SF

May 31, 2025

0.1 The Spark Foundation - Data Science & Business Analytics Internship

0.1.1 Perform ‘Exploratory Data Analysis’ on dataset ‘SampleSuperstore’

[30]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

[31]:

```
df = pd.read_csv("C:/Users/Asus/Downloads/SampleSuperstore.csv")
```

[32]:

```
df
```

[32]:

```
           Ship Mode    Segment        Country             City       State  \
0       Second Class   Consumer  United States        Henderson    Kentucky
1       Second Class   Consumer  United States        Henderson    Kentucky
2       Second Class  Corporate  United States      Los Angeles  California
3     Standard Class   Consumer  United States  Fort Lauderdale     Florida
4     Standard Class   Consumer  United States  Fort Lauderdale     Florida
…               …        …            …               …         …
9989    Second Class   Consumer  United States            Miami     Florida
9990  Standard Class   Consumer  United States       Costa Mesa  California
9991  Standard Class   Consumer  United States       Costa Mesa  California
9992  Standard Class   Consumer  United States       Costa Mesa  California
9993    Second Class   Consumer  United States      Westminster  California

      Postal Code Region         Category Sub-Category     Sales  Quantity  \
0           42420  South        Furniture    Bookcases  261.9600         2
1           42420  South        Furniture       Chairs  731.9400         3
2           90036   West  Office Supplies       Labels   14.6200         2
3           33311  South        Furniture       Tables  957.5775         5
4           33311  South  Office Supplies      Storage   22.3680         2
…           …    …               …         …       …       …
9989        33180  South        Furniture  Furnishings   25.2480         3
9990        92627   West        Furniture  Furnishings   91.9600         2
9991        92627   West       Technology       Phones  258.5760         2
9992        92627   West  Office Supplies        Paper   29.6000         4
9993        92683   West  Office Supplies   Appliances  243.1600         2
```

```
      Discount    Profit
0         0.00   41.9136
1         0.00  219.5820
2         0.00    6.8714
3         0.45 -383.0310
4         0.20    2.5164
…          …        …
9989      0.20    4.1028
9990      0.00   15.6332
9991      0.20   19.3932
9992      0.00   13.3200
9993      0.00   72.9480

[9994 rows x 13 columns]
```

[33]:
```
df.shape
```

[33]:
```
(9994, 13)
```

[34]:
```
df.head()
```

[34]:
```
        Ship Mode    Segment        Country             City       State  \
0    Second Class   Consumer  United States        Henderson    Kentucky
1    Second Class   Consumer  United States        Henderson    Kentucky
2    Second Class  Corporate  United States      Los Angeles  California
3  Standard Class   Consumer  United States  Fort Lauderdale     Florida
4  Standard Class   Consumer  United States  Fort Lauderdale     Florida

   Postal Code Region         Category Sub-Category     Sales  Quantity  \
0        42420  South        Furniture    Bookcases  261.9600         2
1        42420  South        Furniture       Chairs  731.9400         3
2        90036   West  Office Supplies       Labels   14.6200         2
3        33311  South        Furniture       Tables  957.5775         5
4        33311  South  Office Supplies      Storage   22.3680         2

   Discount    Profit
0      0.00   41.9136
1      0.00  219.5820
2      0.00    6.8714
3      0.45 -383.0310
4      0.20    2.5164
```

[35]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 13 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   int64
 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   int64
 11  Discount      9994 non-null   float64
 12  Profit        9994 non-null   float64
dtypes: float64(3), int64(2), object(8)
memory usage: 1015.1+ KB
```

[36]:

```
df.describe()
```

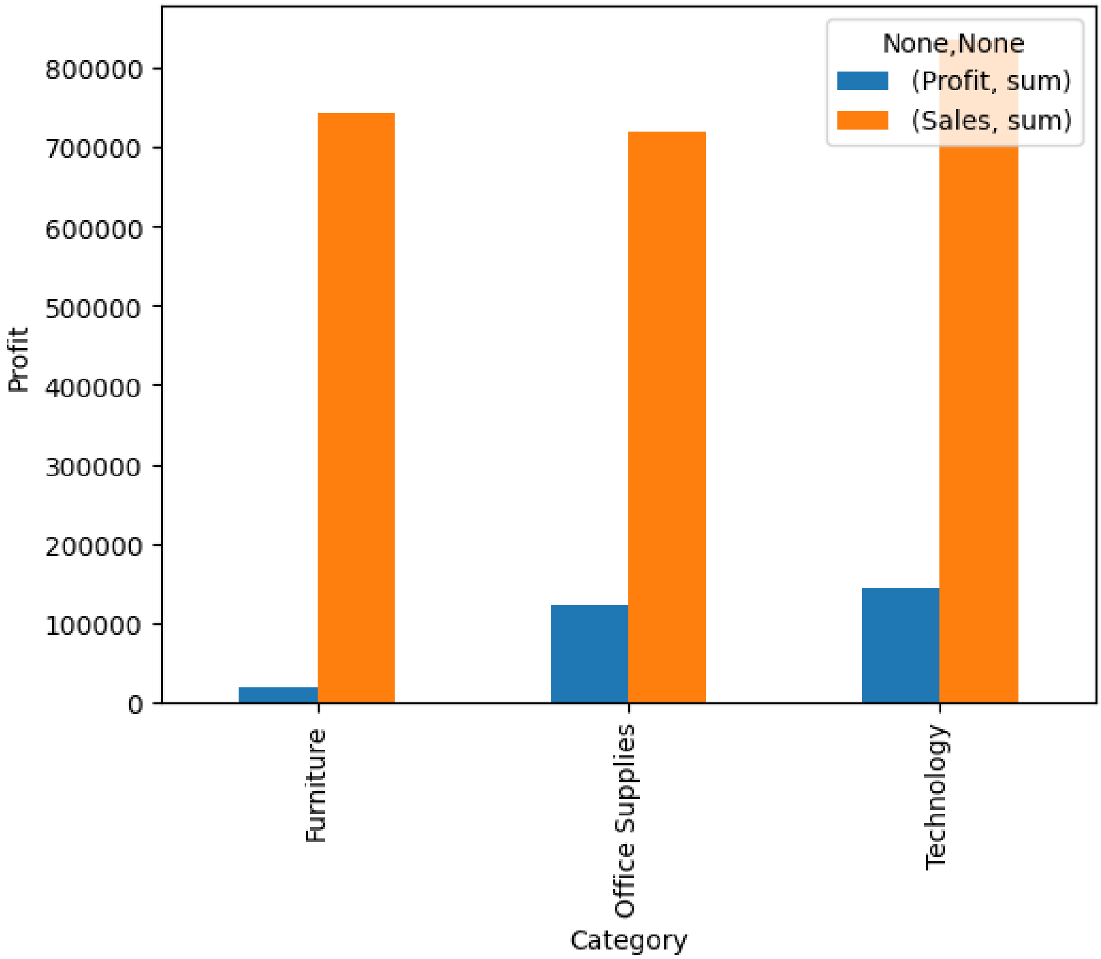
[36]:

```
        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
```

[39]:

