

Skill 9

Name : Shraddha Pawar

Batch : ML 18

Title :Visualize the dataset for presentation

Problem statement

Visualize the sample superstore dataset

```
In [34]: # Import Libraries

import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
```

```
In [35]: # Read dataset

df = pd.read_excel("SampleSuperstore.csv.xlsx")
df
```

Out[35]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category
0	Second Class	Consumer	United States	Henderson	Kentucky	42420.0	South	Furniture	Bookcases
1	Second Class	Consumer	United States	Henderson	Kentucky	42420.0	South	Furniture	Chairs
2	Second Class	Corporate	United States	Los Angeles	California	90036.0	West	Office Supplies	Labels
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311.0	South	Furniture	Tables
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311.0	South	Office Supplies	Storage
...
9989	Second Class	Consumer	United States	Miami	Florida	33180.0	South	Furniture	Furnishings
9990	Standard Class	Consumer	United States	Costa Mesa	California	92627.0	West	Furniture	Furnishings
9991	Standard Class	Consumer	United States	Costa Mesa	California	92627.0	West	Technology	Phones
9992	Standard Class	Consumer	United States	Costa Mesa	California	92627.0	West	Office Supplies	Paper
9993	Second Class	Consumer	United States	Westminster	California	92683.0	West	Office Supplies	Appliances

9994 rows × 14 columns



Data Information and insights

In [36]: df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 14 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Ship Mode              9994 non-null   object
1   Segment                9994 non-null   object
2   Country                9994 non-null   object
3   City                   9994 non-null   object
4   State                  9994 non-null   object
5   Postal Code            9994 non-null   float64
6   Region                 9994 non-null   object
7   Category               9994 non-null   object
8   Sub-Category           9994 non-null   object
9   Sales                  9994 non-null   float64
10  Quantity               9994 non-null   float64
11  Discount               9994 non-null   float64
12  Profit                 9994 non-null   float64
13  Profit/Loss            9994 non-null   object
dtypes: float64(5), object(9)
memory usage: 1.1+ MB
```

In [37]: df.head()

Out[37]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category	S
0	Second Class	Consumer	United States	Henderson	Kentucky	42420.0	South	Furniture	Bookcases	261.5
1	Second Class	Consumer	United States	Henderson	Kentucky	42420.0	South	Furniture	Chairs	731.5
2	Second Class	Corporate	United States	Los Angeles	California	90036.0	West	Office Supplies	Labels	14.6
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311.0	South	Furniture	Tables	957.5
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311.0	South	Office Supplies	Storage	22.5

In [38]: df.columns

Out[38]: Index(['Ship Mode', 'Segment', 'Country', 'City', 'State', 'Postal Code', 'Region', 'Category', 'Sub-Category', 'Sales', 'Quantity', 'Discount', 'Profit', 'Profit/Loss'], dtype='object')

In [39]: df.shape

Out[39]: (9994, 14)

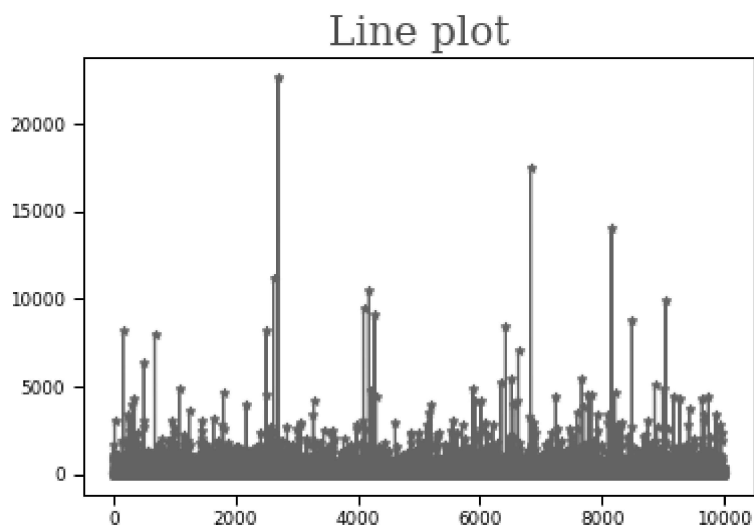
```
In [40]: df.isnull().sum()
```

