GRIP : The Sparks Foundation

Data Science & Business Analytics

Author: Ume Salma Khan

Task 3: Perform 'Exploratory Data Analysis' on dataset 'SampleSuperstore' This task is about Exploratory Data Analysis - Retail where the task focuses on a business manager who will try to find out weak areas where he can work to make more profit.

In [57]:
```
# Importing required libraries
import warnings
warnings.filterwarnings('ignore')
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

In [58]:
```
df = pd.read_csv('SampleSuperstore.csv')
df.head()
```

Out[58]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.9600	2	0.00	41.9136
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731.9400	3	0.00	219.5820
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	14.6200	2	0.00	6.8714
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775	5	0.45	-383.0310
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.3680	2	0.20	2.5164

In [3]:
```
df.shape
```

Out[3]:
```
(9994, 13)
```

In [4]:
```
df.describe()
```

Out[4]:

	Postal Code	Sales	Quantity	Discount	Profit
count	9994.000000	9994.000000	9994.000000	9994.000000	9994.000000
mean	55190.379428	229.858001	3.789574	0.156203	28.656896
std	32063.693350	623.245101	2.225110	0.206452	234.260108
min	1040.000000	0.444000	1.000000	0.000000	-6599.978000
25%	23223.000000	17.280000	2.000000	0.000000	1.728750
50%	56430.500000	54.490000	3.000000	0.200000	8.666500
75%	90008.000000	209.940000	5.000000	0.200000	29.364000
max	99301.000000	22638.480000	14.000000	0.800000	8399.976000

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
Ship Mode       0
Segment         0
Country         0
City            0
State           0
Postal Code     0
Region          0
Category        0
Sub-Category    0
Sales           0
Quantity        0
Discount        0
Profit          0
dtype: int64
```

In [6]:
```
df.columns
```

Out[6]:
```
Index(['Ship Mode', 'Segment', 'Country', 'City', 'State', 'Postal Code',
       'Region', 'Category', 'Sub-Category', 'Sales', 'Quantity', 'Discount',
       'Profit'],
      dtype='object')
```

In [7]:
```
df.duplicated().sum()
```

Out[7]:
```
17
```

In [59]:
```
df = pd.read_csv('SampleSuperstore.csv')
df
```

Out[59]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	Sales	Quantity	Discount	Profit
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.9600	2	0.00	41.9136
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	731.9400	3	0.00	219.5820
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	14.6200	2	0.00	6.8714
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775	5	0.45	-383.0310
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.3680	2	0.20	2.5164
...	...	...	...	...	...	...	...	...	...	...	...	...	...
9989	Second Class	Consumer	United States	Miami	Florida	33180	South	Furniture	Furnishings	25.2480	3	0.20	4.1028
9990	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Furniture	Furnishings	91.9600	2	0.00	15.6332
9991	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Technology	Phones	258.5760	2	0.20	19.3932
9992	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Office Supplies	Paper	29.6000	4	0.00	13.3200
9993	Second Class	Consumer	United States	Westminster	California	92683	West	Office Supplies	Appliances	243.1600	2	0.00	72.9480

9994 rows × 13 columns

EXPLORATORY DATA ANALYSIS

DATA VISUALIZATION:

In [ ]:
```
# Visualizing the dataset using pairplot
sns.pairplot(df)
```

Out[ ]:
```
<seaborn.axisgrid.PairGrid at 0x10b5ee80>
```

In [ ]:
```
# Correlation analysis
df.corr()
sns.heatmap(df.corr(), annot=True)
plt.show()
```

Visualizing Ship Modes:

In [ ]:
```
df['Ship Mode'].value_counts()
```

In [ ]:
```
plt.figure(figsize=(6,6))
plt.title('SHIP MODES')
plt.pie(df['Ship Mode'].value_counts(), labels=df['Ship Mode'].value_counts().index, autopct='%1.1f%%')
plt.show()
sns.countplot(x=df['Ship Mode'])
```

Visualizing Segment:

In [ ]:
```
df['Segment'].value_counts()
```

In [ ]:
```
sns.pairplot(df,hue="Segment")
```

Visualizing Category:

In [ ]:
```
df['Category'].value_counts()
```

In [ ]:
```
plt.figure(figsize=(6,6))
plt.title('Category')
plt.pie(df['Category'].value_counts(), labels = df['Category'].value_counts().index, autopct='%1.1f%%')
plt.show()
sns.countplot(x='Category',data=df,palette='tab10')
```

In [ ]:
```
plt.figure(figsize = (8,8))
plt.title('Sub-Category')
plt.pie(df['Sub-Category'].value_counts(), labels=df['Sub-Category'].value_counts().index,autopct='%1.1f%%')
plt.show()
```

In [ ]:
```
df['State'].value_counts()
```

State-wise Profit:

In [ ]:
```
plt.figure(figsize=(15,15))
stpr = df.groupby(['State'])['Profit'].sum().nlargest(50)
stpr.plot.barh()
```

The above Graph displays that California and New York have the highest Profits while Texas and Ohio have the least profits.

Region-wise Profit:

In [ ]:
```
plt.figure(figsize= (6,6))
plt.title('Region wise Profits')
plt.pie(df['Region'].value_counts(), labels=df['Region'].value_counts().index,autopct='%1.1f%%')
plt.show()
```

Sales, Profit & Discount Interdependency-

In [ ]:
```
plt.style.use('seaborn')
df.plot(kind="scatter",figsize=(12,6), x="Sales",y="Profit",c="Discount",s=20,fontsize=12,colormap='plasma')
plt.ylabel('Profits')
plt.show()
```

More discount leads to more Sales but Lesser the Profits

Profit vs Discount:

In [ ]:
```
sns.lineplot(x='Discount',y='Profit',label='Profit',data=df)
plt.legend()
plt.show()
```

Profit/Loss and Sales of Each State:

In [ ]:
```
pls = df.groupby('State')[['Sales','Profit']].sum().sort_values(by='Sales',ascending=False)
pls[:].plot.bar(color=['red','black'],figsize=(20,10))
plt.title('Profit/Loss and Sales across the States')
plt.xlabel('States')
plt.ylabel('Profit/Loss and Sales')
plt.show()
```

Conclusion

1. Work more on California and New York as they are places of maximum sales.

2. Decrease Discounts in Southern Region to Increase sales.

3. Reduce sales of furniture as it has very less profit compared to other category Sales