objective** is to ensure that the **variation within a cluster is minimized** while the **variation between clusters is maximized**

How Clustering works?

- It works by partitioning data into k clusters, based on feature similarity,
- Clustering algorithms use different distance or similarity or

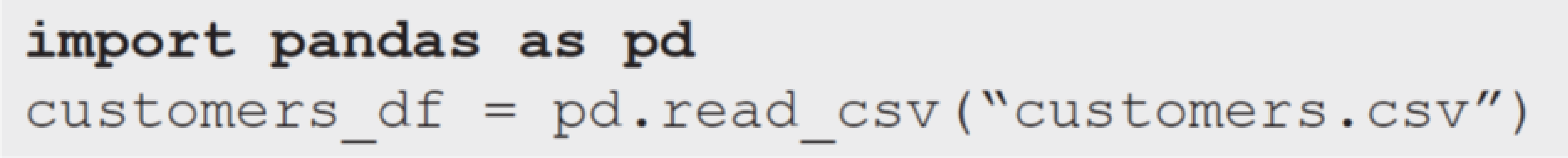
dissimilarity measures to derive different clusters. The type of distance/similarity measure used plays a crucial role in the final cluster formation. Larger distance would imply that observations are far away from one another, whereas higher similarity would indicate that the observations are similar.

Case study: Do clustering operation on customer data

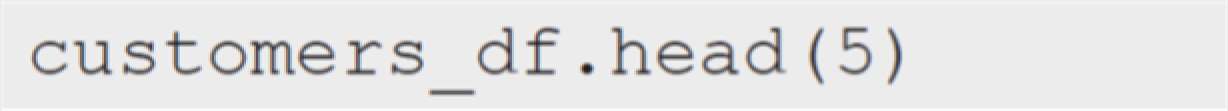
- Established a relationship between age and salary with k mean clustering

K-mean clustering(Case study)..

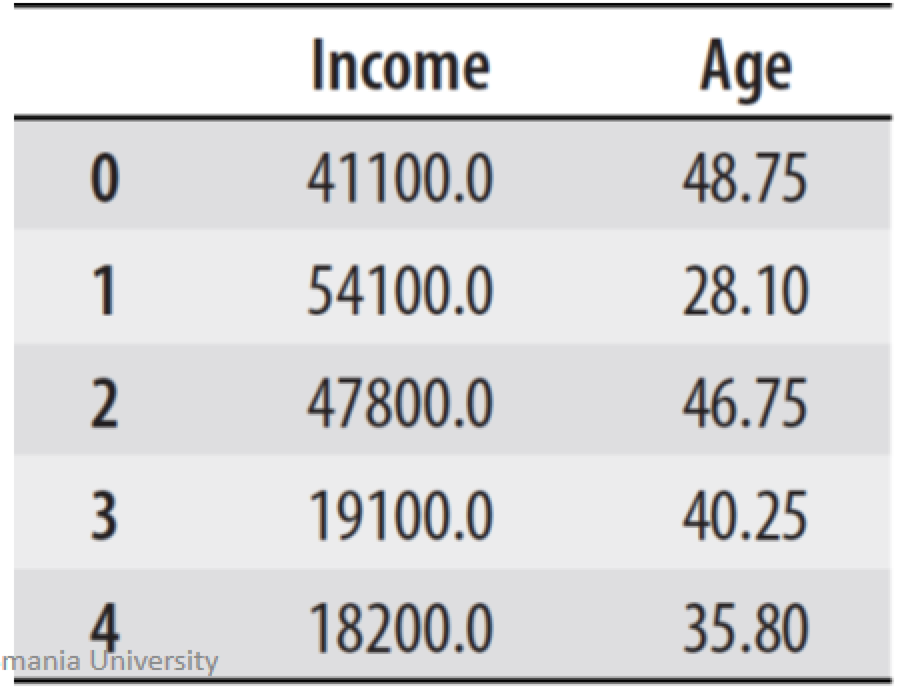
- Loading data



```
import pandas as pd
customers_df = pd.read_csv(“customers.csv”)
```



```
customers_df.head(5)
```



	Income	Age
0	41100.0	48.75
1	54100.0	28.10
2	47800.0	46.75
3	19100.0	40.25
4	18200.0	35.80

