

Assignment no.- 03

Aim-

1. Summary statistics
2. Types of Variables
3. Summary statistics of income grouped by the age groups
4. Display basic statistical details on the iris dataset.

```
In [1]: import pandas as pd
import numpy as np
```

```
In [30]: df=pd.read_csv("C:\\Users\\SS0S03\\Desktop\\data.csv")
```

```
In [31]: df
```

```
Out[31]:
```

| | Unnamed: 0 | customer id | age | income | spending score |
|----|------------|-------------|------|--------|----------------|
| 0 | 0 | 1 | 19.0 | 42.0 | NaN |
| 1 | 1 | 2 | 20.0 | NaN | 55.0 |
| 2 | 2 | 3 | 28.0 | NaN | NaN |
| 3 | 3 | 4 | 29.0 | NaN | NaN |
| 4 | 4 | 5 | 23.0 | NaN | NaN |
| 5 | 5 | 6 | 23.0 | NaN | NaN |
| 6 | 6 | 7 | NaN | NaN | NaN |
| 7 | 7 | 8 | 32.0 | NaN | NaN |
| 8 | 8 | 9 | 43.0 | NaN | NaN |
| 9 | 9 | 10 | 36.0 | NaN | NaN |
| 10 | 10 | 11 | NaN | NaN | NaN |
| 11 | 11 | 12 | 20.0 | NaN | NaN |

```
In [32]: df.mean()
```

```
Out[32]: Unnamed: 0      26.000
customer id      27.000
age              32.425
income           42.000
spending score   55.000
dtype: float64
```

```
In [39]: df.loc[:, 'age '].mean()
```

```
Out[39]: 32.425
```

```
In [40]: df.mean(axis=1)[0:4]
```

```
Out[40]: 0      15.5
1      19.5
2      11.0
3      12.0
dtype: float64
```

```
In [41]: df.median()
```

```
Out[41]: Unnamed: 0      26.0  
customer id      27.0  
age              32.5  
income           42.0  
spending score   55.0  
dtype: float64
```

```
In [43]: df.loc[:, 'age '].median()
```

```
Out[43]: 32.5
```

```
In [44]: df.mode()
```

```
Out[44]:
```

| | Unnamed: 0 | customer id | age | income | spending score |
|----|------------|-------------|------|--------|----------------|
| 0 | 0 | 1 | 29.0 | 42.0 | 55.0 |
| 1 | 1 | 2 | NaN | NaN | NaN |
| 2 | 2 | 3 | NaN | NaN | NaN |
| 3 | 3 | 4 | NaN | NaN | NaN |
| 4 | 4 | 5 | NaN | NaN | NaN |
| 5 | 5 | 6 | NaN | NaN | NaN |
| 6 | 6 | 7 | NaN | NaN | NaN |
| 7 | 7 | 8 | NaN | NaN | NaN |
| 8 | 8 | 9 | NaN | NaN | NaN |
| 9 | 9 | 10 | NaN | NaN | NaN |
| 10 | 10 | 11 | NaN | NaN | NaN |
| 11 | 11 | 12 | NaN | NaN | NaN |

```
In [46]: df.loc[:, 'age '].mode()
```

```
Out[46]: 0    29.0  
Name: age , dtype: float64
```

```
In [47]: df.min()
```

```
Out[47]: Unnamed: 0      0.0  
customer id      1.0  
age             19.0  
income          42.0  
spending score   55.0  
dtype: float64
```

```
In [49]: df.loc[:, 'age '].min(skipna = False)
```

```
Out[49]: nan
```

```
In [50]: df.max()
```

```
Out[50]: Unnamed: 0      52.0  
customer id      53.0  
age              50.0  
income           42.0  
spending score   55.0  
dtype: float64
```

```
In [52]: df.loc[:, 'age '].max(skipna = False)
```

```
Out[52]: nan
```

```
In [53]: df.std()
```

```
Out[53]: Unnamed: 0      15.443445  
customer id      15.443445  
age              9.747814  
income           NaN  
spending score   NaN  
dtype: float64
```

```
In [54]: df.loc[:, 'age '].std()
```

```
Out[54]: 9.747813693073532
```

```
In [55]: df.std(axis=1)[0:4]
```

```
Out[55]: 0      19.706175  
1      25.225648  
2      14.730920  
3      14.730920  
dtype: float64
```

```
In [57]: df.groupby(['customer id'])['age '].mean()
```

```
Out[57]: customer id  
1      19.0  
2      20.0  
3      28.0  
4      29.0  
5      23.0  
6      23.0  
7      NaN  
8      32.0  
9      43.0  
10     36.0  
11     NaN  
12     20.0  
13     19.0  
14     23.0  
15     49.0  
16     43.0  
17     NaN  
18     47.0  
19     19.0
```

```
In [71]: df_u=df.rename(columns= {'income':' new income'},inplace=False)
df_u.groupby(['age ']).income.mean()
```

```
Out[71]: age
19.0    42.0
20.0     NaN
21.0     NaN
22.0     NaN
23.0     NaN
28.0     NaN
29.0     NaN
31.0     NaN
32.0     NaN
33.0     NaN
34.0     NaN
36.0     NaN
37.0     NaN
38.0     NaN
40.0     NaN
43.0     NaN
45.0     NaN
47.0     NaN
48.0     NaN
```

```
In [73]: from sklearn import preprocessing
enc = preprocessing.OneHotEncoder()
enc_df = pd.DataFrame(enc.fit_transform(df[['age ']]).toarray())
enc_df
```

```
Out[73]:
```

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ... | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 1.0 | 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 | 0.0 | 1.0 | 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 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 3 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 4 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 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 |
| 5 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 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 |
| 6 | 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 |
| 7 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 8 | 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 | 1.0 | 0.0 | 0.0 | 0.0 |
| 9 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 10 | 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 |

```
In [74]: df_encode = df_u.join(enc_df)
df_encode
```

```
Out[74]:
```

| | Unnamed: 0 | customer id | age | income | spending score | 0 | 1 | 2 | 3 | 4 | ... | 11 | 12 | 13 |
|----|------------|-------------|------|--------|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 0 | 1 | 19.0 | 42.0 | NaN | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 |
| 1 | 1 | 2 | 20.0 | NaN | 55.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 |
| 2 | 2 | 3 | 28.0 | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 |
| 3 | 3 | 4 | 29.0 | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 |
| 4 | 4 | 5 | 23.0 | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | ... | 0.0 | 0.0 | 0.0 |
| 5 | 5 | 6 | 23.0 | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | ... | 0.0 | 0.0 | 0.0 |
| 6 | 6 | 7 | NaN | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 |
| 7 | 7 | 8 | 32.0 | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 |
| 8 | 8 | 9 | 43.0 | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 |
| 9 | 9 | 10 | 36.0 | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 1.0 | 0.0 | 0.0 |
| 10 | 10 | 11 | NaN | NaN | NaN | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 |

```
In [91]: col_names = ['Sepal_Length', 'Sepal_Width', 'Petal_Length', 'Petal_Width', 'Species']
```

```
In [93]: iris = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data')
```

```
In [94]: irisSet = (iris['Species'] == 'Iris-setosa')
```

```
In [95]: print('Iris-setosa')
print(iris[irisSet].describe())
```

```
Iris-setosa
```

| | Sepal_Length | Sepal_Width | Petal_Length | Petal_Width |
|-------|--------------|-------------|--------------|-------------|
| count | 50.00000 | 50.000000 | 50.000000 | 50.00000 |
| mean | 5.00600 | 3.418000 | 1.464000 | 0.24400 |
| std | 0.35249 | 0.381024 | 0.173511 | 0.10721 |
| min | 4.30000 | 2.300000 | 1.000000 | 0.10000 |
| 25% | 4.80000 | 3.125000 | 1.400000 | 0.20000 |
| 50% | 5.00000 | 3.400000 | 1.500000 | 0.20000 |
| 75% | 5.20000 | 3.675000 | 1.575000 | 0.30000 |
| max | 5.80000 | 4.400000 | 1.900000 | 0.60000 |

```
In [96]: irisVer = (iris['Species'] == 'Iris-versicolor')
```

```
In [97]: print('Iris-versicolor')
print(iris[irisVer].describe())
```

Iris-versicolor

| | Sepal_Length | Sepal_Width | Petal_Length | Petal_Width |
|-------|--------------|-------------|--------------|-------------|
| count | 50.000000 | 50.000000 | 50.000000 | 50.000000 |
| mean | 5.936000 | 2.770000 | 4.260000 | 1.326000 |
| std | 0.516171 | 0.313798 | 0.469911 | 0.197753 |
| min | 4.900000 | 2.000000 | 3.000000 | 1.000000 |
| 25% | 5.600000 | 2.525000 | 4.000000 | 1.200000 |
| 50% | 5.900000 | 2.800000 | 4.350000 | 1.300000 |
| 75% | 6.300000 | 3.000000 | 4.600000 | 1.500000 |
| max | 7.000000 | 3.400000 | 5.100000 | 1.800000 |

Name- Mansi Nirbhavane Rollno.- 13251