```
plt.figure(figsize= (10,16))
df.groupby('Category')[['Profit','Sales']].agg(['sum']).plot.bar()
plt.ylabel('Profit')
plt.show()
```

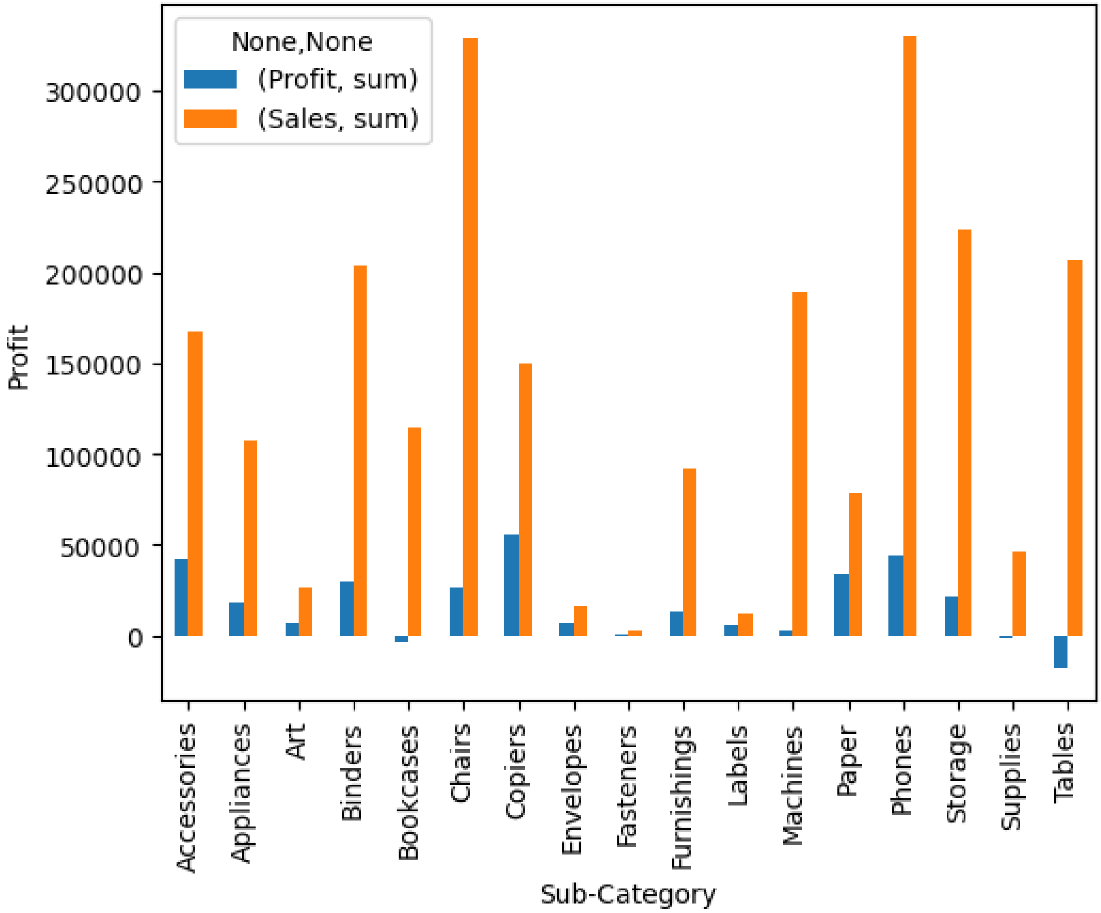
```
<Figure size 1000x1600 with 0 Axes>
```



[40]:

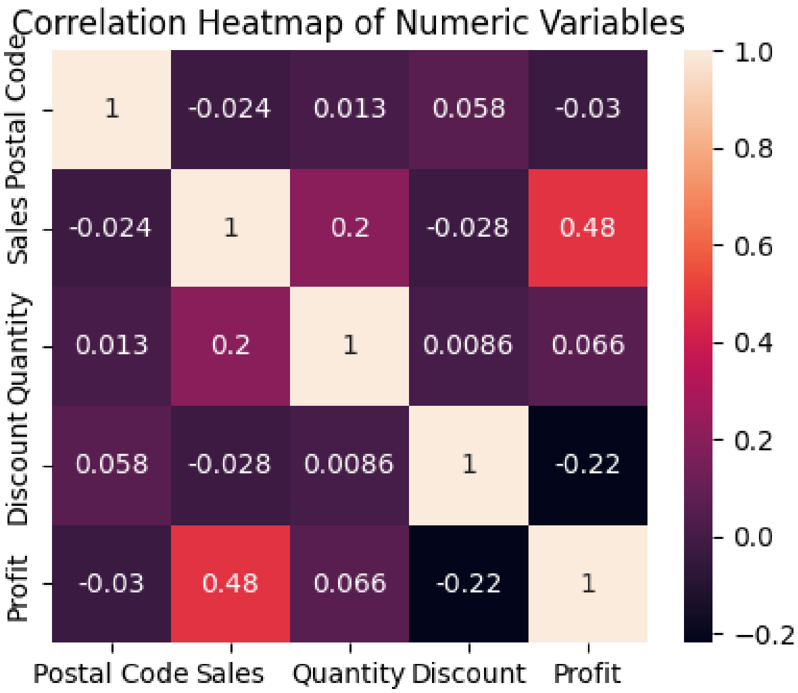
