

Assignment date	21 october 2022
Student Name	Varshini P
Student Roll Number	721719104089
Maximum Marks	2 Marks

The screenshot shows a Jupyter Notebook titled "assignment.4" running on a local host. The notebook contains two code cells. The first cell imports pandas and numpy. The second cell reads a CSV file named "Mall_Customers.csv" and displays the first five rows of the data. The data includes columns for CustomerID, Gender, Age, Annual Income (k\$), and Spending Score (1-100).

Download the dataset

```
In [5]: import pandas as pd
import numpy as np
```

Load the dataset

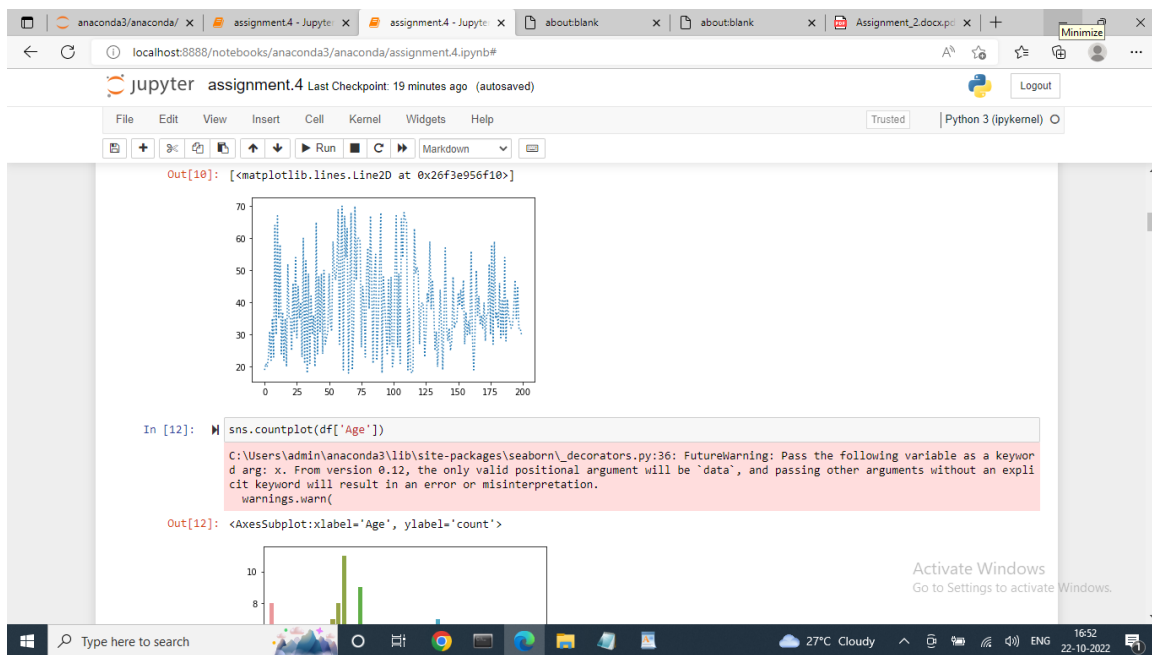
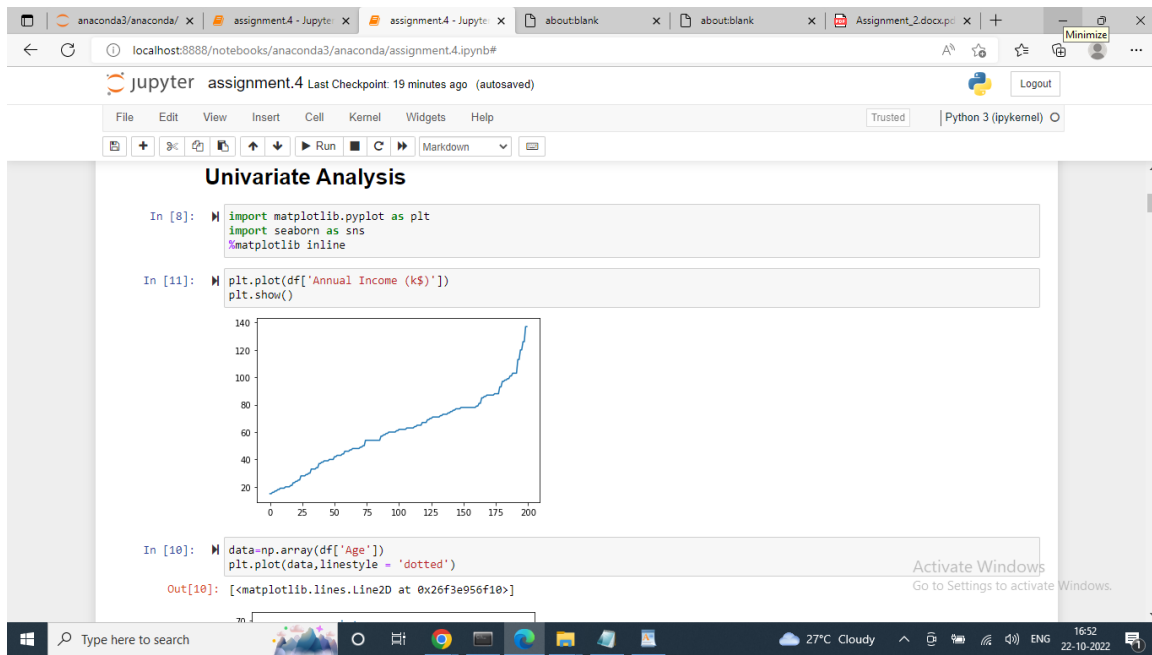
```
In [7]: df=pd.read_csv('Mall_Customers.csv')
df.head()
```

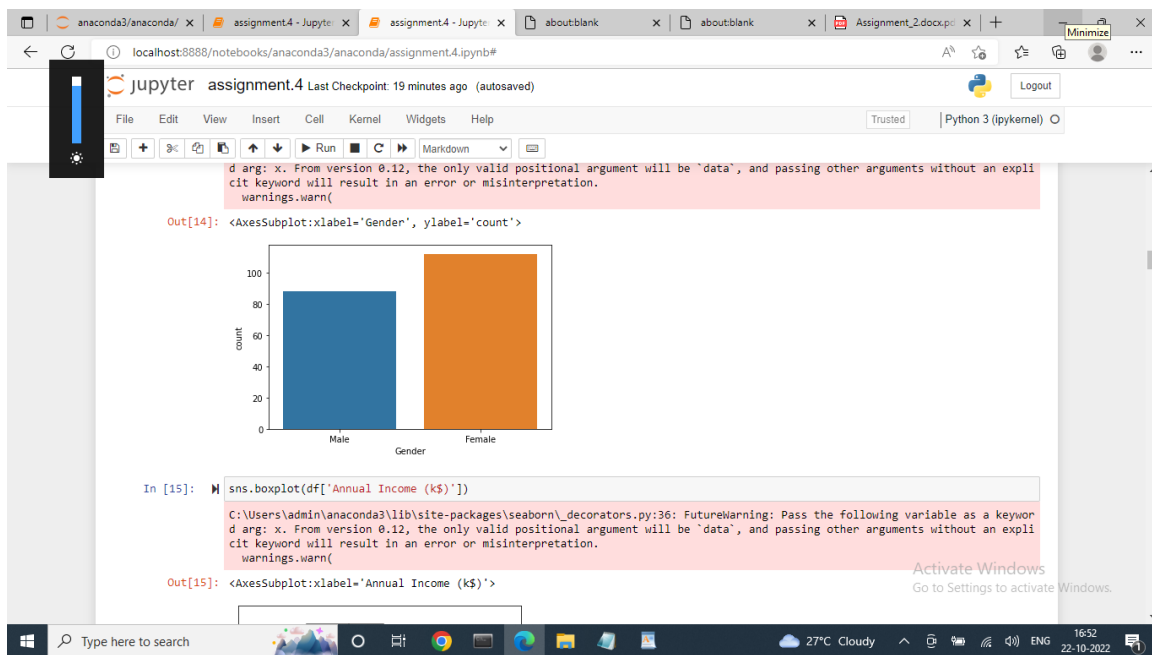
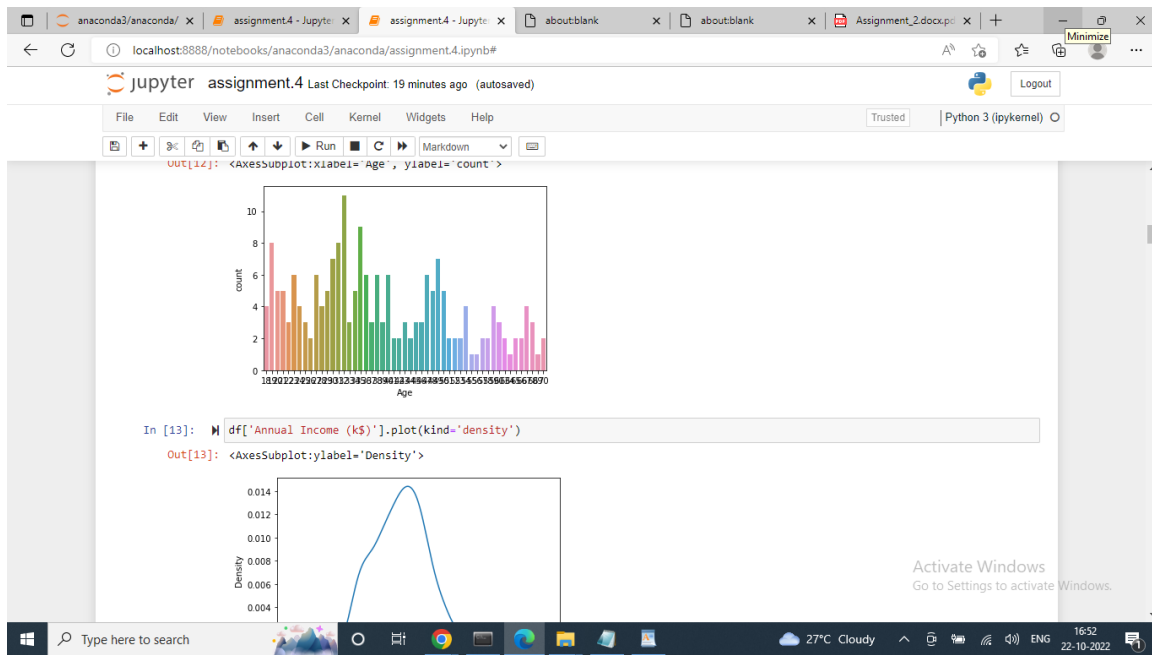
Out[7]:

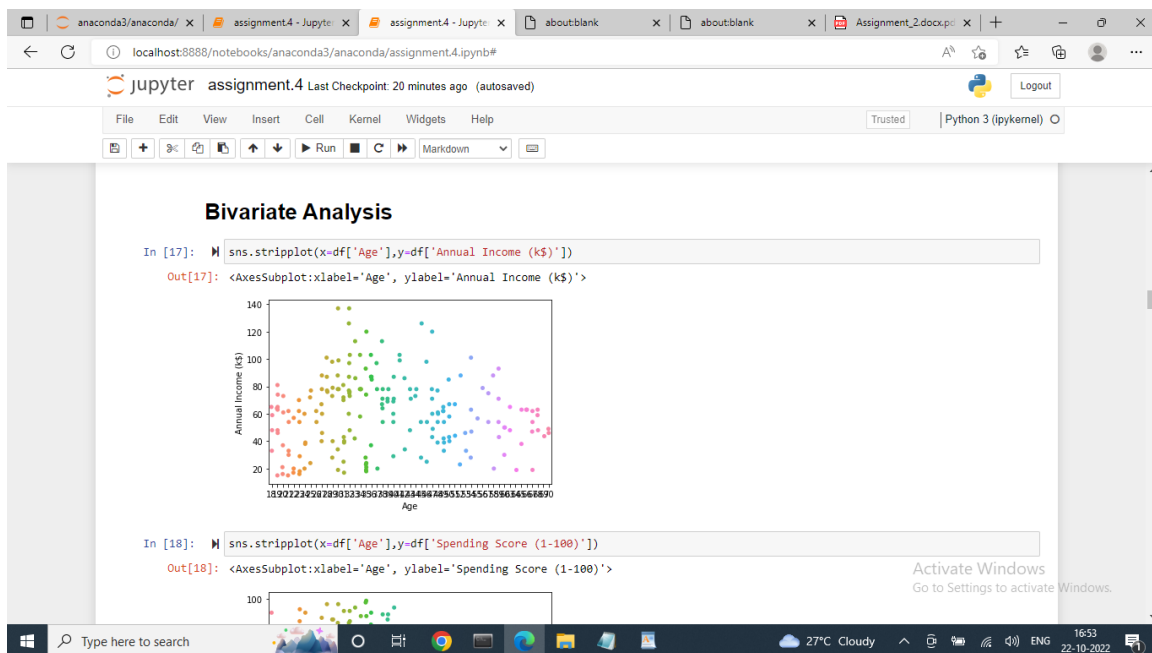
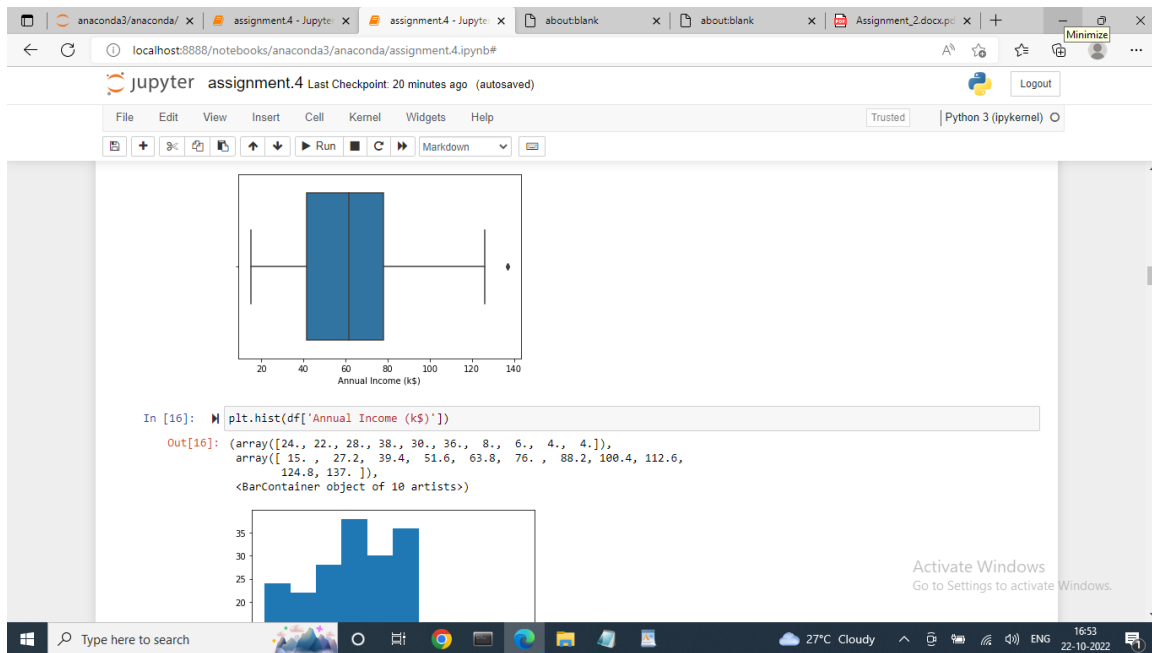
	CustomerID	Gender	Age	Annual Income (k\$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

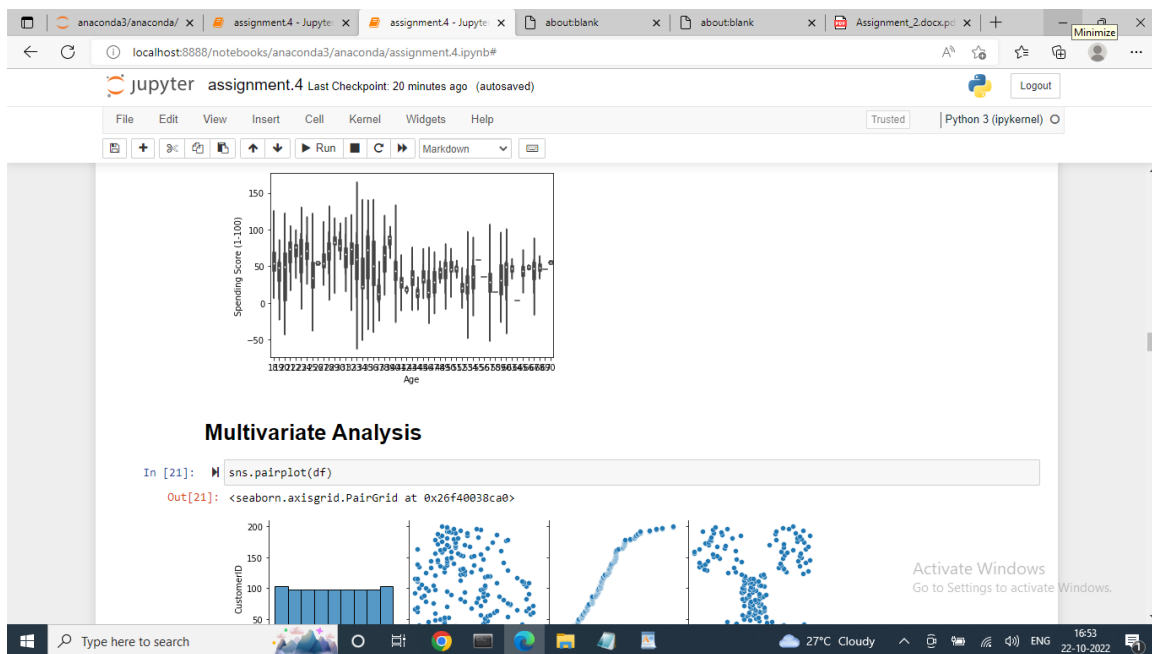
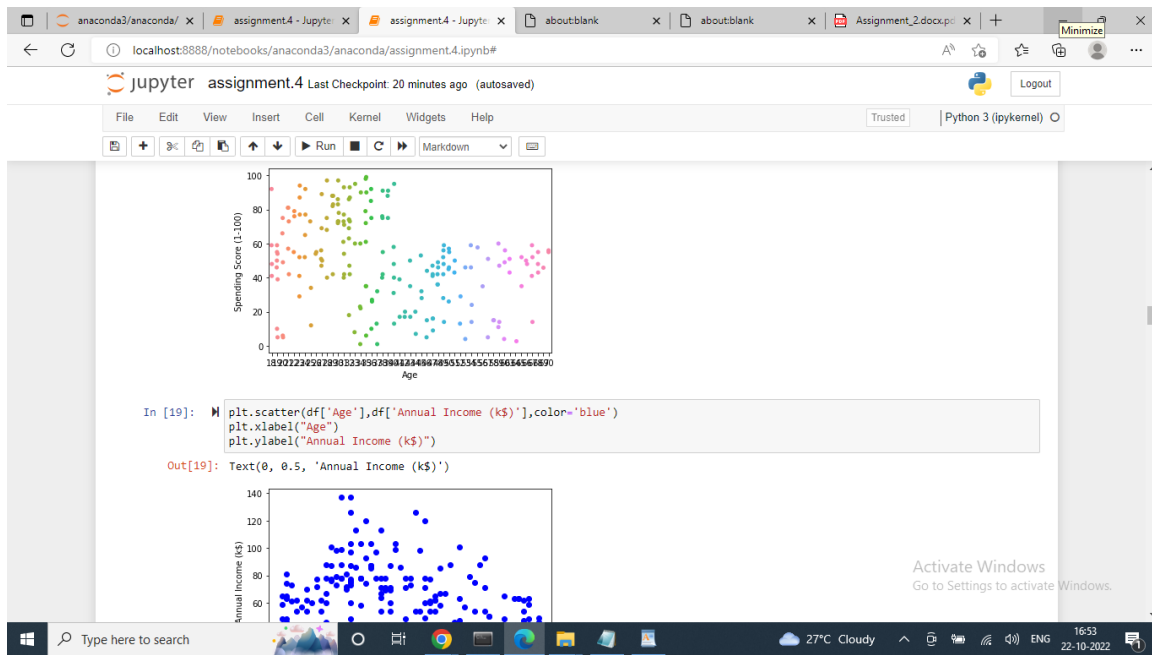
Perform Below Visualisations

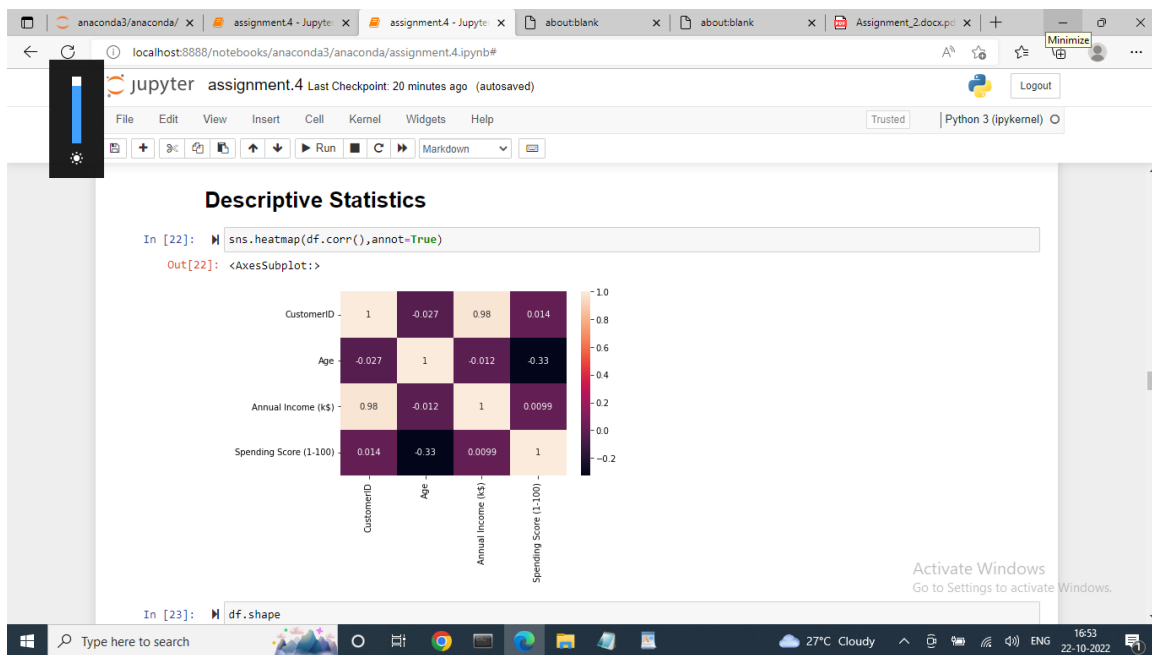
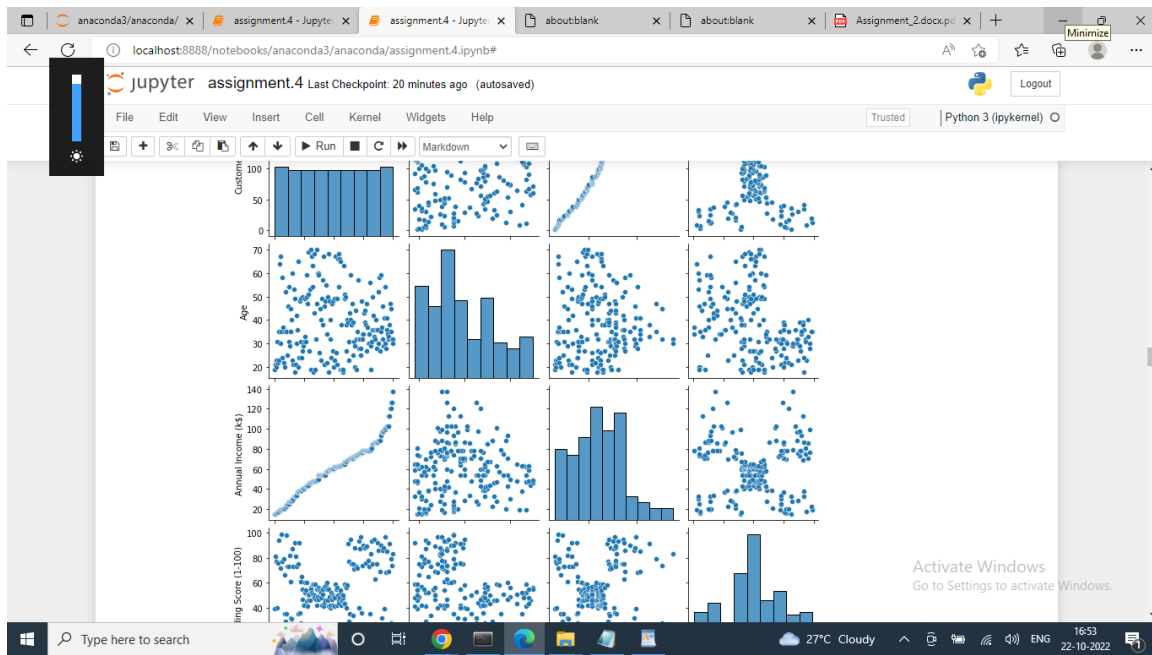
Univariate Analysis











anaconda3/anaconda/ x assignment4 - Jupyter x assignment4 - Jupyter x about:blank x about:blank x Assignment_2.docx:po 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	200.000000	200.000000	200.000000	200.000000
mean	100.500000	36.000000	61.500000	50.000000
std	99.600000	16.000000	90.000000	19.600000
min	0.000000	18.000000	18.000000	4.000000
25%	50.000000	26.000000	39.300000	39.300000
50%	100.500000	36.000000	61.500000	50.000000
75%	150.500000	46.000000	93.700000	60.700000
max	200.000000	56.000000	160.000000	96.000000

Activate Windows
Go to Settings to activate Windows.

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

C:\Users\admin\AppData\Local\Temp\ipykernel_7900\530051474.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
...

Activate Windows
Go to Settings to activate Windows.

Type here to search 27°C Cloudy 16:54 22-10-2022

anaconda3/anaconda/ x assignment4 - Jupyter: x assignment4 - Jupyter: x about:blank x about:blank x Assignment_2.docx:po 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)

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.



Type here to search 27°C Cloudy 16:54 22-10-2022

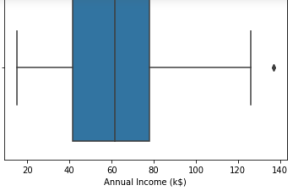

```

anaconda3/anaconda/ x assignment4 - Jupyter: x assignment4 - Jupyter: x about:blank x about:blank x Assignment_2.docx:po 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)



```
In [35]: 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[35]: <AxesSubplot:xlabel='Annual Income (k$)'

Activate Windows  
Go to Settings to activate Windows.



Type here to search 27°C Cloudy 16:54 22-10-2022


```


anaconda3/anaconda/ x assignment4 - Jupyter: x assignment4 - Jupyter: x aboutblank x aboutblank 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)

Encoding Categorical Values

```
In [37]: numerical_data = df.select_dtypes(include=[np.number])
categorical_data = df.select_dtypes(exclude=[np.number])
print("Number of numerical variables: ", numerical_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.

Type here to search 27°C Cloudy 16:54 22-10-2022

anaconda3/anaconda/ x assignment4 - Jupyter: x assignment4 - Jupyter: x aboutblank x aboutblank 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)

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  1.38981187]
[ 1.42895978  1.12815215  1.42906343 -1.36651894]
[ 1.4462805  -0.88640526  1.42906343  1.46745499]
[ 1.46360123 -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 -1.28867582]
[ 1.55020485 -0.88640526  1.61991057  1.35090831]
[ 1.56752558 -0.88640526  1.61991057 -1.05594645]
[ 1.5848463  -0.88640526  1.61991057  0.72984534]
[ 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 -1.25065425]
[ 1.7234121  1.12815215  2.91767117  1.27334719]
```

Activate Windows
Go to Settings to activate Windows.

Type here to search 27°C Cloudy 16:54 22-10-2022

anaconda3/anaconda/ x assignment4 - Jupyter: x assignment4 - Jupyter: x about:blank x about:blank x Assignment_2.docx:po 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)

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

```
[ 1.41163905 -0.88648526 1.390894 1.38981187]
[ 1.42895978 1.12815215 1.42906343 -1.36651894]
[ 1.44628805 -0.88648526 1.42906343 1.46745499]
[ 1.46360123 -0.88648526 1.46723286 -0.43480148]
[ 1.48092195 1.12815215 1.46723286 1.81684904]
[ 1.49824268 -0.88648526 1.54357172 -1.01712489]
[ 1.5155634 1.12815215 1.54357172 0.69102378]
[ 1.53288413 -0.88648526 1.61991057 -1.28887582]
[ 1.55020485 -0.88648526 1.61991057 1.35899031]
[ 1.56752558 -0.88648526 1.61991057 -1.05594645]
[ 1.5848463 -0.88648526 1.61991057 0.72984534]
[ 1.60216702 1.12815215 2.00160487 -1.63826986]
[ 1.61948775 -0.88648526 2.00160487 1.58391968]
[ 1.63680847 -0.88648526 2.26879087 -1.32769738]
[ 1.6541292 -0.88648526 2.26879087 1.11806095]
[ 1.67144992 -0.88648526 2.49780745 -0.86183865]
[ 1.68877065 1.12815215 2.49780745 0.92395314]
[ 1.70609137 1.12815215 2.91767117 -1.25005425]
[ 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	1.128152	-1.738999	-0.434801
1	-1.706091	1.128152	-1.738999	1.195704

Activate Windows
Go to Settings to activate Windows.

Type here to search 27°C Cloudy 16:54 22-10-2022

anaconda3/anaconda/ x assignment4 - Jupyter: x assignment4 - Jupyter: x about:blank x about:blank x Assignment_2.docx:po 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)

```
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	1.128152	-1.738999	-0.434801
1	-1.706091	1.128152	-1.738999	1.195704
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	0.923953
198	1.706091	1.128152	2.917671	-1.250054
199	1.723412	1.128152	2.917671	1.273347

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.

Type here to search 27°C Cloudy 16:54 22-10-2022

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)

```
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`
Out[48]: (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)`
 `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()`

Activate Windows
Go to Settings to activate Windows.

Type here to search 27°C Cloudy 16:54 22-10-2022

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)

```
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()

The Elbow Method Graph

Activate Windows
Go to Settings to activate Windows.

Type here to search 27°C Cloudy 16:54 22-10-2022

