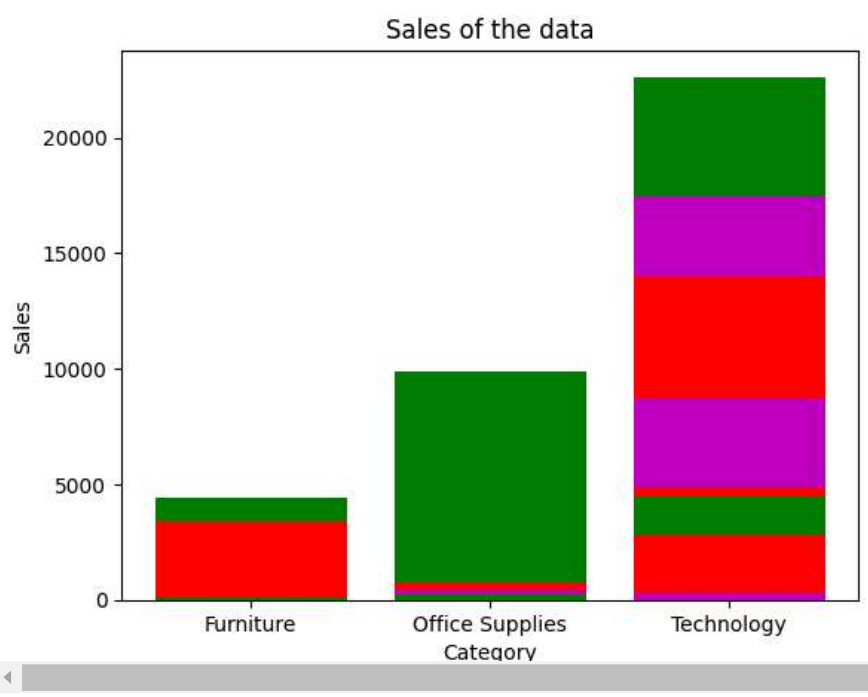
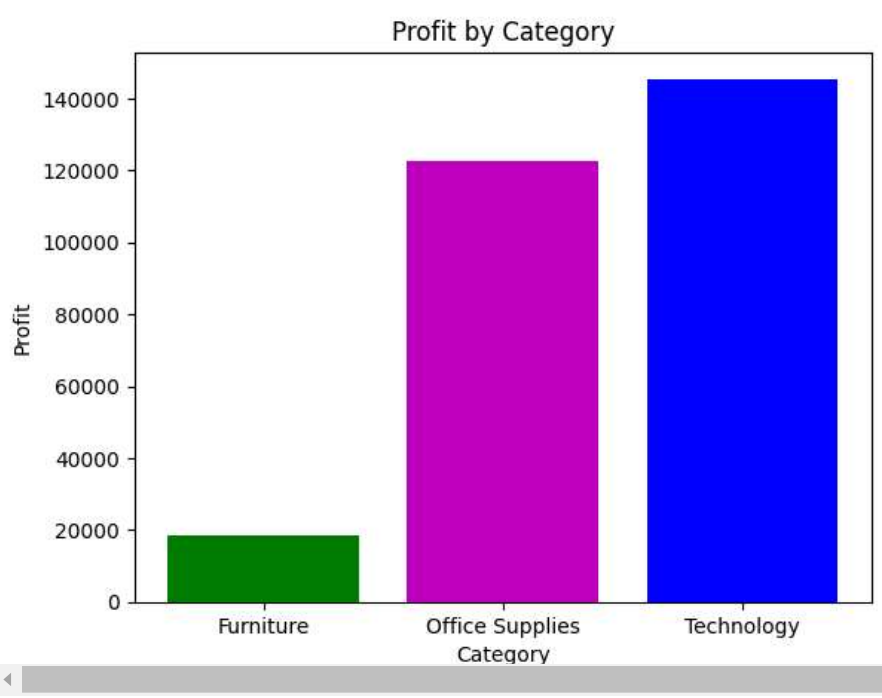
```
import pandas as pd
superstore=pd.read_excel("/content/Superstore.xls")
import plotly.express as px
import plotly
import numpy as np
import sys
import matplotlib
import matplotlib.pyplot as plt
from matplotlib import style
```

```
no=superstore['Ship Mode']
n1=superstore['no']
Sales=superstore['Sales']
```

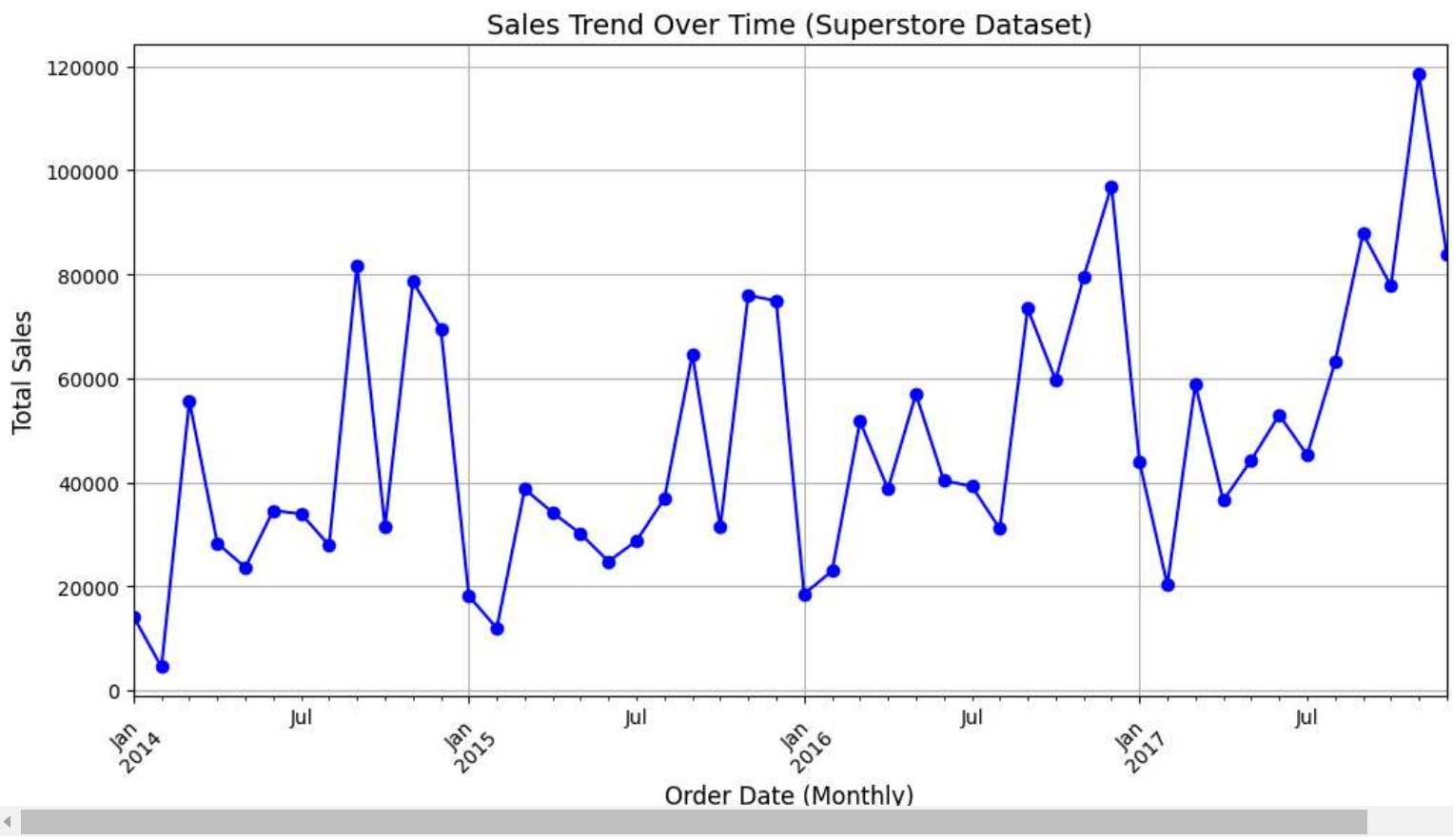
```
plt.bar(superstore['Category'],superstore['Sales'],color=['g','m','r'])
plt.xlabel('Category')
plt.ylabel('Sales')
plt.title('Sales of the data')
plt.show()
```



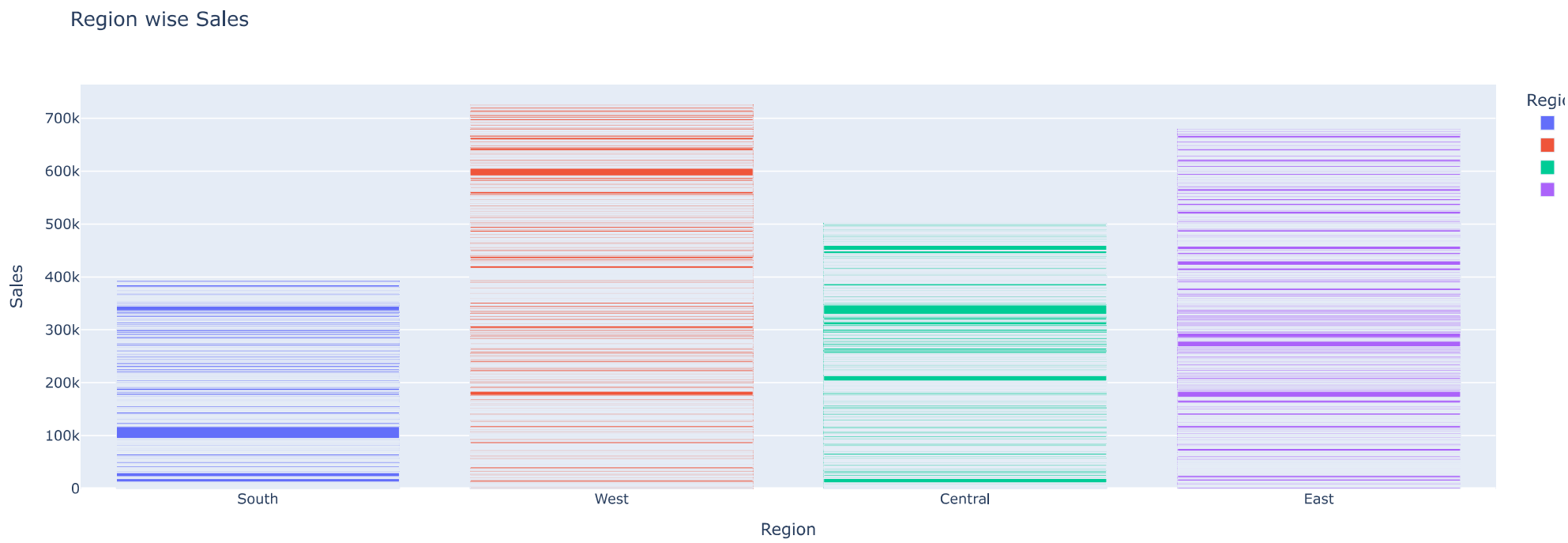

```
category_profit = superstore.groupby('Category')['Profit'].sum()
plt.bar(category_profit.index, category_profit.values, color=['g', 'm', 'b'])  # Add more colors if needed
plt.xlabel('Category')
plt.ylabel('Profit')
plt.title('Profit by Category')
plt.show()
```



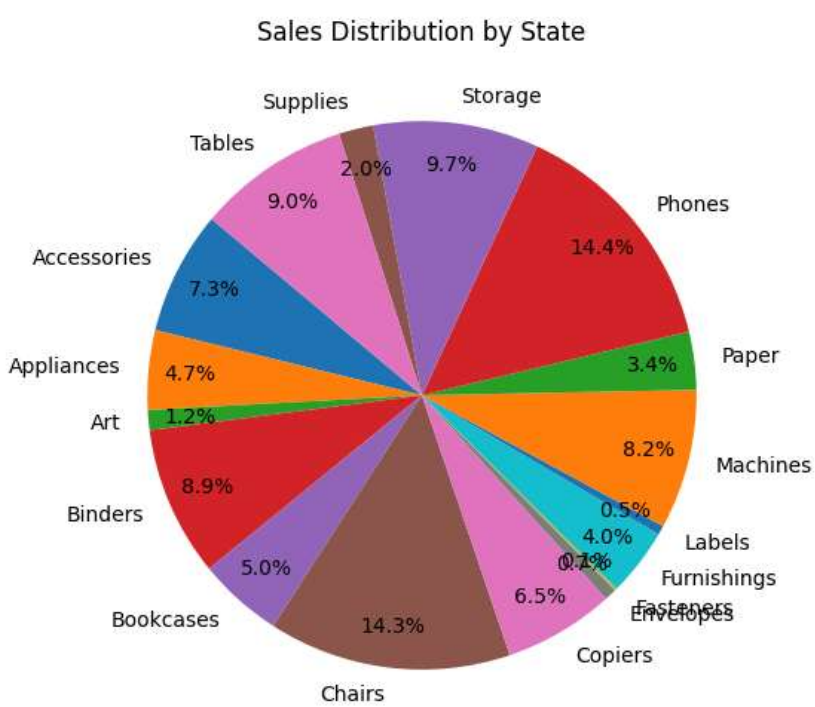

```
superstore['Order Date'] = pd.to_datetime(superstore['Order Date'], errors='coerce')
superstore = superstore.dropna(subset=['Order Date'])
sales_trend = superstore.groupby(superstore['Order Date'].dt.to_period("M"))['Sales'].sum()
plt.figure(figsize=(12, 6))
sales_trend.plot(marker='o', linestyle='-', color='b')

plt.title("Sales Trend Over Time (Superstore Dataset)", fontsize=14)
plt.xlabel("Order Date (Monthly)", fontsize=12)
plt.ylabel("Total Sales", fontsize=12)
plt.grid(True)
plt.xticks(rotation=45)
plt.show()
```



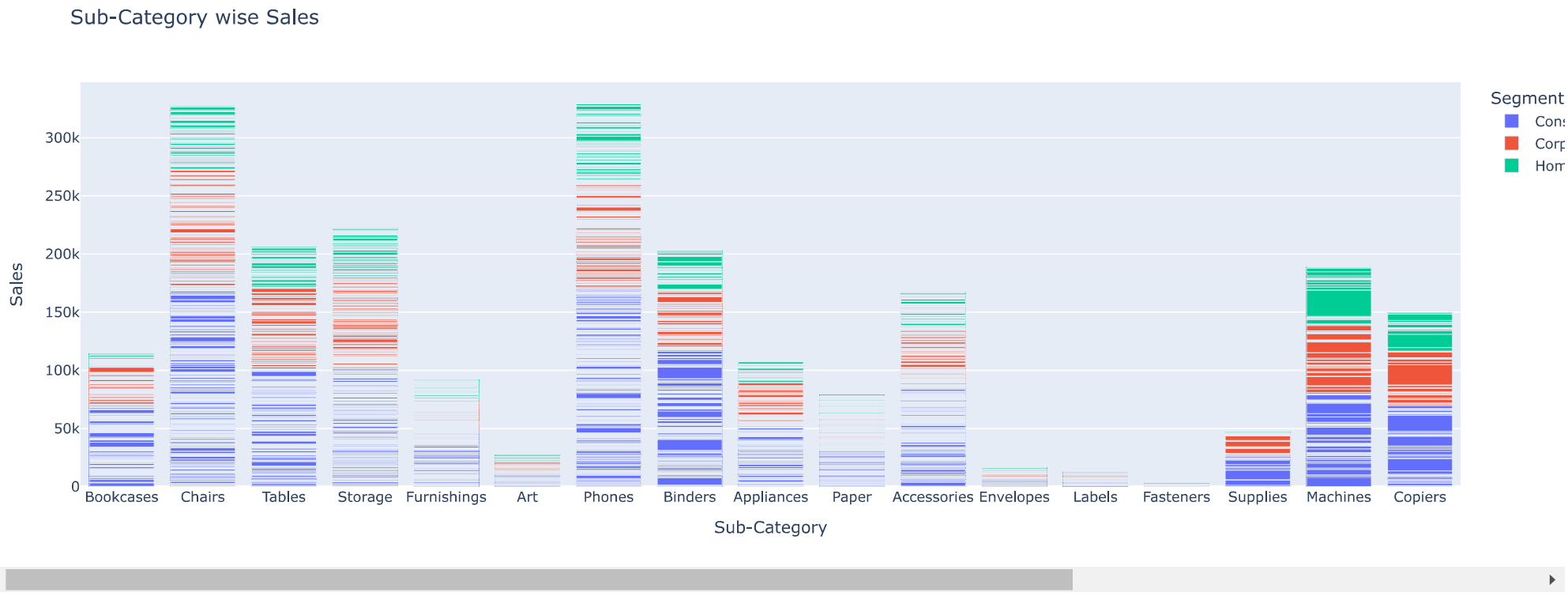

```
fig=px.bar(superstore,x='Region',y='Sales',color='Region',title='Region wise Sales')
fig.show()
```




```
state_sales = superstore.groupby('Sub-Category')['Sales'].sum()
plt.figure(figsize=(7, 6))
plt.pie(state_sales,labels=state_sales.index, autopct='%1.1f%%',startangle=140,pctdistance=0.85,)
plt.title('Sales Distribution by State')
plt.show()
```

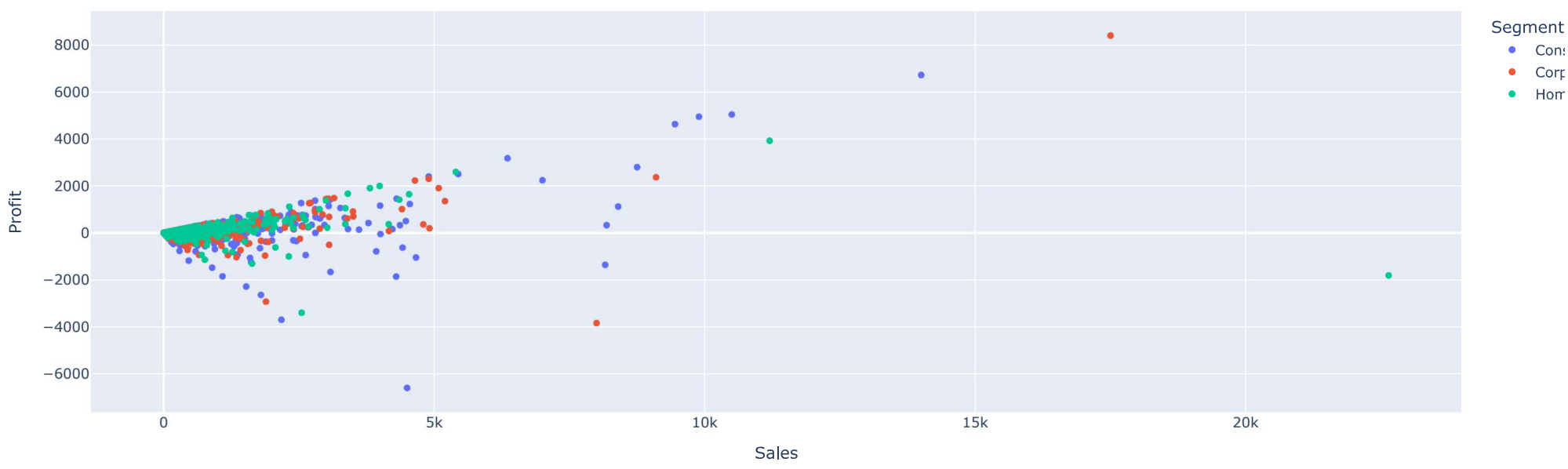



```
px.bar(superstore,x='Sub-Category',y='Sales',color='Segment',title='Sub-Category wise Sales')
```

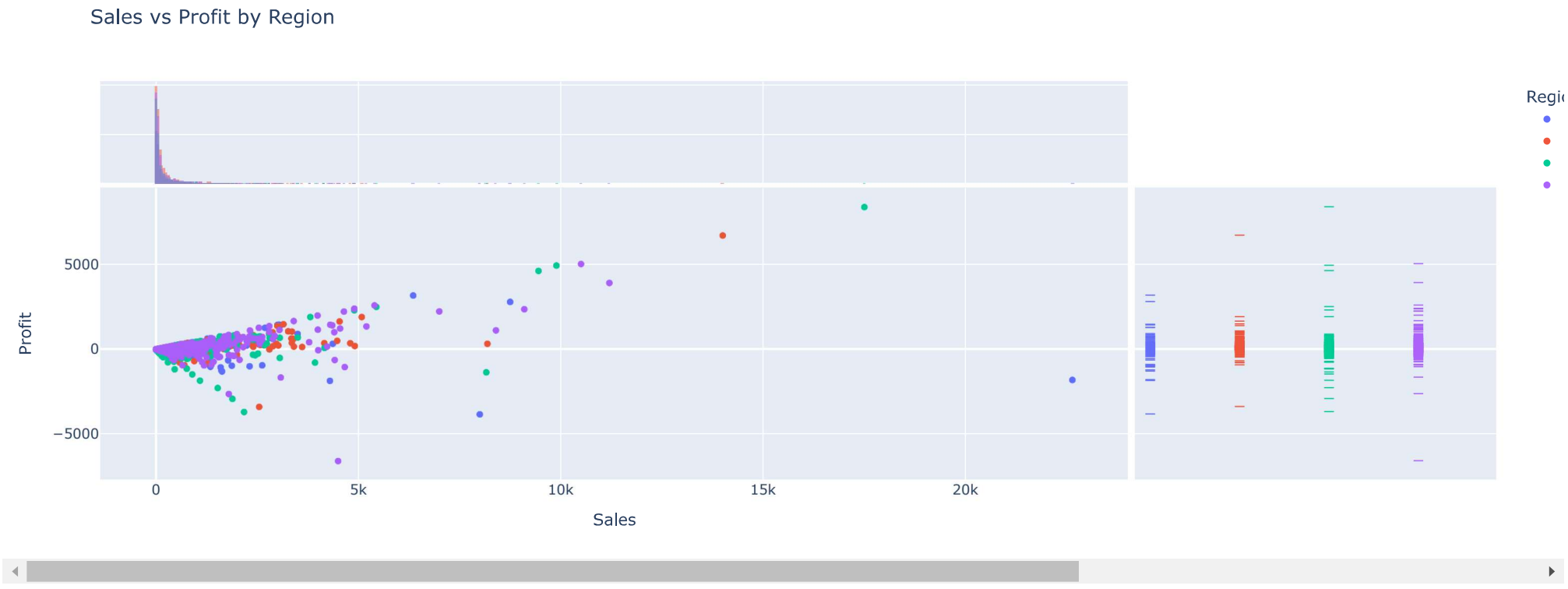



```
selected_cities = ['New York City', 'Los Angeles', 'San Francisco','Marlborough','Bryan','Garden City']
filtered_data = superstore[superstore['City'].isin(selected_cities)]
Pro = px.line(filtered_data, x='City', y='Profit', color='City', title='Profit Trend for Selected Cities',markers=True)
Pro.show()
```

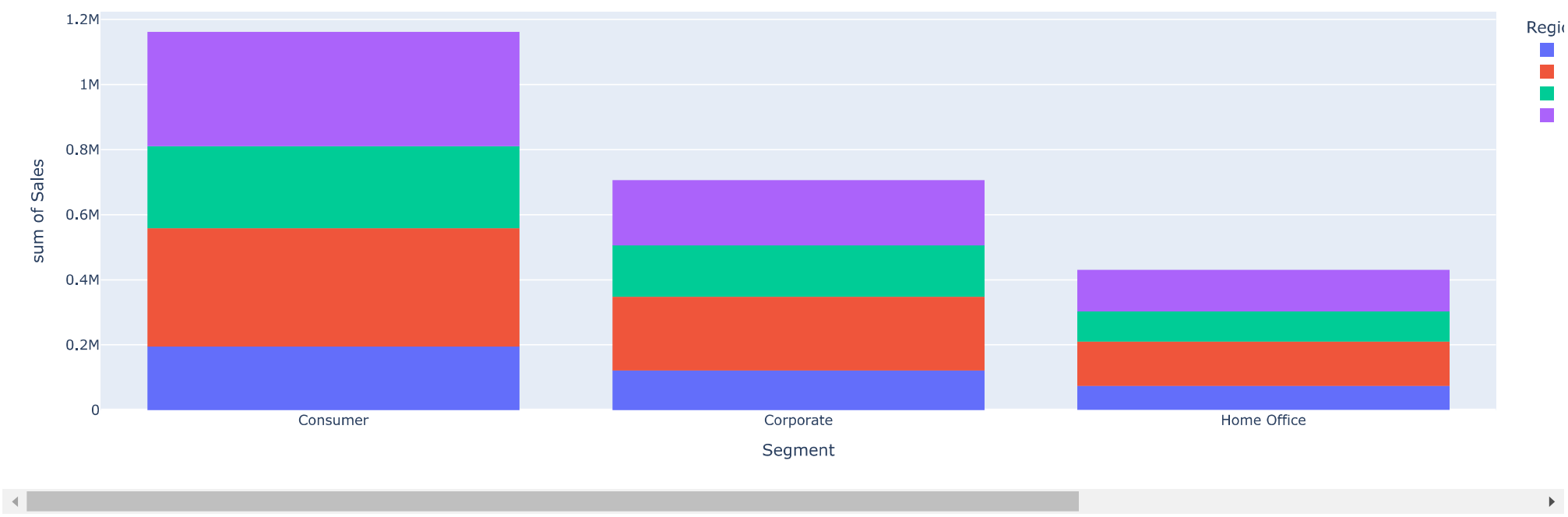
```
px.scatter(superstore,x='Sales',y='Profit',color='Segment',title='Scatterplot for Sales and Profit')
```



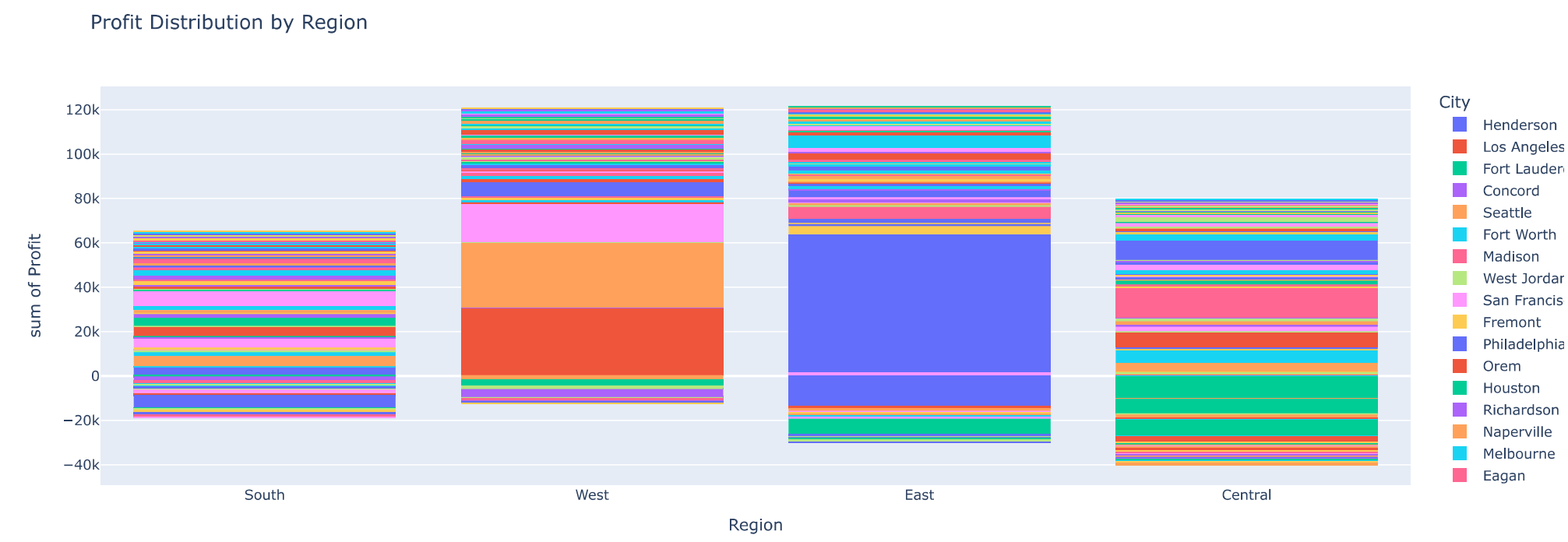

```
fig=px.scatter(superstore,x='Sales',y='Profit',color='Region',marginal_x='histogram',marginal_y='rug',title='Sales vs Profit by Region')
fig.update_layout(xaxis_title="Sales",yaxis_title="Profit",legend_title="Region",)
fig.show()
```



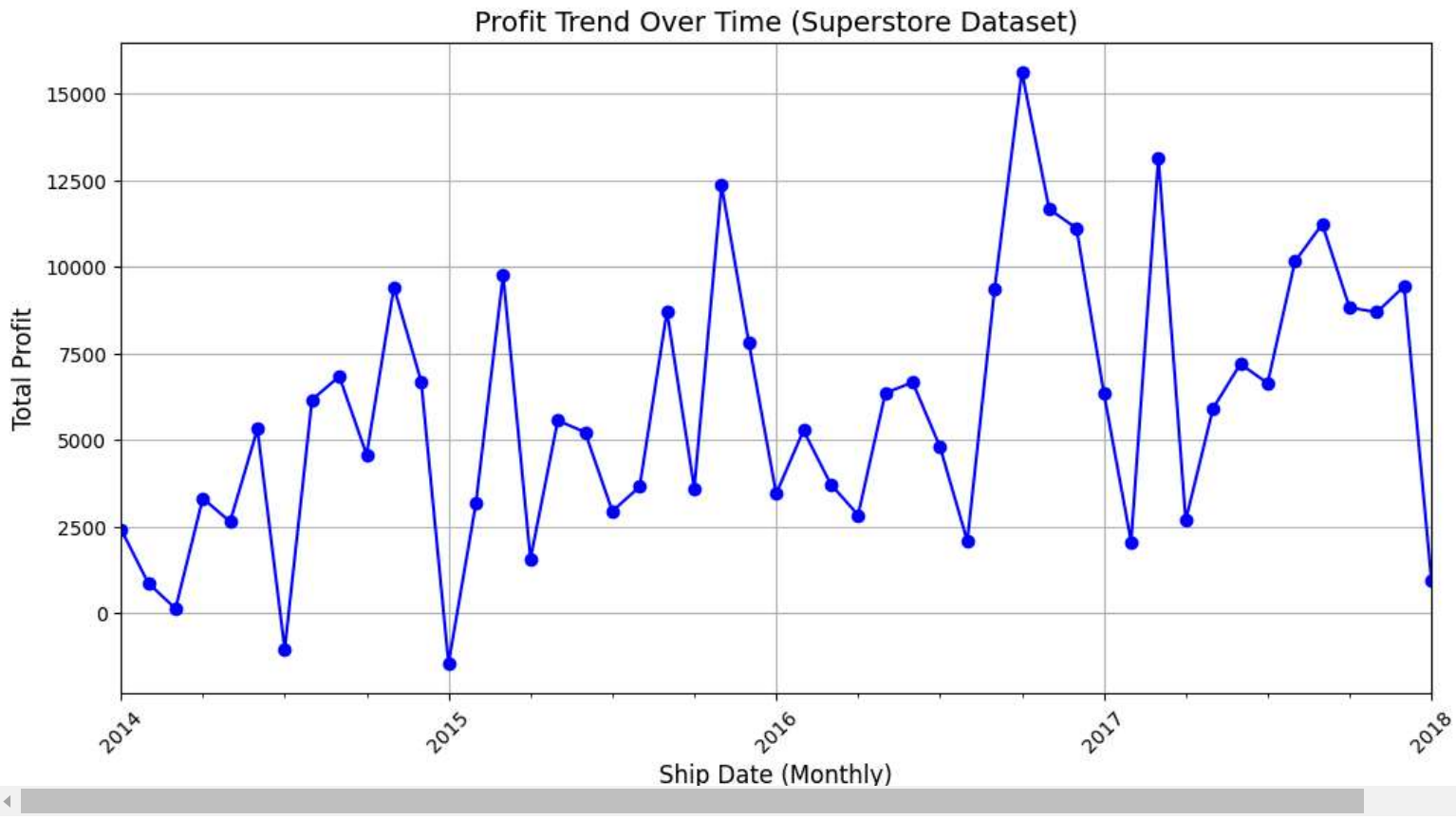

```
px.histogram(superstore,x='Segment',y='Sales',color='Region',title='Sales Distribution by Region')
```



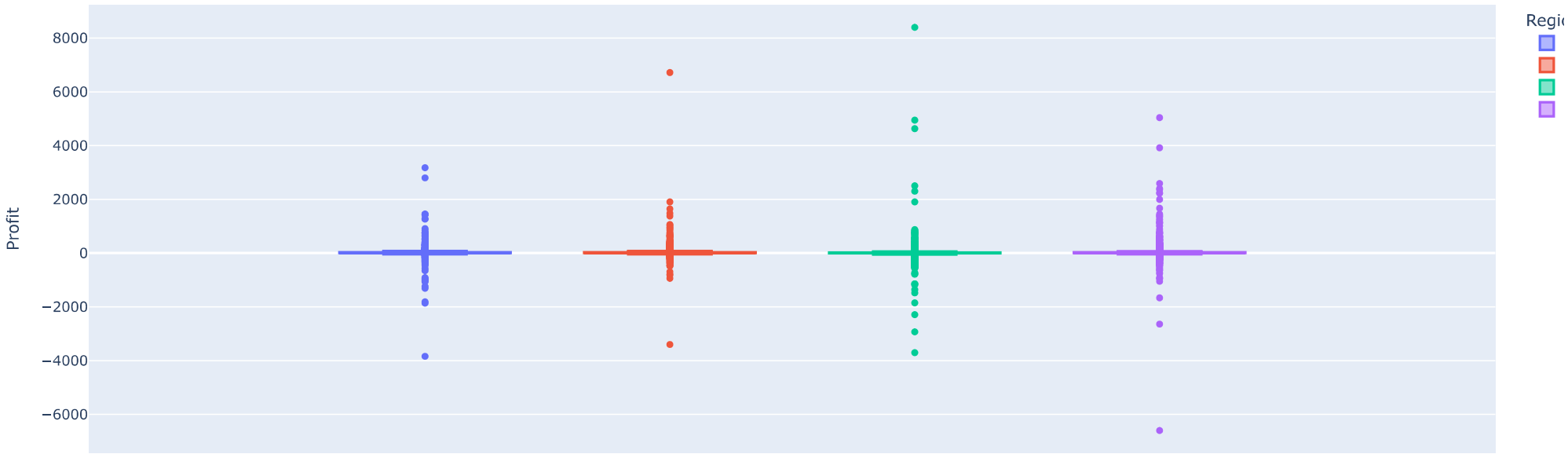

```
px.histogram(superstore,x='Region',y='Profit',color='City',title='Profit Distribution by Region')
```



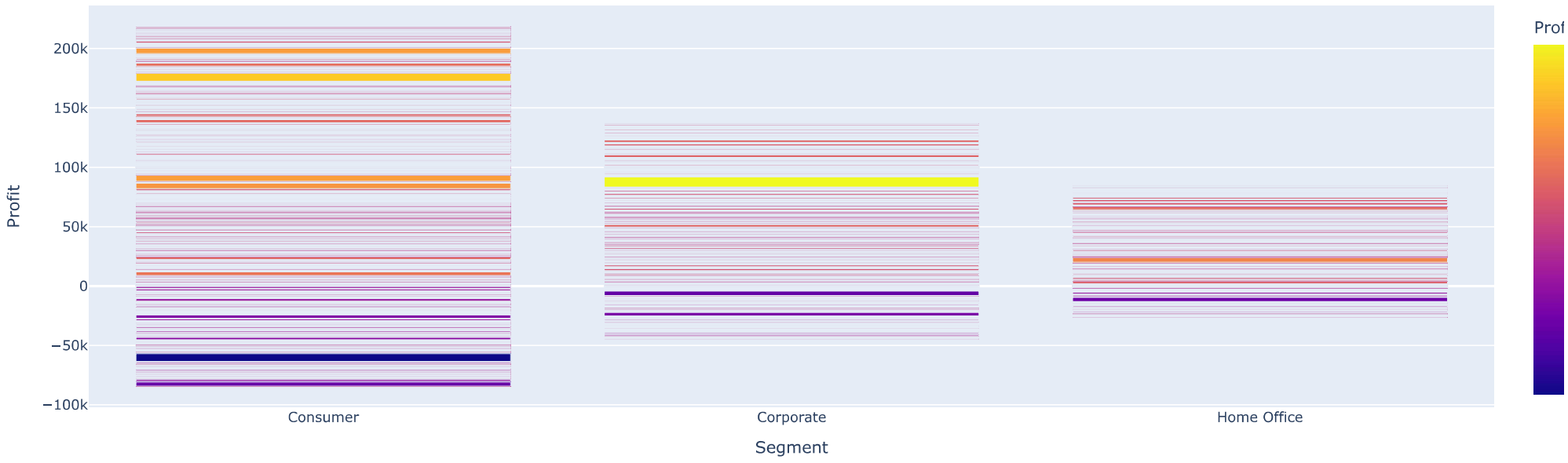

```
superstore['Order Date'] = pd.to_datetime(superstore['Order Date'], errors='coerce')
superstore = superstore.dropna(subset=['Ship Date'])
sales_trend = superstore.groupby(superstore['Ship Date'].dt.to_period("M"))['Profit'].sum()
plt.figure(figsize=(12, 6))
sales_trend.plot(marker='o', linestyle='-', color='b')
plt.title("Profit Trend Over Time (Superstore Dataset)", fontsize=14)
plt.xlabel("Ship Date (Monthly)", fontsize=12)
plt.ylabel("Total Profit", fontsize=12)
plt.grid(True)
plt.xticks(rotation=45)
plt.show()
```



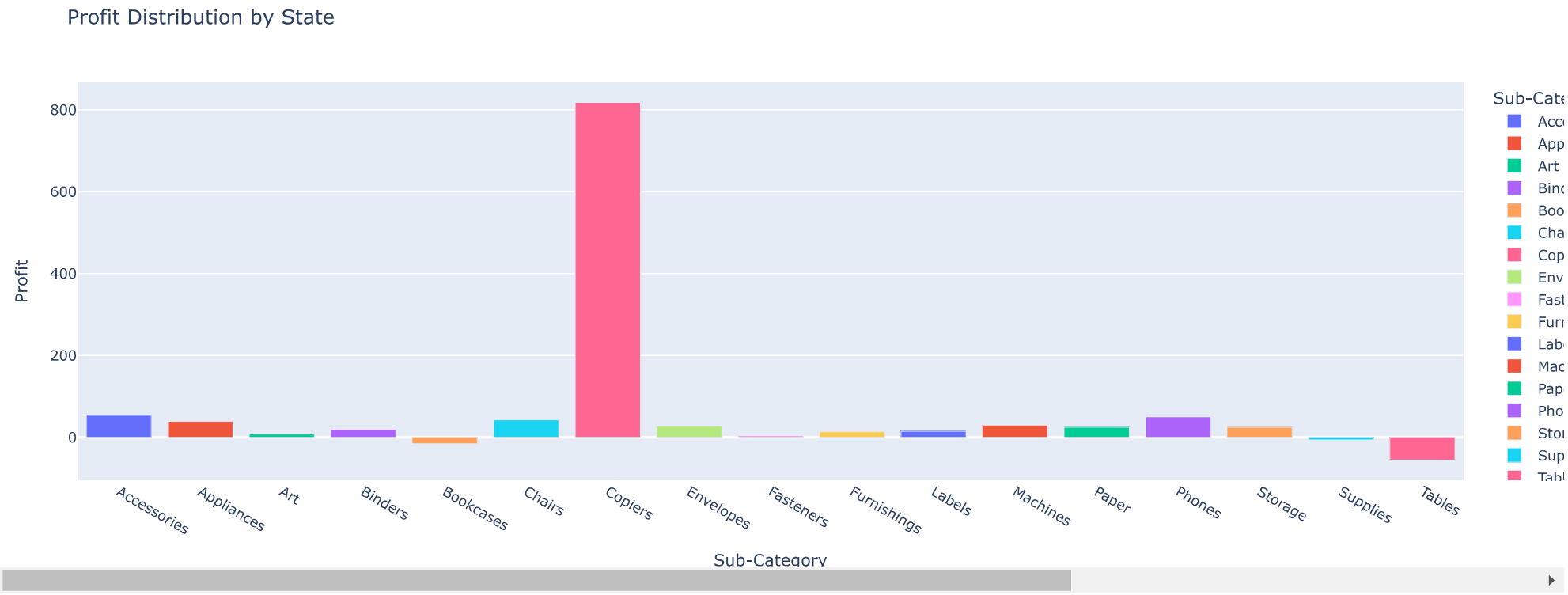

```
px.box(superstore,y='Profit',color='Region')
```




```
px.bar(superstore,x='Segment',y='Profit',color='Profit',title='Profit Distribution by Segment')
```



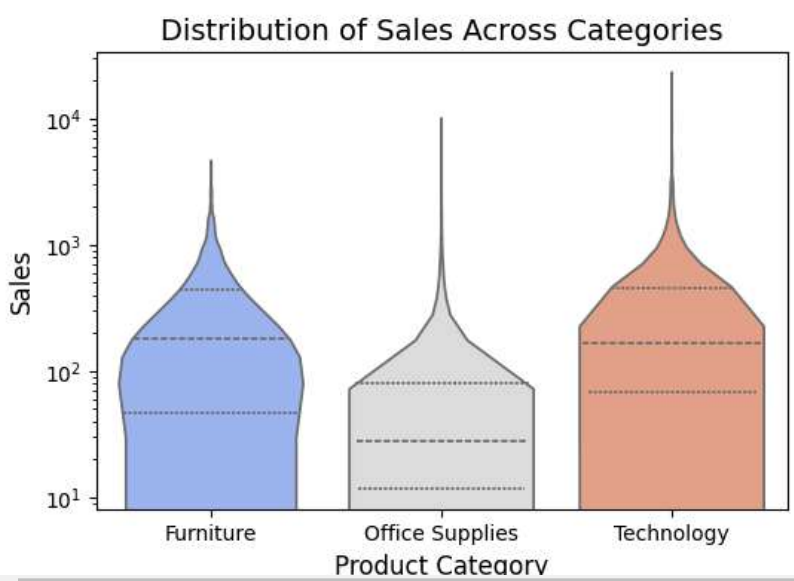

```
a=superstore.groupby('Sub-Category')['Profit'].mean().reset_index()
px.bar(a,x='Sub-Category',y='Profit',color='Sub-Category',title='Profit Distribution by State')
```



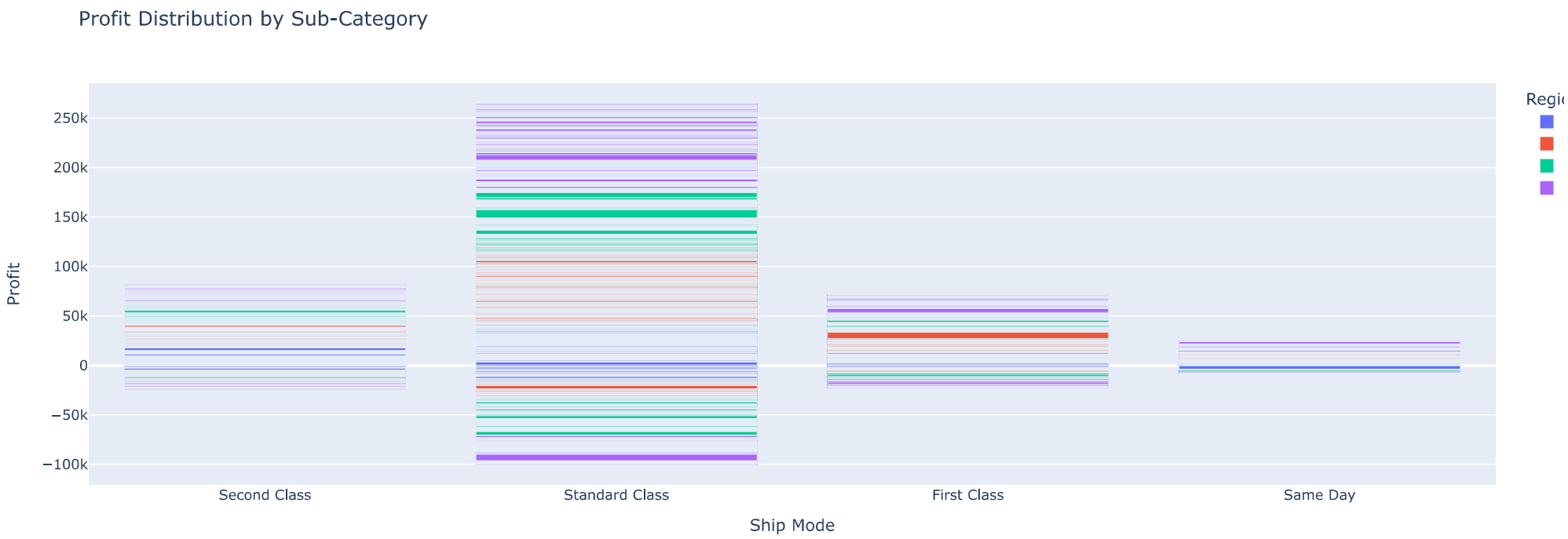

```
a = superstore.groupby('Ship Mode')['Sales'].mean().reset_index()
plt.figure(figsize=(6,4))
fig = px.bar(superstore, x='Ship Mode', y='Sales', color='Region', title='Profit Distribution by Sub-Category')
fig.update_traces(opacity=1.0)
fig.show()
```

```
import seaborn as sns
superstore['Order Date'] = pd.to_datetime(superstore['Order Date'], errors='coerce')
superstore = superstore.dropna(subset=['Order Date'])
plt.figure(figsize=(6,4))
sns.violinplot(x="Category", y="Sales", data=superstore, inner="quartile", palette="coolwarm")
plt.title("Distribution of Sales Across Categories", fontsize=14)
plt.xlabel("Product Category", fontsize=12)
plt.ylabel("Sales", fontsize=12)
plt.yscale("log")
plt.show()
```

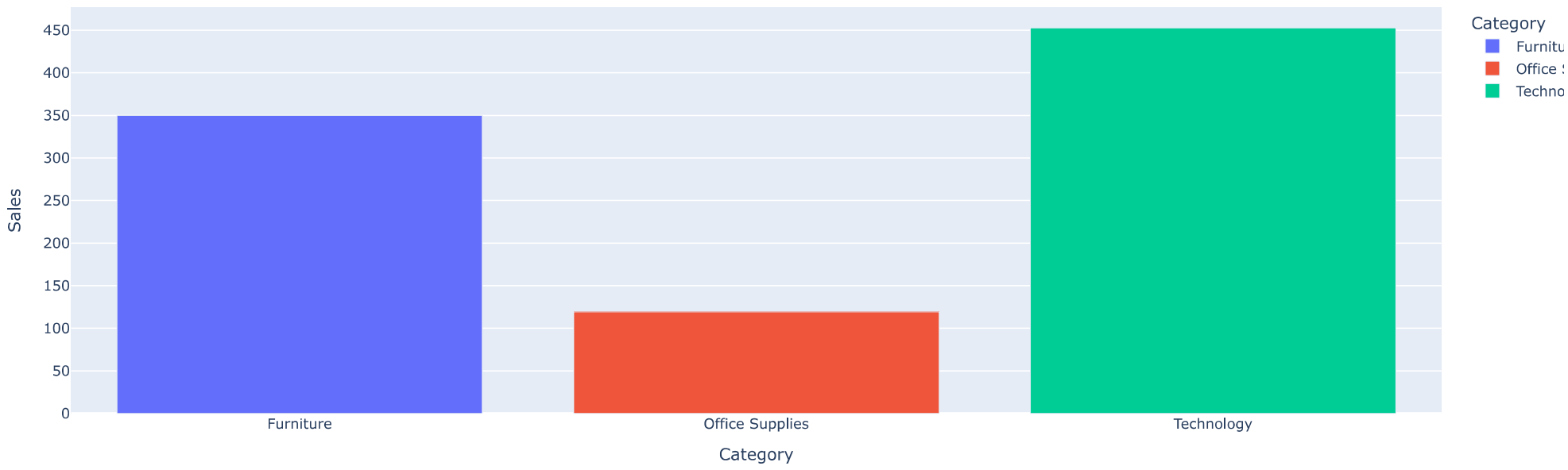
```
<ipython-input-109-f79545287772>:7: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.
```



```
a = superstore.groupby('Ship Mode')['Profit'].mean().reset_index()
plt.figure(figsize=(6,4))
fig = px.bar(superstore, x='Ship Mode', y='Profit', color='Region', title='Profit Distribution by Sub-Category')
fig.update_traces(opacity=1.0)
fig.show()
```




```
a = superstore.groupby('Category')['Sales'].mean().reset_index()
plt.figure(figsize=(6,4))
fig = px.bar(a, x='Category', y='Sales', color='Category', title='Sales Distribution by Category')
fig.show()
```




```
plt.figure(figsize=(6,4))
fig = px.bar(superstore,x="Category",y="Sales",color="Region",title="Stacked Bar Chart : Sales by Category and Region", barmode="stack",  # Stacked bars
    color_discrete_sequence=px.colors.qualitative.Dark24)
fig.update_traces(opacity=1.0)
fig.show()
```