Lab 1

# **Introduction**

In this Lab on Data Visualization, Data Preprocessing, and Statistical Analysis Using Python in Jupyter Notebook, I have used the Walmart sales data and visualization trends, distributions and relationship among the features implemented in the upcoming paragraphs.

# **Methodology**

To implement the data collections, following is the data source information below

Source: Kaggle

Dataset: Walmart Sales Data(Walmart\_sales.csv)

Environment and IDE: Jupyter Notebook extension in VS code.

Libraries used:

* Pandas for data manipulation
* Matplotlib and seaborn for visualization

# **Data Collections**

Python code for loading the data source is given below in the screenshot attached.

A screenshot of a computer

AI-generated content may be incorrect.

# **Data Visualization**

Data visualization is the process for generating the graphical representation of data for communication of the information and insights effectively.

## **4. 1 Scatter Plots: Show relationships between numeric variables**.

To implement the scatter plots, I have used the temperature as x axis and weekly\_sales as y axis with the kind scatter.

**Python code:**

df.plot(x = 'Temperature', y = 'Weekly\_Sales', kind = 'scatter', title='Weekly Sales vs Temperature')

**Output**

A screen shot of a graph

AI-generated content may be incorrect.

## **4.2 Line Plots: Visualize trends over time or sequences.**

To implement the trends over the time, I have sorted the date values and performed the group by with Weekly\_Sales and Unemployement by date as shown in the screenshot below.

**Python Code**

df['Date'] = pd.to\_datetime(df['Date'],dayfirst=True)

df.sort\_values('Date', inplace=True)

df.groupby('Date')['Weekly\_Sales'].sum().plot(kind='line')

df.groupby('Date')['Unemployment'].sum().plot(kind='line')

plt.title('Weekly Sales and Unemployment over time')

A screen shot of a graph

AI-generated content may be incorrect.

## **4.3 Bar Charts: Compare categorical data.**

To compare the categorical data using bar chart, I have performed group by of store for weekly\_sales performed by Walmart.

**Python code:**

df.groupby('Store')['Weekly\_Sales'].mean().plot(kind='bar',figsize=(12,6), title='Average Weekly Sales by Store')

**Output:**

A screenshot of a graph

AI-generated content may be incorrect.

## **4.4 Histograms: Display the distribution of numerical data.**

To display the distribution of numerical data using histogram I have used weekly\_sales of Walmart.

**Python Code**

df['Weekly\_Sales'].plot(kind = 'hist',bins=20,title='Distribution by Weekly Sales')

**Output**

A screenshot of a computer

AI-generated content may be incorrect.

## **4.5 Box Plots: Visualize data spread and identify outliers.**

To implement the data spread and outliers’ identification visualization, I have used box plot kind and seaborn library of python. Below is the code for the same.

**Python code:**

import seaborn as sns

import matplotlib.pyplot as plt

sns.boxplot(x='Holiday\_Flag', y='Weekly\_Sales', data=df)

plt.title('Weekly Sales Distribution on Holiday vs Non-Holiday Weeks')

plt.show()

**Output:**

A graph with a blue rectangle

AI-generated content may be incorrect.

## **4.6 Pie Charts: Represent proportions for categorical data.**

For implementing the pie chart representation for categorical data I have used the pie chart for holiday flag, below is the python code for the same for identify the percentage of holidays and non-holidays in Walmart.

plt.pie(df['Holiday\_Flag'].value\_counts(), labels= ['Holiday', 'No Holiday'])

plt.legend()

plt.title('Holiday vs No Holiday')

**Output**

A screenshot of a graph

AI-generated content may be incorrect.

**Summary**

In data visualization, following is the summary

* we discovered correlation between temperature and weekly sales using scatter
* plot.
* Plotted the trend of sales and unemployment over time using line plots.
* Compared store-wise sales performance using bar charts.
* Checked the distribution of sales and outliers using histogram and box plots.
* Plotted the ratio of holiday vs non-holiday weeks using a pie chart.

**Challenges Faced:**

* Formatting date fields for plotting time-based data.
* Handling missing or badly formatted data entries.

**Conclusion:**

* Visualization aided in uncovering key sales trends.
* The sales data exhibits strong seasonal trends and volatility during holiday weeks.

# **5. Data Preprocessing**

Data preprocessing is one of the most crucial step in data mining pipeline, which is a process of cleaning, transforming and preparing the data for data analysis, that helps to ensure that data is in easy-to-read format.

## **5.1 Handle Missing Values**

Step 1: Before handling the missing values, following the python code for displaying the dataset as mentioned in the screenshot below.

A screenshot of a computer

AI-generated content may be incorrect.

To handle the missing values, firstly I have detected the missing values using the following python code.

A black rectangular object with a black border

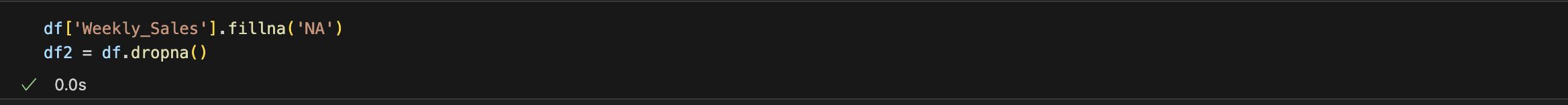
AI-generated content may be incorrect.

For handling the missing values, I am going to use fillna() function to handle the missing store value.

A black screen with a black border

AI-generated content may be incorrect.

For other missing values I have used dropna function as mentioned in the screenshot below.



## **5.2 2. Outlier Detection and Removal:**

To perform outlier detection and removal following is the python code which I have added.

A screenshot of a computer

AI-generated content may be incorrect.

To remove the outlier, and check result, please find the below mentioned python code.

A screen shot of a graph

AI-generated content may be incorrect.

## **5.3 Data Reduction**

### **Apply sampling (by number or percentage) to reduce data size.**

I have applied the sampling by both number and percentage to reduce data size.

A screenshot of a computer

AI-generated content may be incorrect.

### **Perform dimension elimination by dropping less relevant columns.**

A screenshot of a computer

AI-generated content may be incorrect.

## **5.4 Data Scaling and Discretization**

### **Min-Max Scaling**

A screen shot of a computer

AI-generated content may be incorrect.

**6. Statistical Analysis**

## **6.1 General Overview of DataFollowing is the general overview of the data.**

A screenshot of a computer

AI-generated content may be incorrect.

## **6.2 Central Tendency Measures**

A computer screen with green text

AI-generated content may be incorrect.

## **6.3 Dispersion Measures**

A computer screen shot of a code

AI-generated content may be incorrect.

## **6.4 Correlation Analysis**

A screenshot of a computer

AI-generated content may be incorrect.

# **7. GitHub Repository**