```
plt.figure(figsize= (10,16))
df.groupby('Sub-Category')[['Profit','Sales']].agg(['sum']).plot.bar()
plt.ylabel('Profit')
plt.show()
```

```
<Figure size 1000x1600 with 0 Axes>
```



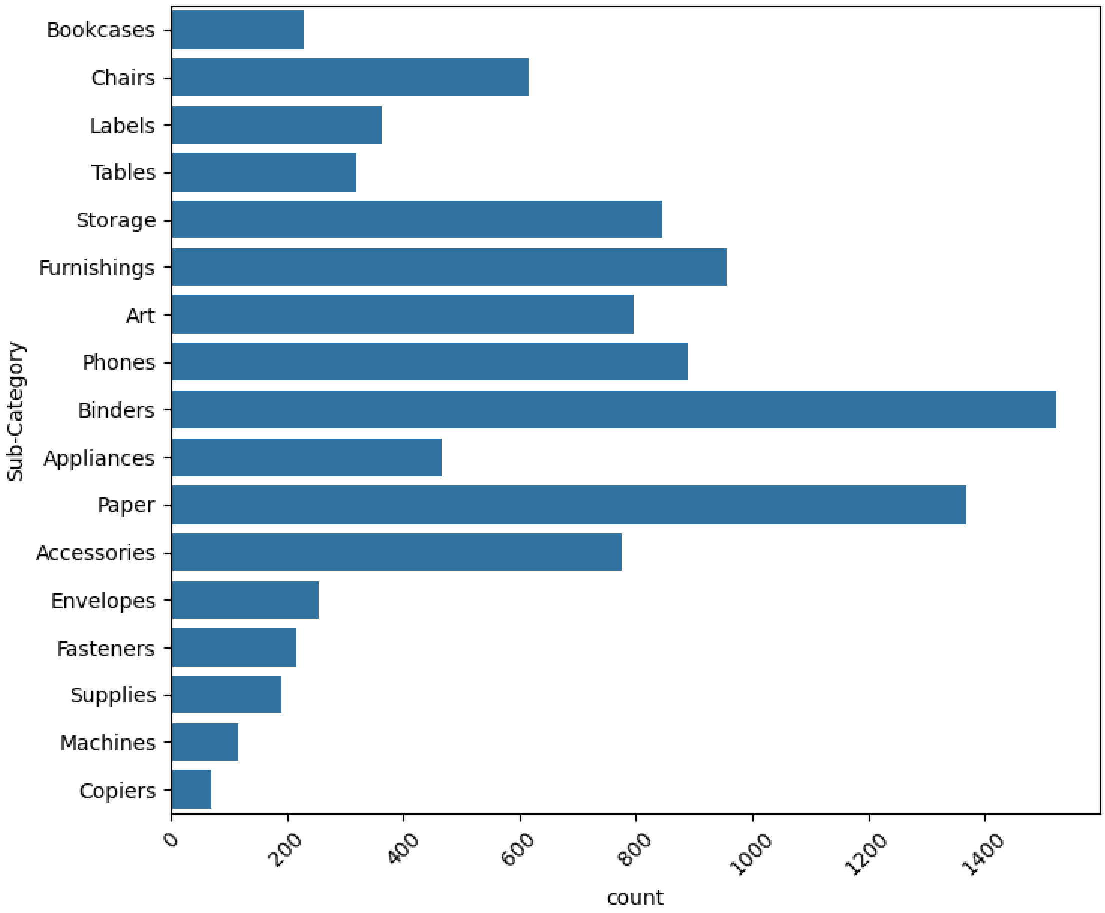
[58]:

```
import seaborn as sns
import matplotlib.pyplot as plt

numeric_df = df.select_dtypes(include=['float64', 'int64'])

numeric_df = numeric_df.dropna()

plt.figure(figsize=(5, 4))
sns.heatmap(numeric_df.corr(), annot=True)
plt.title('Correlation Heatmap of Numeric Variables')
plt.show()
```



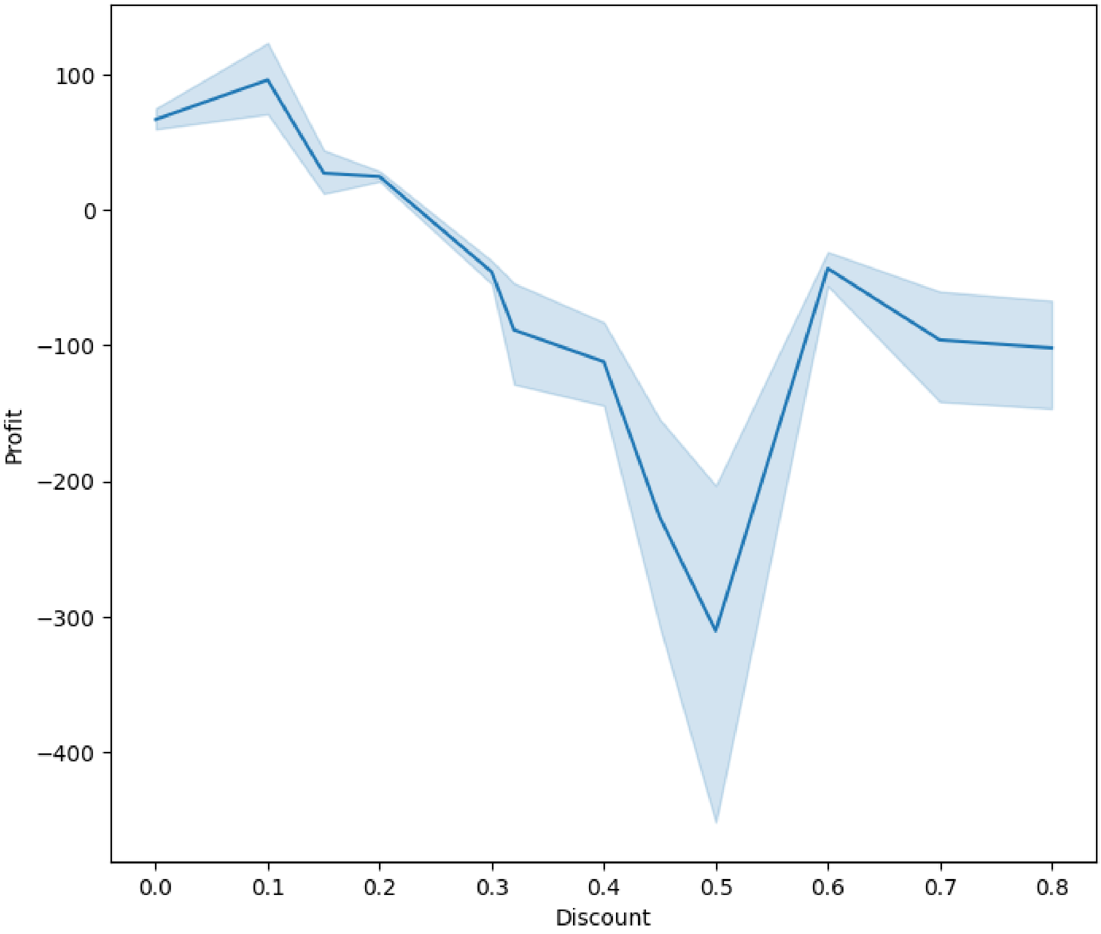
[66]:
```
plt.figure(figsize=(8, 7))
sns.countplot(df['Sub-Category'])
plt.xticks(rotation=45)
```

[66]:
```
(array([   0.,  200.,  400.,  600.,  800., 1000., 1200., 1400., 1600.]),
 [Text(0.0, 0, '0'),
  Text(200.0, 0, '200'),
  Text(400.0, 0, '400'),
  Text(600.0, 0, '600'),
  Text(800.0, 0, '800'),
  Text(1000.0, 0, '1000'),
  Text(1200.0, 0, '1200'),
  Text(1400.0, 0, '1400'),
  Text(1600.0, 0, '1600')])
```



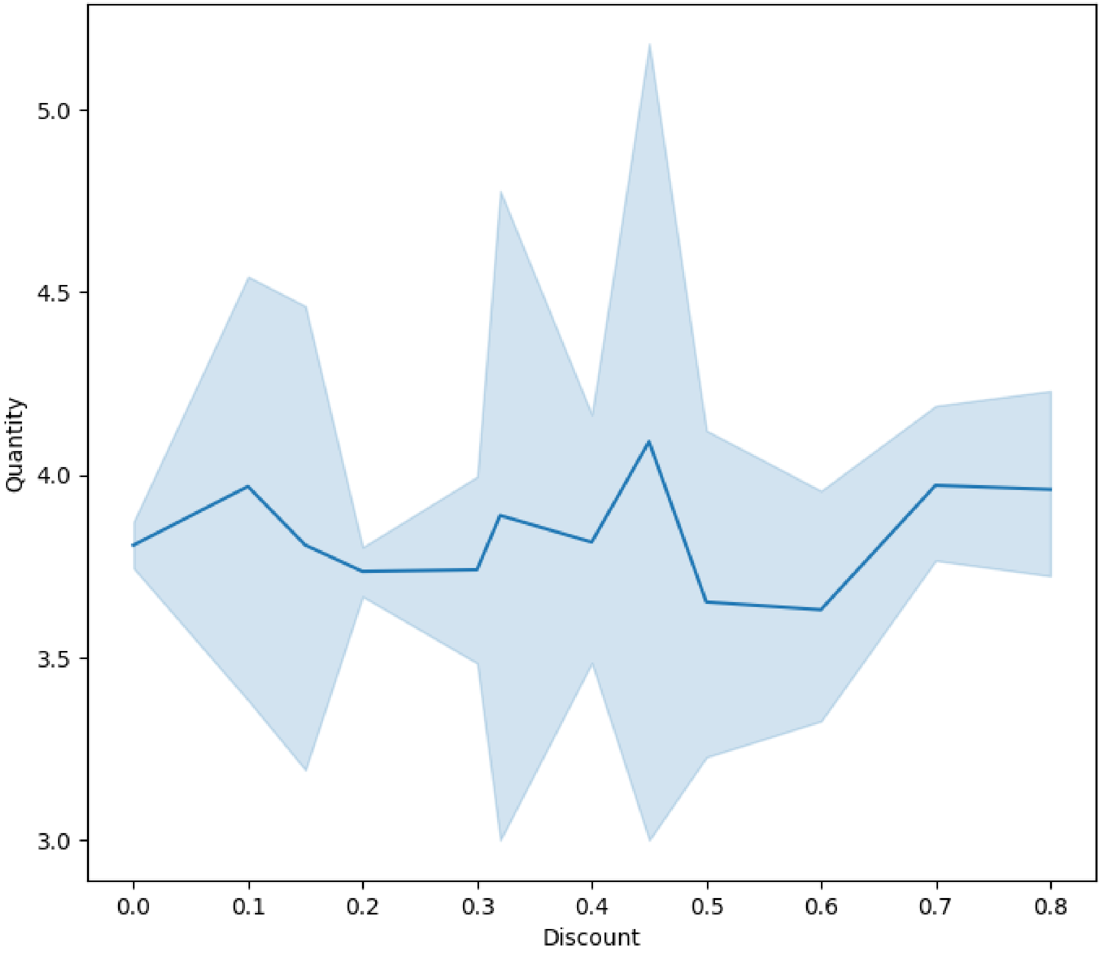
```
[70]: plt.figure(figsize=(8, 7))
      sns.lineplot(x=df['Discount'], y=df['Profit'])
```

```
[70]: <Axes: xlabel='Discount', ylabel='Profit'>
```



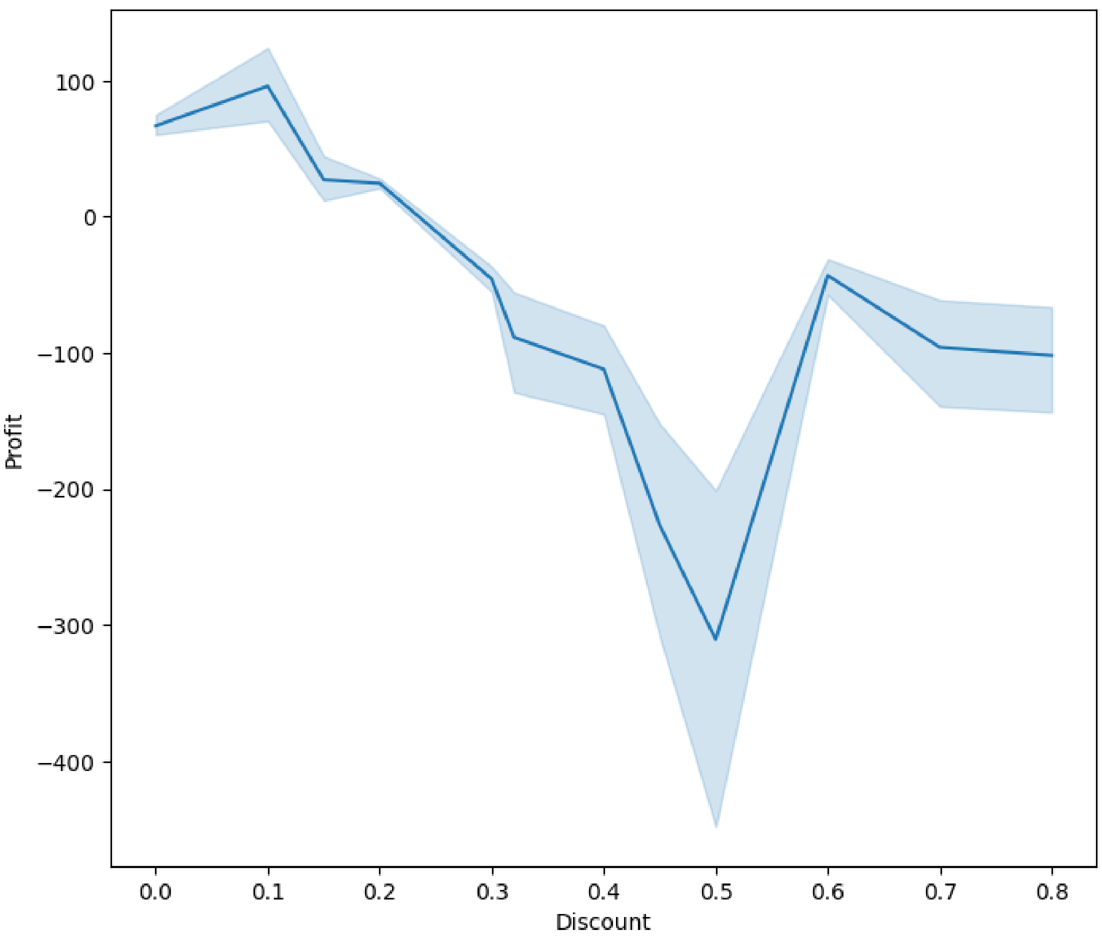
```
[71]: plt.figure(figsize=(8, 7))
      sns.lineplot(x=df['Discount'], y=df['Quantity'])
```

