

ASSIGNMENT-4
APPLIED DATA SCIENCE

Assignment date	06 November 2022
Student Name	Bishal Bhandhari
Student Roll Number	7309730919104132
Maximum Marks	2 Marks

anaconda3/anaconda/ x assignment4 - Jupyter x assignment4 - Jupyter x about:blank x about:blank x Assignment 2.docx.pdf x

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

Minimize

jupyter assignment.4 Last Checkpoint: 18 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted Python 3 (ipykernel)

Download the dataset

In [5]:

```
import pandas as pd
import numpy as np
```

Load the dataset

In [7]:

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

Out [7]:

	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

Perform Below Visualisations

Univariate Analysis

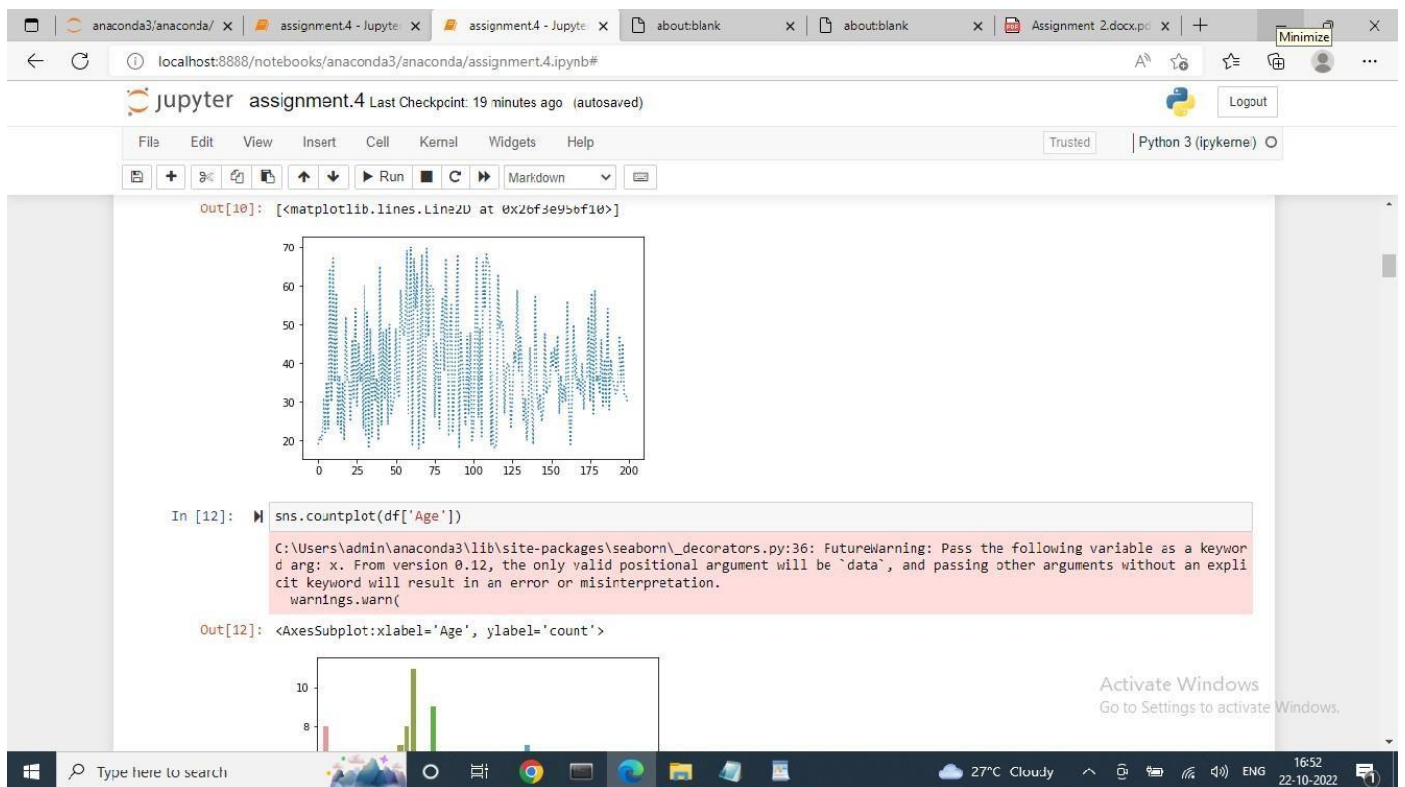
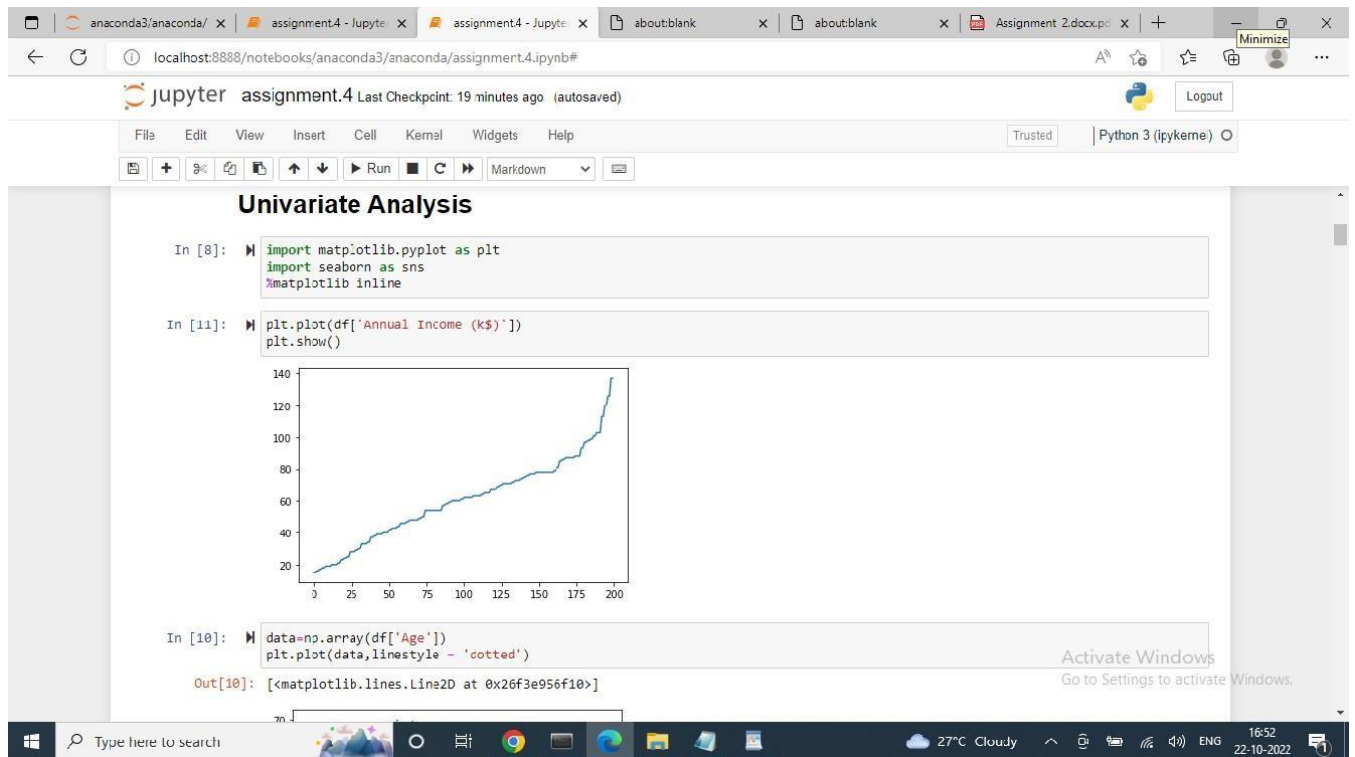
Activate Windows

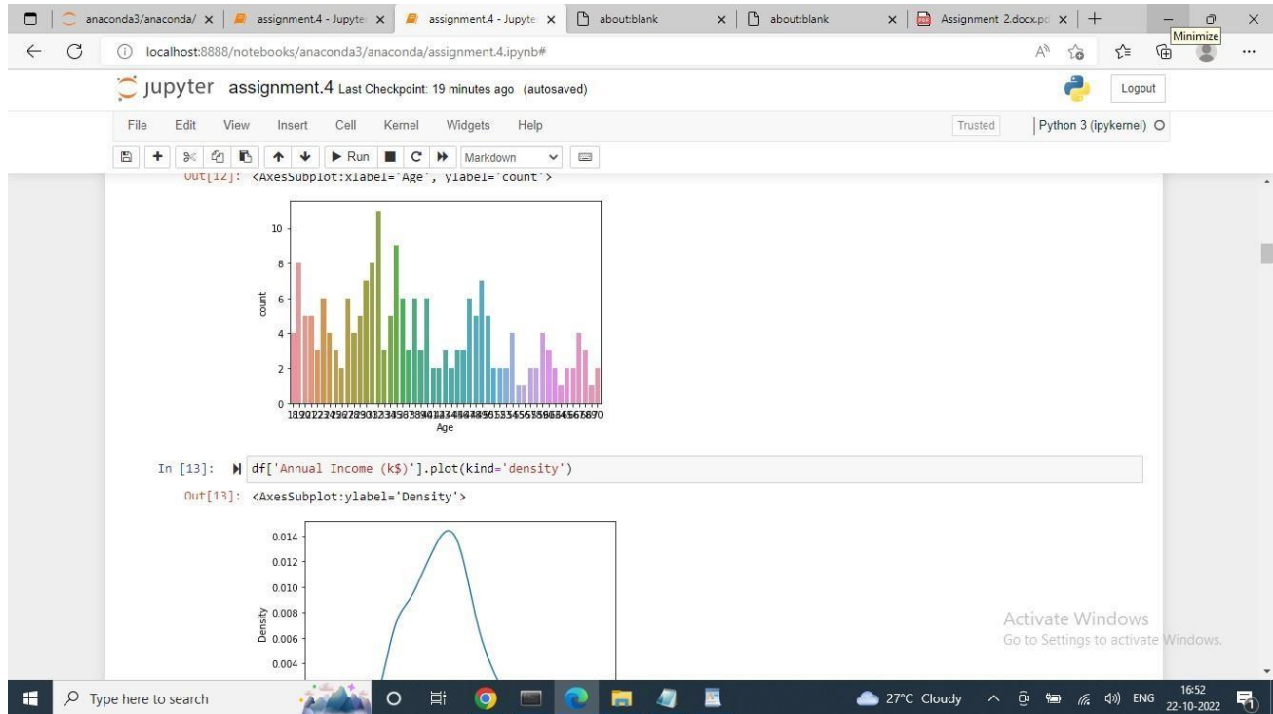
Go to Settings to activate Windows.

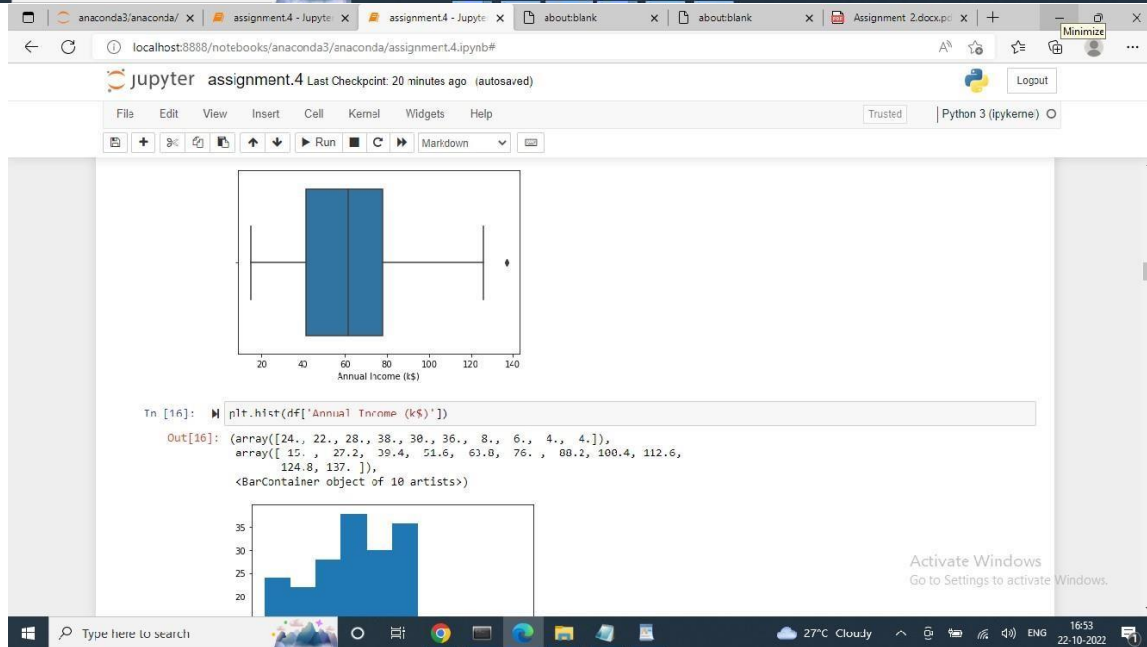
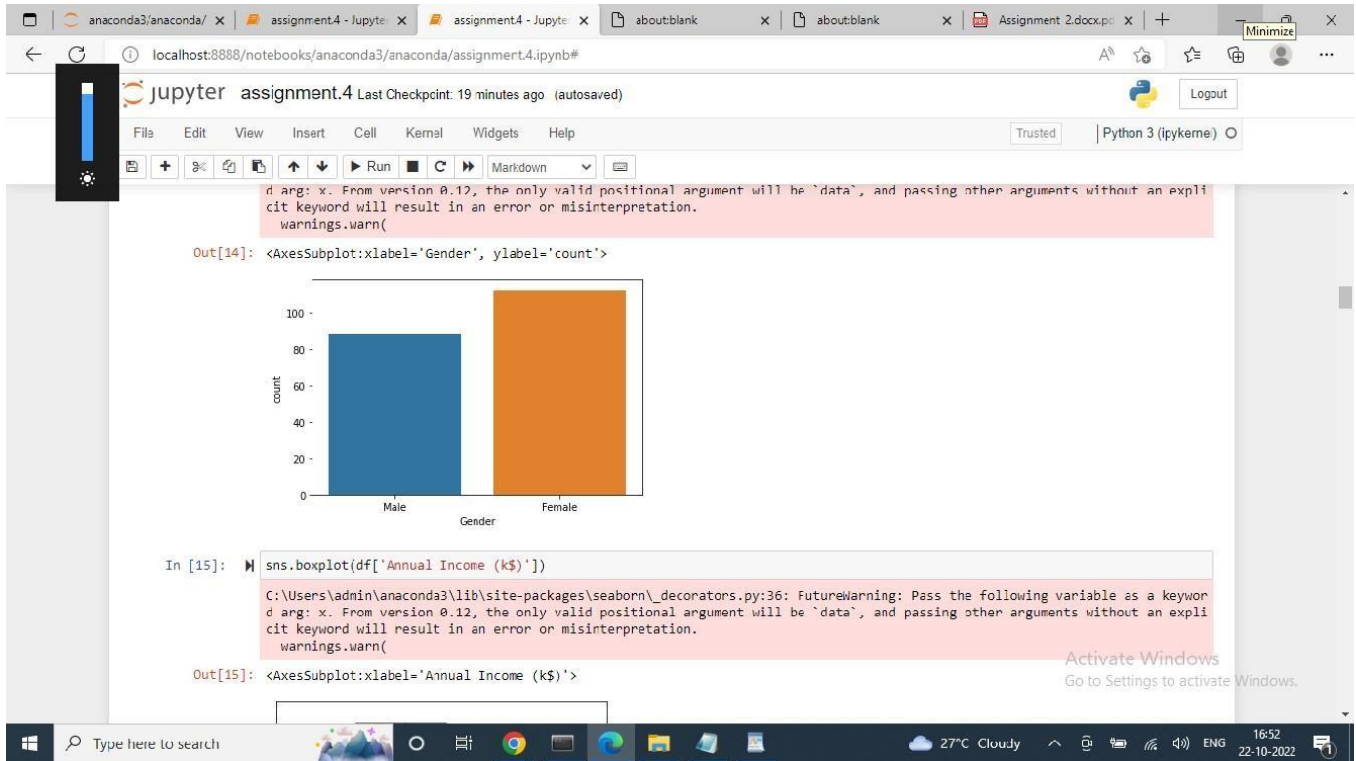
Type here to search

27°C Cloudy

16:51 22-10-2022







anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X about:blank X about:blank X Assignment 2.docx: X + -

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

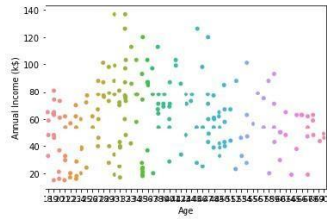
jupyter assignment.4 Last Checkpoint: 20 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

+ - Run C Markdown

Bivariate Analysis

```
In [17]: sns.stripplot(x=df['Age'],y=df['Annual Income (k$)'])
Out[17]: <AxesSubplot:xlabel='Age', ylabel='Annual Income (k$)'
```



```
In [18]: sns.stripplot(x=df['Age'],y=df['Spending Score (1-100)'])
Out[18]: <AxesSubplot:xlabel='Age', ylabel='Spending Score (1-100)'
```



Type here to search

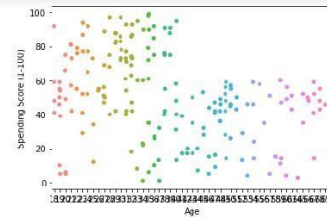
anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X about:blank X about:blank X Assignment 2.docx: X + Minimize X

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

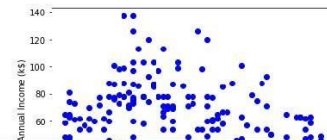
jupyter assignment.4 Last Checkpoint: 20 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

+ - Run C Markdown

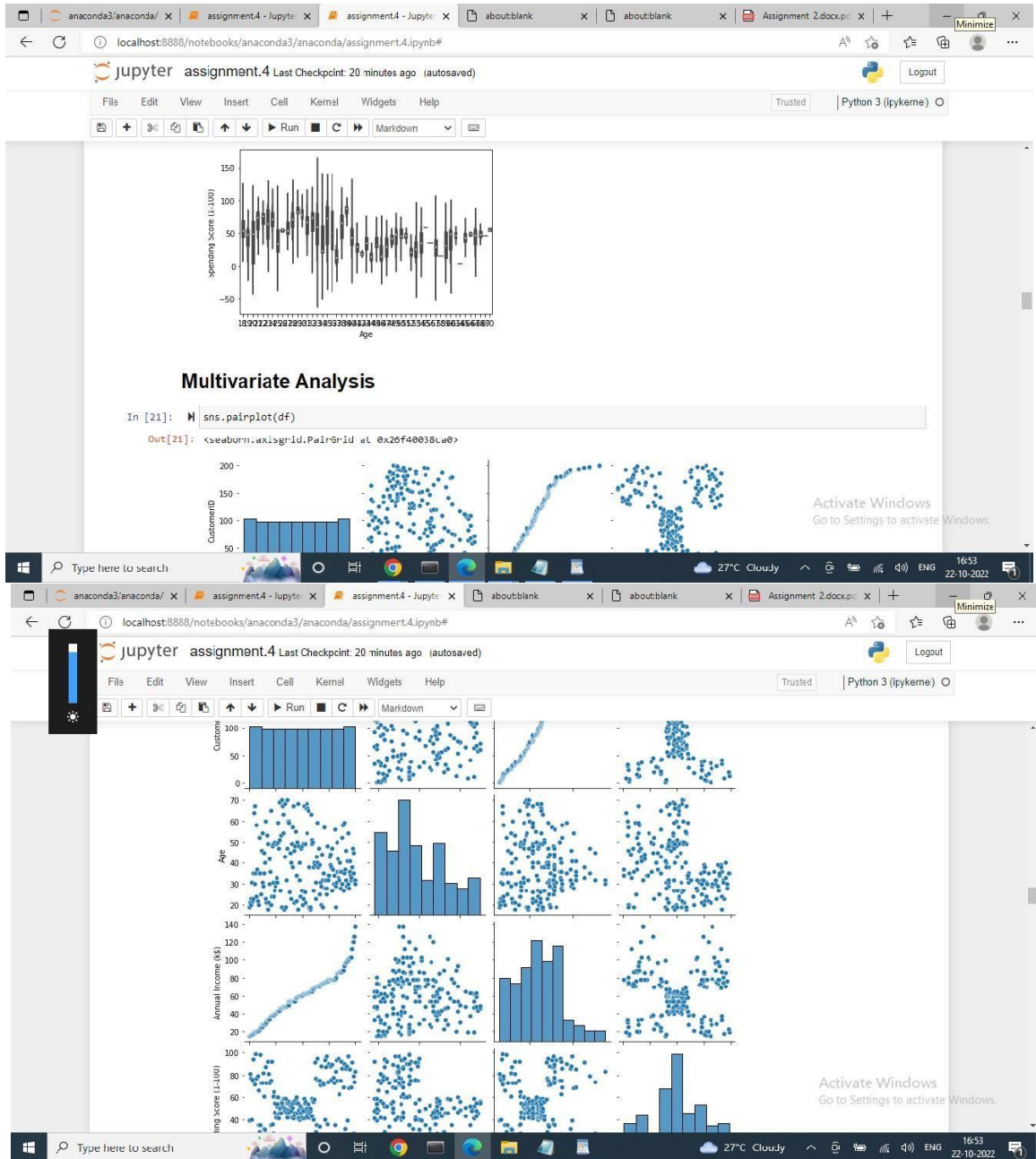


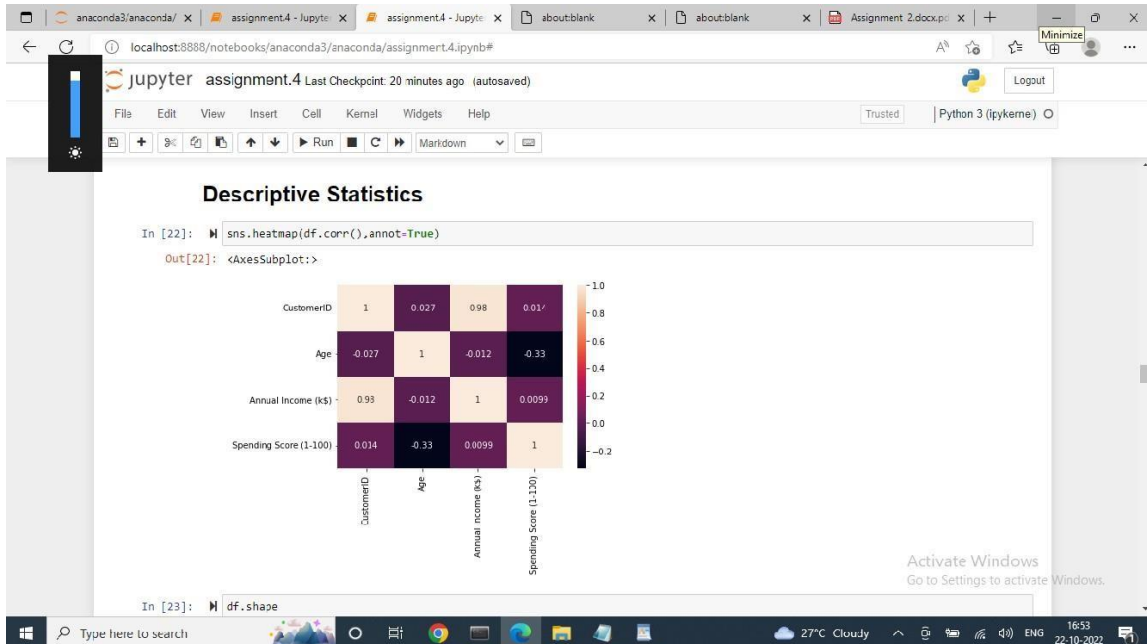
```
In [19]: plt.scatter(df['Age'],df['Annual Income (k$)'],color='blue')
plt.xlabel("Age")
plt.ylabel("Annual Income (k$)")
Out[19]: Text(0, 0.5, 'Annual Income (k$)')
```



Type here to search

27°C Cloudy 16:53 22-10-2022





anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X about:blank X about:blank X Assignment 2.docx: X + - Minimize

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 20 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

```
In [23]: df.shape
```

Out[23]: (200, 5)

```
In [24]: df.isnull().sum()
```

Out[24]:

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

```
In [25]: df.info()
```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
Column Non-Null Count Dtype

0 CustomerID 200 non-null int64
1 Gender 200 non-null object
2 Age 200 non-null int64
3 Annual Income (k\$) 200 non-null int64
4 Spending Score (1-100) 200 non-null int64
dtypes: int64(4), object(1)
memory usage: 7.9+ KB

```
In [27]: df.describe()
```

Out[27]:

	CustomerID	Age	Annual Income (k\$)	Spending Score (1-100)
count	200	200	200	200
mean	100.5	33.8	53.9	48.5
std	85.4	11.5	38.6	14.6
min	1	18	18	1
max	200	70	160	100

Activate Windows
Go to Settings to activate Windows.

Type here to search

anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X about:blank X about:blank X Assignment 2.docx: X + -

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 20 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

In [29]: `df.median()`

```
C:\Users\admin\AppData\Local\Temp\ipykernel_7988\538851474.py:1: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.
df.median()
```

Out[29]:

CustomerID	100.5
Age	36.0
Annual Income (k\$)	61.5
Spending Score (1-100)	50.0
dtype:	float64

In [30]: `df.mode()`

Out[30]:

	CustomerID	Gender	Age	Annual income (k\$)	Spending Score (1-100)
0	1	Female	32.0	54.0	42.0
1	2	NaN	NaN	78.0	NaN
2	3	NaN	NaN	NaN	NaN
3	4	NaN	NaN	NaN	NaN
4	5	NaN	NaN	NaN	NaN
...
195	196	NaN	NaN	NaN	NaN
196	197	NaN	NaN	NaN	NaN
197	198	NaN	NaN	NaN	NaN
198	199	NaN	NaN	NaN	NaN

Activate Windows
Go to Settings to activate Windows.

anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X about:blank X about:blank X Assignment 2.docx: X + -

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 21 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

Check For Missing Values

In [33]: `df.isna().sum()`

Out[33]:

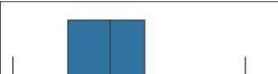
CustomerID	0
Gender	0
Age	0
Annual Income (k\$)	0
Spending Score (1-100)	0
dtype:	int64

Handling Outliers

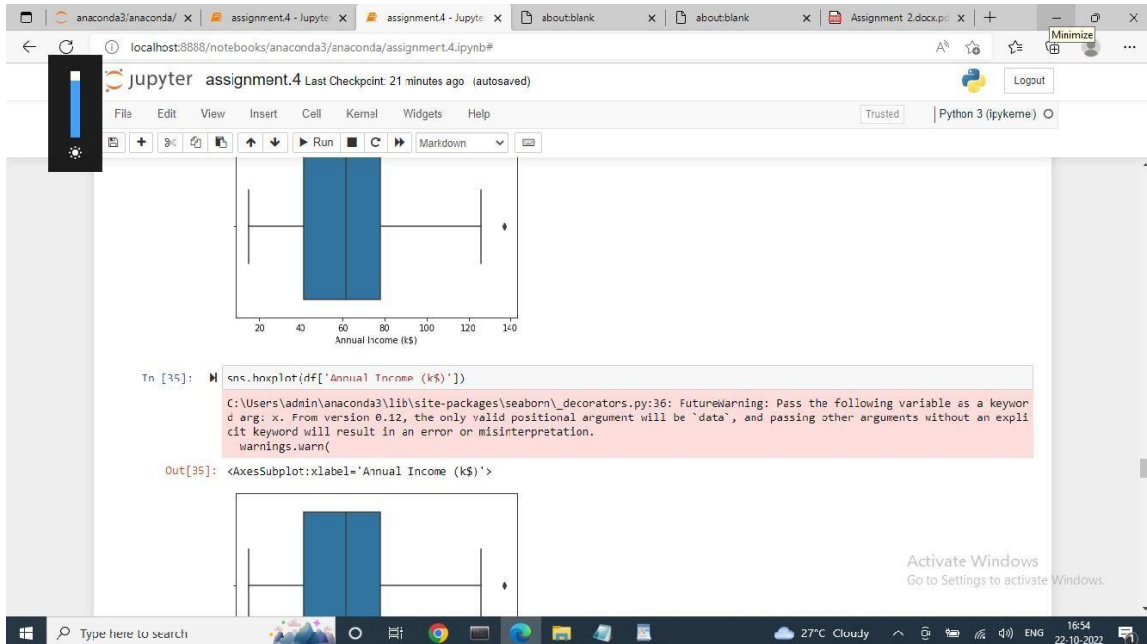
In [34]: `sns.boxplot(df['Annual Income (k$)'])`

```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be 'data', and passing other arguments without an explicit keyword will result in an error or misinterpretation.
warnings.warn()
```

Out[34]: `<AxesSubplot:xlabel='Annual Income (k$)'\>`



Activate Windows
Go to Settings to activate Windows.



anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X about:blank X about:blank X Assignment 2.docx: X + - [Minimize] [Maximize] [Close]

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 21 minutes ago (autosaved) [Logout]

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run [Up] [Down] [Run] [Stop] [Restart] [Markdown]

Encoding Categorical Values

```
In [37]: numeric_data = df.select_dtypes(include=[np.number])  
categorical_data = df.select_dtypes(exclude=[np.number])  
print("Number of numerical variables: ", numeric_data.shape[1])  
print("Number of categorical variables: ", categorical_data.shape[1])  
  
Number of numerical variables: 4  
Number of categorical variables: 1
```

```
In [38]: print("Number of categorical variables: ", categorical_data.shape[1])  
Categorical_variables = list(categorical_data.columns)  
Categorical_variables  
  
Number of categorical variables: 1
```

```
Out[38]: ['Gender']
```

```
In [39]: df['Gender'].value_counts()
```

```
Out[39]: Female    112  
Male           88  
Name: Gender, dtype: int64
```

```
In [40]: from sklearn.preprocessing import LabelEncoder  
le = LabelEncoder()  
label = le.fit_transform(df['Gender'])  
df['Gender'] = label
```

Activate Windows
Go to Settings to activate Windows.

anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X aboutblank X aboutblank X Assignment 2.docx X + Minimize X

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 21 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

Run

Scaling The Data

```
In [42]: X = df.drop("Age",axis=1)
         y = df["Age"]

In [43]: from sklearn.preprocessing import StandardScaler
         object= StandardScaler()
         scale = object.fit_transform(X)
         print(scale)
```

```
[ 1.41163905 -0.88640526  1.390894  1.38581187]
[ 1.42895978  1.12815215  1.42906343 -1.36651894]
[ 1.4162805  -0.88640526  1.42906343  1.46745499]
[ 1.45360123 -0.88640526  1.46723286 -0.43480148]
[ 1.48092195  1.12815215  1.46723286  1.81684904]
[ 1.49824268 -0.88640526  1.54357172 -1.01712489]
[ 1.5155634  1.12815215  1.54357172  0.69102378]
[ 1.53288413 -0.88640526  1.61991057 -1.28887582]
[ 1.55020485 -0.88640526  1.61991057  1.35699031]
[ 1.56752558 -0.88640526  1.61991057 -1.05594645]
[ 1.5848463  -0.88640526  1.61991057  0.72584534]
[ 1.60216702  1.12815215  2.00160487 -1.63826986]
[ 1.61948775 -0.88640526  2.00160487  1.58391968]
[ 1.63680847 -0.88640526  2.26879087 -1.32769738]
[ 1.6541292  -0.88640526  2.26879087  1.11806095]
[ 1.67144992 -0.88640526  2.49780745 -0.86183865]
[ 1.68877065  1.12815215  2.49780745  0.92395314]
[ 1.70609137  1.12815215  2.91767117 -1.25805425]
[ 1.7234121  1.12815215  2.91767117  1.27334719]
```

Activate Windows
Go to Settings to activate Windows.

anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X aboutblank X aboutblank X Assignment 2.docx X + Minimize X

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 21 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

Run

```
object= StandardScaler()
scale = object.fit_transform(X)
print(scale)
```

```
[ 1.41163905 -0.88640526  1.390894  1.38581187]
[ 1.42895978  1.12815215  1.42906343 -1.36651894]
[ 1.4162805  -0.88640526  1.42906343  1.46745499]
[ 1.45360123 -0.88640526  1.46723286 -0.43480148]
[ 1.48092195  1.12815215  1.46723286  1.81684904]
[ 1.49824268 -0.88640526  1.54357172 -1.01712489]
[ 1.5155634  1.12815215  1.54357172  0.69102378]
[ 1.53288413 -0.88640526  1.61991057 -1.28887582]
[ 1.55020485 -0.88640526  1.61991057  1.35699031]
[ 1.56752558 -0.88640526  1.61991057 -1.05594645]
[ 1.5848463  -0.88640526  1.61991057  0.72584534]
[ 1.60216702  1.12815215  2.00160487 -1.63826986]
[ 1.61948775 -0.88640526  2.00160487  1.58391968]
[ 1.63680847 -0.88640526  2.26879087 -1.32769738]
[ 1.6541292  -0.88640526  2.26879087  1.11806095]
[ 1.67144992 -0.88640526  2.49780745 -0.86183865]
[ 1.68877065  1.12815215  2.49780745  0.92395314]
[ 1.70609137  1.12815215  2.91767117 -1.25805425]
[ 1.7234121  1.12815215  2.91767117  1.27334719]
```

```
In [44]: X_scaled = pd.DataFrame(scale, columns = X.columns)
         X_scaled
```

```
Out[44]:
```

	CustomerID	Gender	Annual Income (k\$)	Spending Score (1-100)
0	-1.723412	1128152	-1.738999	-0.434801
1	-1.706091	1128152	-1.738999	1.195704

Activate Windows
Go to Settings to activate Windows.

anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X aboutblank X aboutblank X Assignment 2.docx: X + - Minimize

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 21 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

In [44]: `X_scaled = pd.DataFrame(scale, columns = X.columns)`

Out[44]:

	CustomerID	Gender	Annual Income (K\$)	Spending Score (1-100)
0	-1.723412	1	128152	-1.738999
1	-1.706991	1	128152	-1.738999
2	-1.688771	-0.886405	-1.700830	-1.715913
3	-1.671450	-0.886405	-1.700830	1.040418
4	-1.654129	-0.886405	-1.662660	-0.395980
...
195	1.654129	-0.886405	2.268791	1.118061
196	1.671450	-0.886405	2.497807	-0.861839
197	1.688771	1	128152	2.497807
198	1.706991	1	128152	2.917671
199	1.723412	1	128152	2.917671

200 rows x 4 columns

In [45]: `#train test split
from sklearn.model_selection import train_test_split
split the dataset
X_train, X_test, Y_train, Y_test = train_test_split(X_scaled, Y, test_size=0.20, random_state=0)`

In [48]: `X_train.shape`

Activate Windows
Go to Settings to activate Windows.

anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X aboutblank X aboutblank X Assignment 2.docx: X + - Minimize

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 21 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

`X_train, X_test, Y_train, Y_test = train_test_split(X_scaled, Y, test_size=0.20, random_state=0)`

In [40]: `X_train.shape`

Out[40]: (160, 4)

In [49]: `X_test.shape`

Out[49]: (40, 4)

In [50]: `Y_train.shape`

Out[50]: (160,)

In [51]: `Y_test.shape`

Out[51]: (40,)

#clustering algorithm

In [52]: `x = df.iloc[:, [3, 4]].values`

In [53]: `#finding optimal number of clusters using the elbow method
from sklearn.cluster import KMeans
wcss_list = [] #Initializing the list for the values of WCSS

#Using for loop for iterations from 1 to 10.
for i in range(1, 11):
 kmeans = KMeans(n_clusters=i, init='k-means++', random_state= 42)`

Activate Windows
Go to Settings to activate Windows.

anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X aboutblank X aboutblank X Assignment 2.docx: X + - Minimize X

localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

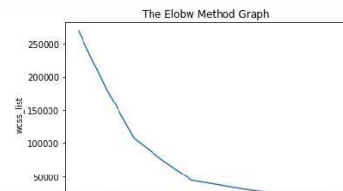
jupyter assignment.4 Last Checkpoint: 21 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

```
from sklearn.cluster import KMeans
wcss_list= [] #Initializing the List for the values of WCSS

#Using for loop for iterations from 1 to 10.
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', random_state= 42)
    kmeans.fit(x)
    wcss_list.append(kmeans.inertia_)
plt.plot(range(1, 11), wcss_list)
plt.title('The Elbow Method Graph')
plt.xlabel('Number of clusters(k)')
plt.ylabel('wcss_list')
plt.show()
```

C:\Users\admin\anaconda3\lib\site-packages\sklearn\cluster_kmeans.py:1036: UserWarning: KMeans is known to have a memory leak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environment variable OMP_NUM_THREADS=1.
warnings.warn(



Activate Windows
Go to Settings to activate Windows.

Type here to search

anaconda3/anaconda/ X assignment4 - Jupyter X assignment4 - Jupyter X aboutblank X aboutblank X Assignment 2.docx: X + - X

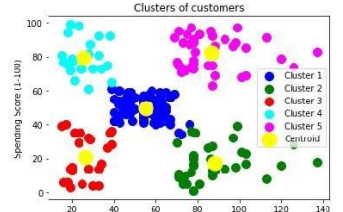
localhost:8888/notebooks/anaconda3/anaconda/assignment4.ipynb#

jupyter assignment.4 Last Checkpoint: 21 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

```
kmeans = KMeans(n_clusters=5, init='k-means++', random_state= 42)
y_predict= kmeans.fit_predict(x)

In [56]: #visualizing the clusters
plt.scatter(x[y_predict == 0, 0], x[y_predict == 0, 1], s = 100, c = 'blue', label = 'Cluster 1') #for first cluster
plt.scatter(x[y_predict == 1, 0], x[y_predict == 1, 1], s = 100, c = 'green', label = 'Cluster 2') #for second cluster
plt.scatter(x[y_predict == 2, 0], x[y_predict == 2, 1], s = 100, c = 'red', label = 'Cluster 3') #for third cluster
plt.scatter(x[y_predict == 3, 0], x[y_predict == 3, 1], s = 100, c = 'cyan', label = 'Cluster 4') #for fourth cluster
plt.scatter(x[y_predict == 4, 0], x[y_predict == 4, 1], s = 100, c = 'magenta', label = 'Cluster 5') #for fifth cluster
plt.scatter(kmeans.cluster_centers[:, 0], kmeans.cluster_centers[:, 1], s = 300, c = 'yellow', label = 'Centroid')
plt.title('Clusters of customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```



Activate Windows
Go to Settings to activate Windows.

Type here to search

