analysis-eda-samplesuperstore-2

April 25, 2024

0.0.1 Step -1: Importing the required Libraries

[3]:

```
# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
%matplotlib inline
from plotnine import *
import warnings
warnings.filterwarnings('ignore')
```

0.0.2 2) Importing the dataset

[4]:

```
sample = pd.read_csv('SampleSuperstore.csv')
```

[5]:

```
sample
```

[5]:

```
           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]
```

[6]:

```
sample.head(5)
```

[6]:

```
        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
```

[7]:

```
sample.tail(5)
```

[7]:

```
           Ship Mode   Segment        Country         City       State  \
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  \
9989        33180  South        Furniture  Furnishings   25.248         3
9990        92627   West        Furniture  Furnishings   91.960         2
9991        92627   West       Technology       Phones  258.576         2
9992        92627   West  Office Supplies        Paper   29.600         4
9993        92683   West  Office Supplies   Appliances  243.160         2

      Discount   Profit
9989       0.2   4.1028
9990       0.0  15.6332
9991       0.2  19.3932
9992       0.0  13.3200
9993       0.0  72.9480
```

0.0.3 Step-3 Check Data types

[8]:

```
sample.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
```

[9]:

```
sample.describe()
```

```
[9]:         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
```

0.0.4 Step - 4 Checking for any Null Values in the columns and duplicates values

```
[10]: sample.isnull().sum()
```

```
[10]: 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
```

```
[11]: ## Checking of Duplicated data
      sample.duplicated().sum()
```

```
[11]: 17
```

```
[12]: ## Deleting Duplicates if any
      sample.drop_duplicates(inplace=True)
```

```
[13]: ## founding out any duplicates left from the sample file
      sample.duplicated().sum()
```

```
[13]: 0
```

```
[14]: ## Displaying the unique data
      sample.nunique()
```

```
[14]: Ship Mode          4
      Segment            3
```

```
Country            1
City             531
State             49
Postal Code      631
Region             4
Category           3
Sub-Category      17
Sales           5825
Quantity          14
Discount          12
Profit          7287
dtype: int64
```

```
[15]: ##Dropping of Irrelevant columns like we have postal code in the sample file
      col =['Postal Code']
      drop =sample.drop(columns=col, axis=1, inplace =True)
```

```
[16]: sample
```

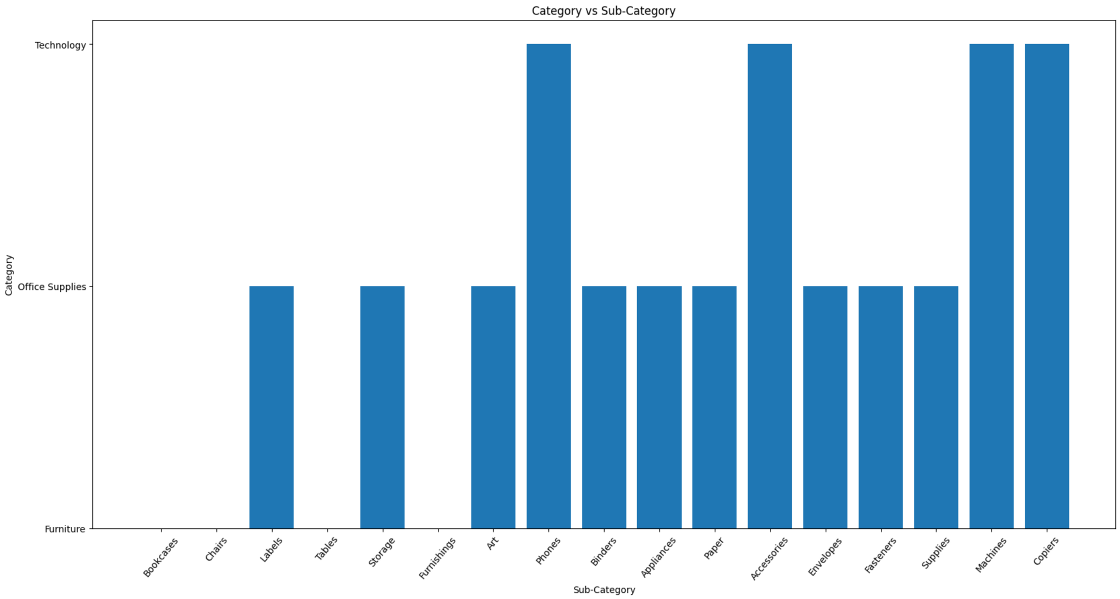
```
[16]:
           Ship Mode    Segment        Country             City       State  \
0       Second Class   Consumer  United States        Henderson    Kentucky
1       Second Class   Consumer  United States        Henderson    Kentucky
2       Second Class  Corporate  United States      Los Angeles  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

     Region         Category Sub-Category     Sales  Quantity  Discount  \
0     South        Furniture    Bookcases  261.9600         2      0.00
1     South        Furniture       Chairs  731.9400         3      0.00
2      West  Office Supplies       Labels   14.6200         2      0.00
3     South        Furniture       Tables  957.5775         5      0.45
4     South  Office Supplies      Storage   22.3680         2      0.20
…     …               …          …       …       …       …
9989  South        Furniture  Furnishings   25.2480         3      0.20
9990   West        Furniture  Furnishings   91.9600         2      0.00
9991   West       Technology       Phones  258.5760         2      0.20
9992   West  Office Supplies        Paper   29.6000         4      0.00
9993   West  Office Supplies   Appliances  243.1600         2      0.00

        Profit
0      41.9136
```

```
1      219.5820
2        6.8714
3     -383.0310
4        2.5164
…          …
9989     4.1028
9990    15.6332
9991    19.3932
9992    13.3200
9993    72.9480

[9977 rows x 12 columns]
```

0.0.5 Step -5 Data Visualisation

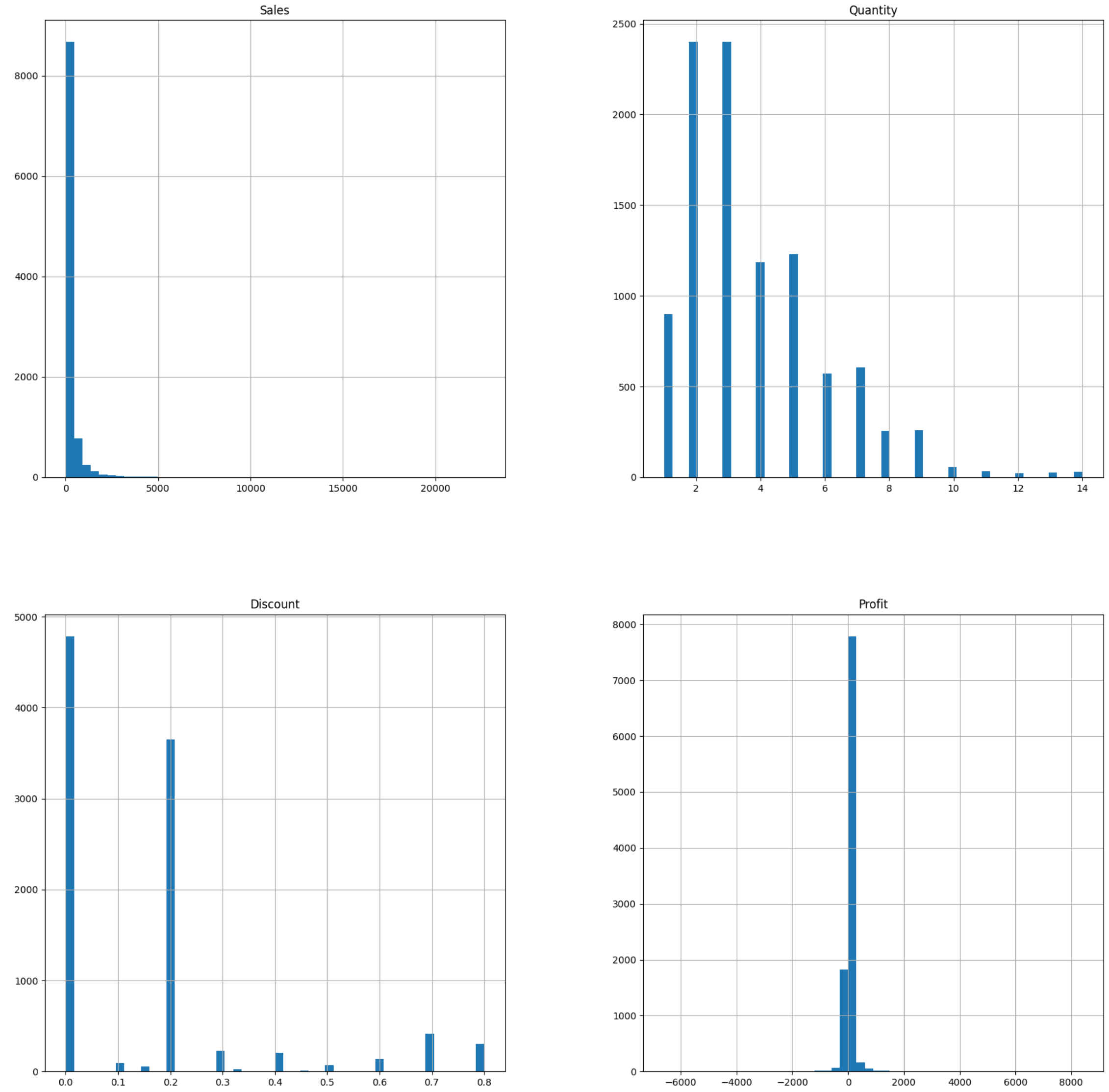
[18]:

```
plt.figure(figsize=(20,10))
plt.bar('Sub-Category','Category', data=sample)
plt.title('Category vs Sub-Category')
plt.xlabel('Sub-Category')
plt.ylabel('Category')
plt.xticks(rotation=50)
plt.show()
```



[19]:

```
sample.hist(bins=50,figsize=(20,20))
plt.show()
```



[20]:

```
## Counting repeatable states in the sample file
sample['State'].value_counts()
```

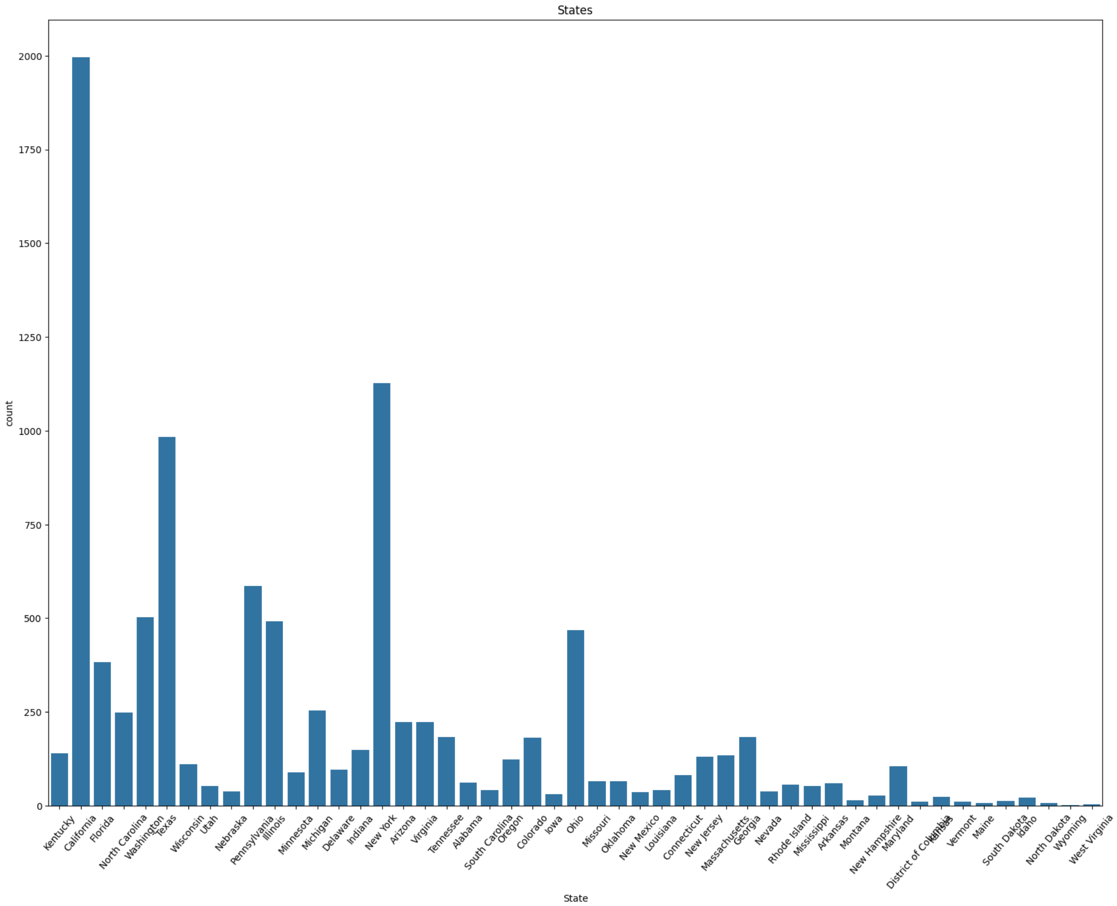
[20]:

```
State
California              1996
New York                1127
Texas                    983
Pennsylvania             586
Washington               502
Illinois                 491
Ohio                     468
Florida                  383
Michigan                 254
```

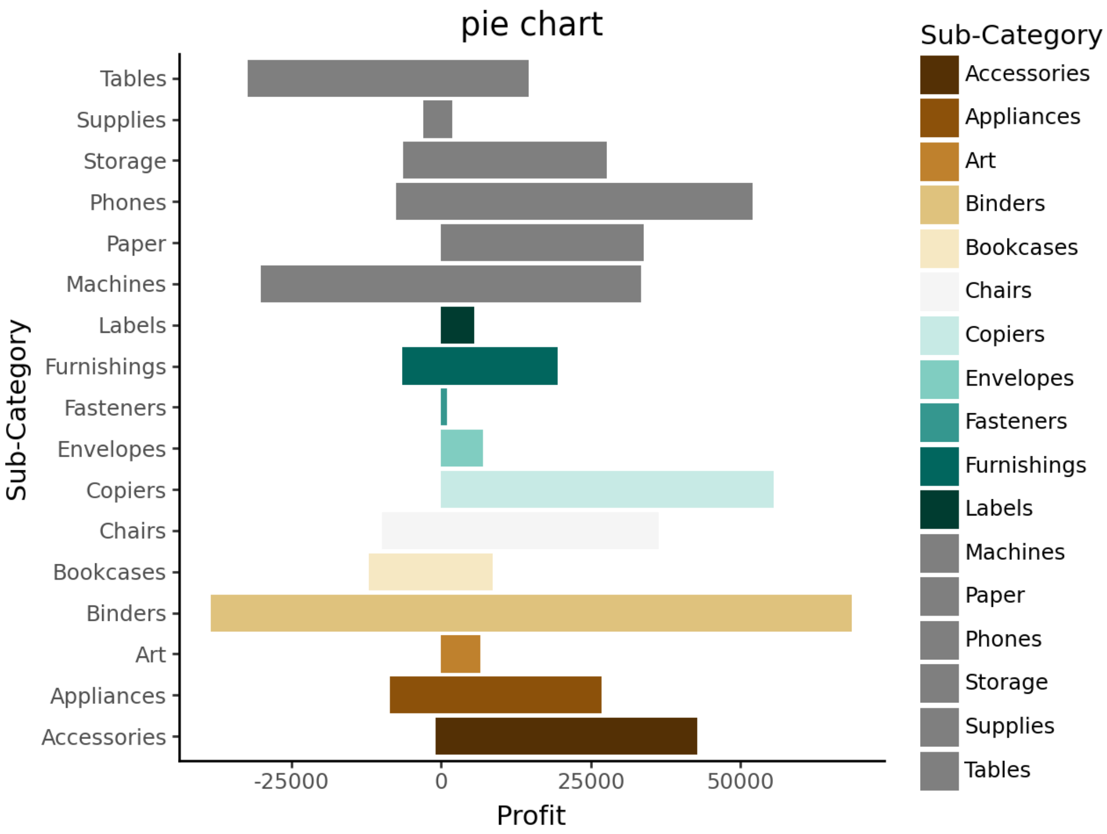
```
North Carolina           249
Arizona                  224
Virginia                 224
Georgia                  184
Tennessee                183
Colorado                 182
Indiana                  149
Kentucky                 139
Massachusetts            135
New Jersey               130
Oregon                   123
Wisconsin                110
Maryland                 105
Delaware                  96
Minnesota                 89
Connecticut               82
Oklahoma                  66
Missouri                  66
Alabama                   61
Arkansas                  60
Rhode Island              56
Utah                      53
Mississippi               53
Louisiana                 42
South Carolina            42
Nevada                    39
Nebraska                  38
New Mexico                37
Iowa                      30
New Hampshire             27
Kansas                    24
Idaho                     21
Montana                   15
South Dakota              12
Vermont                   11
District of Columbia      10
Maine                      8
North Dakota               7
West Virginia              4
Wyoming                    1
Name: count, dtype: int64
```

[21]:

```
plt.figure(figsize=(20,15))
sns.countplot(x=sample['State'])
plt.xticks(rotation=50)
plt.title('States')
plt.show()
```



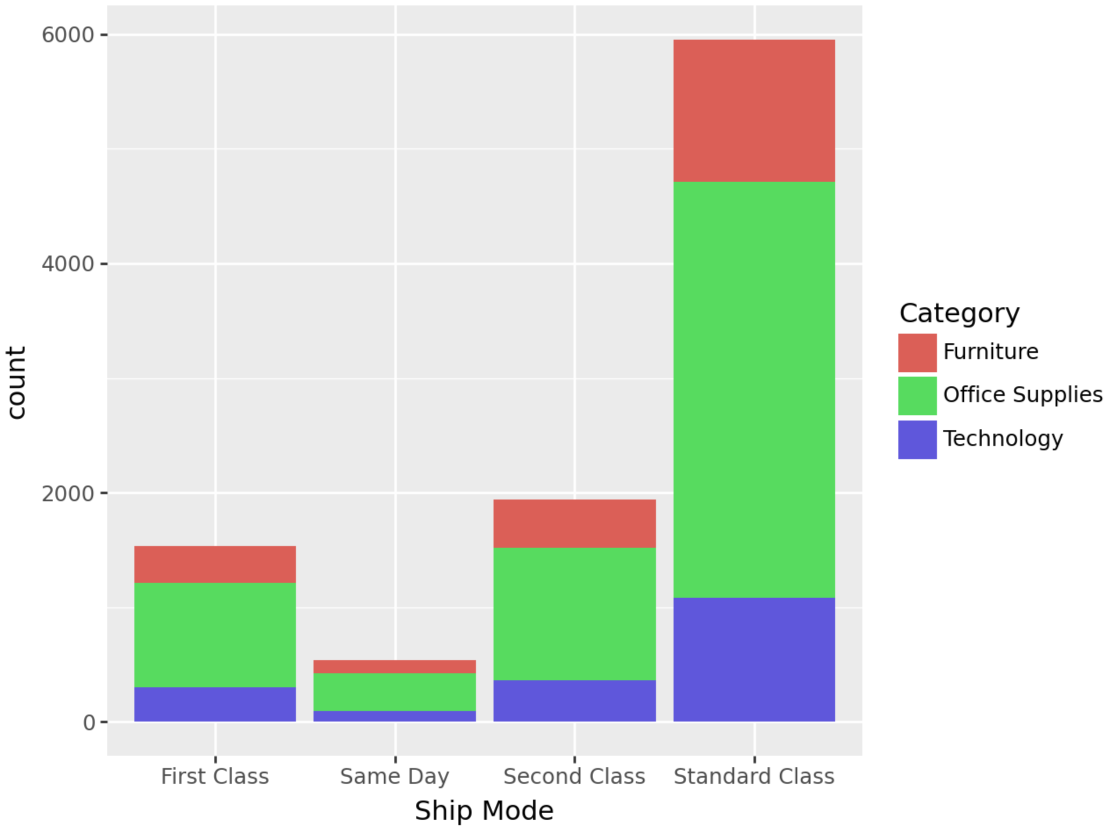
```
[22]: Profit_plot=(ggplot(sample, aes(x='Sub-Category', y='Profit', fill='Sub-Category')) + geom_col() + coord_flip()
      + scale_fill_brewer(type='div', pelette='Spectral') + theme_classic() + ggtitle('pie chart'))
      display(Profit_plot)
```



The chart shows the profit got by the Sub-Categories and the loss

by ggplot checking out the categories and its count, ship modes

```
[23]: ggplot(sample, aes(x='Ship Mode', fill ='Category')) + geom_bar(stat ='count')
```

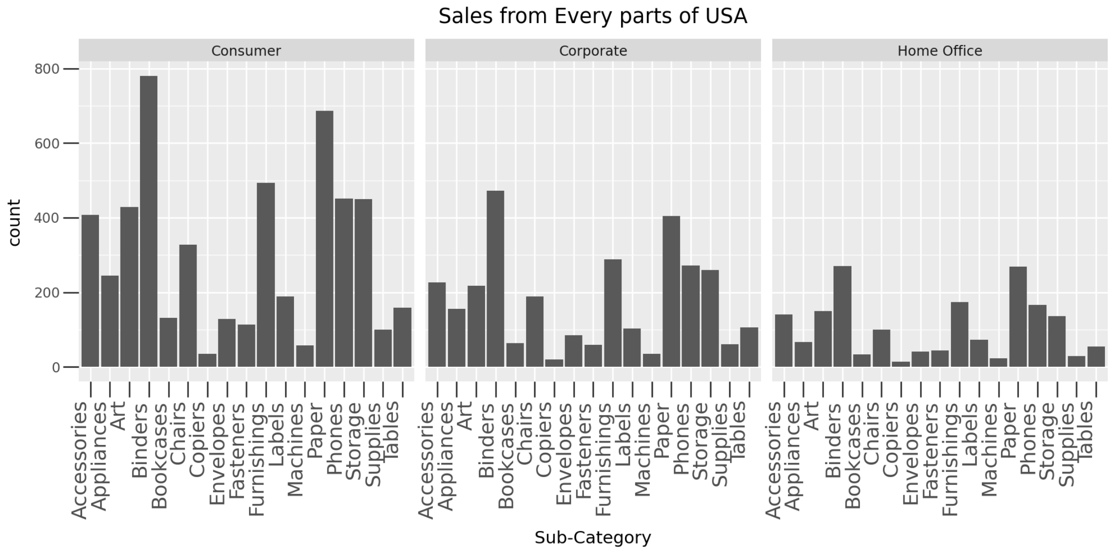


[24]:

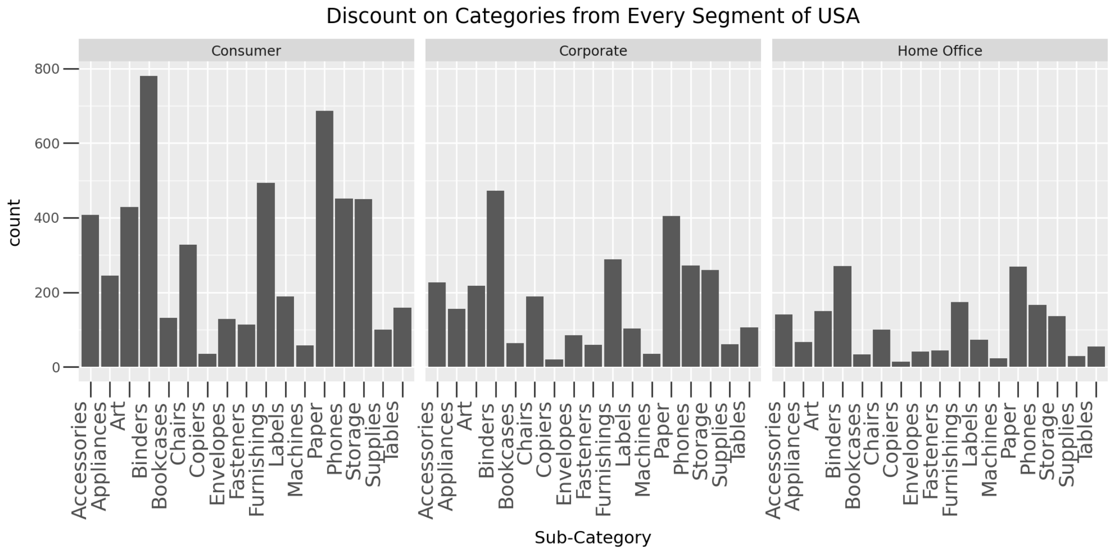
```
flip_xlabels = theme(axis_text_x = element_text(angle=90,␣
 ↪hjust=1),figure_size=(10,5),
                        axis_ticks_length_major=10,axis_ticks_length_minor=5)
```

[25]:

```
(ggplot(sample, aes(x='Sub-Category', fill='Sales'))+ geom_bar()+␣
 ↪facet_wrap(['Segment'])+flip_xlabels + theme(axis_text_x =␣
 ↪element_text(size=13))+ggtitle("Sales from Every parts of USA"))
```

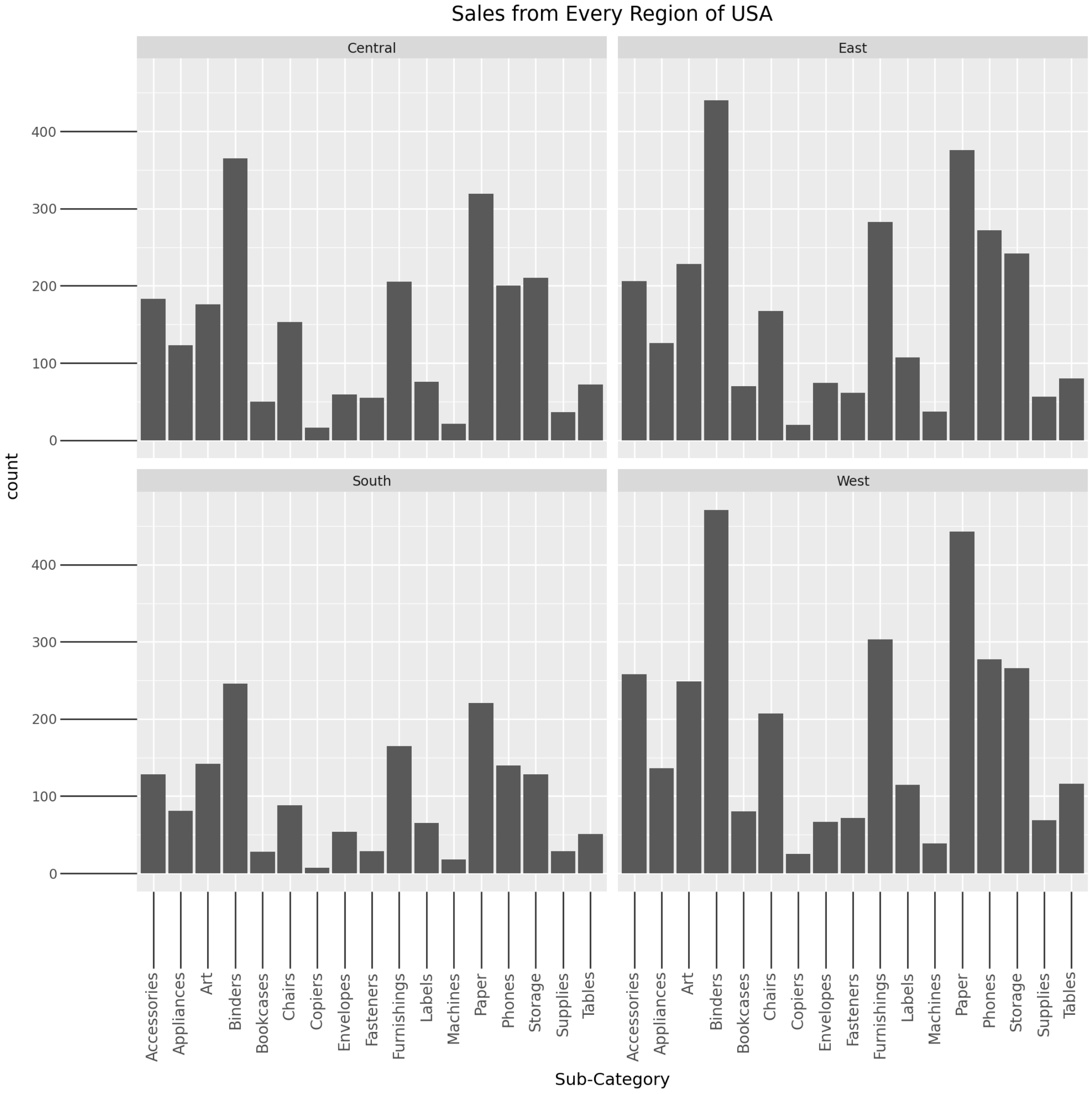


```
[26]: flip_xlabels = theme(axis_text_x = element_text(angle=90,␣
      ↪hjust=1),figure_size=(10,5),
                          axis_ticks_length_major=10,axis_ticks_length_minor=5)
      (ggplot(sample, aes(x='Sub-Category', fill='Discount'))+ geom_bar()+␣
      ↪facet_wrap(['Segment'])+flip_xlabels + theme(axis_text_x =␣
      ↪element_text(size=13))+ggtitle("Discount on Categories from Every Segment of␣
      ↪USA"))
```



[27]:

```
flip_xlabels = theme(axis_text_x = element_text(angle=90,␣
 ↪hjust=10),figure_size=(10,10),
                    axis_ticks_length_major=50,axis_ticks_length_minor=50)
(ggplot(sample, aes(x='Sub-Category', fill='Sales'))+ geom_bar()+␣
 ↪facet_wrap(['Region'])+flip_xlabels + theme(axis_text_x =␣
 ↪element_text(size=10))+ggtitle("Sales from Every Region of USA"))
```



[30]:

```
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
```

```
[31]: state_code ={
          'Alabama': 'AL',
          'Alaska': 'AK',
          'American Samoa': 'AS',
          'Arizona': 'AZ',
          'Arkansas': 'AR',
          'California': 'CA',
          'Colorado': 'CO',
          'Connecticut': 'CT',
          'Delaware': 'DE',
          'District of Columbia': 'DC',
          'Florida': 'FL',
          'Georgia': 'GA',
          'Guam': 'GU',
          'Hawaii': 'HI',
          'Idaho': 'ID',
          'Illinois': 'IL',
          'Indiana': 'IN',
          'Iowa': 'IA',
          'Kansas': 'KS',
          'Kentucky': 'KY',
          'Louisiana': 'LA',
          'Maine': 'ME',
          'Maryland': 'MD',
          'Massachusetts': 'MA',
          'Michigan': 'MI',
          'Minnesota': 'MN',
          'Mississippi': 'MS',
          'Missouri': 'MO',
          'Montana': 'MT',
          'Nebraska': 'NE',
          'Nevada': 'NV',
          'New Hampshire': 'NH',
          'New Jersey': 'NJ',
          'New Mexico': 'NM',
          'New York': 'NY',
          'North Carolina': 'NC',
          'North Dakota': 'ND',
          'Northern Mariana Islands':'MP',
          'Ohio': 'OH',
          'Oklahoma': 'OK',
          'Oregon': 'OR',
          'Pennsylvania': 'PA',
          'Puerto Rico': 'PR',
          'Rhode Island': 'RI',
          'South Carolina': 'SC',
          'South Dakota': 'SD',
    'Tennessee': 'TN',
    'Texas': 'TX',
    'Utah': 'UT',
    'Vermont': 'VT',
    'Virgin Islands': 'VI',
    'Virginia': 'VA',
    'Washington': 'WA',
    'West Virginia': 'WV',
    'Wisconsin': 'WI',
    'Wyoming': 'WY'
}
sample['state_code'] =sample.State.apply(lambda x: state_code[x])
```

[32]:

```
state_data = sample[['Sales', 'Profit', 'state_code']].groupby(['state_code']).
  ↪sum()
```

[33]:

```
fig =go.Figure(data=go.Choropleth(locations=state_data.index, z= state_data.
  ↪Sales, locationmode ='USA-states', colorscale='greens', colorbar_title␣
  ↪='Sales in USD',))
fig.update_layout(title_text = 'Total States-Wise Sales', geo_scope='usa',␣
  ↪height=800,)
fig.show()
```

Now lets Analysis the sales of few states in terms of high profit, medium, loss

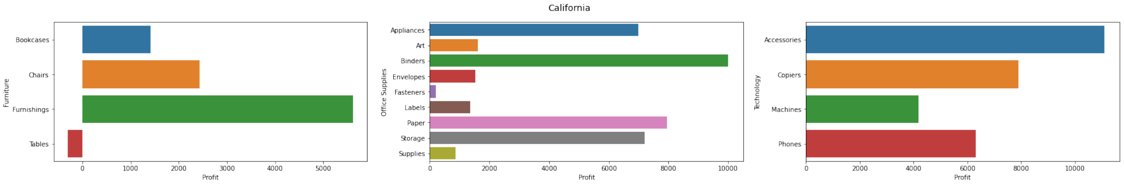
[ ]:

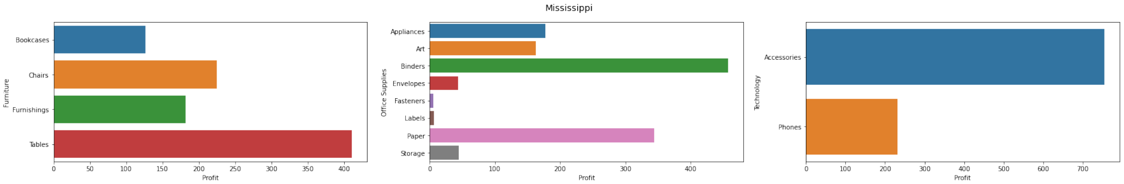
```
##### After the Analysis we can get the answers for questions like
1) Which state has most sale and what Category has more demand (i.e CALIFORNIA␣
  ↪HAS Highest SALE)
2) What are the products that are going into loss due to less or no demand(i.e␣
  ↪Taxes)
3) Which product need more improvement to make the sales drive higher
```

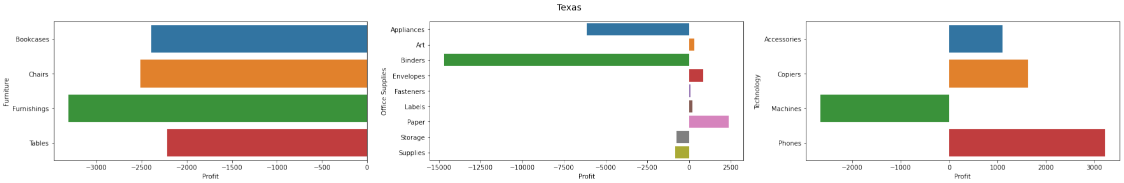
[209]:

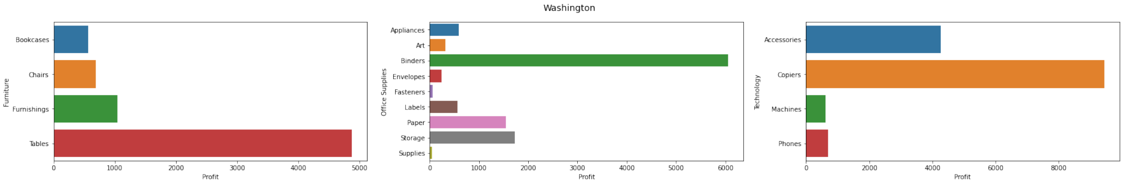
```
def state_data_viewer(states):
    """plots the turnover generated by different product categories and␣
  ↪sub-categories for the list of given states
    Args:
        states -List all the states you want to plot for
        Returns:
        None
        """
    product_data = sample.groupby(['State'])
    for state in states:
        data = product_data.get_group(state).groupby(['Category'])
        fig,ax =plt.subplots(1, 3, figsize= (30,4))
        fig.suptitle(state, fontsize=14)
        ax_index =0
        for cat in ['Furniture', 'Office Supplies', 'Technology']:
            cat_data = data.get_group(cat).groupby(['Sub-Category']).sum()
            sns.barplot(x=cat_data.Profit, y= cat_data.index, ax =ax[ax_index])
            ax[ax_index].set_ylabel(cat)
            ax_index+=1

        fig.show()
```

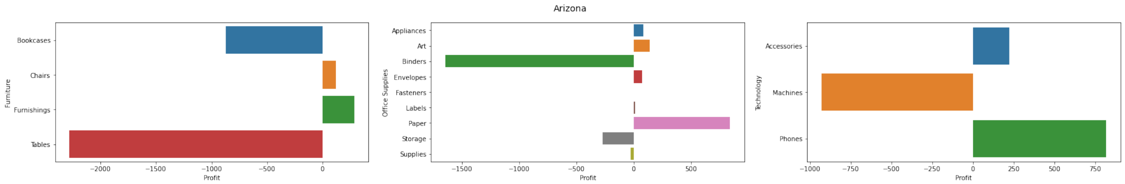
[212]:
```
states =['California', 'Mississippi', 'Texas','Washington','Arizona']
state_data_viewer(states)
```











After seeing the Charts and Visualization, We can improve the profit in other states where we see low sale by giving discount.