Identification of the problem:

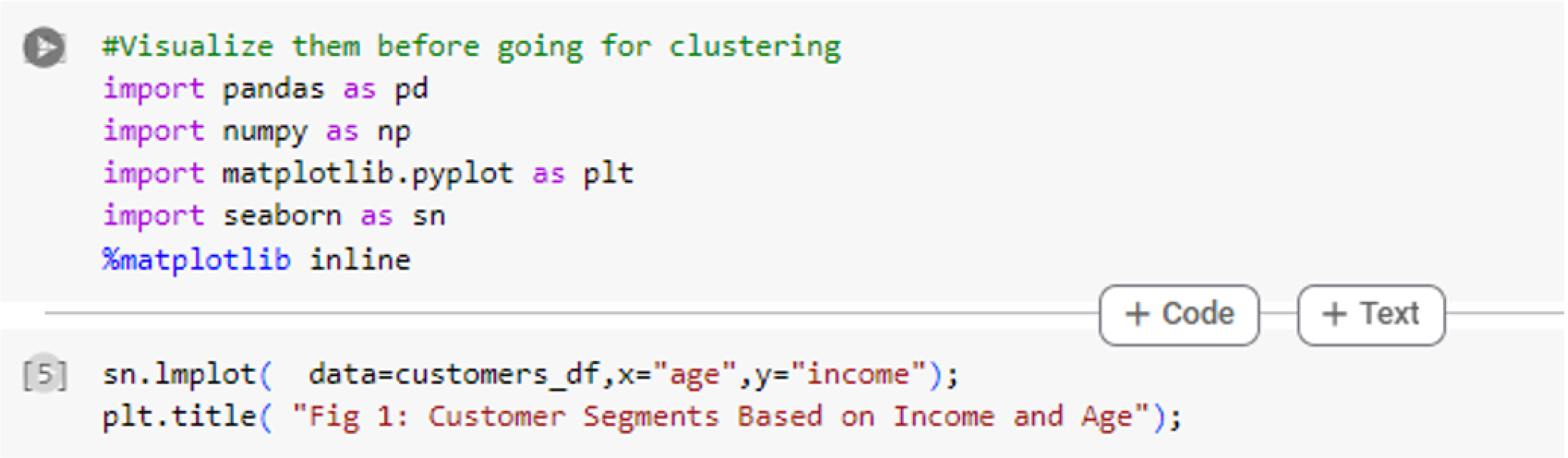
- Group the people as per their age
- Group the people as per their income

Identification of the problem:..

- Consider grouping as per their income:
  - Low income with low age
  - Medium income with medium age
  - High income with high age etc..

K-mean clustering(Case study)..

- Visualizing the relationship



```
#Visualize them before going for clustering
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sn
%matplotlib inline
```

```
sn.lmplot(  data=customers_df,x="age",y="income");
plt.title( "Fig 1: Customer Segments Based on Income and Age");
```

K-mean clustering(Case study)..

- Scatterplot

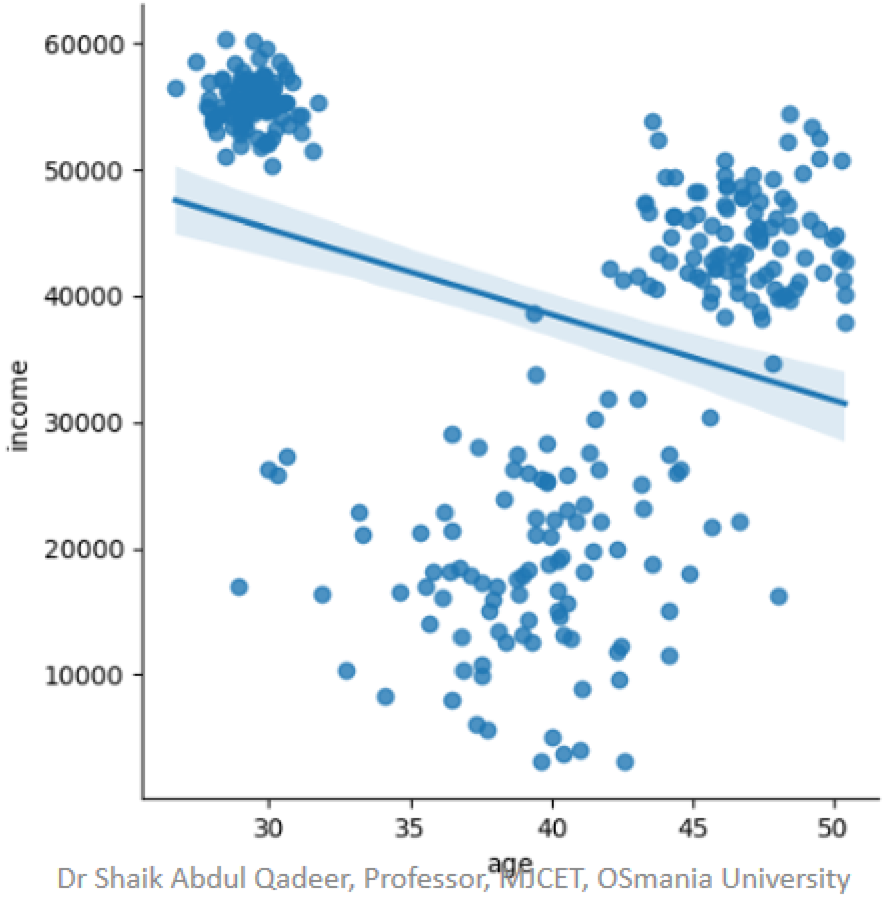


Fig 1: Customer Segments Based on Income and Age

Finding similarities using distance

- Clustering techniques assume that there are subsets in the data that are similar or homogeneous. One approach for measuring similarity is through distances measured using different metrics. Few distance measures used in clustering are discussed in the following sections.

Euclidean distance

- Euclidean distance between two observations X_1 and X_2 with n features can be calculated as

$$D(X_1, X_2) = \sqrt{\Sigma(X_{i1} - X_{i2})^2}$$

- Where X_{i1} and X_{i2} are the values of the i^{th} feature for first observation

and second observation, respectively.

Other method of distance measurement

- Minkowski Distance
- Jaccard Similarity Coefficient
- Cosine Similarity
- Gower’s Similarity Coefficient

Procedure of k-mean clustering

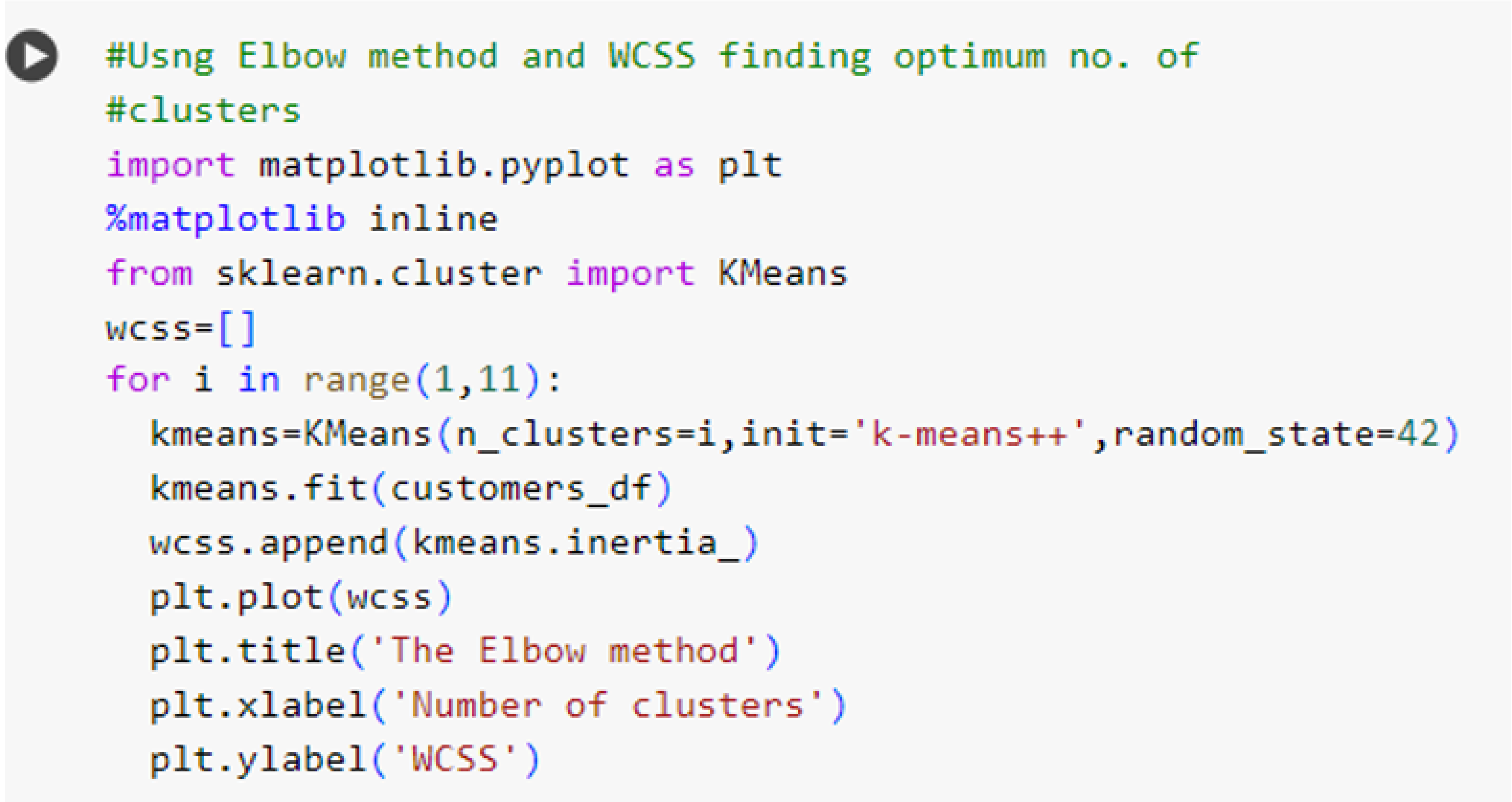
- The following steps are used in *K*-means clustering algorithm:

  1. Decide the value of *K*.
  2. Choose *K* observations from the data that are likely to be in different clusters. Choose observations that are farthest.
  3. The *K* observations selected in step 2 are the centroids of those clusters.
  4. For remaining observations, find the cluster closest to the centroid. Add the new observation (say observation *j*) to the cluster with the closest centroid. Adjust the centroid after adding a new observation to the cluster. The closest centroid is chosen based upon an appropriate distance measure.
  5. Repeat step 4 until all observations are assigned to a cluster.

Method of finding exact number of clusters

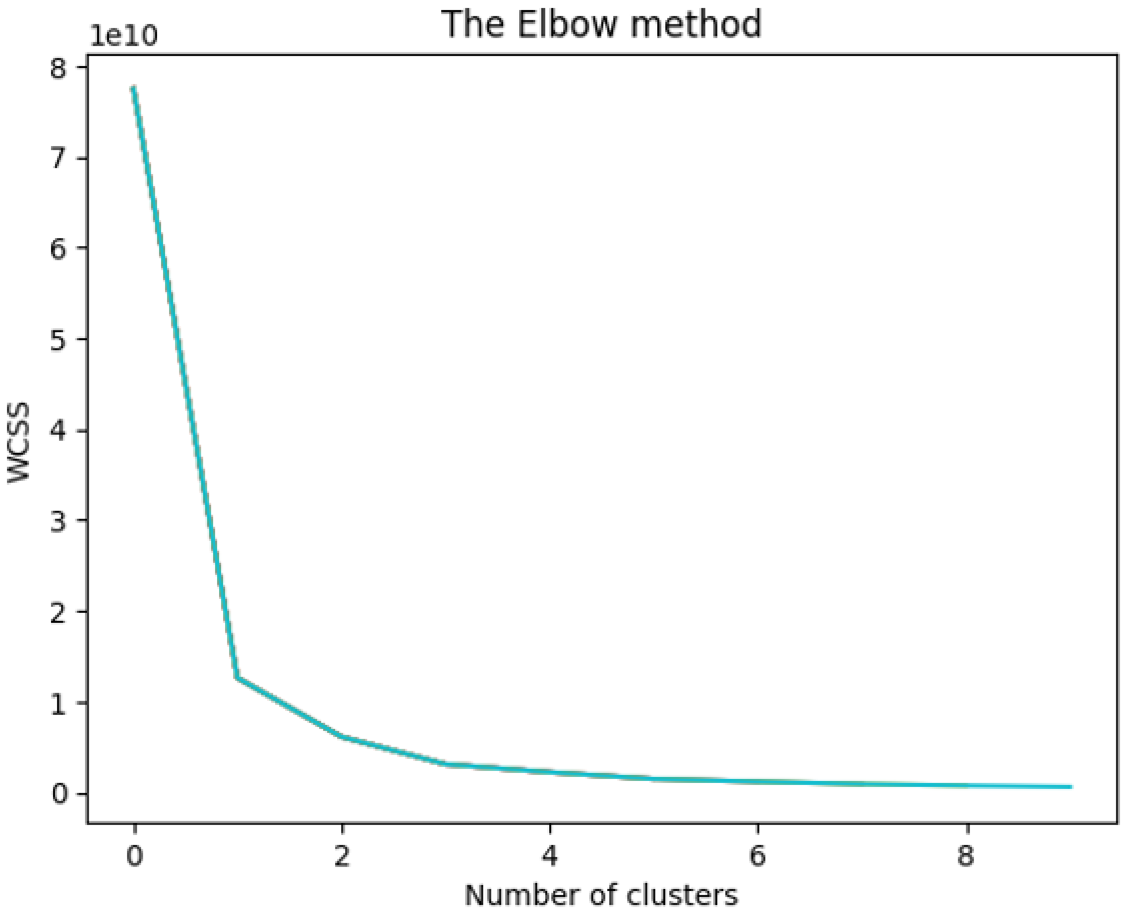
- Although the number of clusters is often arbitrary
- But there is a procedure to find the optimal number
- Ex: Elbow method and WCSS(within cluster sums of square)

Method of finding exact number of clusters…

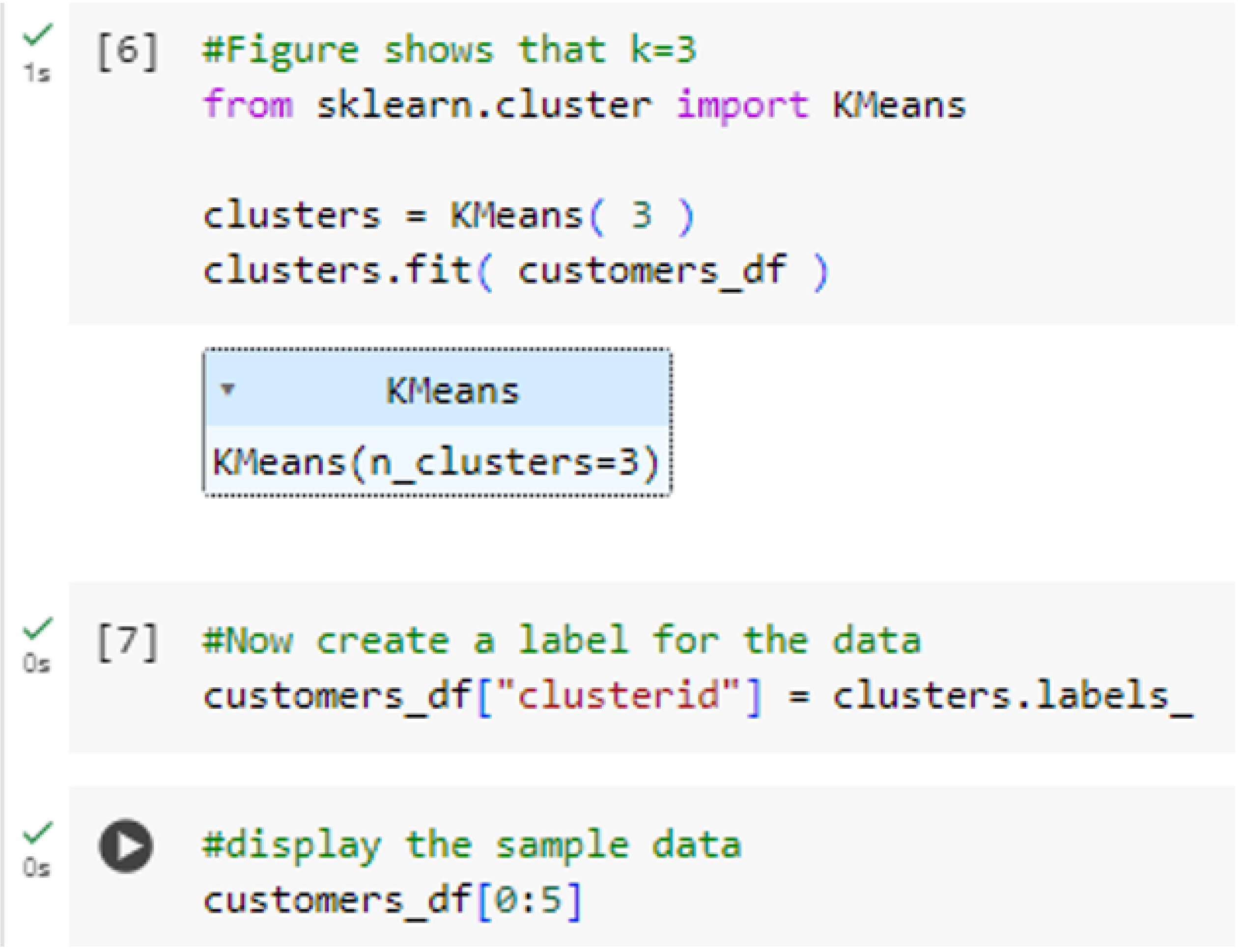


```
#Usng Elbow method and WCSS finding optimum no. of
#clusters
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn.cluster import KMeans
wcss=[]
for i in range(1,11):
  kmeans=KMeans(n_clusters=i,init='k-means++',random_state=42)
  kmeans.fit(customers_df)
  wcss.append(kmeans.inertia_)
  plt.plot(wcss)
  plt.title('The Elbow method')
  plt.xlabel('Number of clusters')
  plt.ylabel('WCSS')
```

Method of finding exact number of clusters…



Creating empty cluster: From above test k can be selected as 3, and add labels to it



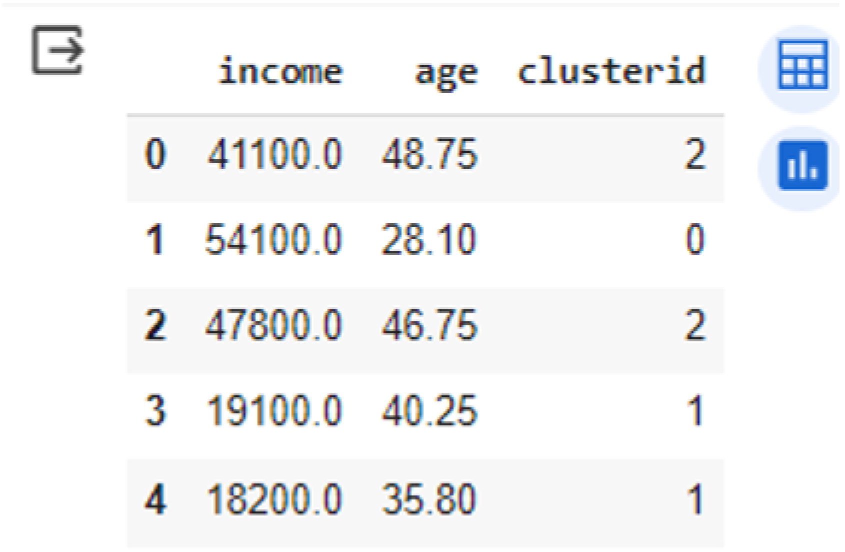
```
[6]  #Figure shows that k=3
     from sklearn.cluster import KMeans

     clusters = KMeans( 3 )
     clusters.fit( customers_df )
```

```
▾        KMeans
KMeans(n_clusters=3)
```

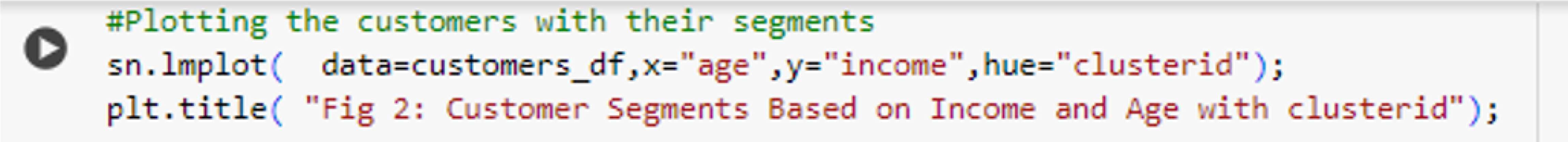
```
[7]  #Now create a label for the data
     customers_df["clusterid"] = clusters.labels_
```

```
     #display the sample data
     customers_df[0:5]
```

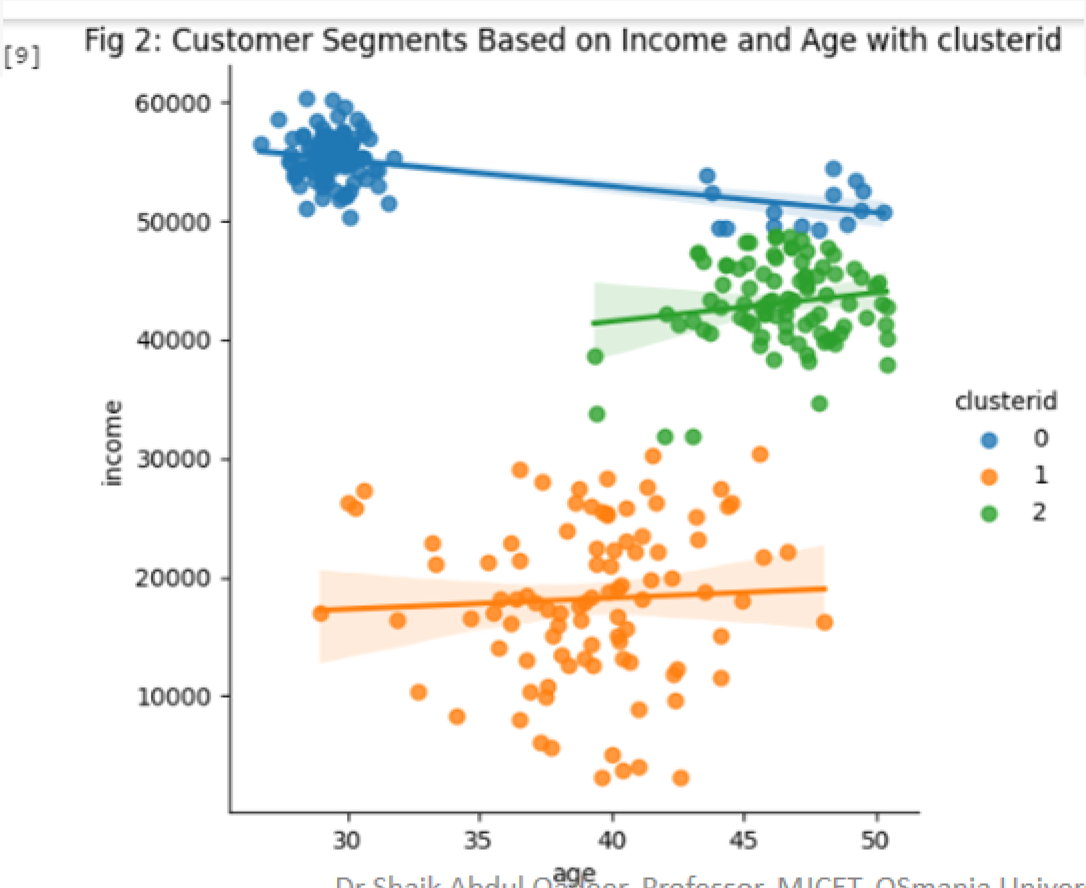


	income	age	clusterid
0	41100.0	48.75	2
1	54100.0	28.10	0
2	47800.0	46.75	2
3	19100.0	40.25	1
4	18200.0	35.80	1

Plotting Customers with their Segments



```
#Plotting the customers with their segments
sn.lmplot(  data=customers_df,x="age",y="income",hue="clusterid");
plt.title( "Fig 2: Customer Segments Based on Income and Age with clusterid");
```


[9]
Fig 2: Customer Segments Based on Income and Age with clusterid
income
age
clusterid
● 0
● 1
● 2