```
Out[40]: 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  
Profit/Loss  0  
dtype: int64
```

Data Visualization

Univariate analysis

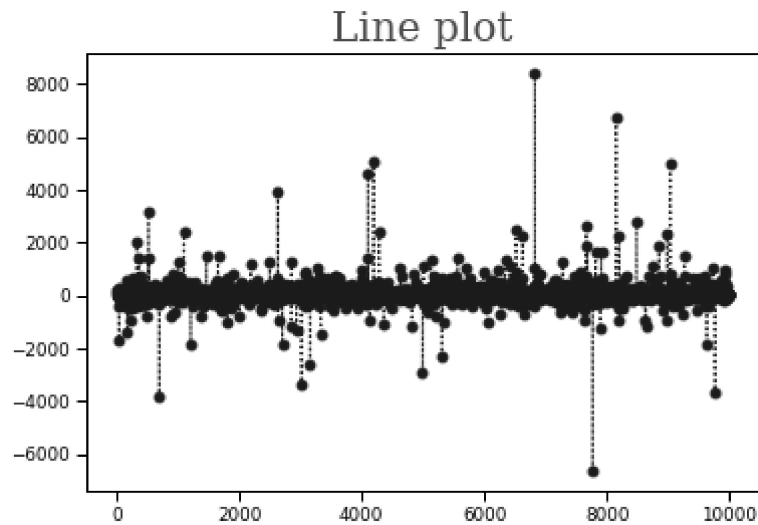
```
In [41]: plt.plot("Sales",data=df,marker = "*")  
  
font1 = {'family':'serif','color':'green','size':20}  
plt.title("Line plot",fontdict = font1)  
plt.show()
```



The above plot shows the sales of the suprestore.

Maximum sales is upto 5000.

```
In [42]: plt.plot("Profit",data=df,marker = "o",color="b", linestyle = "dotted")
font1 = {'family':'serif','color':'green','size':20}
plt.title("Line plot",fontdict = font1)
plt.show()
```

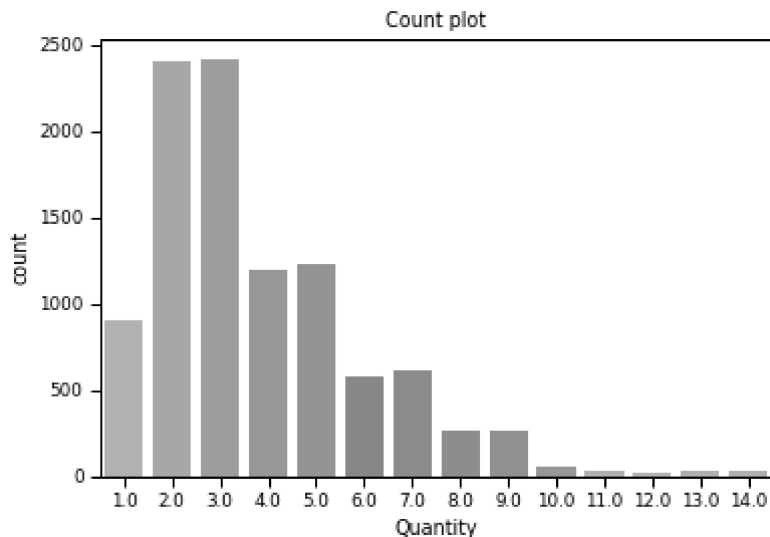


The above plot shows the profit and loss of superstore.

```
In [43]: # Countplot

sns.countplot(x='Quantity', data=df )
plt.title("Count plot")
```

```
Out[43]: Text(0.5, 1.0, 'Count plot')
```



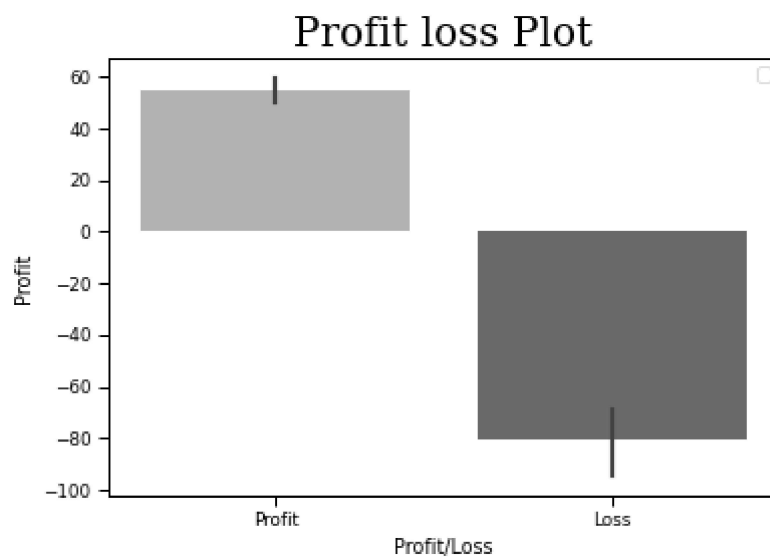
Maximum 2 to 3 products purchased in superstore.

Bivariate analysis

```
In [44]: sns.set_context('paper')

# create plot
sns.barplot(x = 'Profit/Loss', y = 'Profit', data = df,
            palette = 'PuRd')
font1 = {'family':'serif','color':'black','size':20}
plt.title("Profit loss Plot",fontdict = font1)
plt.legend()
plt.show()
```

No handles with labels found to put in legend.



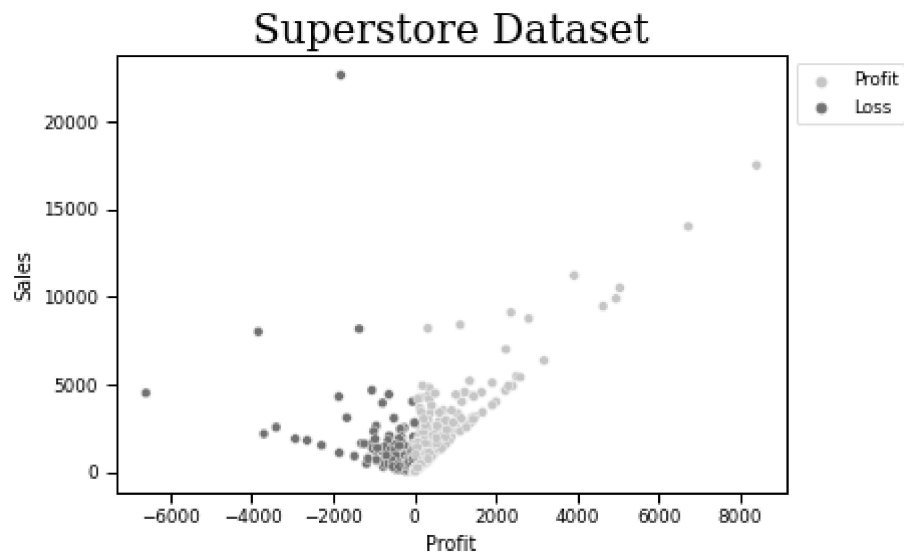
From the above plot, loss of superstore is maximum than profit.

```
In [45]: fig = px.pie(df, values='Sales', names='Category')  
fig.show()
```

In [46]: *# Scatter plot*

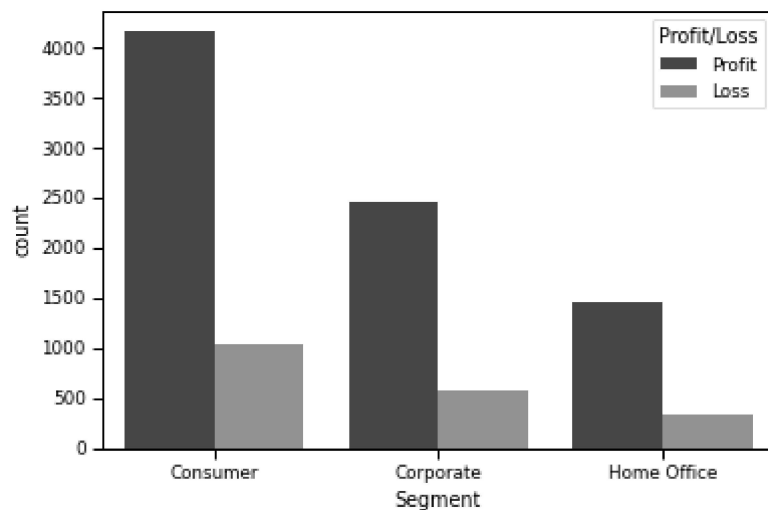
```
sns.scatterplot(x='Profit', y='Sales', hue='Profit/Loss', data=df, palette = 'YlOrRd')
font1 = {'family':'serif','color':'black','size':20}
plt.title("Superstore Dataset",fontdict = font1)
plt.legend(bbox_to_anchor=(1,1), loc=2)
```

Out[46]: <matplotlib.legend.Legend at 0x2a79835a700>



In [47]: `sns.set_context('paper')`

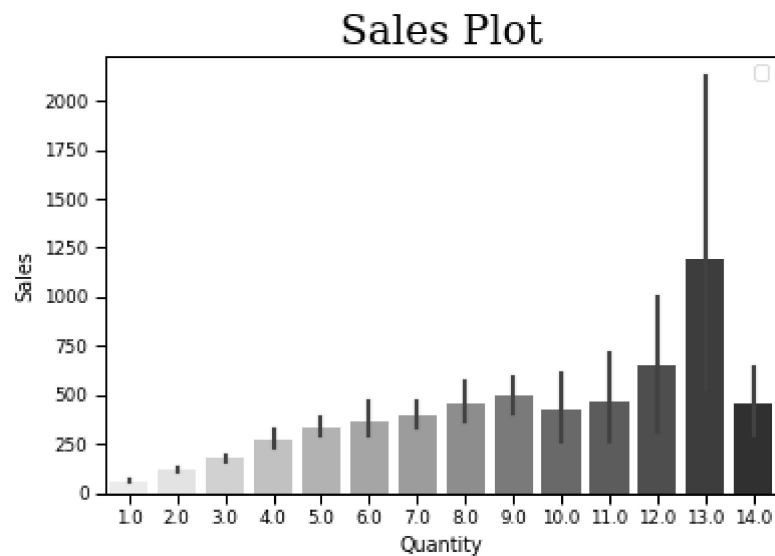
```
# create plot
sns.countplot(x = 'Segment', hue = 'Profit/Loss', data = df, palette = 'magma')
plt.show()
```




```
In [48]: sns.set_context('paper')

# create plot
sns.barplot(x = 'Quantity', y = 'Sales', data = df,
            palette = 'PuRd')
font1 = {'family':'serif','color':'black','size':20}
plt.title("Sales Plot",fontdict = font1)
plt.legend()
plt.show()
```

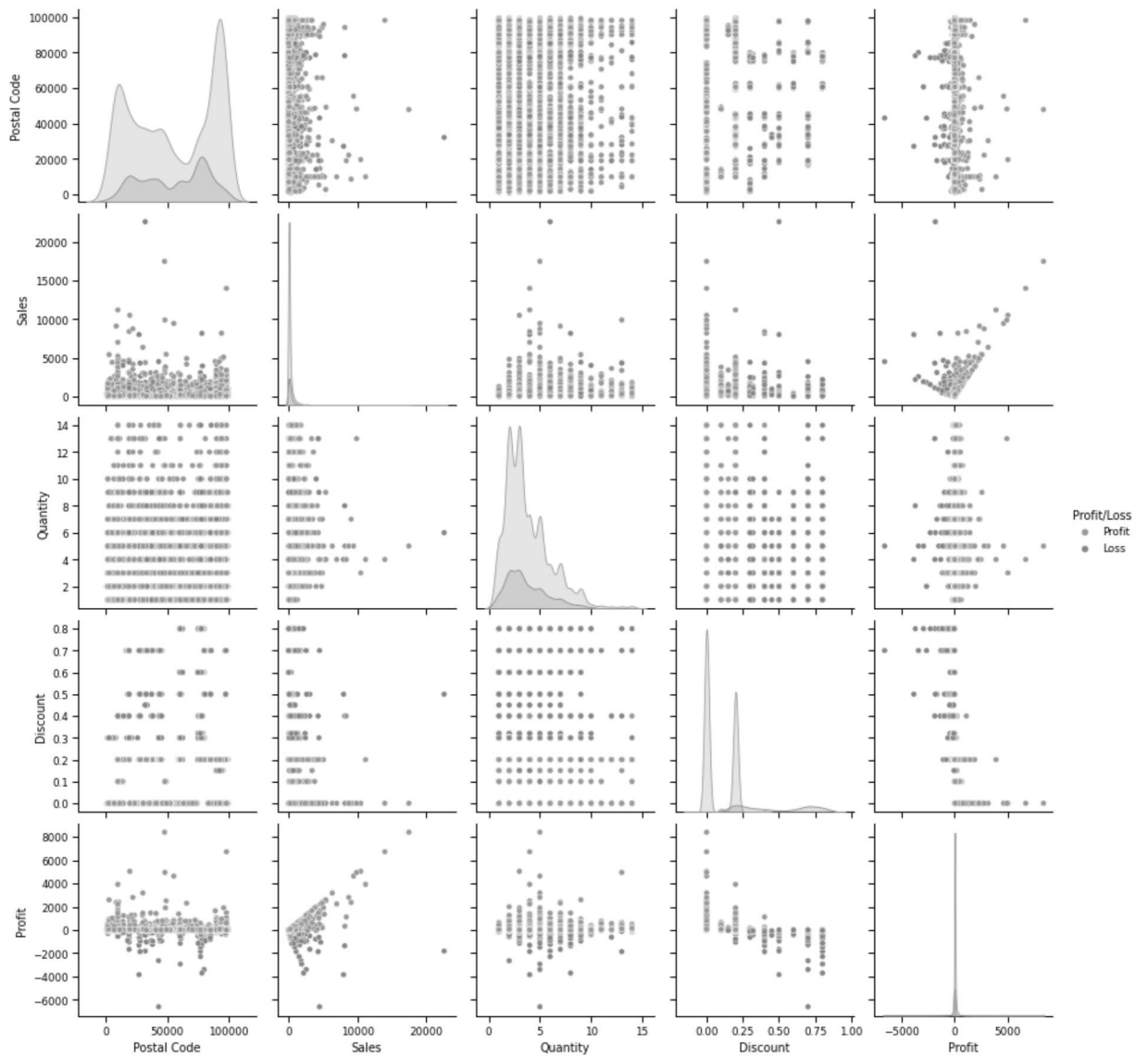
No handles with labels found to put in legend.



Multivariate Analysis

