jce5amulc

October 1, 2023

1 Kaggle Connection & DataFrame setup

[380]:
```
!pip install -q kaggle
```

[381]:
```
!mkdir ~/.kaggle
```

```
mkdir: cannot create directory ‘/root/.kaggle’: File exists
```

[382]:
```
!cp kaggle.json ~/.kaggle
```

[383]:
```
! kaggle datasets download -d vjchoudhary7/customer-segmentation-tutorial-in-python
```

```
Warning: Your Kaggle API key is readable by other users on this system! To fix
this, you can run 'chmod 600 /root/.kaggle/kaggle.json'
customer-segmentation-tutorial-in-python.zip: Skipping, found more recently
modified local copy (use --force to force download)
```

[384]:
```
!unzip /content/customer-segmentation-tutorial-in-python.zip
```

```
Archive:  /content/customer-segmentation-tutorial-in-python.zip
replace Mall_Customers.csv? [y]es, [n]o, [A]ll, [N]one, [r]ename: n
```

2 Pre-Processing

[385]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

[386]:
```
df = pd.read_csv('./Mall_Customers.csv')
df.head()
```

[386]:
```
   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
```

[387]:

```
df.describe()
```

[387]:

```
       CustomerID         Age  Annual Income (k$)  Spending Score (1-100)
count  200.000000  200.000000          200.000000              200.000000
mean   100.500000   38.850000           60.560000               50.200000
std     57.879185   13.969007           26.264721               25.823522
min      1.000000   18.000000           15.000000                1.000000
25%     50.750000   28.750000           41.500000               34.750000
50%    100.500000   36.000000           61.500000               50.000000
75%    150.250000   49.000000           78.000000               73.000000
max    200.000000   70.000000          137.000000               99.000000
```

[388]:

```
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
```

[389]:

```
df.isnull().values.any()
```

[389]:

```
False
```

[390]:

```
df.shape
```

[390]:

```
(200, 5)
```

[391]:

```
# Dropping 'CustomerID' as it has no impact or connection to dataset or data␣
 ↪values
df.drop(['CustomerID'], axis=1, inplace=True)
```

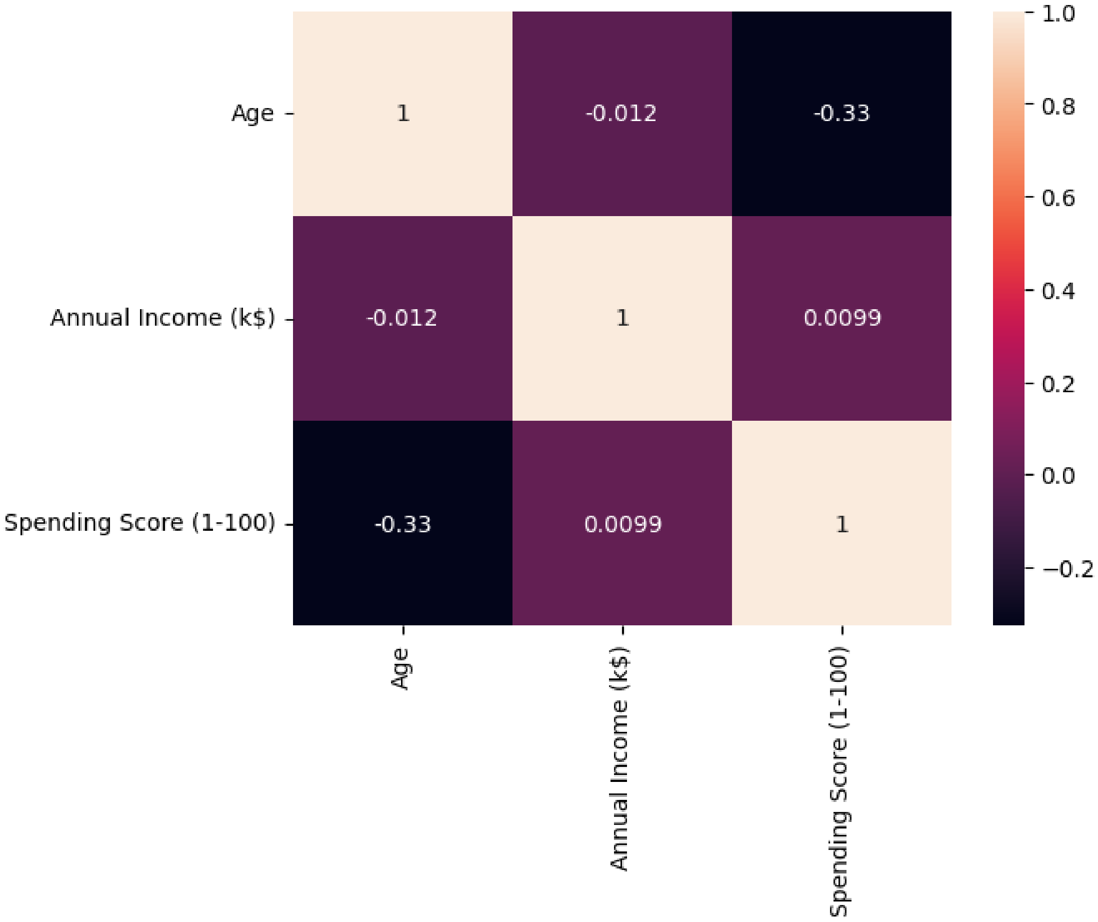
[392]:

```
sns.heatmap(df.corr(), annot=True)
```

```
<ipython-input-392-6dc1c4c1753e>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
```

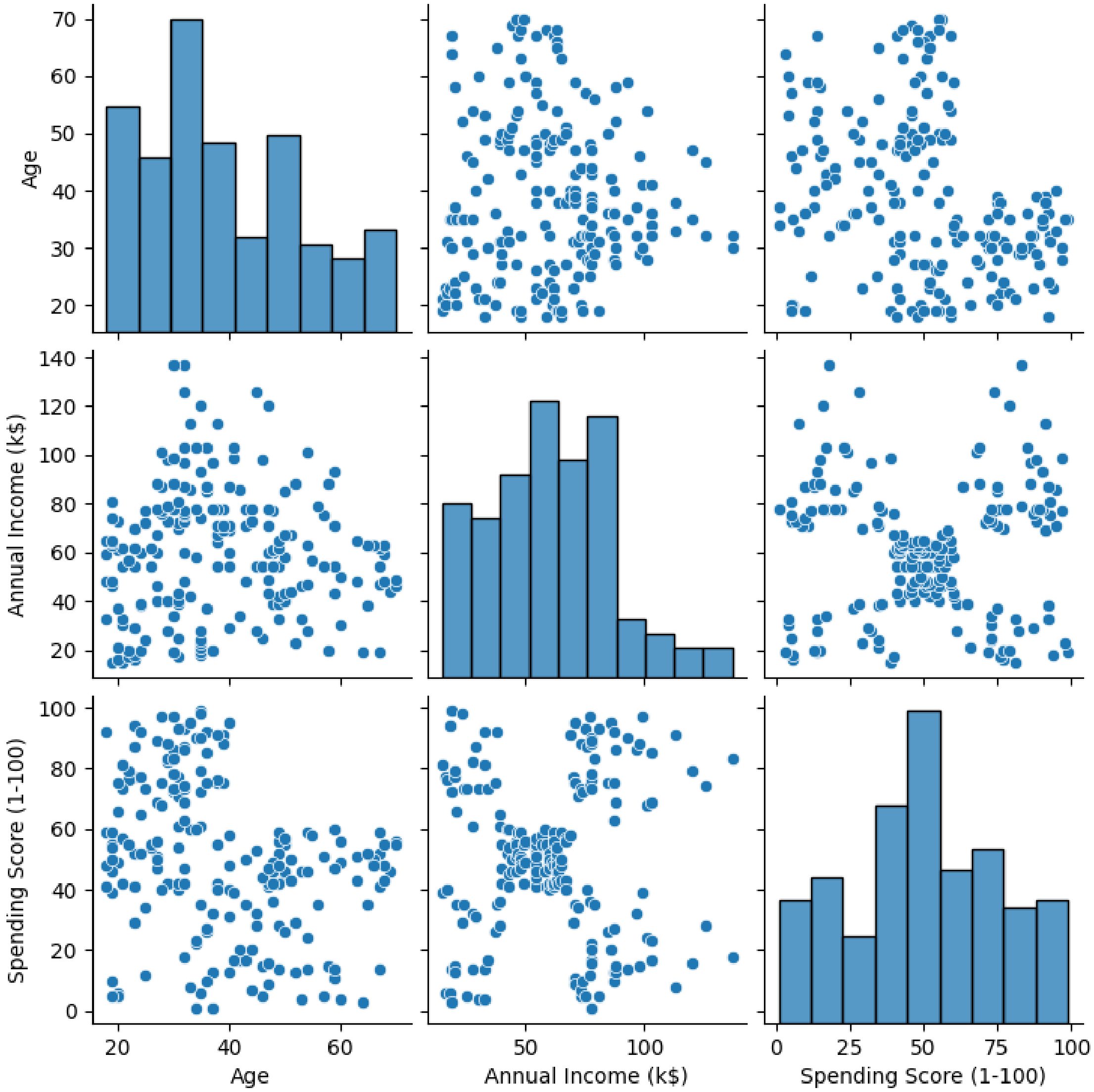
```
to silence this warning.
  sns.heatmap(df.corr(), annot=True)
```

[392]:
```
<Axes: >
```



[393]:
```
sns.pairplot(df)
```

[393]:
```
<seaborn.axisgrid.PairGrid at 0x7c71dd8216f0>
```



3 Converting Categorical Data (Columns) to Numerical

[394]:
```
df['Gender'].value_counts()
```

[394]:
```
Female    112
Male       88
Name: Gender, dtype: int64
```

[395]:
```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
```

[396]:

```
# Label Encoding 'Gender' column
# '1' == 'Male' && 0 == 'Female'
df['Gender'] = le.fit_transform(df.Gender)
```

[397]:

```
df.head()
```

[397]:

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

[398]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 4 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Gender                  200 non-null    int64
 1   Age                     200 non-null    int64
 2   Annual Income (k$)      200 non-null    int64
 3   Spending Score (1-100)  200 non-null    int64
dtypes: int64(4)
memory usage: 6.4 KB
```

4 Scaling Dataset values for uniformity

[399]:

```
from sklearn.preprocessing import MinMaxScaler
```

[400]:

```
df = pd.DataFrame(MinMaxScaler().fit_transform(df), columns=df.columns)
```

[401]:

```
df.head()
```

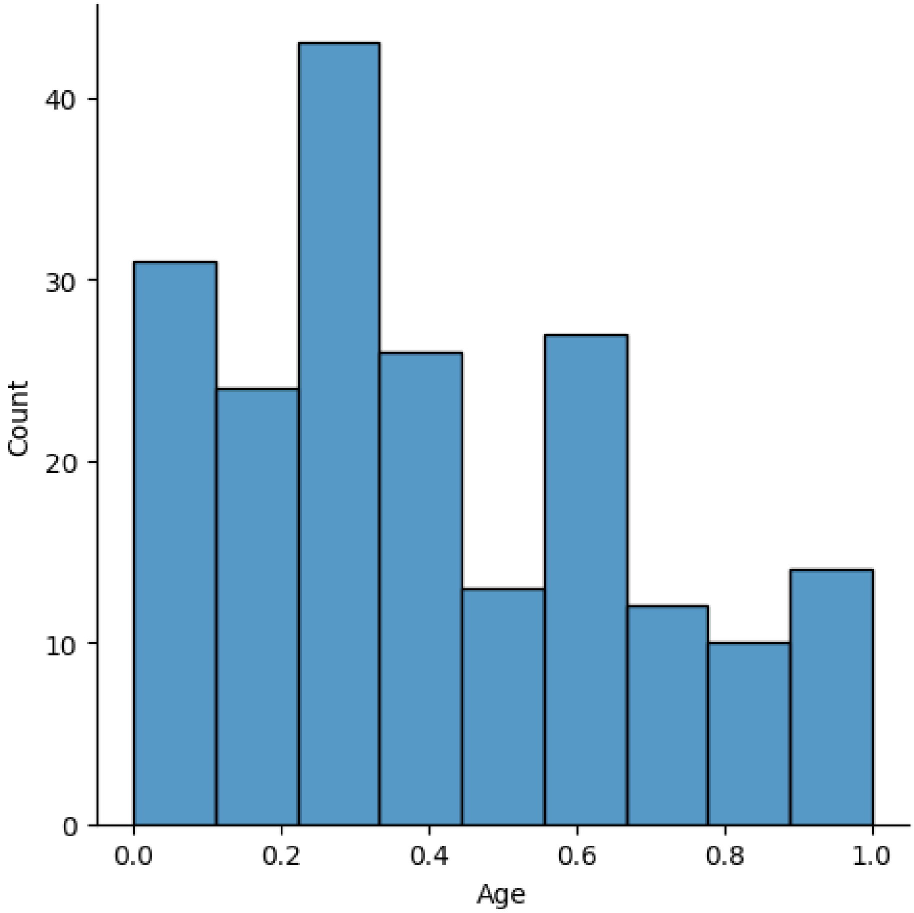
[401]:

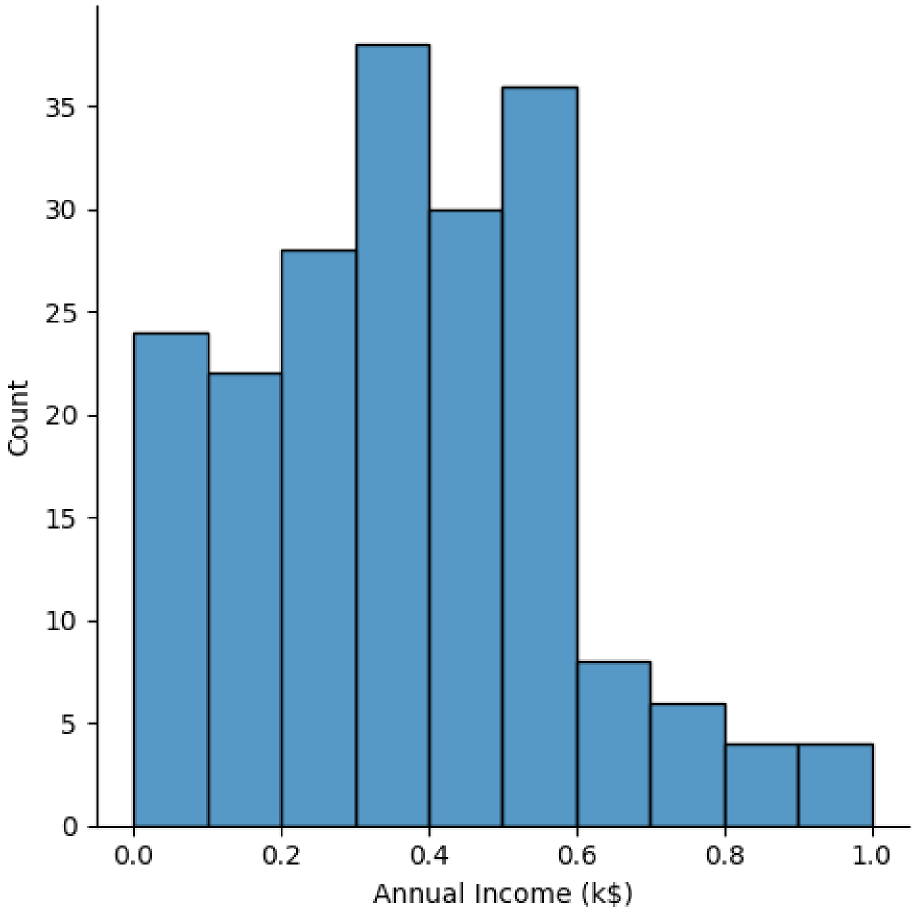
```
   Gender       Age  Annual Income (k$)  Spending Score (1-100)
0     1.0  0.019231            0.000000                0.387755
1     1.0  0.057692            0.000000                0.816327
2     0.0  0.038462            0.008197                0.051020
3     0.0  0.096154            0.008197                0.775510
4     0.0  0.250000            0.016393                0.397959
```

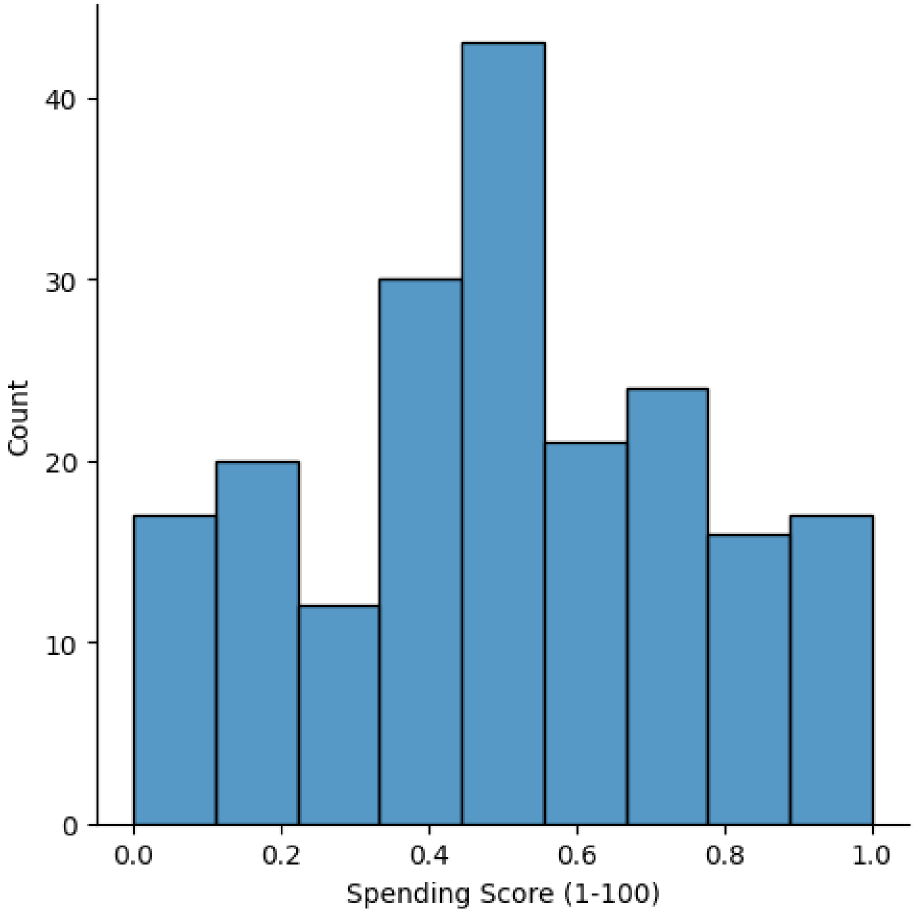
5 Data Analysis, Outlier Detection & Outlier Elimination

```
[402]: sns.displot(df['Age'])
       sns.displot(df['Annual Income (k$)'])
       sns.displot(df['Spending Score (1-100)'])
```

```
[402]: <seaborn.axisgrid.FacetGrid at 0x7c71db8662f0>
```



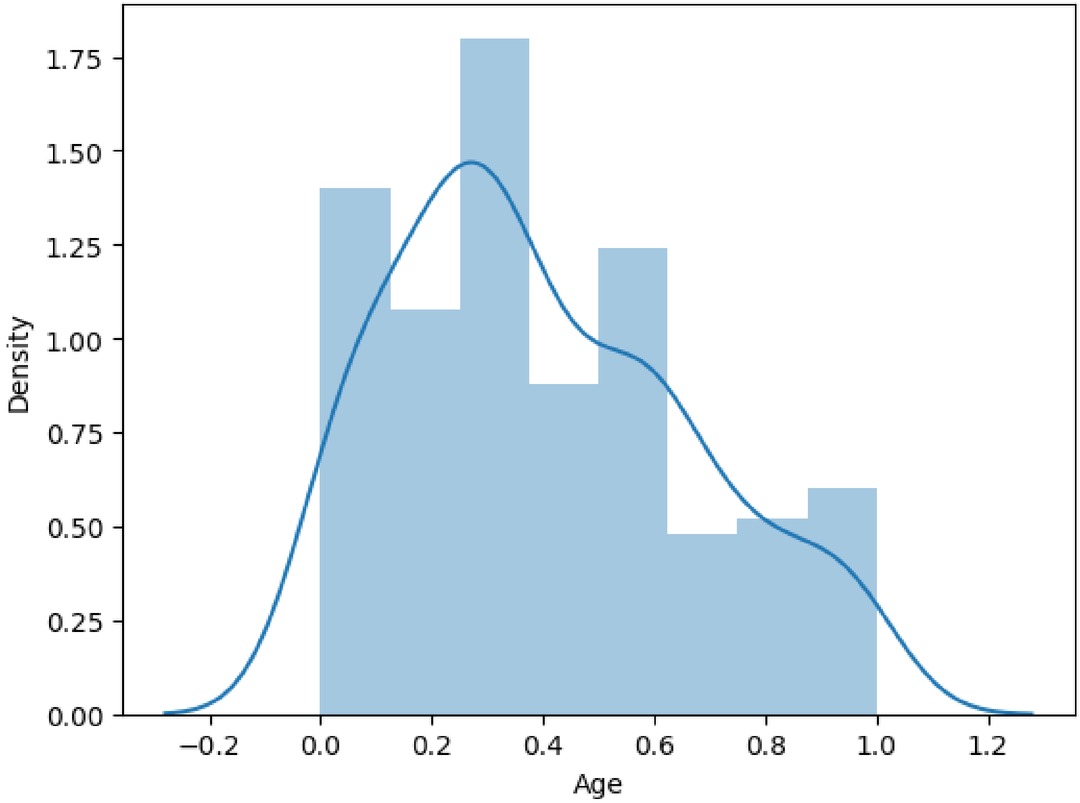

Count
0
5
10
15
20
25
30
35
0.0
0.2
0.4
0.6
0.8
1.0
Annual Income (k$)



[403]:
```
sns.distplot(df['Age'])
```

```
<ipython-input-403-0fafe04ea3f6>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df['Age'])
```

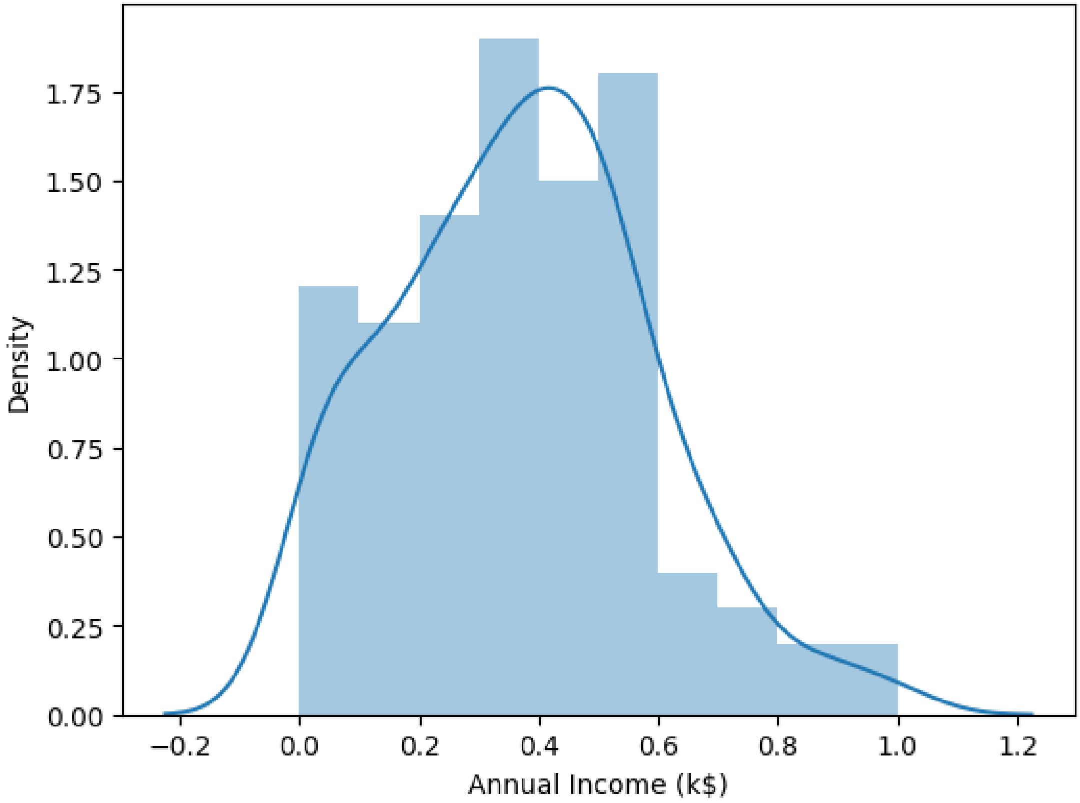
[403]:
```
<Axes: xlabel='Age', ylabel='Density'>
```



```
[404]: sns.distplot(df['Annual Income (k$)'])
```

```
<ipython-input-404-5c9bfeb4bab1>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df['Annual Income (k$)'])
```

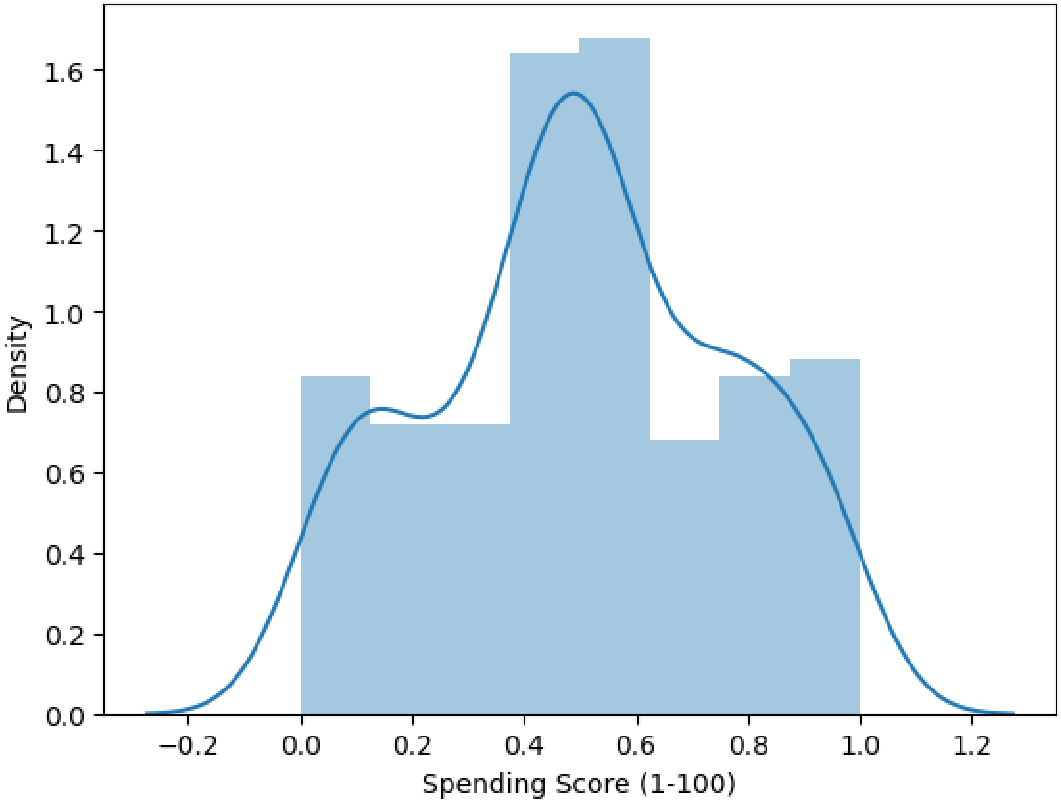
```
[404]: <Axes: xlabel='Annual Income (k$)', ylabel='Density'>
```



[405]:
```
sns.distplot(df['Spending Score (1-100)'])
```

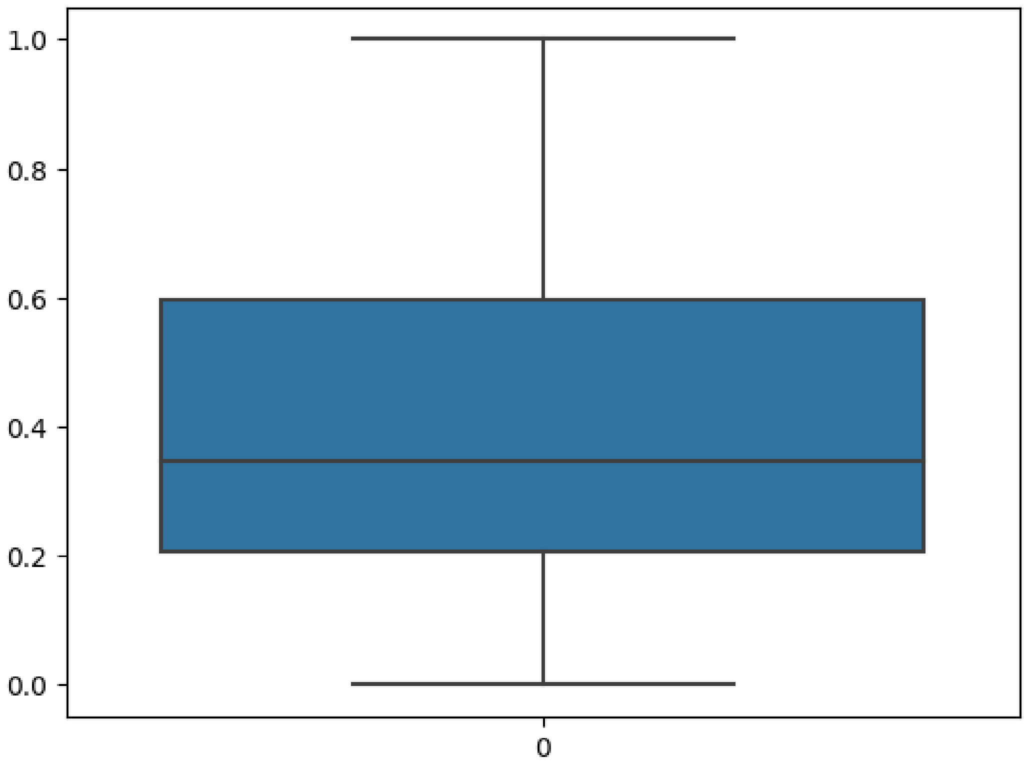
```
<ipython-input-405-beed7b40d5ab>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df['Spending Score (1-100)'])
```

[405]:
```
<Axes: xlabel='Spending Score (1-100)', ylabel='Density'>
```



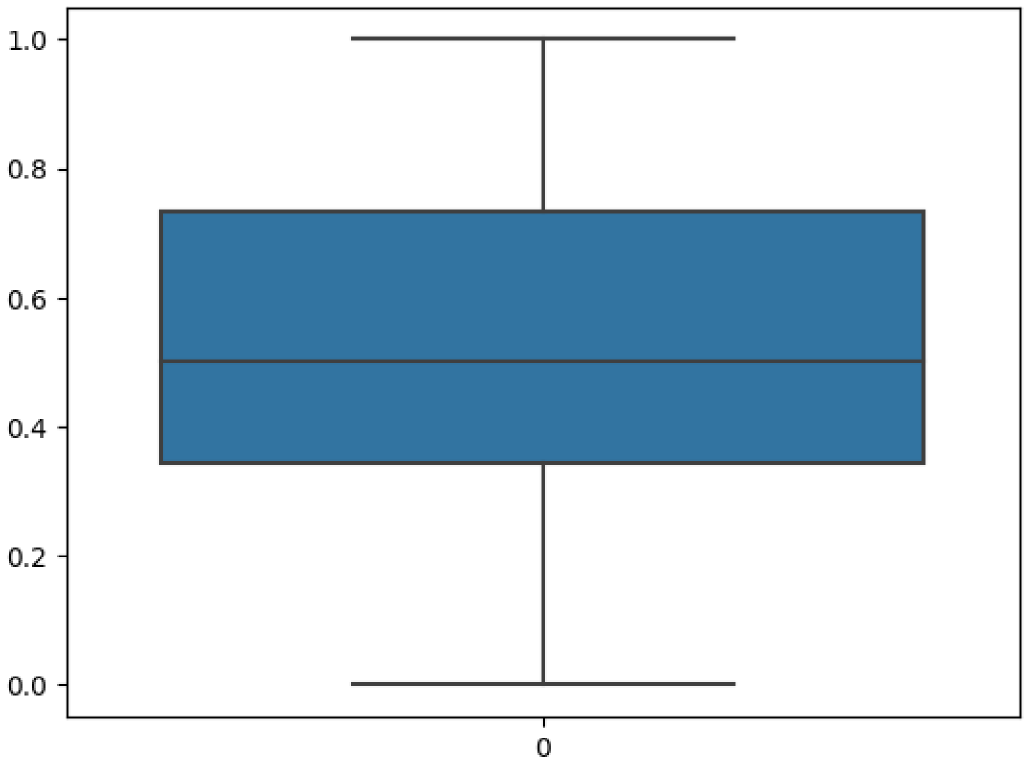
[406]:
```
sns.boxplot(df.Age)
```

[406]:
```
<Axes: >
```



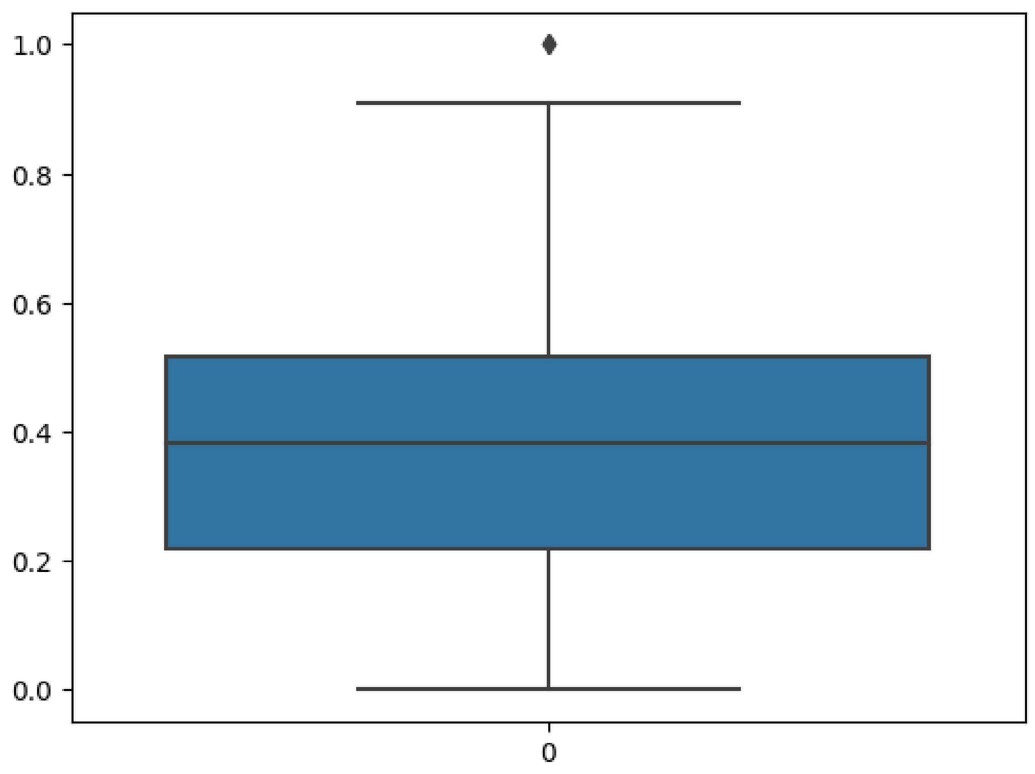
[407]:
```
sns.boxplot(df['Spending Score (1-100)'])
```

[407]:
```
<Axes: >
```



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

```
[408]: <Axes: >
```



[409]:
```
Q1 = df['Annual Income (k$)'].quantile(0.25)
Q3 = df['Annual Income (k$)'].quantile(0.75)
```

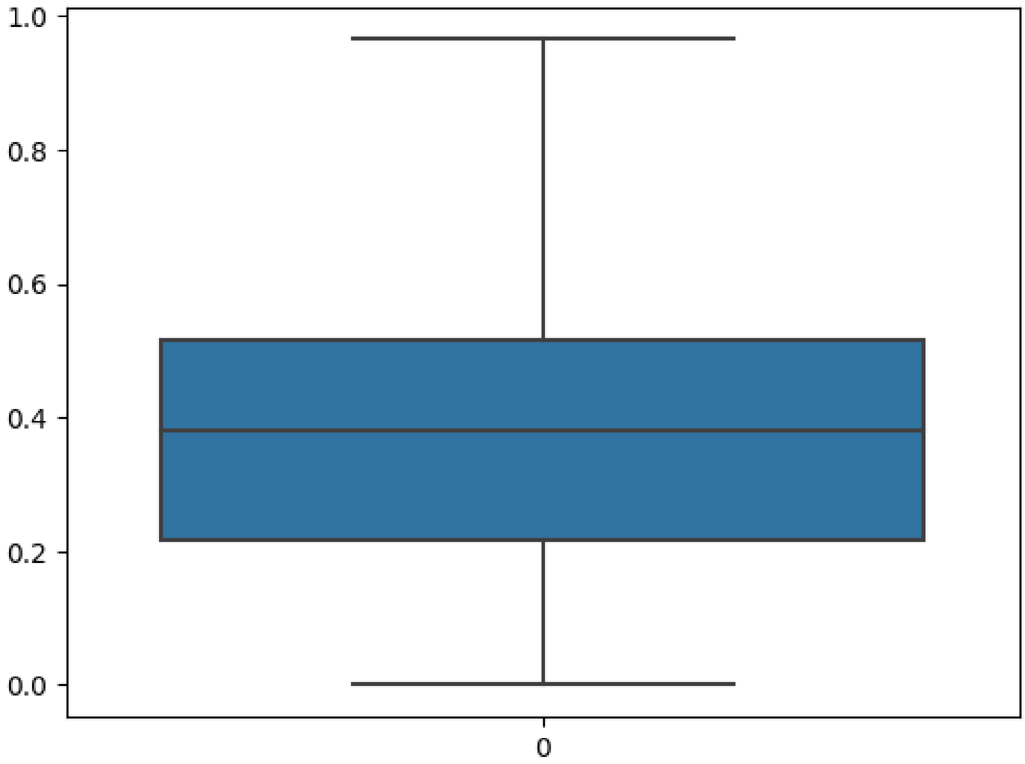
[410]:
```
IQR = Q3 - Q1
whisker_width = 1.5
```

[411]:
```
lower_whisker = Q1 - (whisker_width*IQR)
upper_whisker = Q3 + (whisker_width*IQR)
```

[412]:
```
df['Annual Income (k$)'] = np.where(df['Annual Income (k$)'] > upper_whisker,␣
 ↪upper_whisker, np.where(df['Annual Income (k$)'] < lower_whisker,␣
 ↪lower_whisker, df['Annual Income (k$)']))
```

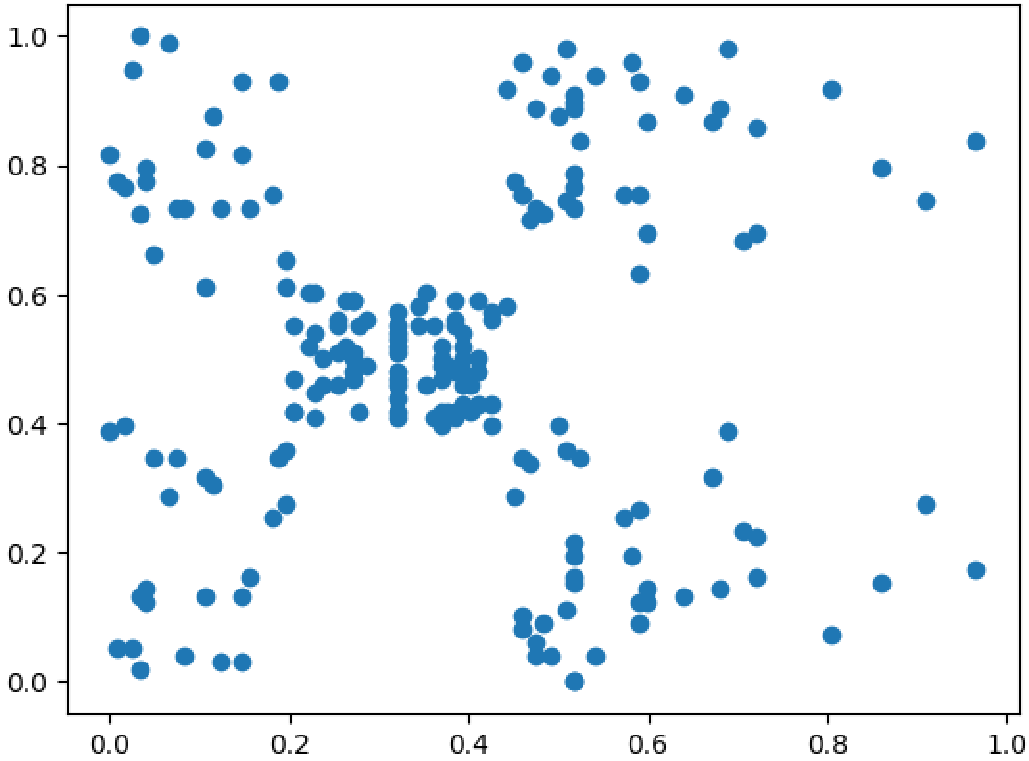
[413]:
```
sns.boxplot(df['Annual Income (k$)'])
```

[413]:
```
<Axes: >
```



[414]:
```
plt.scatter(df['Annual Income (k$)'], df['Spending Score (1-100)'])
```

[414]:
```
<matplotlib.collections.PathCollection at 0x7c71db350670>
```



```
[415]: X_train = df.drop(['Spending Score (1-100)'], axis=1)
       Y_train = df['Spending Score (1-100)']
```

```
[416]: X_train.head(), Y_train.head()
```

```
[416]: (   Gender       Age  Annual Income (k$)
        0     1.0  0.019231            0.000000
        1     1.0  0.057692            0.000000
        2     0.0  0.038462            0.008197
        3     0.0  0.096154            0.008197
        4     0.0  0.250000            0.016393,
        0    0.387755
        1    0.816327
        2    0.051020
        3    0.775510
        4    0.397959
        Name: Spending Score (1-100), dtype: float64)
```

6 Finding Elbow Point (Possible 'K' value)

```
[417]: from sklearn.cluster import KMeans
```

```
[418]: k_rng = range(1,40)
        sse = []

        for k in k_rng:
          km = KMeans(n_clusters=k)
          km.fit(df[['Annual Income (k$)', 'Spending Score (1-100)']])
          sse.append(km.inertia_)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
```

```
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
```

```
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
```

```
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
```

[419]:

```
sse
```

[419]:

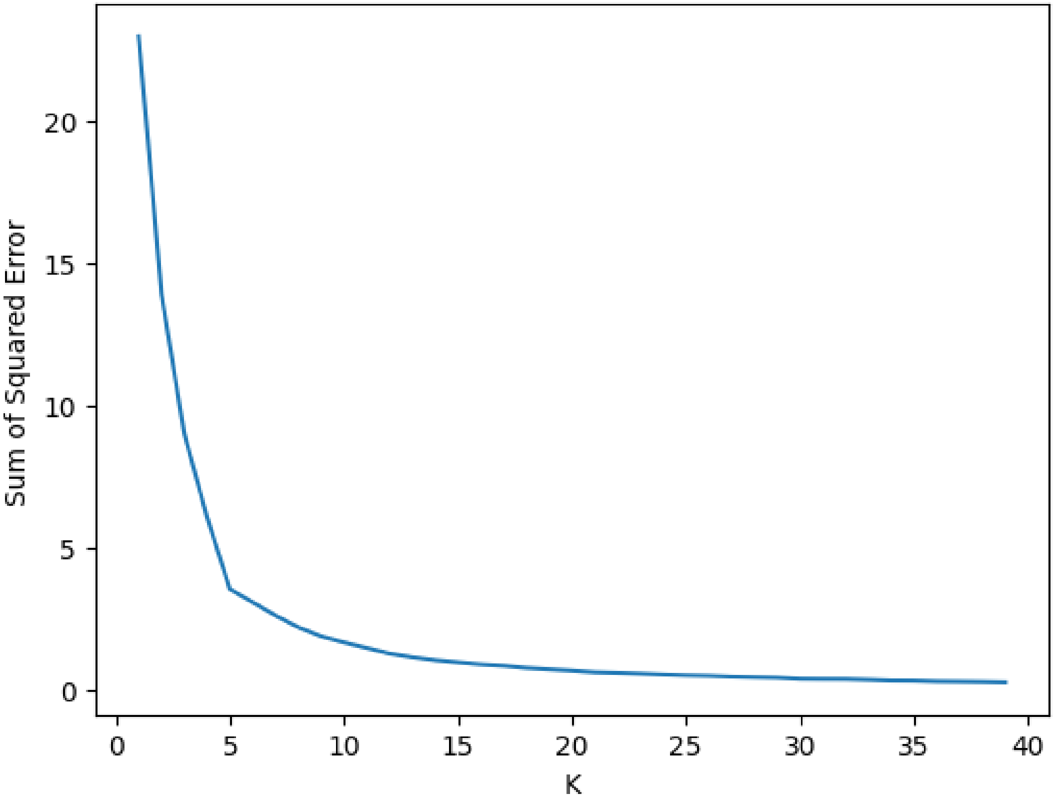
```
[22.955815982449476,
 13.908672064927337,
 8.995430225375209,
 6.055304436065896,
 3.5287926379212995,
 3.0667030475951083,
 2.603378920756281,
 2.187885090648535,
 1.870681765951311,
 1.65554733172543,
 1.4601876173540953,
 1.2648508391006437,
 1.1345272740631709,
 1.0315255123213856,
 0.9552656259840362,
 0.8832398005986595,
 0.8364995074086296,
 0.7597145770475486,
 0.7125788697448221,
 0.6661188281903597,
 0.6081924334679754,
 0.5868774306356352,
 0.5586492443579925,
 0.5255947055689602,
 0.4992976622795663,
 0.48431656019645386,
 0.45355573875679756,
 0.4349334492297876,
 0.4189017353108364,
 0.37528629789300777,
 0.369173284144015,
 0.3668613072634828,
 0.3457038348766507,
 0.31685030651587986,
 0.31137225211091013,
 0.287577555697374,
 0.27961820956283134,
 0.2697865474553408,
 0.2549457476255709]
```

[420]:

```
plt.xlabel('K')
plt.ylabel('Sum of Squared Error')
plt.plot(k_rng, sse)
```

[420]:

```
[<matplotlib.lines.Line2D at 0x7c71db3d0070>]
```



```
[421]: kmeans = KMeans(n_clusters=5)
       kmeans
```

```
[421]: KMeans(n_clusters=5)
```

```
[422]: z = kmeans.fit_predict(df[['Annual Income (k$)','Spending Score (1-100)']])
       z
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
```

```
[422]: array([3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4,
              3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 0,
              3, 4, 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, 2, 1, 2, 0, 2, 1, 2, 1, 2,
              0, 2, 1, 2, 1, 2, 1, 2, 1, 2, 0, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2,
              1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2,
              1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2,
              1, 2], dtype=int32)
```

[423]:

```
df_2 = df
```

[424]:

```
df_2['Cluster'] = z
df_2.head()
```

[424]:

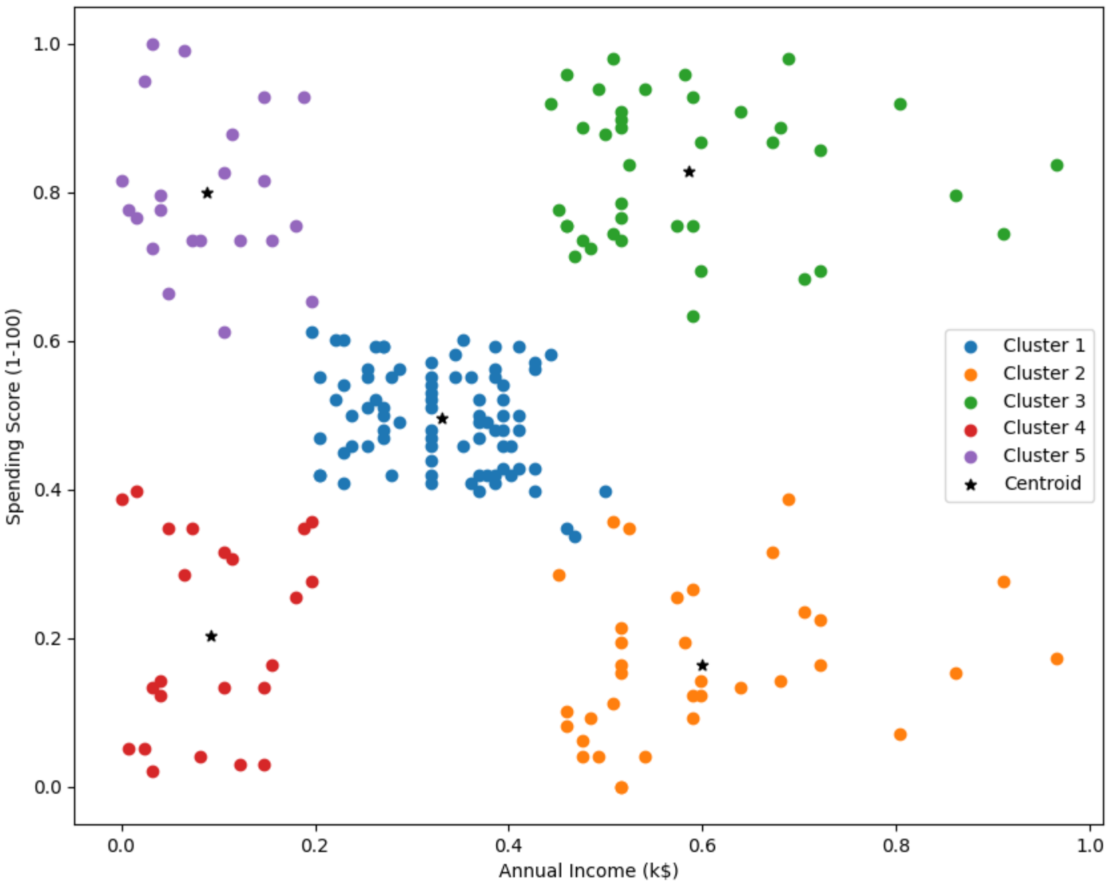
```
   Gender       Age  Annual Income (k$)  Spending Score (1-100)  Cluster
0     1.0  0.019231            0.000000                0.387755        3
1     1.0  0.057692            0.000000                0.816327        4
2     0.0  0.038462            0.008197                0.051020        3
3     0.0  0.096154            0.008197                0.775510        4
4     0.0  0.250000            0.016393                0.397959        3
```

[425]:

```
plt.figure(figsize=(10,8))

df1 = df_2[df_2.Cluster==0]
df2 = df_2[df_2.Cluster==1]
df3 = df_2[df_2.Cluster==2]
df4 = df_2[df_2.Cluster==3]
df5 = df_2[df_2.Cluster==4]


plt.scatter(df1['Annual Income (k$)'], df1['Spending Score (1-100)'],␣
 ↪label='Cluster 1')
plt.scatter(df2['Annual Income (k$)'], df2['Spending Score (1-100)'],␣
 ↪label='Cluster 2')
plt.scatter(df3['Annual Income (k$)'], df3['Spending Score (1-100)'],␣
 ↪label='Cluster 3')
plt.scatter(df4['Annual Income (k$)'], df4['Spending Score (1-100)'],␣
 ↪label='Cluster 4')
plt.scatter(df5['Annual Income (k$)'], df5['Spending Score (1-100)'],␣
 ↪label='Cluster 5')

plt.scatter(kmeans.cluster_centers_[:,0], kmeans.cluster_centers_[:,1],␣
 ↪color='black', marker='*', label='Centroid')

plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
```

[425]:

```
<matplotlib.legend.Legend at 0x7c71db2697e0>
```



7 Train - Test Split

```
[426]: df.drop(['Cluster'], axis=1, inplace=True)
```

```
[427]: df.head()
```

```
[427]:    Gender       Age  Annual Income (k$)  Spending Score (1-100)
       0     1.0  0.019231            0.000000                0.387755
       1     1.0  0.057692            0.000000                0.816327
       2     0.0  0.038462            0.008197                0.051020
       3     0.0  0.096154            0.008197                0.775510
       4     0.0  0.250000            0.016393                0.397959
```

```
[428]: from sklearn.model_selection import train_test_split
```

```
[429]: X_train = df.drop(['Spending Score (1-100)'], axis=1)
       Y_train = df['Spending Score (1-100)']
```

[430]:
```
# X_train, Y_train = train_test_split(df, test_size=0.5)
```

[431]:
```
# print(X_train.shape)
# print(Y_train.shape)
```

[432]:
```
# print(X_train.head())
```

8 Logistic Regression Modeling

[443]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
```

[444]:
```
lr = LogisticRegression()
svc = SVC()
```

[435]:
```
xtest, xtrain, ytest, ytrain = train_test_split(X_train, Y_train, test_size=0.
 ↪25)
```

[437]:
```
xtest.shape, xtrain.shape
```

[437]:
```
((150, 3), (50, 3))
```

[441]:
```
ytest.shape, ytrain.shape
```

[441]:
```
((150,), (50,))
```

I don't understand what to do next as I am not able to find the solution to the error

[447]:
```
lr.fit(xtrain, ytrain)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-447-dad6f18d5f54> in <cell line: 1>()
----> 1 lr.fit(xtrain, ytrain)

/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py in␣
 ↪fit(self, X, y, sample_weight)
   1202                 accept_large_sparse=solver not in ["liblinear", "sag",␣
 ↪"saga"],
   1203             )
-> 1204             check_classification_targets(y)
   1205             self.classes_ = np.unique(y)
   1206

/usr/local/lib/python3.10/dist-packages/sklearn/utils/multiclass.py in␣
 ↪check_classification_targets(y)
```

```
    216         "multilabel-sequences",
    217     ]:
--> 218         raise ValueError("Unknown label type: %r" % y_type)
    219 
    220 

ValueError: Unknown label type: 'continuous'
```

```
[ ]:
```