```
[71]: <Axes: xlabel='Discount', ylabel='Quantity'>
```



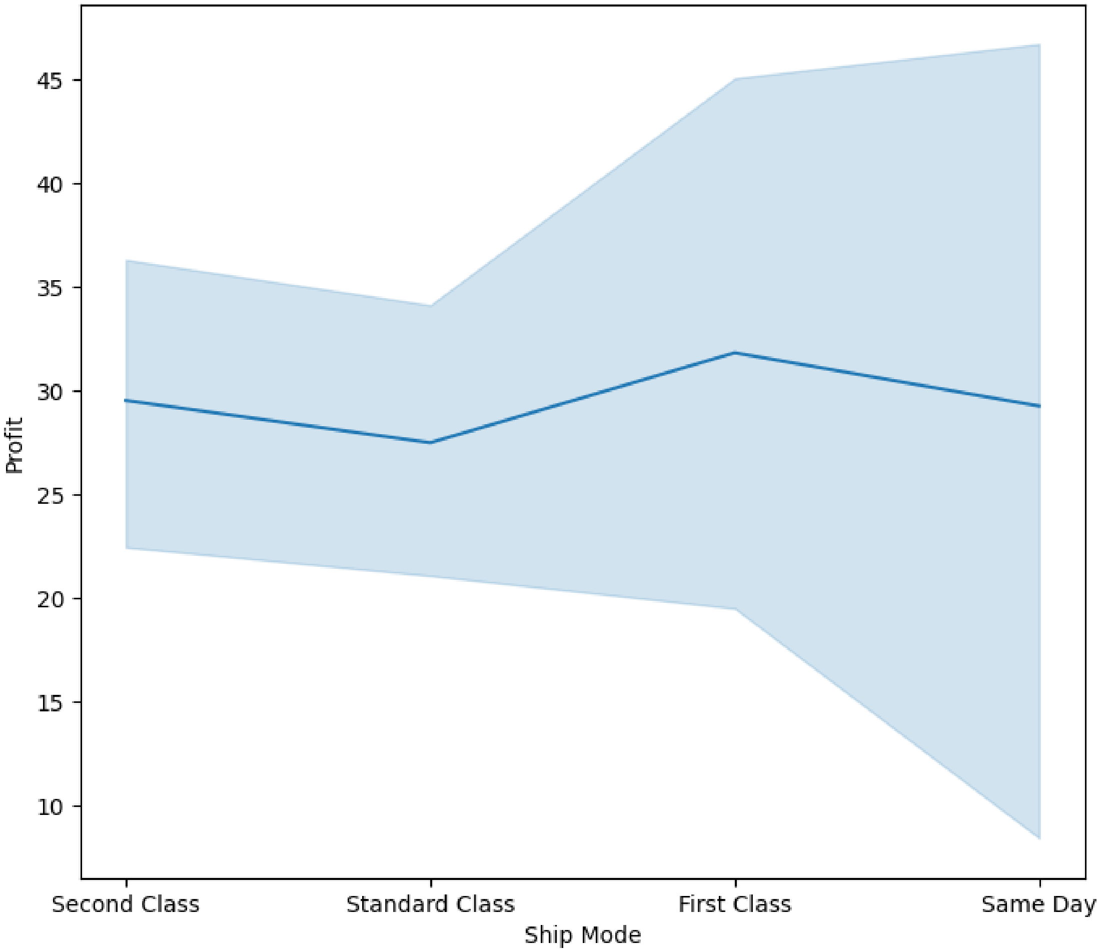
```
[72]: plt.figure(figsize=(8, 7))
      sns.lineplot(x=df['Discount'], y=df['Profit'])
```

```
[72]: <Axes: xlabel='Discount', ylabel='Profit'>
```

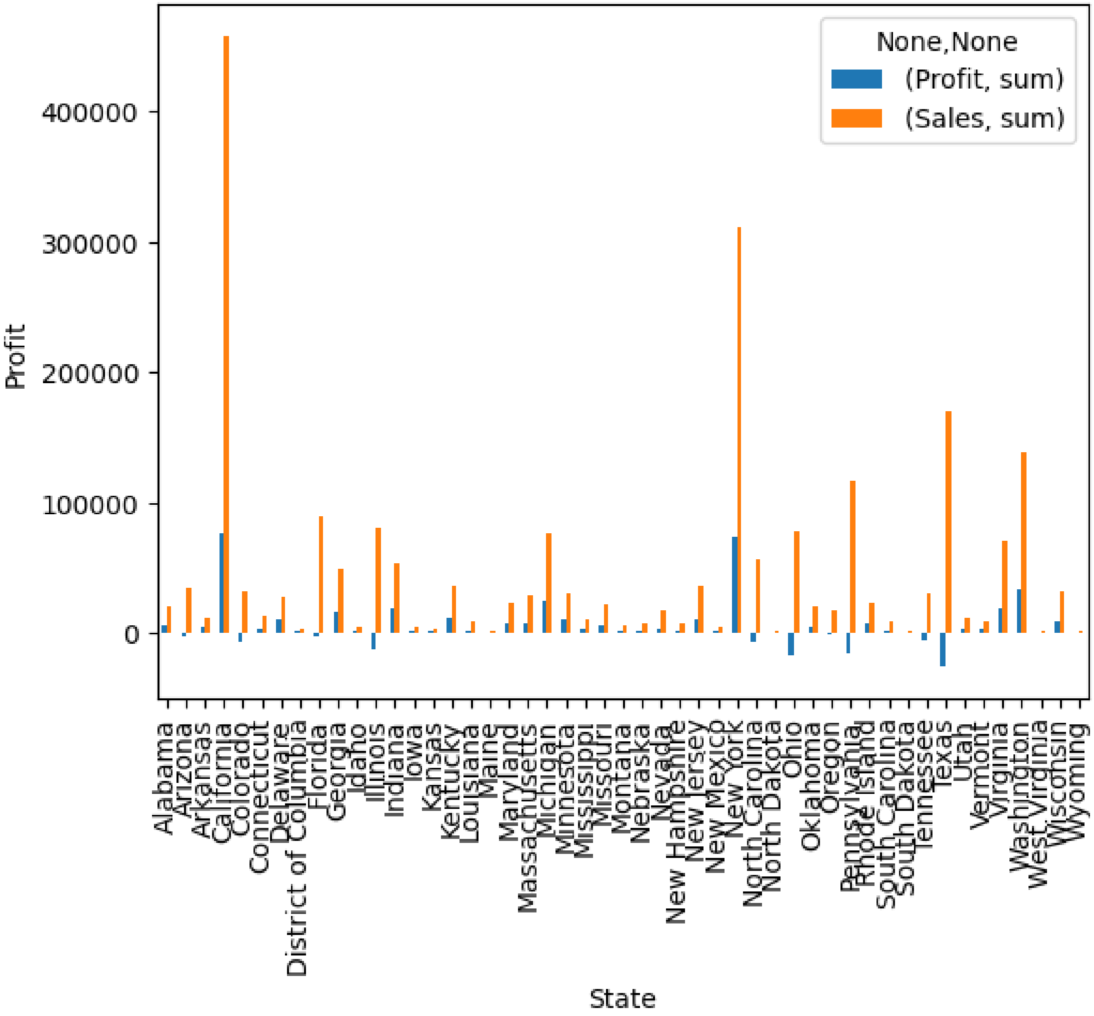


```
[74]: plt.figure(figsize=(8, 7))
      sns.lineplot(x=df['Ship Mode'], y=df['Profit'])
```

```
[74]: <Axes: xlabel='Ship Mode', ylabel='Profit'>
```



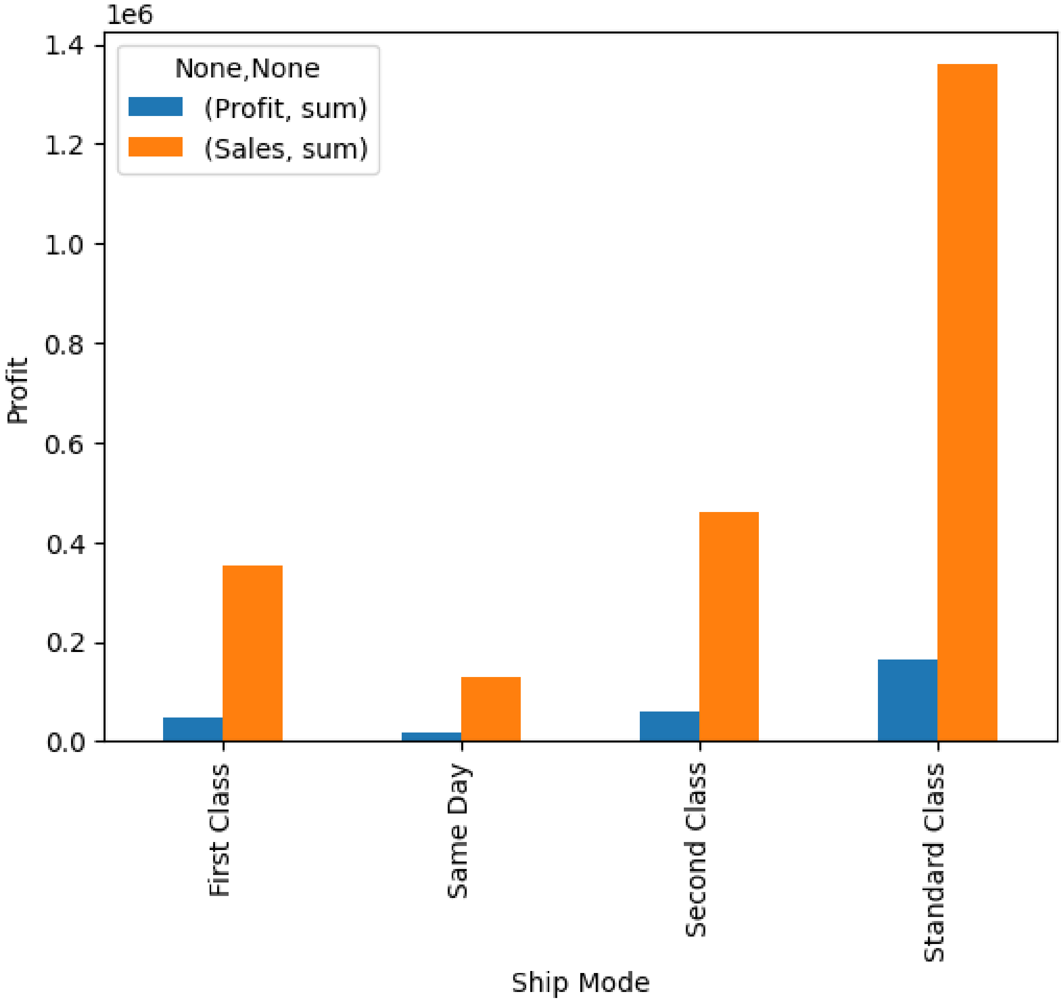
[75]:
```
df.groupby('State')[['Profit','Sales']].agg(['sum']).plot.bar()
plt.ylabel('Profit')
plt.show()
```



[76]:

```
plt.figure(figsize= (10,16))
df.groupby('Ship Mode')[['Profit','Sales']].agg(['sum']).plot.bar()
plt.ylabel('Profit')
plt.show()
```

```
<Figure size 1000x1600 with 0 Axes>
```


1e6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Profit
None,None
(Profit, sum)
(Sales, sum)
First Class
Same Day
Second Class
Standard Class
Ship Mode

[ ]: