practical-3

May 9, 2025

```
[1]: import pandas as pd
     import statistics as st
```

```
[2]: df=pd.read_csv("Mall_Customers.csv")
```

```
[3]: df
```

```
[3]:      CustomerID   Genre  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
     ..          …     …   …                 …                     …
     195         196  Female   35                 120                      79
     196         197  Female   45                 126                      28
     197         198    Male   32                 126                      74
     198         199    Male   32                 137                      18
     199         200    Male   30                 137                      83

     [200 rows x 5 columns]
```

```
[4]: df.mean(numeric_only=True)
```

```
[4]: CustomerID                100.50
     Age                        38.85
     Annual Income (k$)         60.56
     Spending Score (1-100)     50.20
     dtype: float64
```

```
[5]: df.loc[:,'Age'].mean()
```

```
[5]: np.float64(38.85)
```

```
[6]: df.mean(axis=1,numeric_only=True)[0:4]
```

```
[6]: 0    18.50
     1    29.75
     2    11.25
     3    30.00
     dtype: float64
```

```
[7]: df.mean(axis=1,numeric_only=True)[0:4]
```

```
[7]: 0    18.50
     1    29.75
     2    11.25
     3    30.00
     dtype: float64
```

```
[8]: df.median(numeric_only=True)
```

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

```
[9]: df.loc[:,'Age'].median()
```

```
[9]: np.float64(36.0)
```

```
[10]: df.median(axis=1,numeric_only=True)[0:4]
```

```
[10]: 0    17.0
      1    18.0
      2    11.0
      3    19.5
      dtype: float64
```

```
[11]: df.mode()
```

```
[11]:      CustomerID   Genre   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
```

```
199         200      NaN    NaN                   NaN                       NaN

[200 rows x 5 columns]
```

[12]:
```
df.loc[:,'Age'].mode()
```

[12]:
```
0    32
Name: Age, dtype: int64
```

[13]:
```
df.min()
```

[13]:
```
CustomerID                     1
Genre                     Female
Age                           18
Annual Income (k$)            15
Spending Score (1-100)         1
dtype: object
```

[14]:
```
df.loc[:,'Age'].min(skipna = False)
```

[14]:
```
np.int64(18)
```

[15]:
```
df.max()
```

[15]:
```
CustomerID                 200
Genre                     Male
Age                         70
Annual Income (k$)         137
Spending Score (1-100)      99
dtype: object
```

[16]:
```
df.loc[:,'Age'].max(skipna = False)
```

[16]:
```
np.int64(70)
```

[17]:
```
df.std(numeric_only=True)
```

[17]:
```
CustomerID                57.879185
Age                       13.969007
Annual Income (k$)        26.264721
Spending Score (1-100)    25.823522
dtype: float64
```

[18]:
```
df.loc[:,'Age'].std()
```

[18]:
```
np.float64(13.96900733155888)
```

[19]:
```
df.std(axis=1,numeric_only=True)[0:4]
```

[19]:
```
0    15.695010
1    35.074920
2     8.057088
3    32.300671
dtype: float64
```

[20]:
```
df.groupby(['Genre'])['Age'].mean()
```

[20]:
```
Genre
Female    38.098214
Male      39.806818
Name: Age, dtype: float64
```

[21]:
```
df_u = df.rename(columns={'Annual Income k$': 'Income'}, inplace=False)
```

[22]:
```
df_u.groupby('Genre')['Annual Income (k$)'].mean()
```

[22]:
```
Genre
Female    59.250000
Male      62.227273
Name: Annual Income (k$), dtype: float64
```

[23]:
```
print(df.columns)
```

```
Index(['CustomerID', 'Genre', 'Age', 'Annual Income (k$)',
       'Spending Score (1-100)'],
      dtype='object')
```

[24]:
```
from sklearn import preprocessing
enc = preprocessing.OneHotEncoder()
enc_df = pd.DataFrame(enc.fit_transform(df[['Genre']]).toarray())
enc_df
```

[24]:
```
       0    1
0    0.0  1.0
1    0.0  1.0
2    1.0  0.0
3    1.0  0.0
4    1.0  0.0
..   …    …
195  1.0  0.0
196  1.0  0.0
197  0.0  1.0
198  0.0  1.0
199  0.0  1.0
```

```
[200 rows x 2 columns]
```

```
[25]: df_encode =df_u.join(enc_df)
      df_encode
```

```
[25]:
     CustomerID   Genre  Age  Annual Income (k$)  Spending Score (1-100)    0  \
0             1    Male   19                  15                      39  0.0
1             2    Male   21                  15                      81  0.0
2             3  Female   20                  16                       6  1.0
3             4  Female   23                  16                      77  1.0
4             5  Female   31                  17                      40  1.0
..          …     …  …                 …                     …  …
195         196  Female   35                 120                      79  1.0
196         197  Female   45                 126                      28  1.0
197         198    Male   32                 126                      74  0.0
198         199    Male   32                 137                      18  0.0
199         200    Male   30                 137                      83  0.0

       1
0    1.0
1    1.0
2    0.0
3    0.0
4    0.0
..   …
195  0.0
196  0.0
197  1.0
198  1.0
199  1.0

[200 rows x 7 columns]
```

```
[ ]:
```