GRIP - MAY23 @ The Sparks Foundation

Name:- Hitesh Yadav

Task2 :- Performing EDA on dataset SampleSuperstore.

Problem statement:-

- As a business manager, try to find out the weak areas where you can work to make profit.
- What all business problems you can drive by exploring the data?

Importing the basic libraries

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
%matplotlib inline
```

Importing dataset

In [2]:

```
df = pd.read_csv("SampleSuperstore.csv")
df.head()
```

Out[2]:

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage

In [3]:

```
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
```

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.columns
```

Out[5]:

```
Index(['Ship Mode', 'Segment', 'Country', 'City', 'State', 'Postal Code',
       'Region', 'Category', 'Sub-Category', 'Sales', 'Quantity', 'Discoun
t',
       'Profit'],
      dtype='object')
```

In [6]:

```
df['Ship Mode'].unique()
```

Out[6]:

```
array(['Second Class', 'Standard Class', 'First Class', 'Same Day'],
      dtype=object)
```

In [7]:

```
df['Country'].unique()
```

Out[7]:

```
array(['United States'], dtype=object)
```

In [8]:

```
# there is only single country so we can del this column.
df.drop("Country",axis = 1,inplace= True)
```

In [ ]:

```

```

In [9]:

```
df["Total_sale"] = df['Sales']*df['Quantity']
df.head()
```

Out[9]:

	Ship Mode	Segment	City	State	Postal Code	Region	Category	Sub-Category	Sales
0	Second Class	Consumer	Henderson	Kentucky	42420	South	Furniture	Bookcases	261.9600
1	Second Class	Consumer	Henderson	Kentucky	42420	South	Furniture	Chairs	731.9400
2	Second Class	Corporate	Los Angeles	California	90036	West	Office Supplies	Labels	14.6200
3	Standard Class	Consumer	Fort Lauderdale	Florida	33311	South	Furniture	Tables	957.5775
4	Standard Class	Consumer	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	22.3680

In [10]:

```
# now we do not need the sales and quatity column.
df.drop(["Sales","Quantity"],axis = 1,inplace = True)
```

In [ ]:

```
```

In [11]:

```
top10_state = df.groupby(by = 'State').agg({"Total_sale":"sum"}).sort_values(by = "Total
top10_state
```

Out[11]:

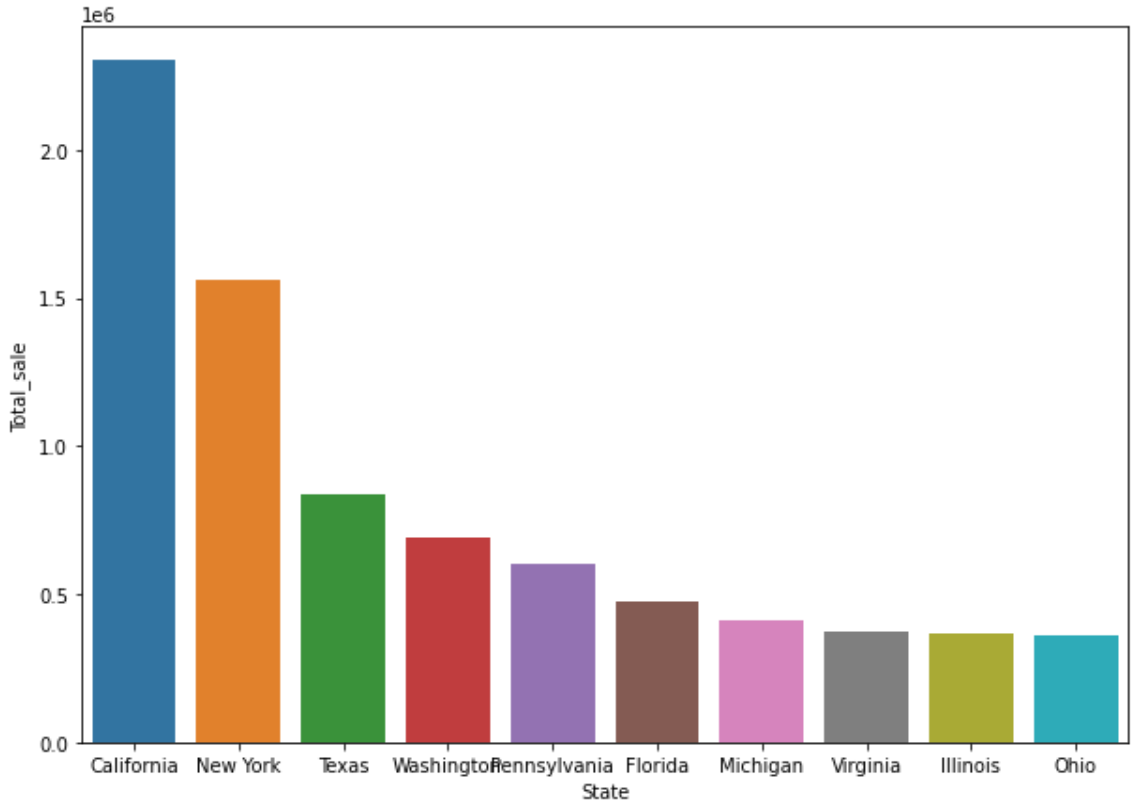
	State	Total_sale
0	California	2.301218e+06
1	New York	1.561073e+06
2	Texas	8.340883e+05
3	Washington	6.923602e+05
4	Pennsylvania	6.021270e+05
5	Florida	4.720913e+05
6	Michigan	4.103509e+05
7	Virginia	3.764538e+05
8	Illinois	3.650843e+05
9	Ohio	3.625338e+05

In [12]:

```
# Checking the country on the bases of the country.

plt.figure(figsize=(10,7))
sns.barplot(x = "State" ,y = 'Total_sale',data = top10_state)
```

Out[12]:

```
<AxesSubplot:xlabel='State', ylabel='Total_sale'>
```



We can conclude that the California state has the maximum sales then New York and then Texas.

In [13]:

```
top10_subcategory = df.groupby('Sub-Category')['Profit'].sum().to_frame("Total_profit").
top10_subcategory
```

Out[13]:

	Sub-Category	Total_profit
0	Copiers	55617.8249
1	Phones	44515.7306
2	Accessories	41936.6357
3	Paper	34053.5693
4	Binders	30221.7633
5	Chairs	26590.1663
6	Storage	21278.8264
7	Appliances	18138.0054
8	Furnishings	13059.1436
9	Envelopes	6964.1767

In [22]:

```
loss_category = df.groupby('Sub-Category')['Profit'].sum().to_frame("Total_profit").sort
loss_category
```

Out[22]:

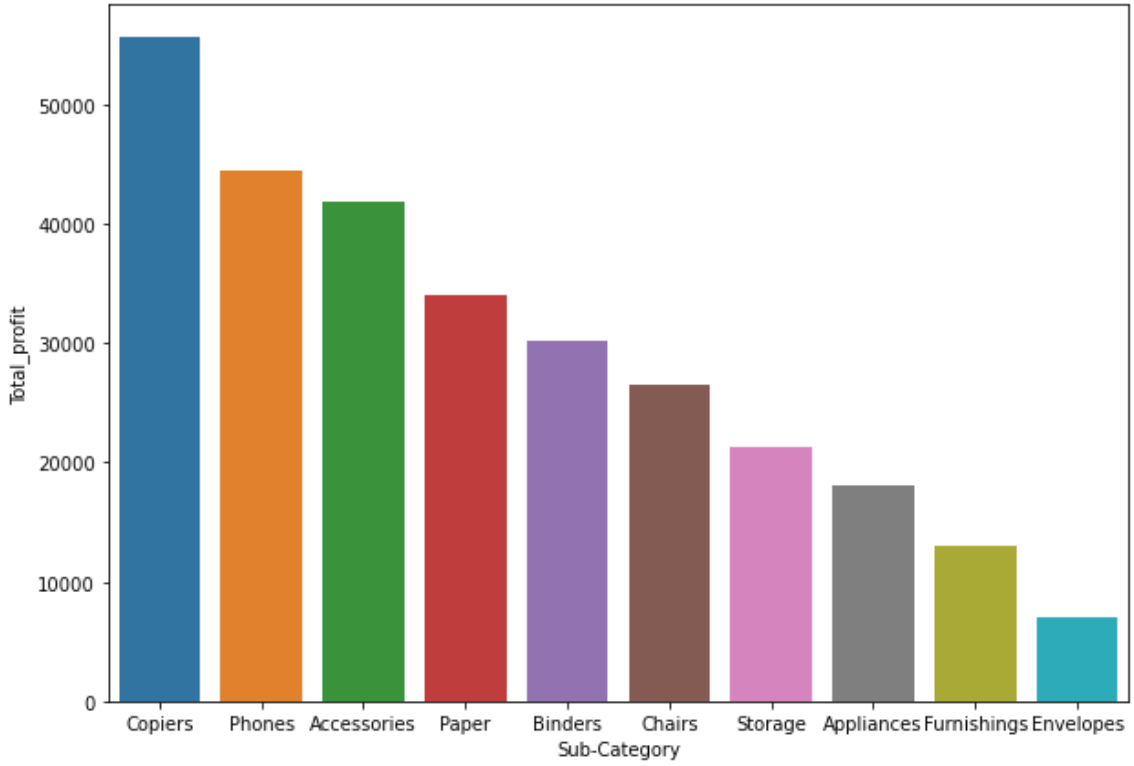
	Sub-Category	Total_profit
0	Tables	-17725.4811
1	Bookcases	-3472.5560
2	Supplies	-1189.0995
3	Fasteners	949.5182
4	Machines	3384.7569
5	Labels	5546.2540
6	Art	6527.7870
7	Envelopes	6964.1767
8	Furnishings	13059.1436
9	Appliances	18138.0054

In [14]:

```
plt.figure(figsize=(10,7))
sns.barplot(x = 'Sub-Category' , y = 'Total_profit',data = top10_subcategory)
```

Out[14]:

```
<AxesSubplot:xlabel='Sub-Category', ylabel='Total_profit'>
```



Copiers make the maximum profit.

Tables , Bookcases and Supplies are the category which gives us loss.

In [ ]:

```
```

In [15]:

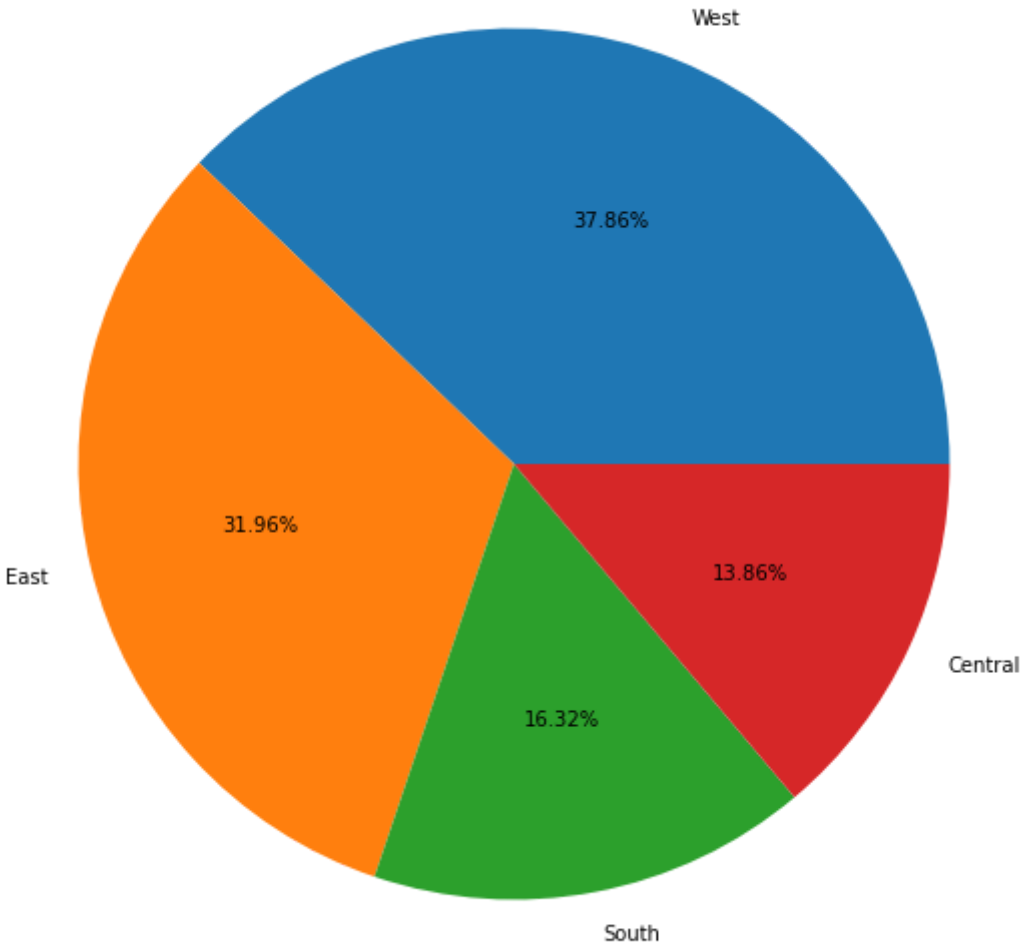
```
Region_profit = df.groupby("Region")['Profit'].sum().to_frame("Total_profit").sort_value
Region_profit
```

Out[15]:

	Region	Total_profit
0	West	108418.4489
1	East	91522.7800
2	South	46749.4303
3	Central	39706.3625

In [16]:

```
plt.figure(figsize=(10,10))
plt.pie('Total_profit',labels = "Region",data = Region_profit,autopct = "%.2f%%")
plt.show()
```



West region has the maximum profit.

In [17]:

```
category_profit = df.groupby('Category')['Profit'].sum().reset_index().sort_values(by =
```

In [18]:

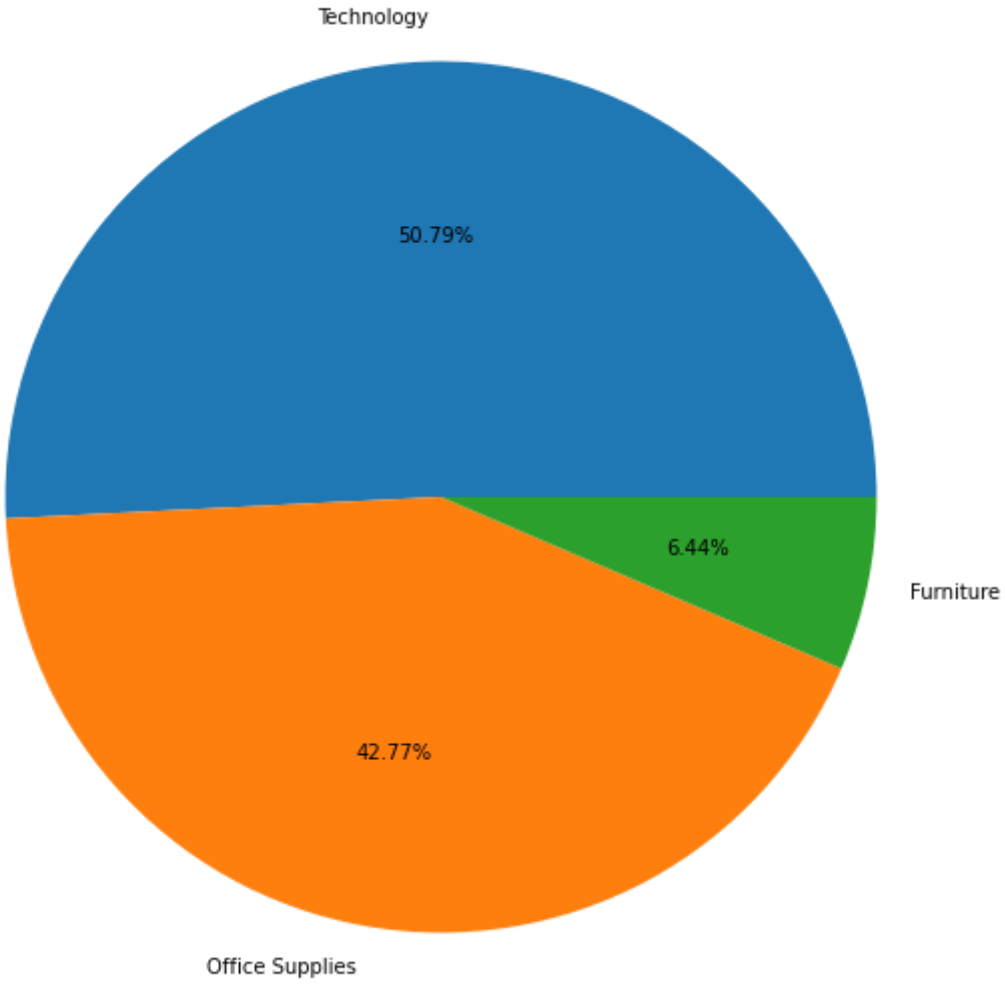
```
category_profit
```

Out[18]:

	Category	Profit
2	Technology	145454.9481
1	Office Supplies	122490.8008
0	Furniture	18451.2728

In [19]:

```
plt.figure(figsize=(10,10))
plt.pie('Profit',labels = "Category",data = category_profit,autopct = "%.2f%%")
plt.show()
```



In [20]:

```
df["Ship Mode"].value_counts()
```

Out[20]:

```
Standard Class    5968
Second Class      1945
First Class       1538
Same Day           543
Name: Ship Mode, dtype: int64
```

Maximum number of buyer prefer standard class shipping.

In [21]:

```
df.head()
```

Out[21]:

	Ship Mode	Segment	City	State	Postal Code	Region	Category	Sub-Category	Discount
0	Second Class	Consumer	Henderson	Kentucky	42420	South	Furniture	Bookcases	0.00
1	Second Class	Consumer	Henderson	Kentucky	42420	South	Furniture	Chairs	0.00
2	Second Class	Corporate	Los Angeles	California	90036	West	Office Supplies	Labels	0.00
3	Standard Class	Consumer	Fort Lauderdale	Florida	33311	South	Furniture	Tables	0.45
4	Standard Class	Consumer	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	0.20

In [27]:

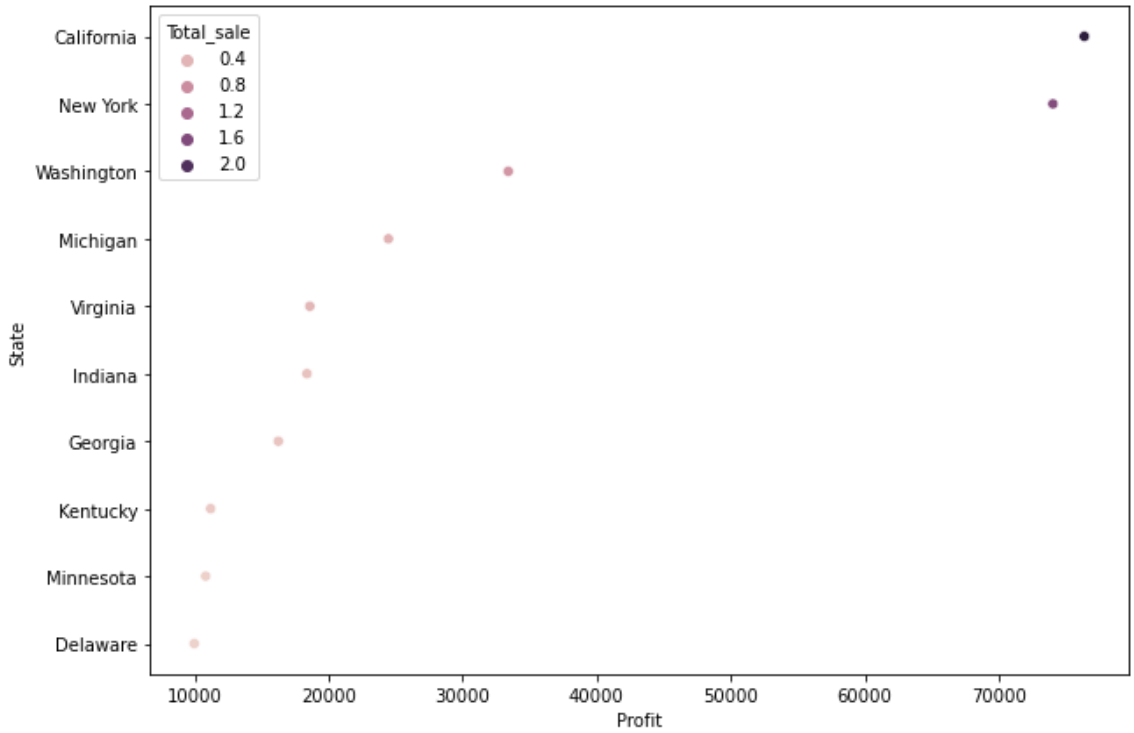
```
# sales and profit on the bases of the state.

max_profit = df.groupby("State").agg({"Profit":"sum","Total_sale":"sum"}).reset_index().
max_profit
```

Out[27]:

	State	Profit	Total_sale
3	California	76381.3871	2.301218e+06
30	New York	74038.5486	1.561073e+06
45	Washington	33402.6517	6.923602e+05
20	Michigan	24463.1876	4.103509e+05
44	Virginia	18597.9504	3.764538e+05
12	Indiana	18382.9363	2.670377e+05
9	Georgia	16250.0433	2.422653e+05
15	Kentucky	11199.6966	1.972006e+05
21	Minnesota	10823.1874	1.349785e+05
6	Delaware	9977.3748	1.156586e+05

In [32]:

```
plt.figure(figsize=(10,7))
sns.scatterplot(y = "State",x = "Profit",hue = "Total_sale",data = max_profit)
plt.show()
```



California makes the highest sales and highest profit followed by New York

In [ ]:

```

```

In [26]:

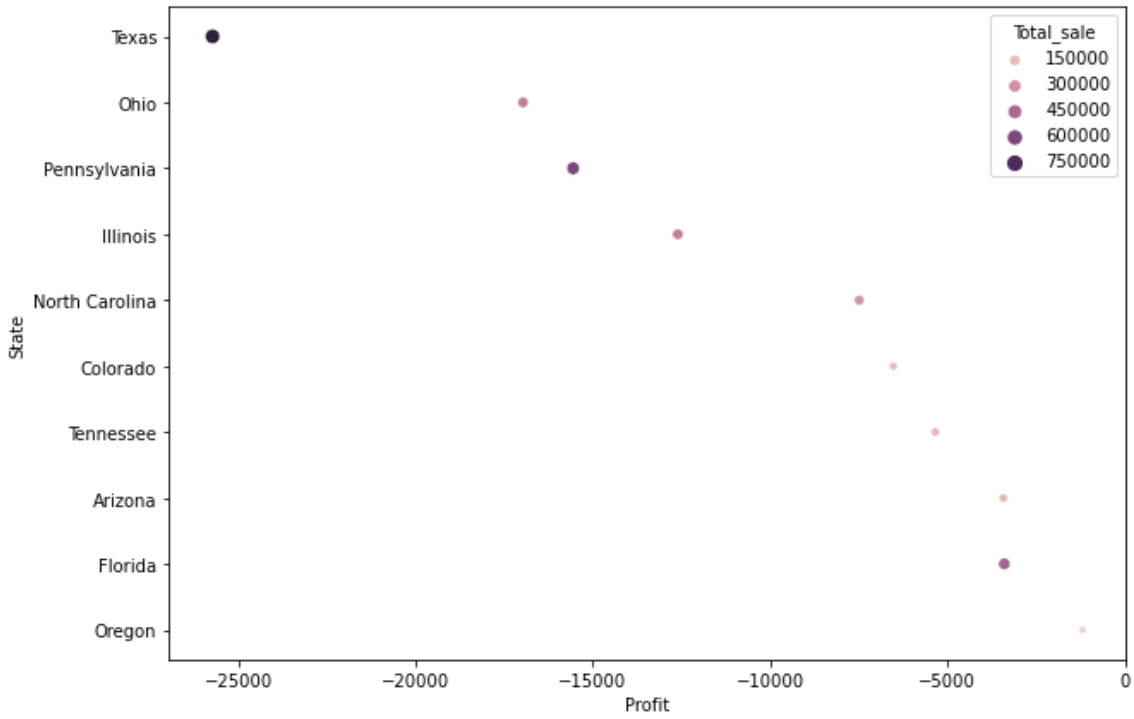
```
Loss_state = df.groupby("State").agg({"Profit":"sum","Total_sale":"sum"}).reset_index().
Loss_state
```

Out[26]:

	State	Profit	Total_sale
41	Texas	-25729.3563	834088.3314
33	Ohio	-16971.3766	362533.7980
36	Pennsylvania	-15559.9603	602126.9600
11	Illinois	-12607.8870	365084.2910
31	North Carolina	-7490.9122	282777.5440
4	Colorado	-6527.8579	145591.1180
40	Tennessee	-5341.6936	158654.9030
1	Arizona	-3427.9246	170003.3770
8	Florida	-3399.3017	472091.2830
35	Oregon	-1190.4705	79009.0640

In [34]:

```
plt.figure(figsize=(10,7))
sns.scatterplot(y = "State",x = "Profit",hue = "Total_sale",data = Loss_state,size = "To
plt.show()
```



Observations

- California is giving the Maximum sales followed by New York and Texas.
- The copier in the sub-category gives the maximum profit of 55.617k.
- Table sub-category is giving the maximum loss of 17.725k.

- West region is giving the maximum profit of 37.8%.
- Technology is the category which is giving us the maximum profit.
- Californina makes the highest profit and sales.
- Texas makes the sales but there is loss in the Texas of about 25.729k.

In [ ]: