

## TASK 1:

Untitled6.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

les

..  
sample\_data  
Stores.csv


+ Code + Text

```
[1] import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
[3] data = pd.read_csv('/content/Stores.csv')
data.head()
```

	Store ID	Store_Area	Items_Available	Daily_Customer_Count	Store_Sales
0	1	1659	1961	530	66490
1	2	1461	1752	210	39820
2	3	1340	1609	720	54010
3	4	1451	1748	620	53730
4	5	1770	2111	450	46620

Next steps: [Generate code with data](#) [View recommended plots](#)


0s  `data.tail()`

	Store ID	Store_Area	Items_Available	Daily_Customer_Count	Store_Sales
891	892	1582	1910	1080	66390
892	893	1387	1663	850	82080
893	894	1200	1436	1060	76440
894	895	1299	1560	770	96610
895	896	1174	1429	1110	54340


0s [6] `data.shape`

 (896, 5)

0s [7] `data.info()`

 <class 'pandas.core.frame.DataFrame'>  
 RangeIndex: 896 entries, 0 to 895  
 Data columns (total 5 columns):  
 #    Column                      Non-Null Count    Dtype  
 --- ---  
 0    Store ID                      896 non-null      int64  
 1    Store\_Area                    896 non-null      int64  
 2    Items\_Available               896 non-null      int64  
 3    Daily\_Customer\_Count         896 non-null      int64  
 4    Store\_Sales                   896 non-null      int64  
 dtypes: int64(5)  
 memory usage: 35.1 KB

0s [8] `data = data.dropna()`

1s  `from sklearn.preprocessing import StandardScaler`

`features = data[['Store_Area', 'Items_Available', 'Daily_Customer_Count', 'Store_Sales']]`

`scaler = StandardScaler()`

`features_scaled = scaler.fit_transform(features)`

```

1s import seaborn as sns

plt.figure(figsize=(8, 6))

plt.subplot(2, 2, 1)
sns.histplot(data['Store_Area'], kde=True)
plt.title('Store Area Distribution')

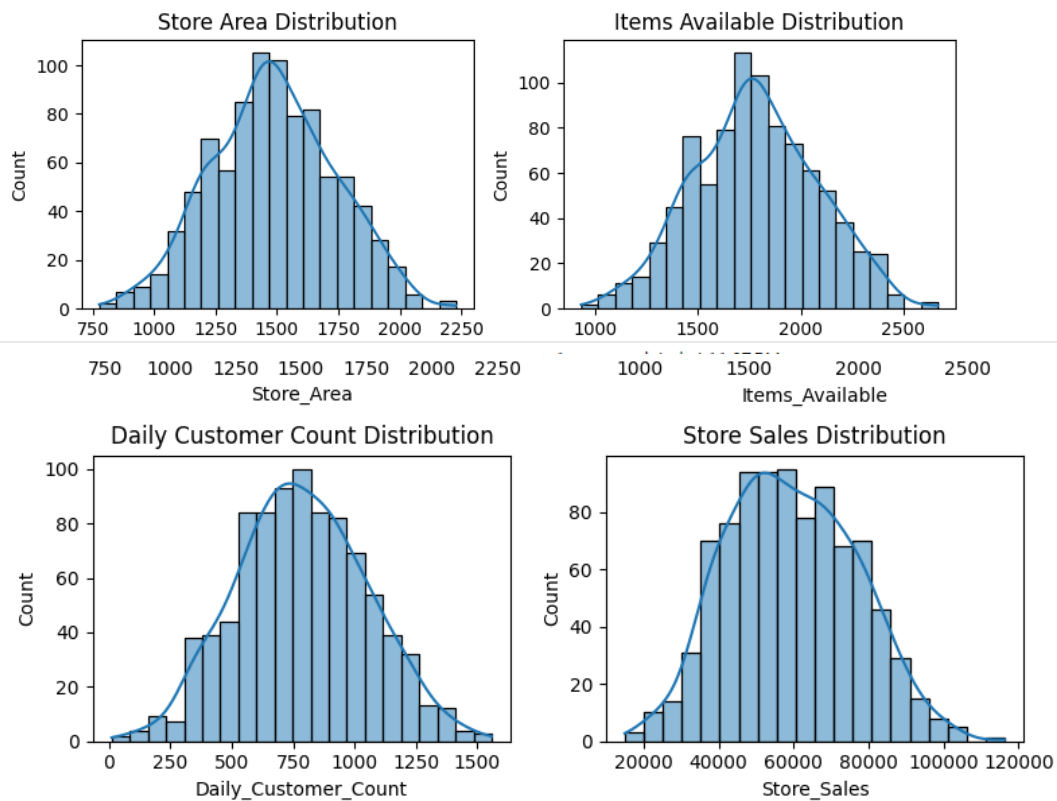
plt.subplot(2, 2, 2)
sns.histplot(data['Items_Available'], kde=True)
plt.title('Items Available Distribution')

plt.subplot(2, 2, 3)
sns.histplot(data['Daily_Customer_Count'], kde=True)
plt.title('Daily Customer Count Distribution')

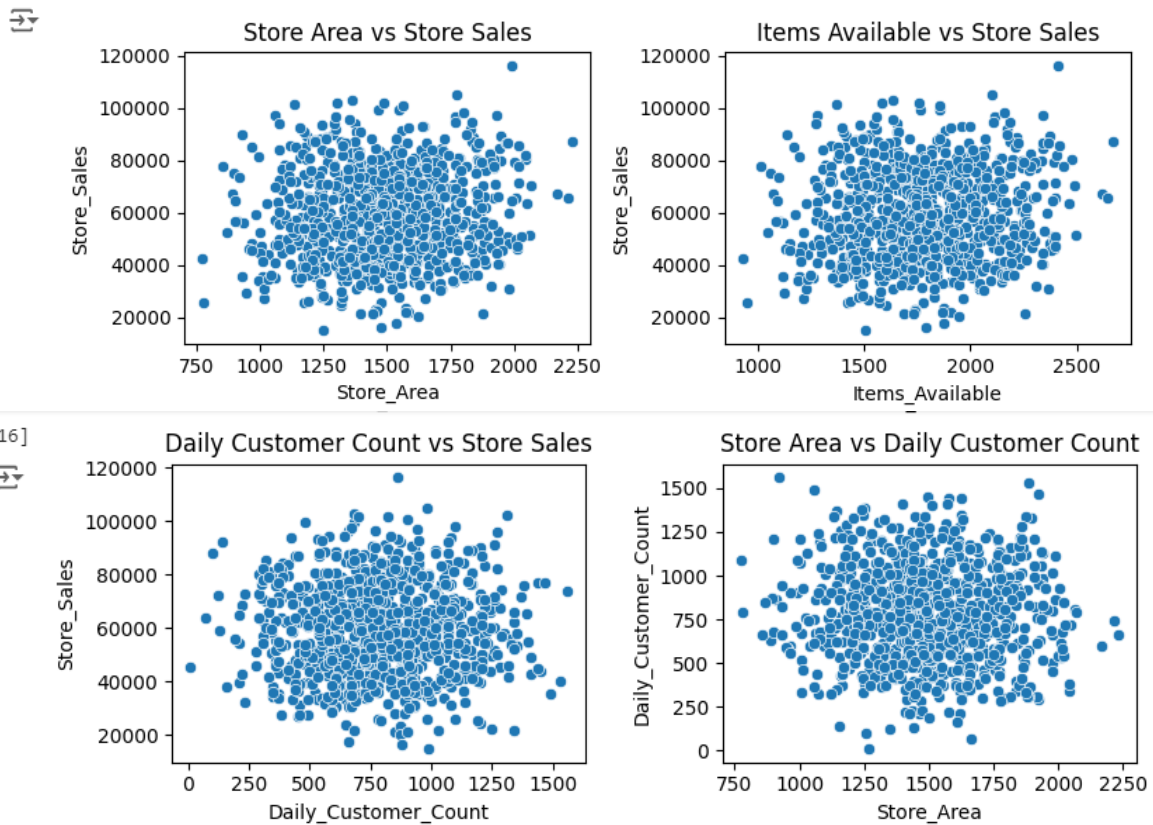
plt.subplot(2, 2, 4)
sns.histplot(data['Store_Sales'], kde=True)
plt.title('Store Sales Distribution')

plt.tight_layout()
plt.show()

```



```
✓ [16] plt.figure(figsize=(8, 6))  
1s  
  
plt.subplot(2, 2, 1)  
sns.scatterplot(x='Store_Area', y='Store_Sales', data=data)  
plt.title('Store Area vs Store Sales')  
  
plt.subplot(2, 2, 2)  
sns.scatterplot(x='Items_Available', y='Store_Sales', data=data)  
plt.title('Items Available vs Store Sales')  
  
plt.subplot(2, 2, 3)  
sns.scatterplot(x='Daily_Customer_Count', y='Store_Sales', data=data)  
plt.title('Daily Customer Count vs Store Sales')  
  
plt.subplot(2, 2, 4)  
sns.scatterplot(x='Store_Area', y='Daily_Customer_Count', data=data)  
plt.title('Store Area vs Daily Customer Count')  
  
plt.tight_layout()  
plt.show()
```



```
[17] #HEATMAP
plt.figure(figsize=(8, 6))
sns.heatmap(data[['Store_Area', 'Items_Available', 'Daily_Customer_Count', 'Store_Sales']].corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```

