We at The Data Monk hold the vision to make sure everyone in the IT industry has an equal stand to work in an open domain such as analytics. Analytics is one domain where there is no formal under-graduation degree and which is achievable to anyone and everyone in the World.

We are a team of 30+ mentors who have worked in various product-based companies in India and abroad, and we have come up with this idea to provide study materials directed to help you crack any analytics interview.

Every one of us has been interviewing for at least the last 6 to 8 years for different positions like Data Scientist, Data Analysts, Business Analysts, Product Analysts, Data Engineers, and other senior roles. We understand the gap between having good knowledge and converting an interview to a top product-based company.

Rest assured that if you follow our different mediums like our blog cum questions-answer portal www.TheDataMonk.com , our youtube channel - The Data Monk, and our e-books, then you will have a very strong candidature in whichever interview you participate in.

There are many blogs that provide free study materials or questions on different analytical tools and technologies, but we concentrate mostly on the questions which are asked in an interview. We have a set of 100+ books which are available both on Amazon and on The Data Monk e-shop page

We would recommend you to explore our website, youtube channel, and e-books to understand the type of questions covered in our articles. We went for the question-answer approach both on our website as well as our e-books just because we feel that the best way to go from beginner to advance level is by practicing a lot of questions on the topic.

We have launched a series of 50 e-books on our website on all the popular as well as niche topics. Our range of material ranges from SQL, Python, and Machine Learning algorithms to ANN, CNN, PCA, etc.

We are constantly working on our product and will keep on updating it. It is very necessary to go through all the questions present in this book.

Give a rating to the book on Amazon, do provide your feedback and if you want to help us grow then please subscribe to our Youtube channel.

K-means Algorithm

Q1. What is Machine Learning?

ANS. Machine learning is a subset of artificial intelligence that provides the system the ability to work, learn and improve from experience without being explicitly programmed.

In Simple words, Machine Learning is known for learning from experience. Machine learning is a process where the learner performs a task and improves from experience automatically with the help of a machine learning algorithm.

Advantages of Machine Learning:

- Machine Learning is accurate as compared to humans.
- Machine Learning is capable discovering patterns and trends.
- Once Machine learning is automated it does not require any human interference.
- Machine Learning has a wide range of applications.

Disadvantages of Machine Learning:

- For training the model machine learning algorithm requires a lot of data.
- Machine Learning algorithm needs enough time for learning.
- Machine Learning requires massive resources for proper functioning.
- It is very difficult to get perfect accuracy.

Q2. How does machine learning works?

ANS. Machine learning works in the following ways:

- The raw data is imported into the system.
- The data in the real world is very messy so it is important the clean the data this is known as data cleaning.
- During the process of data cleaning we perform various types of EDA(Exploratory data analysis) to better understand the data.
- After all this is done we divide our data set into two parts train set and test set.
- Training phase - Training data is used to train the model.
- Prediction phase - Predicting using the model.
- Validation phase - Testing data is used to validate the model accuracy, goodness of fit and other various factors.

Q3. Different type of machine learning algorithm?

ANS. Types of machine learning algorithm are:

1. Supervised Learning Algorithm: In this type of algorithms we have the independent variable(X) as well as the dependent variable(Y) we can also say that the data is labelled and the algorithms learns to predict the output based on the input data.
   Example: Linear Regression, Logistic regression, KNN, Decision Tree, SVM etc.
2. Unsupervised Learning Algorithm: In this type of algorithms we only have the independent variable(X) or we can also say that the data is unlabelled and the algorithms tried to learn the structure, similarity and patterns from the input data.
   Example: K-means clustering, Hierarchical clustering etc.
3. Reinforcement Learning Algorithm: In this type of algorithms we have the independent variable(X) as well as the dependent variable(Y) this algorithms learns from itself and never repeats a mistake.
   Example: Recurrent Neural Network(RNN), Convolutional Neural Network(CNN), Artificial Neural Network(ANN).

Q4. What is Artificial Intelligence?

ANS. Artificial Intelligence is a subset of Computer Science that focuses on building systems that can behave like human beings. Artificial Intelligence helps in building a system or machine that has the capacity to understand the things like a human mind. Artificial Intelligence uses various algorithms for automating a process or decision-making.

Examples of Artificial Intelligence:

- Smart assistants like Google, Siri, and Alexa.
- Manufacturing of robots.
- Conversational bots for customer support and services.
- Recommendation systems on social media or e-commerce websites.

Q5. What is K-means?

ANS. K-means clustering is an unsupervised machine learning algorithm. K-means is the most common type of clustering technique. The process of k-means clustering is very simple it tries to classify the given dataset into number of clusters(groups) given by the k. Suppose we take k = 3 in this case 3 clusters are formed. It tries to find the pattern, similarity to create the clusters.

Example of k-means:

Suppose I have a data set of a customers where the columns in the data are Annual income and spending scores of the individuals. We have to find that depending on the annual income and spending scores what kind of group does the individual falls for example if the individual annual income is low and spending score is high then he will be grouped as careless. Let's take another example if the individual annual score is high and the spending score is precise not that low not that high then he will be place in the careful and so on. So basically we are creating different subgroups based on the similarity of the data.

Q6. What is clustering?

ANS. Clustering is an Exploratory data analysis technique. It is used to discover subgroups from a dataset based on similar pattern. The data points in each subgroups must be very similar to each other and the data points in the other subgroups must be different. This groups are formed based on the Euclidean distance formula. Clustering can be used in market segmentation where we try to find the customers with similar characteristics.

Q7. How does k-means clustering works?

ANS. The k-means clustering works in the following ways:

1. First we have to select the number of clusters by specifying k.
2. Selecting random points or centroids.
3. Assigning each data points to the nearest centroids.
4. Calculating the sum of the squared distance between data points and all centroids and assigning the new clusters to each data points.
5. Now repeat the third step which is nothing but assigning the data points to the nearest clusters.
6. In case any reassignment happens repeat the step forth or else done.
7. Our model is ready.

Q8. What does k represents in k-means clustering?

ANS. k in k-means clustering represents number of clusters.

Q9. What is a cluster?

ANS. A cluster is nothing but the collection of data points aggregated together because of some similarities.

Q10. What is a centroid?

ANS. A centroid is a point that represents the centre of the cluster.

Q11. What does k-means term denotes?

ANS. k in k-means denotes the number of centroids or number of clusters we want to create from the data. Means in k-means denotes averaging of the data.

Q12. How to find the optimal value of k in k-means?

ANS. To find the optimal(minimal) value of k in k means the most popular method is known as elbow method.

To find the optimal value we use elbow method on the K-means clustering algorithm using a for loop on a range of value from 1 to 11.

First we perform k-means clustering on all the values from the range 1 to 11. For each k value we calculate the Within-Cluster Sum of Squared Error.

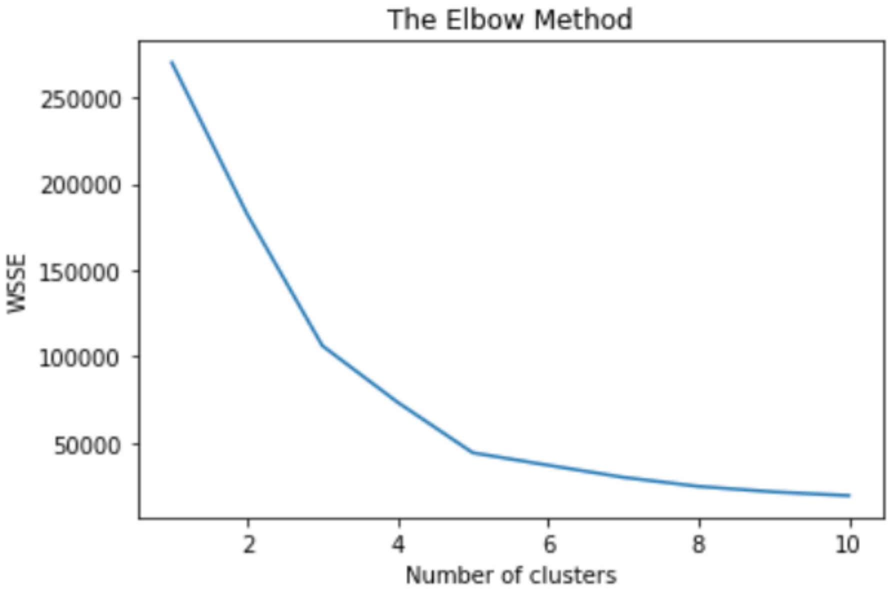
After this we plot a line graph for the value 1 to 11 which is nothing but the number of clusters against wsse(Within-Cluster Sum of Squared Error).

The point at which the line graph suddenly falls(elbow point) that is the optimal value of k in k-means clustering or the optimal number of cluster.

CODE FOR ELBOW METHOD:

```
# Using the elbow method to find the optimal number of clusters

from sklearn.cluster import KMeans
wsse = []
for i in range(1, 11):

   kmeans = KMeans(n_clusters = i, random_state = 10)

   kmeans.fit(X)

   wsse.append(kmeans.inertia_)
plt.plot(range(1, 11), wsse)
plt.title('The Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WSSE')
plt.show()
```

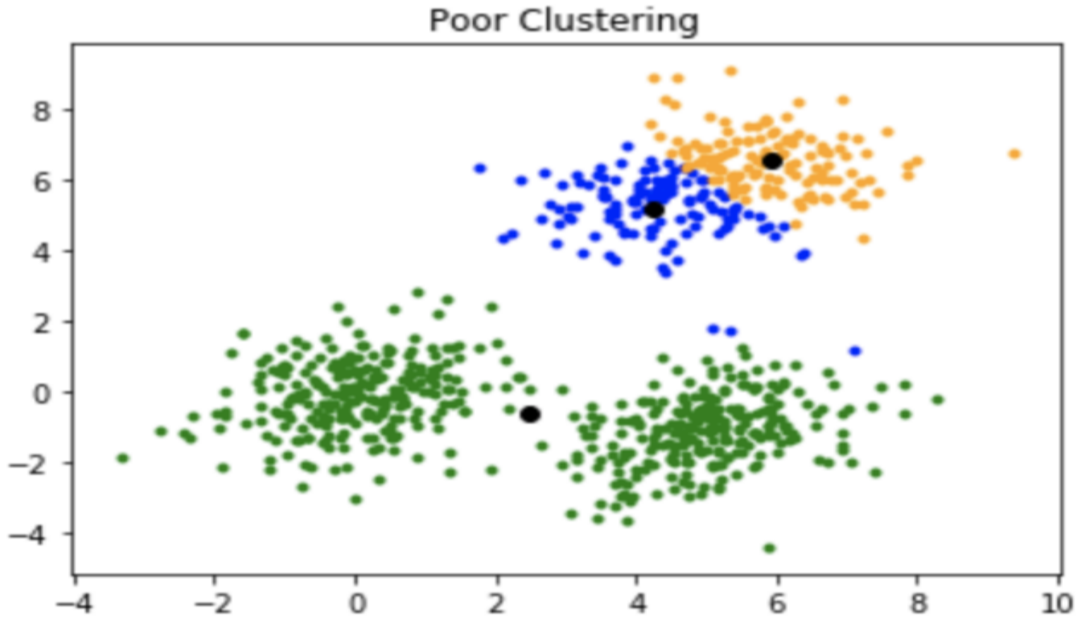
OUTPUT:



Q13. What is the drawback of K-means clustering algorithm?

ANS. The drawback of K-means clustering algorithm is that it is sensitive for initialization of the centroids. Suppose if a centroid is initialized to a very far data point it might end up with no other data points associated with it or it might consider more than one cluster with a single centroid and vice versa more than one centroids might be initialized into the common cluster this results in poor clustering. Poor initialization of centroids might give us poor results in clustering.

Example:



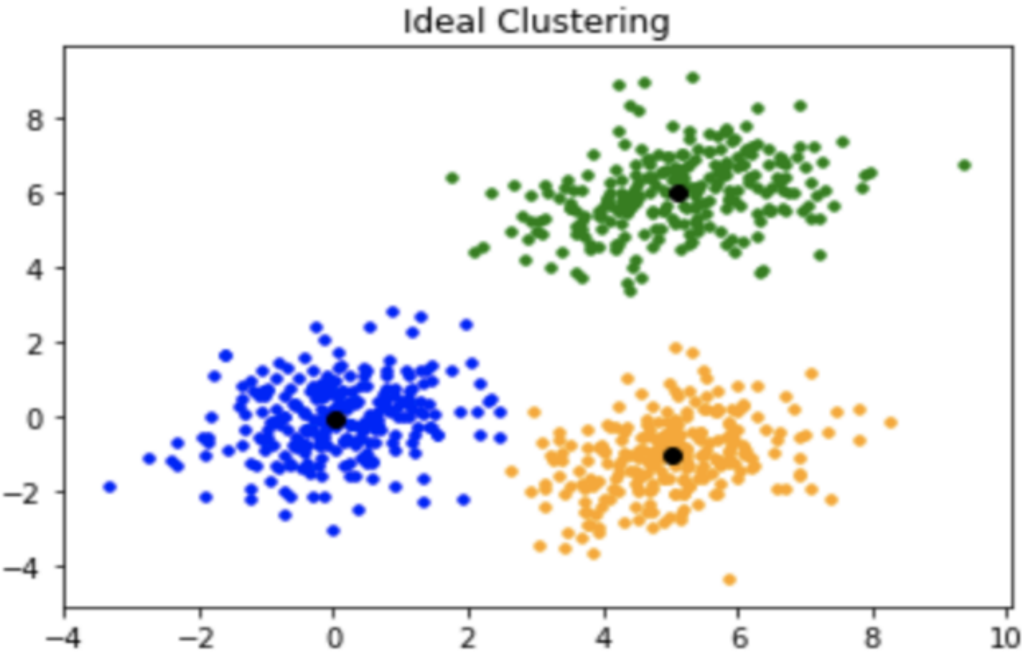
Q14 What are the different ways to solve the problem of initialization sensitivity in k-means algorithm?

ANS. There are two ways to solve the problem of initialization sensitivity in k-means algorithm:

- Repeat k-means: In this case the algorithm repeats itself again and again initializing the centroids thus creating the clusters with small intracluster distance and large intercluster distance.
- K-means++: K-means++ algorithm uses a smart initialization process that deals with the problem of initialization sensitivity in k-means algorithm.

Q15. What is K-means++ algorithm?

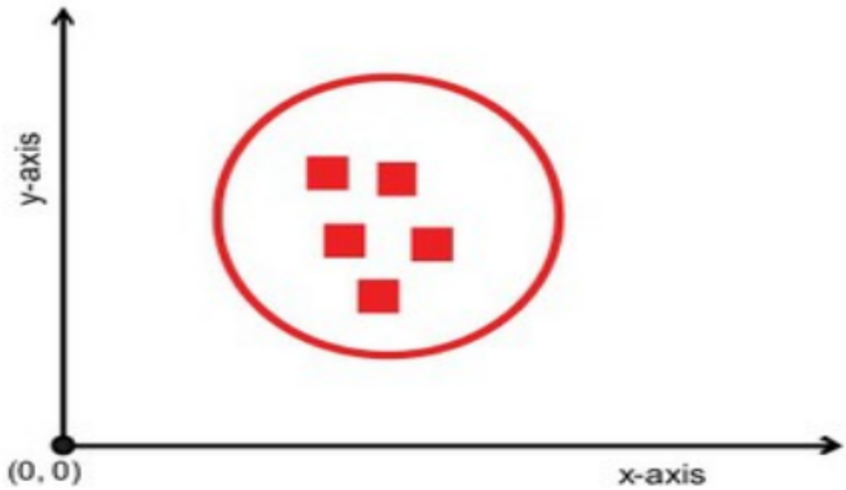
ANS. To deal with the drawback of k-means algorithm which is the initialization problem of centroids we use k-means ++. K-means++ algorithm does a smart initialization of the centroids and tries to improve the quality of the clusters. Only the initialization problem is different from the standard k-means algorithm other than that everything else is the same. To solve the above problem we used k-means++ algorithm which results the following output



Q16. What do you mean by intracluster distance?

ANS. Intracluster distance is nothing but the distance between the two data points or the members of the same cluster. This gives us the idea that how well the distance measures are able to bring the items together. The intracluster distance between the members of the cluster should be small as compared to the intercluster distance. The intracluster distance should be as small as possible so that it is able to bring similar data points together.

Example:



Q17. What do you mean by intercluster distance?

ANS. Intercluster distance is nothing but the distance between the two data points or the members of the different clusters. The intercluster distance between the members of the cluster should be big as compared to the intracluster distance. The distance should be maximum so that it can distinguish that the two points belong to different clusters.

Q18. Difference between K-means and K-means++ algorithm?

ANS. The K-means and the K-means++ algorithm are clustering techniques that comes under Unsupervised Learning. K-means++ algorithm is used to overcome the drawback of the k-means algorithm. The K-means++ algorithm gives a more intelligent initialization of centroids by which the cluster takes place and therefore it improves the nature of clusters. Besides the initialization, there is no other differences and they are almost the same.

Q19. Difference between Classification and Clustering?

ANS. Difference between classification and clustering are as follow:

- Classification is used for Supervised Learning Algorithm and Clustering is used for Unsupervised Learning Algorithm.
- Classification have labelled data associated with it whereas Clustering is associated with unlabelled data.
- Classification is a process where the inputs are classified based on their corresponding labels and Clustering is a process where grouping is done on the basis of similarity.
- In Classification we have labels so in this case there is need of training dataset and testing dataset for evaluating the accuracy of the model created whereas in case of clustering there is no need of training dataset and testing dataset.
- Classification is much more complex as compared to the clustering because there are many levels in classification technique and in clustering only grouping is done.
- Examples of classification are Logistic Regression, Support Vector Machine, Naïve Bayes, Decision Tree, Random Forest, K-nearest neighbours, etc.
- Examples of clustering are k-means clustering algorithm, Hierarchical Clustering, etc.

Q20. What are the advantages of k-means clustering algorithm?

ANS. The advantages of k-means clustering algorithm are:

- It is easy to understand and easy to implement.
- It works on unlabelled data.
- Working with large number of variables k-means can be computationally faster.
- Guaranteed convergence.
- Works better with spherical clusters.
- Easy to interpret and flexible.

Q21. What are the Disadvantages of k-means clustering algorithm?

ANS. The disadvantages of k-means clustering algorithm are:

- We are suppose to choose the value of k manually which is nothing but the total number of clusters.

- K-mean does not work well when the clusters are of different size and different density. To get a better result the cluster must be spherical and equally sized.
- In K-means data should be numerical if not then some pre-processing of the data in necessary.
- K-means algorithm cannot handle outliers and noisy data.
- We cannot pass a very huge dataset if we do so the results may be poor or the computer may crash.
- If there are two data points which are overlapping then it is unable to differentiate that there are two clusters.

Q22. Difference between KNN and K-means algorithm?

ANS. Sometime we get confused by the K in both the algorithm so the differences are as follow:

- KNN is a supervised learning algorithm whereas K-means is an unsupervised learning algorithm.
- KNN need labelled data to train, test and evaluate whereas K-means need unlabelled data there is no need of training and testing.
- K in KNN indicates number of nearest neighbours and k in K-means indicated total number of clusters or groups.
- KNN can be used for both classification and regression whereas K-means is used for clustering.
- There are many differences between KNN and K-means but there is one similarity that both the algorithm works on distance metrics.

Q23. What are the different types of distance metrics used in K-means algorithm?

ANS. The different types of distance metrics in K-means algorithm are:

- Euclidean: The Euclidean distance determines the distance between two points. If we have a point A and point B the Euclidean distance is an ordinary straight line. It is the distance between the two points in Euclidean space.
- Manhattan: The Manhattan distance is nothing but the (Manhattan distance between two points (a1,b1) and (a2,b2) is |a1 - a2| + |b1 - b2|) simple sum of the distance between two points measured along axes at right angles.

Q24. What are the different types of clustering?

ANS. The different types of clustering are:

- Hierarchical Clustering: Hierarchical clustering is a technique that uses a tree-like structure. Hierarchical clustering is an unsupervised clustering algorithm that creates clusters that have predominant ordering from top to bottom.

- Partitioning Clustering: Partitioning Clustering is a clustering technique that classifies the data into several parts as denoted by the k. Suppose if k=2 the two clusters will be created k1, k2. The objects in the clusters will be different from each cluster but the objects within each cluster will be similar.

Q25. What are the applications of K-means Clustering Algorithm?

ANS. The applications of K-means Clustering Algorithm are:

- Academic performance of the students: K-means helps in categorizing the students into different grades like A1, B1, C1 based on the marks obtained by the students.
- Search engines: K-means helps the search engines like when a search is performed the results are grouped therefore the search engines uses clustering techniques.
- Customer Segmentation: To better understand the customer in the market the owner uses customer segmentation to understand which customer they should target with the help of clustering technique and to understand the customer behaviour.

Q26. Steps for performing K-means Clustering Algorithm in Python.

ANS. The steps for performing K-means clustering in Python are as follows:

Step 1:

Select k data points as the initial cluster centers.(Randomly)

Step 2:

Find the euclidean distance of each data point towards each cluster centres.

Step 3:

Assign each data point to the nearest cluster.

Step 4:

Recompute the new cluster centres by taking mean of the data points belonging to that cluster.

Step 5:

Repeat step 2 to 4.

Step 6:

Stop the process when zero convergence is reached.

End result:

You get the data points clustered into k clusters.

Implementation of K-means Clustering Algorithm with Python:

Description of the dataset: The data set which we are using for the k-means clustering is a customers dataset in which we have columns like CustomerId, Gender, Age, Annual Income and Spending score. For K-means we are only using the two column named Annual Income and Spending score. We have to find that depending on the annual income and spending scores what kind of group does the individual falls for example if the individual annual income is low and spending score is high then he will be grouped as careless. Let’s take another example if the individual annual score is high and the spending score is precise not that low not that high then he will be place in the careful and so on. So basically we are creating different subgroups based on the similarity they have.

- Importing the required Libraries:

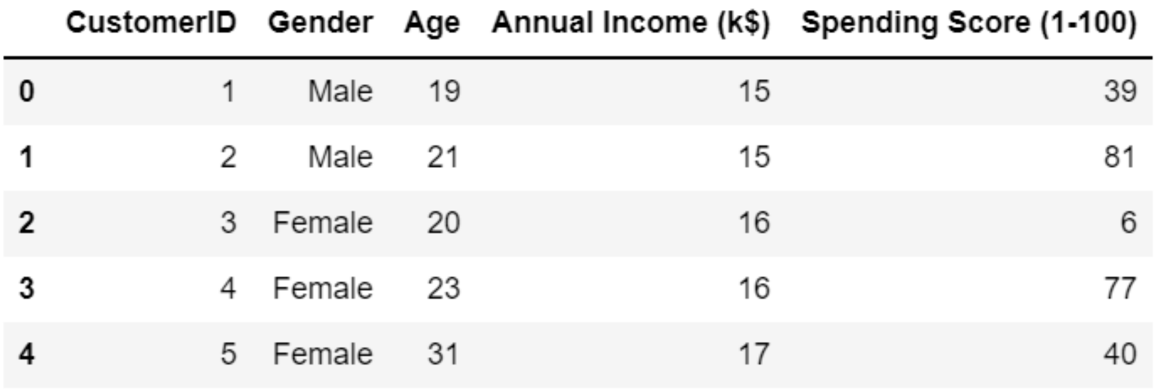
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

- Importing the dataset as DataFrame:

```
df = pd.read_csv(r'Mall_Customers.csv')
```

- Checking the dataset using the head() function:

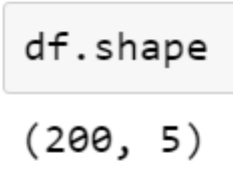
```
df.head()
```



	CustomerID	Gender	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

- Checking the size of the dataset using shape function:

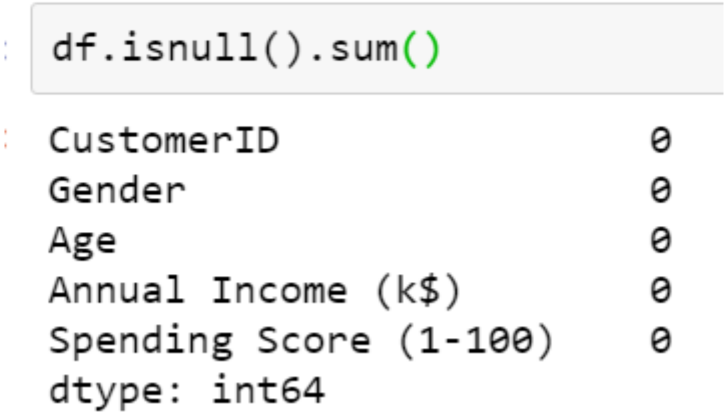
df.shape



```
df.shape
```

```
(200, 5)
```

- Checking for missing values in the dataset using isnull() function:
  df.isnull().sum()

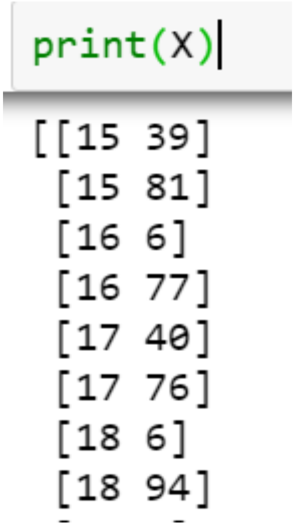


```
df.isnull().sum()
```

```
CustomerID                0
Gender                    0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
```

- Converting the data frame in array because array are much faster then data frame in building models the columns which we are passing are Annual Income (k$) and Spending Score(1-100) because these are the two columns which we will be using for k-means clustering:
  X = df.values[:, [3,4]]

- Printing the array created above:
  print(X)

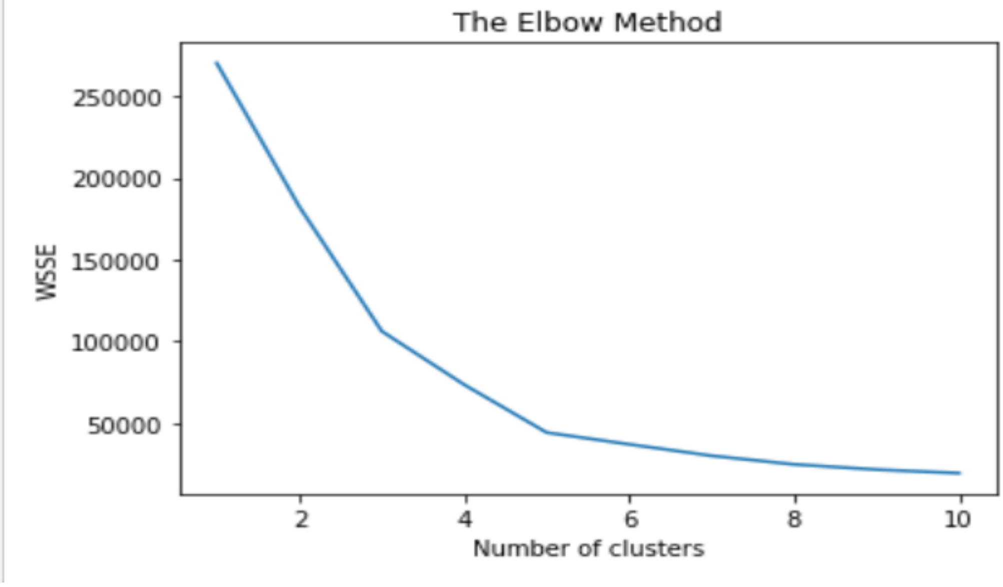


```
print(X)
```

```
[[15 39]
 [15 81]
 [16 6]
 [16 77]
 [17 40]
 [17 76]
 [18 6]
 [18 94]
```

- Using the elbow method to find the optimal number of clusters:

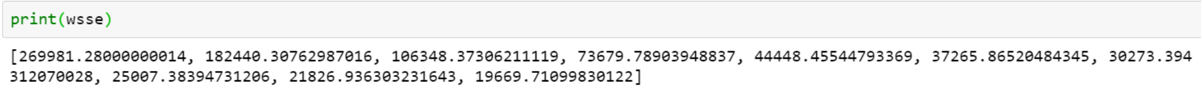
```
from sklearn.cluster import KMeans
wsse = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters = i, random_state = 10)
    kmeans.fit(X)
    wsse.append(kmeans.inertia_)
plt.plot(range(1, 11), wsse)
plt.title('The Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WSSE')
plt.show()
```



The point at which the line graph suddenly falls(elbow point) that is the optimal value of k in k-means clustering or the optimal number of cluster.

In this case the optimal number of cluster is k = 5.

- Printing within-cluster sum of squared error(wsse)
  print(wsse)

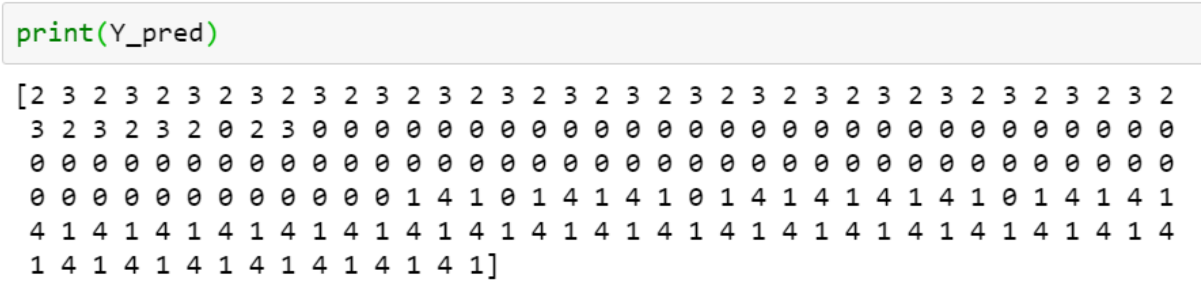


```
print(wsse)
```

```
[269981.28000000014, 182440.30762987016, 106348.37306211119, 73679.78903948837, 44448.45544793369, 37265.86520484345, 30273.394
312070028, 25007.38394731206, 21826.936303231643, 19669.71099830122]
```

- Fitting k-means to the dataset:
  kmeans = KMeans(n_clusters=5, random_state=10)
  Y_pred = kmeans.fit_predict(X)

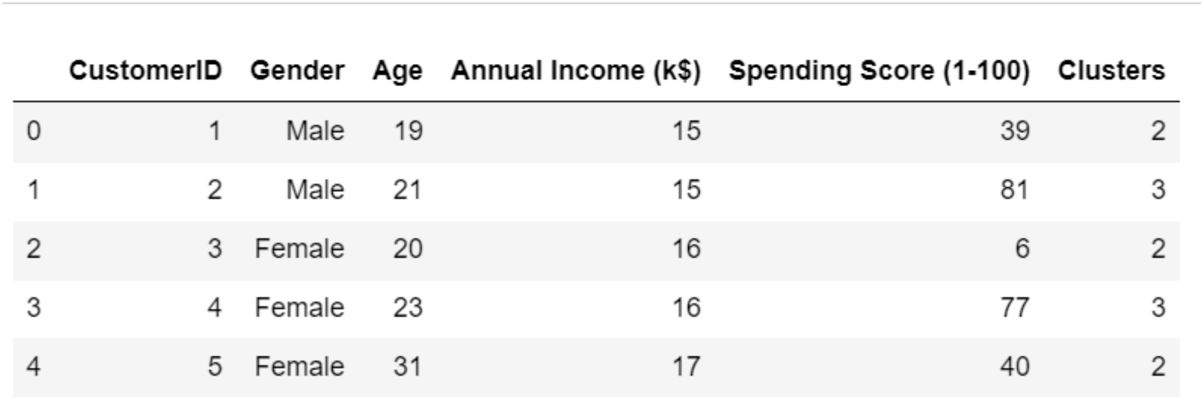
- Printing the Y_pred:
  print(Y_pred)



```
print(Y_pred)
```

```
[2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2
 3 2 3 2 3 2 0 2 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 1 4 1 0 1 4 1 4 1 0 1 4 1 4 1 4 1 4 1 0 1 4 1 4 1
 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4
 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1]
```

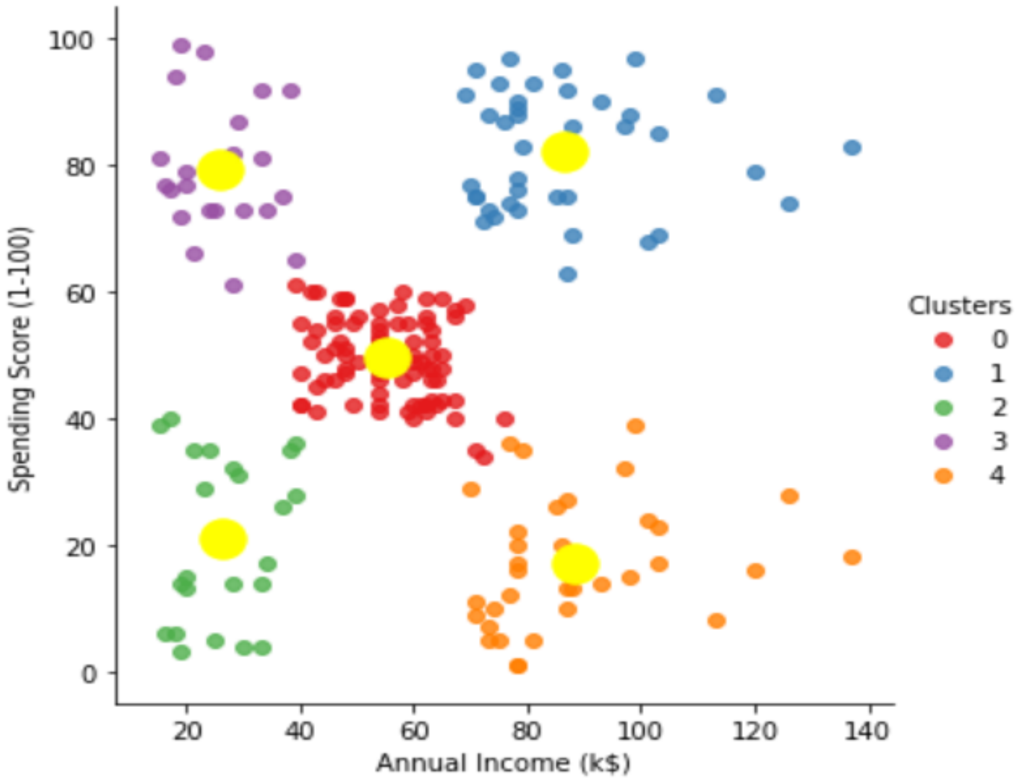
- Now we are creating a new column name Clusters which is the value of Y_pred and storing it in the original dataframe:
  df['Clusters']=Y_pred
  df.head()



	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Clusters
0	1	Male	19	15	39	2
1	2	Male	21	15	81	3
2	3	Female	20	16	6	2
3	4	Female	23	16	77	3
4	5	Female	31	17	40	2

- Plotting the clusters:
  import seaborn as sns

```
sns.lmplot( data=df, x='Annual Income (k$)', y='Spending Score (1-100)',
fit_reg=False, # No regression line
hue='Clusters',palette="Set1")
plt.scatter(kmeans.cluster_centers_[:, 0],
kmeans.cluster_centers_[:, 1],
s = 300, c = 'yellow')
plt.show()
```



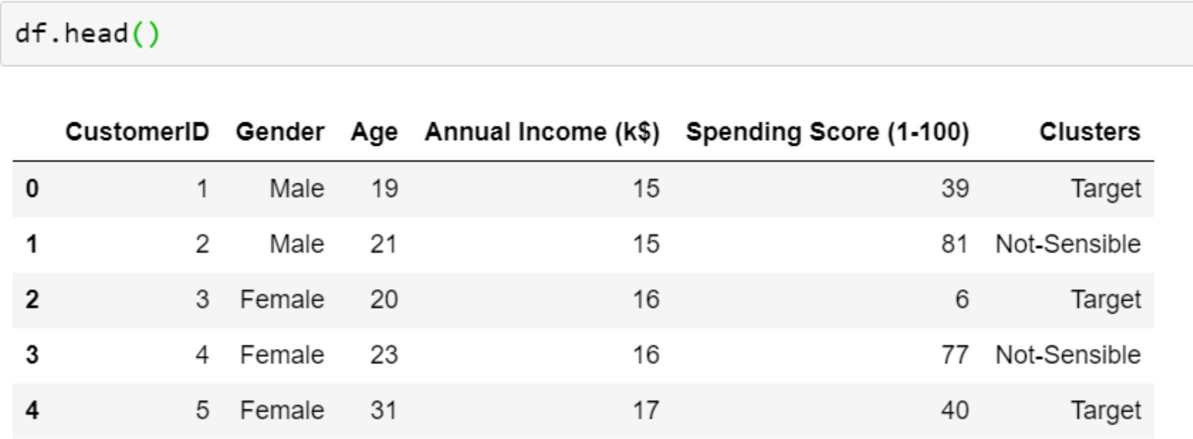
We have created a scatter plot on the model kmeans where we have taken n_clusters = 5 so 5 cluster are plotted in the above scatter polt.

- Manually mapping or assigning the names of the clusters created:

  ```
  df['Clusters']=df.Clusters.map({0:"Careless",1:"Standard",2:"Target",3:'Not-Sensible',4:"Careful"})
  ```

Now we have manually assigned the name if the clusters as the clusters have properties like the cluster 0 which is named careless because the income is low but the spending score is high.

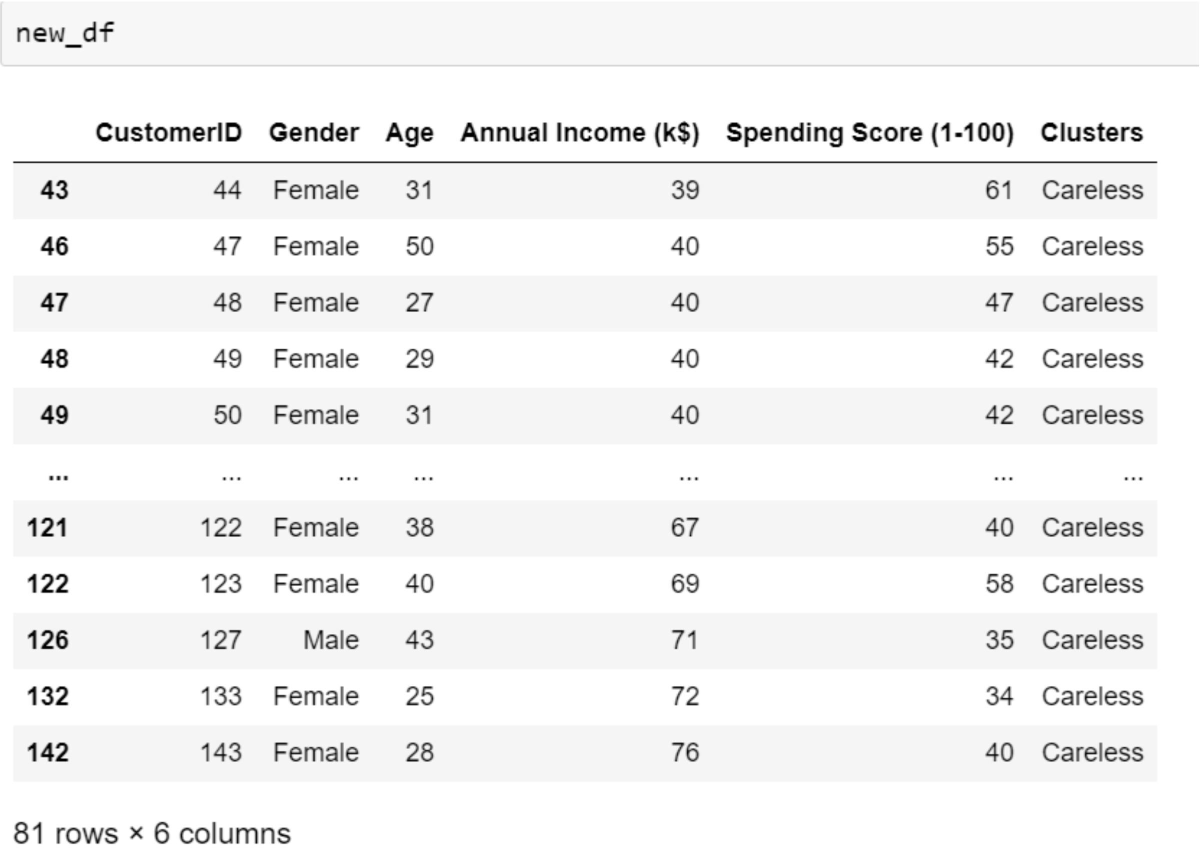
df.head()



```
df.head()
```

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Clusters
0	1	Male	19	15	39	Target
1	2	Male	21	15	81	Not-Sensible
2	3	Female	20	16	6	Target
3	4	Female	23	16	77	Not-Sensible
4	5	Female	31	17	40	Target

- If we want to check for a particular category of clusters we can check that in the following ways:
  new_df=df[df["Clusters"]=="Careless"]
  new_df



```
new_df
```

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Clusters
43	44	Female	31	39	61	Careless
46	47	Female	50	40	55	Careless
47	48	Female	27	40	47	Careless
48	49	Female	29	40	42	Careless
49	50	Female	31	40	42	Careless
...	...	...	...	...	...	...
121	122	Female	38	67	40	Careless
122	123	Female	40	69	58	Careless
126	127	Male	43	71	35	Careless
132	133	Female	25	72	34	Careless
142	143	Female	28	76	40	Careless

81 rows × 6 columns

- We can save the result in the excel file in the following ways: new_df.to_excel("CarelessCustomers.xlsx",index=False)