```
In [49]: sns.pairplot(data = df,hue = 'Profit/Loss',diag_kind = "kde",kind = "scatter",pa
```

```
Out[49]: <seaborn.axisgrid.PairGrid at 0x2a7983ac850>
```



In [50]: *# Correlation plot*

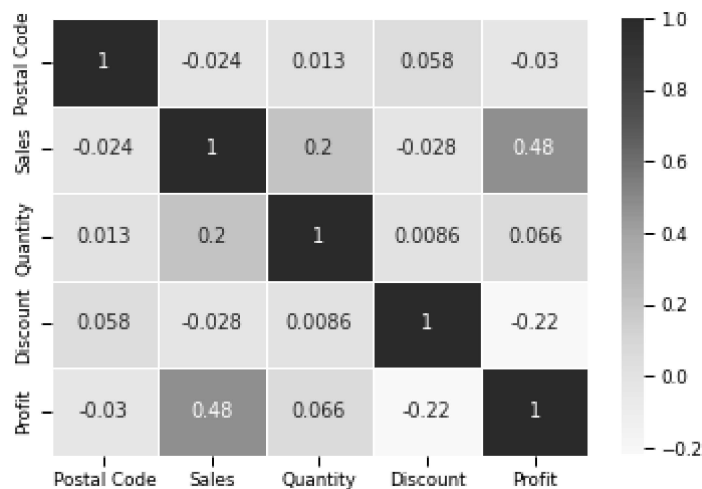
```
corr=df.corr()
corr
```

Out[50]:

	Postal Code	Sales	Quantity	Discount	Profit
Postal Code	1.000000	-0.023854	0.012761	0.058443	-0.029961
Sales	-0.023854	1.000000	0.200795	-0.028190	0.479064
Quantity	0.012761	0.200795	1.000000	0.008623	0.066253
Discount	0.058443	-0.028190	0.008623	1.000000	-0.219487
Profit	-0.029961	0.479064	0.066253	-0.219487	1.000000

In [51]: `sns.heatmap(corr,annot=True,linewidths=.5,cmap="YlOrRd")`

Out[51]: <AxesSubplot:>



Conclusion:

The above analysis provides some results:

Maximum sales is upto 5000.

Maximum 2 to 3 products purchased in superstore.

Loss of superstore is maximum than profit.

In []:

