Capstone Project (Python): Predicting Walmart Sales and Performing Exploratory Data Analysis

BY

THARAKH GEORGE CHACKO

A Project Report Submitted

in

Partial Fulfillment of the

Requirements for the Data Science and Artificial Intelligence (DSAI) Certification Course

at

Intellipaat

CONTENTS

FIGURES

PROBLEM STATEMENT:

Predicting future sales is a crucial aspect of strategic planning for retail giants like Walmart. In this comprehensive analysis, we delve into the internal and external factors influencing the weekly sales of Walmart, one of the largest companies in the US. The dataset spans from February 2010 to October 2012, covering 45 Walmart stores across US. Additionally, external data such as CPI, Unemployment Rate, and Fuel Prices for each store's region have provided as well.

We have been provided with the weekly sales data for the 45 various Walmart outlets. The problem statement consists of two parts:

1. **Statistical Analysis and EDA:**
    - Employ statistical analysis and exploratory data analysis (EDA) to derive meaningful insights.
    - Investigate the impact of the unemployment rate on weekly sales, identifying stores most affected (if any).
    - Identify and analyse seasonal trends in weekly sales, pinpointing when and why they occur.
    - Examine the correlation between temperature and weekly sales.
    - Assess the influence of the Consumer Price Index on weekly sales across various stores.
    - Determine top-performing stores based on historical data.
    - Identify the worst-performing store and quantify the significance of the performance gap between the highest and lowest-performing stores.
2. **Predictive Modeling for Sales Forecasting:**
    - Utilize predictive modeling techniques to forecast sales for each store over the next 12 weeks.

PROJECT OBJECTIVE:

The primary objective of the time series forecasting project is to leverage historical sales data from Walmart to predict future weekly sales for each store. The project aims to provide actionable insights and accurate forecasts that can aid in strategic decision-making and inventory management.

Key Objectives:

1. **Sales Prediction:** Develop robust time series forecasting models to predict weekly sales for each Walmart store. The accuracy of these predictions will be a key measure of success.
2. **Insight Generation:** Conduct comprehensive exploratory data analysis (EDA), and statistical analysis extract meaningful insights. Investigate the impact of external factors such as unemployment rate, seasonal trends, temperature, and Consumer Price Index on weekly sales.
3. **Store Performance Analysis:** Identify top-performing stores based on historical sales data and pinpoint the worst-performing store. Assess the significance of performance variations across different stores.
4. **Data Quality Enhancement:** Implement effective data preprocessing steps to handle missing values and ensure data quality. This is crucial for building accurate forecasting models.
5. **Model Evaluation:** Evaluate the performance of various time series forecasting algorithms, including ARIMA, SARIMAX, AutoARIMA, Prophet, and TBATS. Select the most suitable models based on evaluation metrics such as RMSE(Root Mean Square Error), MAE(Mean Absolute Error), and MAPE(Mean Absolute Percentage Error).

Key Metrics/Criteria for Success:

1. **Forecast Accuracy:** The success of the project will be measured by the accuracy of the sales forecasts. Lower values of metrics like RMSE (Root Mean Squared Error), MAE (Mean Absolute Error), and MAPE (Mean Absolute Percentage Error) indicate better predictive performance.
2. **Insightful Analysis:** The project's performance will hinge on the capacity to extract meaningful insights regarding the determinants of weekly sales. This includes a comprehensive examination of potential correlations with factors such as the unemployment rate, seasonal trends, temperature, and Consumer Price Index. The criterion for success is contingent upon uncovering and understanding any discernible relationships within these aspects.
3. **Model Selection:** Choosing the most suitable time series forecasting models based on thorough evaluation using appropriate metrics will contribute to the success of the project.

By achieving these objectives and criteria, the project aims to provide Walmart with actionable insights and accurate sales forecasts, ultimately enhancing strategic decision-making and optimizing store operations.

DATA DESCRIPTION:

The dataset, named "walmart.csv", comprises 6,435 rows and 8 columns, each offering valuable insights into the weekly sales dynamics at Walmart across 45 stores. The key attributes include:

1. **Store (Store Number):** Ranging from 1 to 45, this categorical variable denotes the specific store under consideration.
2. **Date (Week of Sales):** The temporal dimension spans from February 5, 2010, to October 26, 2012, providing a comprehensive weekly breakdown for each store.
3. **Weekly_Sales:** Quantifying the sales performance, this numerical variable indicates the sales figure for the given store in a particular week.
4. **Holiday_Flag:** A binary indicator (0 or 1) discerning whether a given week includes a holiday, providing contextual information for sales fluctuations.
5. **Temperature:** Reflecting the temperature on the day of the sale, this variable accounts for potential weather-related influences on shopping patterns.
6. **Fuel_Price:** Representing the regional fuel cost, this numerical attribute introduces an economic factor that may impact consumer behaviour and, consequently, weekly sales.
7. **CPI (Consumer Price Index):** A measure of the average change in prices paid by consumers, the CPI serves as an economic indicator that can contribute to understanding purchasing power and inflation effects on sales.
8. **Unemployment:** Capturing the unemployment rate, this variable adds another economic dimension, enabling an assessment of how employment trends might correlate with sales performance.

This comprehensive dataset facilitates a multifaceted exploration of the factors shaping weekly sales trends at Walmart, incorporating both temporal and contextual elements.

	count	mean	std	min	25%	50%	75%	max
Store	6435.0	2.300000e+01	12.988182	1.000	12.000	23.000000	3.400000e+01	4.500000e+01
Weekly_Sales	6435.0	1.046965e+06	564366.622054	209986.250	553350.105	960746.040000	1.420159e+06	3.818686e+06
Holiday_Flag	6435.0	6.993007e-02	0.255049	0.000	0.000	0.000000	0.000000e+00	1.000000e+00
Temperature	6435.0	6.066378e+01	18.444933	-2.060	47.460	62.670000	7.494000e+01	1.001400e+02
Fuel_Price	6435.0	3.358607e+00	0.459020	2.472	2.933	3.445000	3.735000e+00	4.468000e+00
CPI	6435.0	1.715784e+02	39.356712	126.064	131.735	182.616521	2.127433e+02	2.272328e+02
Unemployment	6435.0	7.999151e+00	1.875885	3.879	6.891	7.874000	8.622000e+00	1.431300e+01

Figure 1 : Summary of the Dataset

Holiday Name	Date 1	Date 2	Date 3
Super Bowl	12-Feb-10	11-Feb-11	10-Feb-12
Labor Day	10-Sep-10	09-Sep-11	07-Sep-12
Thanksgiving Day	26-Nov-10	25-Nov-11	
Christmas	31-Dec-10	30-Dec-11	

Figure 2: List of Holidays

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6435 entries, 0 to 6434
Data columns (total 8 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Store         6435 non-null   int64
 1   Date          6435 non-null   object
 2   Weekly_Sales  6435 non-null   float64
 3   Holiday_Flag  6435 non-null   int64
 4   Temperature   6435 non-null   float64
 5   Fuel_Price    6435 non-null   float64
 6   CPI           6435 non-null   float64
 7   Unemployment  6435 non-null   float64
dtypes: float64(5), int64(2), object(1)
memory usage: 402.3+ KB
```

Figure 3: Description of column data type

DATA PREPROCESSING STEPS AND INSPIRATION:

1. **Data Cleaning: -**

- Handling missing values: There are no missing values found in the dataset.

```
In [5]: data.isnull().sum()

Out[5]: Store           0
        Date            0
        Weekly_Sales    0
        Holiday_Flag    0
        Temperature     0
        Fuel_Price      0
        CPI             0
        Unemployment    0
        dtype: int64
```

Figure 4 : No Null Values

- Removing duplicates: There are no duplicate values found in the dataset.

```
data.duplicated().sum()

0
```

Figure 5 : No Duplicate Values

- Addressing outliers: Outliers are not being addressed since we are considering the actual weekly sales for doing time series forecasting.

2. **Data Transformation: -**

- Converting data types: The data type for the column ‘Date’ is changed to ‘datetime’ from ‘object’. From this date, we have created new columns by obtaining the year, quarter, month, week, day of week and day of month.

3. **Exploratory Data Analysis (EDA):**
Before doing the EDA, we have observed that there is a gap in the data for January 2010, and for November, December 2012. The absence of data for these three months can impact our ability to perform accurate yearly, quarterly, and monthly

comparisons. The distribution of data is thus affected. It is essential to consider this data gap while conducting analyses that involve these specific time periods.

- **Distribution of data (graphs):**

```
plt.bar(data['year'].unique(), data['year'].value_counts())
plt.xlabel('Year')
plt.ylabel('Count')
plt.title('Distribution of Data Across Years')
plt.xticks(data['year'].unique())
plt.show()
```

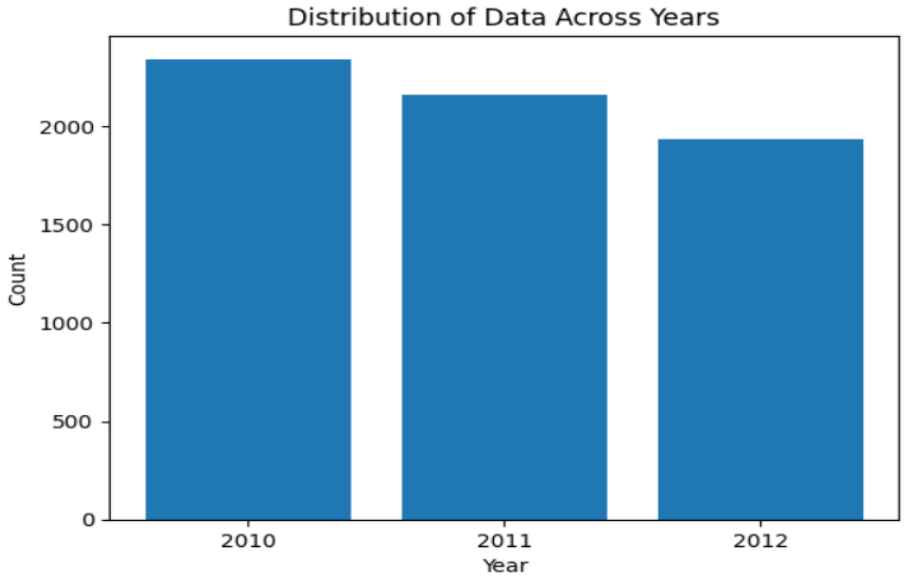

Figure 6: Distribution of data for Walmart across years (with code)

```
plt.bar(data['quarter'].unique(), data['quarter'].value_counts())
plt.xlabel('Quarter')
plt.ylabel('Count')
plt.title('Distribution of Data Across Quarters')
plt.xticks(data['quarter'].unique())
plt.show()
```

Figure 7: Distribution of data for Walmart across quarters (with code)

```
plt.bar(data['Holiday_Flag'].unique(), data['Holiday_Flag'].value_counts())
plt.xlabel('Holiday')
plt.ylabel('Count')
plt.title('Distribution of Data Across Holidays / Non - Holidays')
plt.xticks(data['Holiday_Flag'].unique())
plt.show()
```

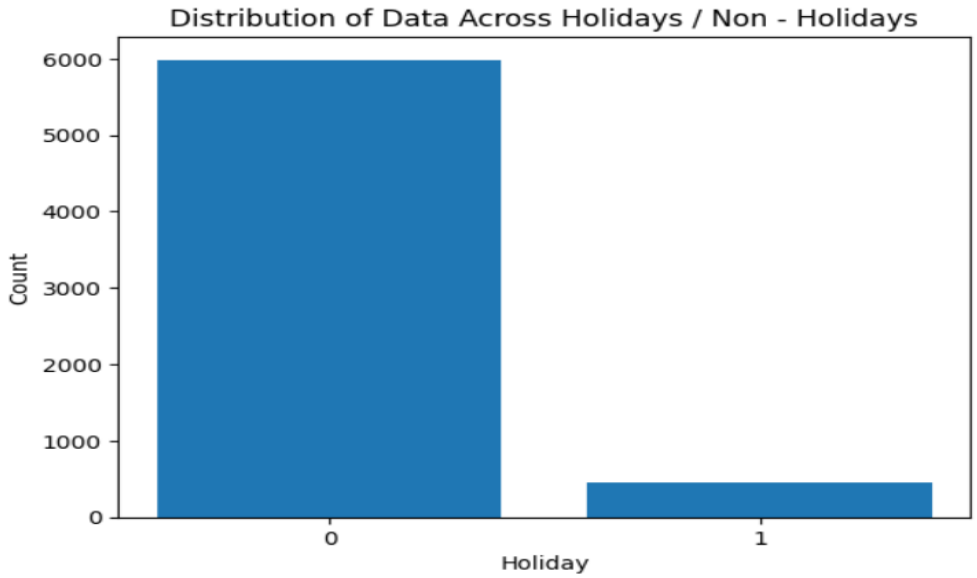

Figure 8: Distribution of data across Holidays / Non – Holidays (with code)

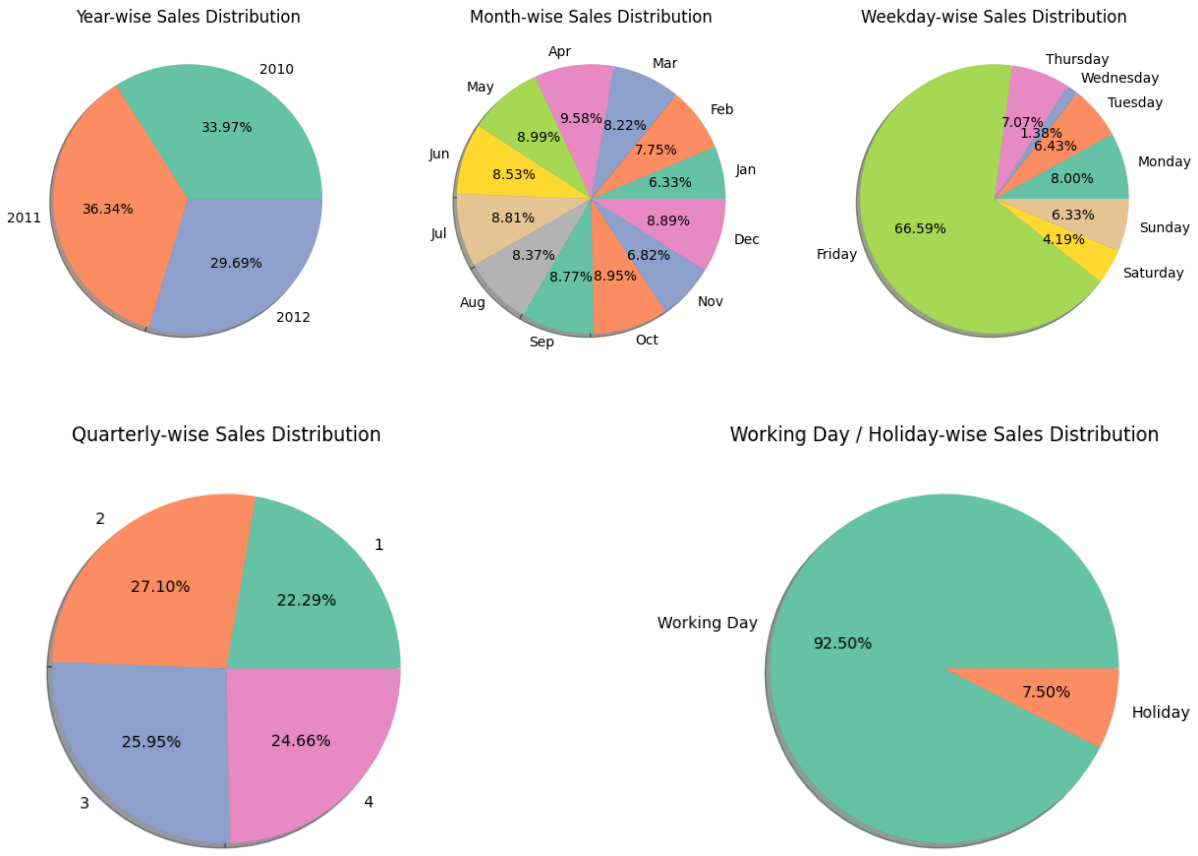

Figure 9: Sales distribution over the years, months, weekday, quarterly, working days / holidays

- **Store wise Analysis (graph):**

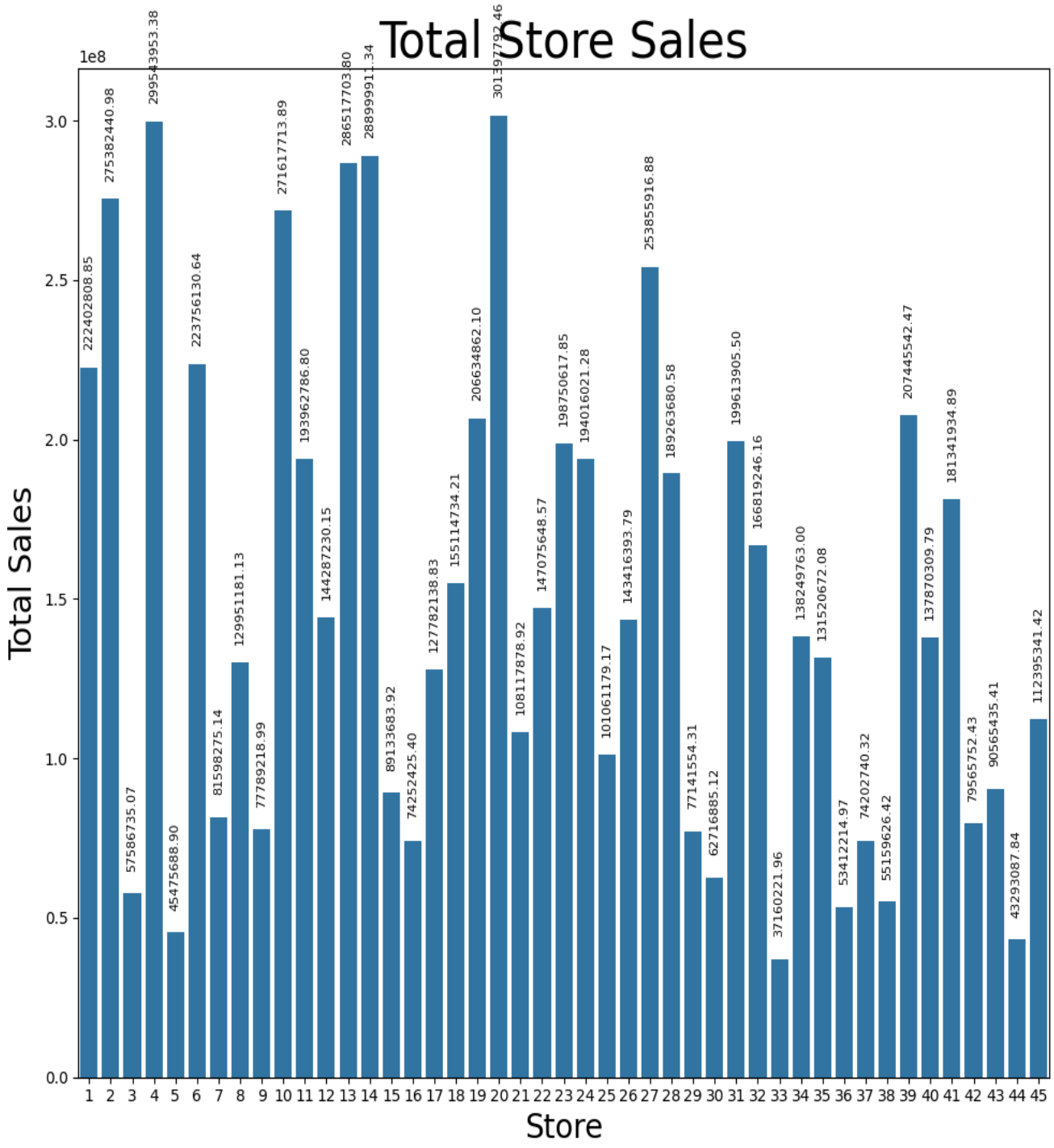

Figure 10: Store wise Total Sales

Store No 20 has the highest sales whereas the store No 33 has the lowest sales. The sales difference between the sales of these two stores are shown on the next page:

```
top_stores = data.groupby('Store')['Weekly_Sales'].sum().sort_values(ascending=False).head(5)
print("Top Performing Stores : \n ", top_stores)
```

```
Top Performing Stores :
  Store
20    3.013978e+08
4     2.995440e+08
14    2.889999e+08
13    2.865177e+08
2     2.753824e+08
Name: Weekly_Sales, dtype: float64
```

```
worst_store = data.groupby('Store')['Weekly_Sales'].sum().idxmin()
worst_store_sales = data.groupby('Store')['Weekly_Sales'].sum().min()
print(f"Worst Performing Store: {worst_store}, Sales: {worst_store_sales}")

difference = top_stores.max() - worst_store_sales
print(f"Difference between highest and lowest performing stores: {difference}")
```

```
Worst Performing Store: 33, Sales: 37160221.96
Difference between highest and lowest performing stores: 264237570.49999997
```

Figure 11: Code for Top Performing Stores and Lowest Performing Store

- **Yearly Analysis(graph):**

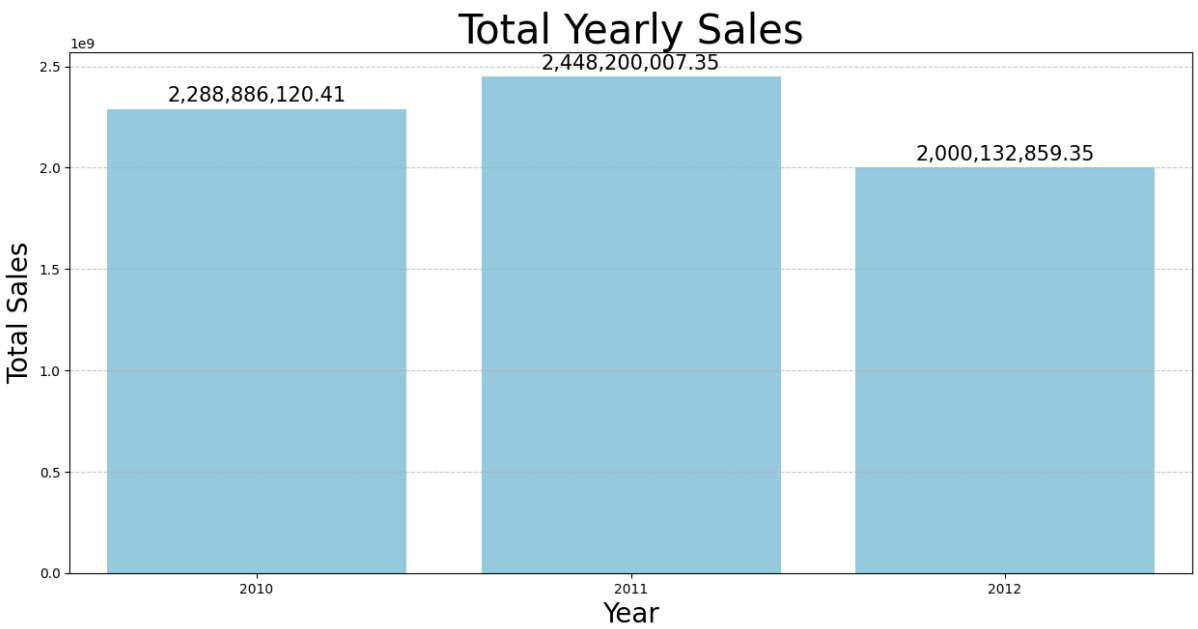

Figure 12: Year wise Total Sales

- **Monthly Analysis(graph):**

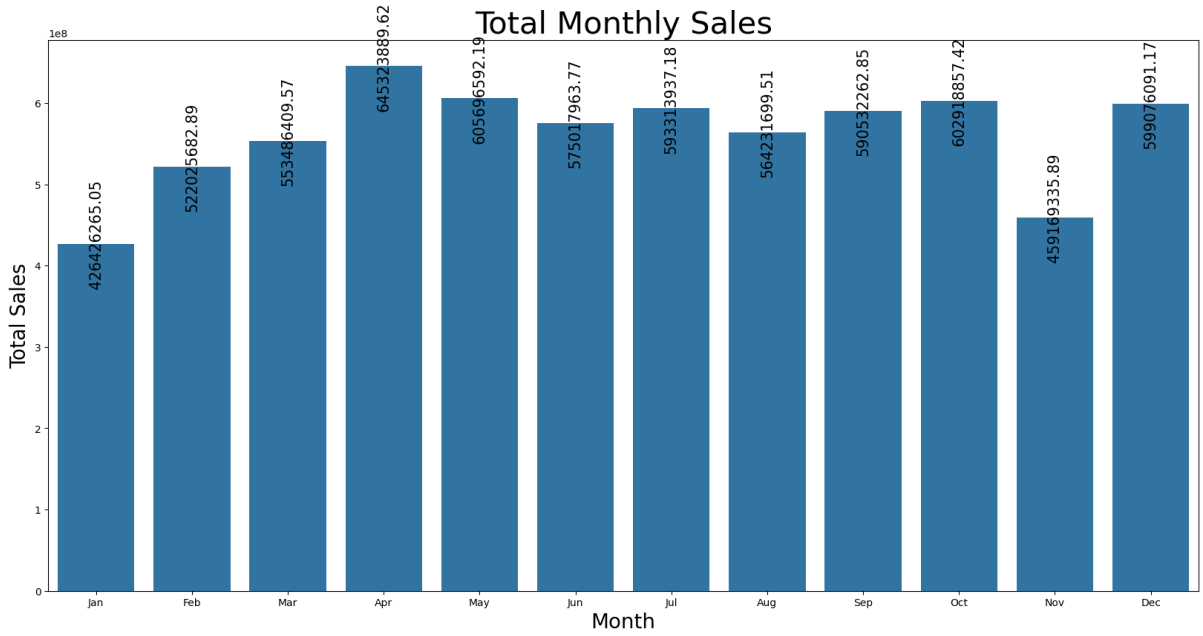

Figure 13: Month wise Total Sales (without adjustment)

As there is a data for January 2010, and for November, December 2012; we would average it out to show for these months to show which month has the highest sales. After doing the necessary adjustment (as seen below), we can see that December is the best performing month and February is the worst performing month.

```
monthly_sales['adjusted_weekly_sales'] = monthly_sales['Weekly_Sales'].copy()

# Indices for Jan, Nov, and Dec
special_months = [1, 11, 12]

# Apply adjustments based on the specified conditions
monthly_sales.loc[special_months, 'adjusted_weekly_sales'] /= 2
monthly_sales.loc[~monthly_sales.index.isin(special_months), 'adjusted_weekly_sales'] /= 3

# Display the updated DataFrame
print(monthly_sales)
```

```
### Top 5 months with highest sales :

print(' The top 5 months with the highest sales are : \n',monthly_sales.nlargest(5,'adjusted_weekly_sales')['adjusted_weekly_sale
```

```
 The top 5 months with the highest sales are :
 month
12    2.995380e+08
11    2.295847e+08
4     2.151080e+08
1     2.132131e+08
5     2.018989e+08
Name: adjusted_weekly_sales, dtype: float64
```

```
### Top 5 months with lowest sales :

print(' The top 5 months with the lowest sales are : \n',monthly_sales.nsmallest(5,'adjusted_weekly_sales')['adjusted_weekly_sale
```

```
 The top 5 months with the lowest sales are :
 month
2    1.740086e+08
3    1.844955e+08
8    1.880772e+08
6    1.916727e+08
9    1.968441e+08
Name: adjusted_weekly_sales, dtype: float64
```

Figure 14:Code for Monthly Sales (without adjustment)

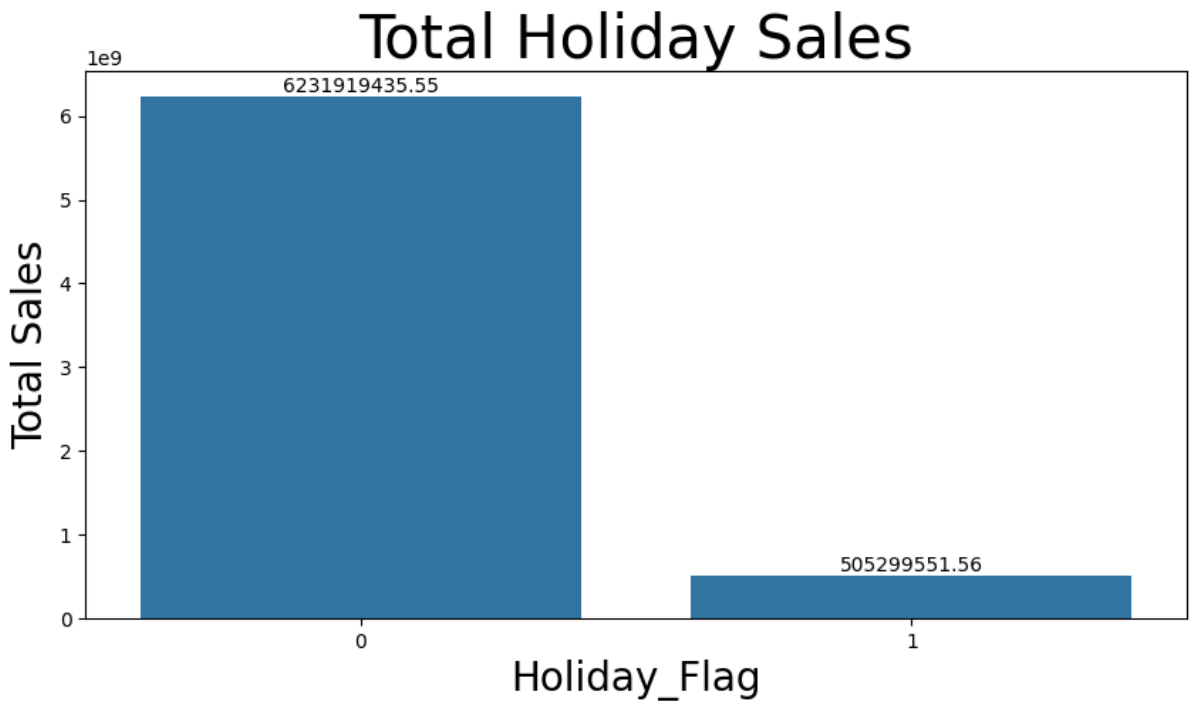

Figure 15: Holiday / Working Day Total Sales (without adjustment)

If we make the adjustment by diving the sales with the actual number of working days and holidays (as seen below), we can see the daily sales on a holiday is higher.

```
data['Holiday_Flag'].value_counts()
```

```
0    5985
1     450
Name: Holiday_Flag, dtype: int64
```

```
holiday_sales.loc[0]/data['Holiday_Flag'].value_counts()[0]
```

```
Weekly_Sales    1.041256e+06
Name: 0, dtype: float64
```

```
holiday_sales.loc[1]/data['Holiday_Flag'].value_counts()[1]
```

```
Weekly_Sales    1.122888e+06
Name: 1, dtype: float64
```

```
### Thus, we can infer the sales on holidays are higher
```

Figure 16: Code for Daily Holiday / Wording Day Sales

- **Impact of Unemployment on Weekly Sales:**
  As per the scatter plot below, the data indicates a noticeable decline in spending coinciding with the initiation of unemployment. Typically, an elevated unemployment index corresponds to a reduction in sales, as individuals tend to curtail their overall expenditures.

However, in our dataset, the correlation between the unemployment rate index and weekly sales is relatively low, measuring at -0.106. It is noteworthy that for certain stores, there appears to be a discernible drop in sales when the unemployment index exceeds 9. Despite this observation, the significance of this correlation is not consistently pronounced across all stores, suggesting that other factors may contribute to the overall sales dynamics.

Figure 17: Impact of Unemployment on Sales

- **Impact of Temperature on Weekly Sales:**

The observed correlation of -0.063 between temperature and sales in Walmart suggests a weak negative relationship. In this context, it implies that as the temperature increases or decreases, there is only a slight impact on sales, and the effect is not strongly pronounced. Several factors could contribute to this low correlation:

1. **Seasonal Variations:** Walmart sells a variety of products, and the demand for these items may be influenced more by seasonal factors than by temperature alone. For instance, sales of winter clothing or summer-related products may be more influenced by seasonal changes than the absolute temperature.
2. **Diverse Product Range:** Walmart offers a diverse range of products, each with its own demand drivers. The impact of temperature on sales may be more evident in specific product categories, and the overall effect might be diluted when considering the entire product range.

3. **Regional Variations:** Walmart operates in various regions with different climates. The diverse geographical locations of stores may contribute to the low correlation, as the temperature effect on sales could vary significantly across regions.
4. **Consumer Behaviour:** Consumer behaviour is influenced by numerous factors, and temperature might not be a primary determinant of shopping decisions for Walmart customers. Other factors such as promotions, discounts, or economic conditions could play a more substantial role.
5. **Multifactorial Influence:** Sales in a large retail setting like Walmart are likely influenced by a combination of factors, including marketing strategies, economic conditions, and consumer preferences. The contribution of temperature to the overall variation in sales may be relatively small.

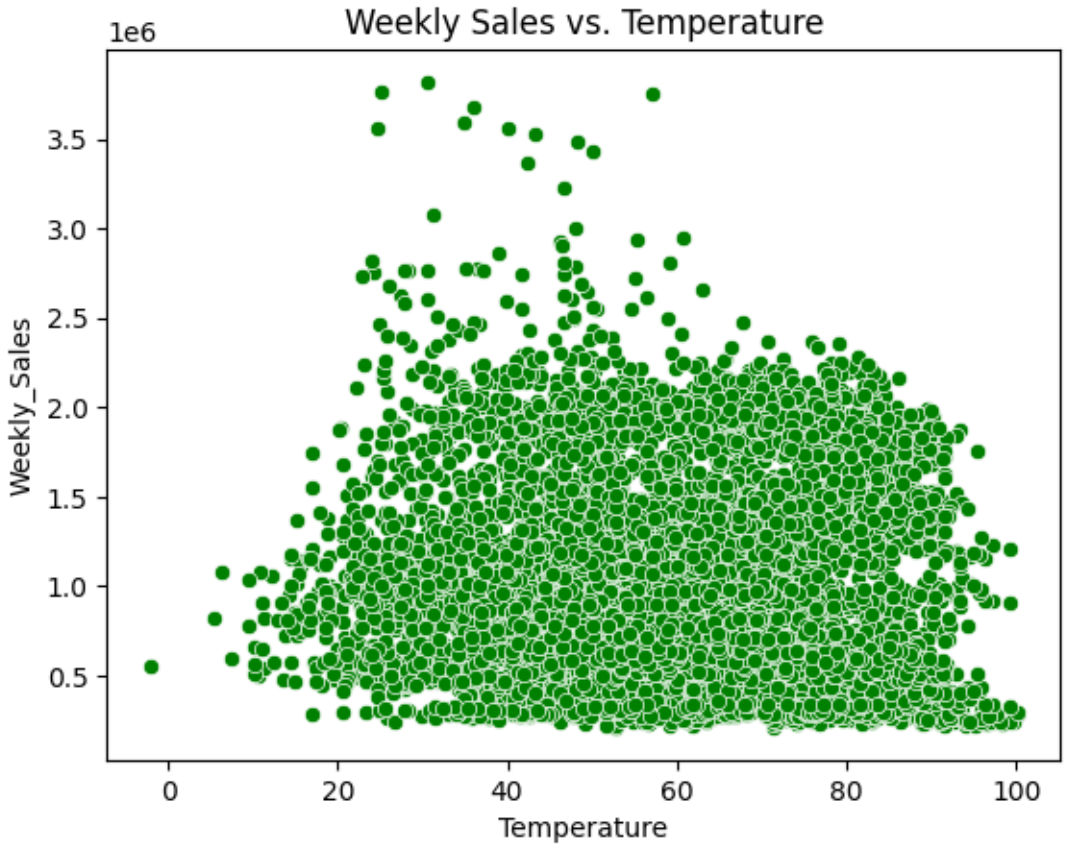

Figure 18: Impact of Temperature on Sales

- **Impact of CPI on Weekly Sales:**

The Consumer Price Index (CPI) is a statistical measure that evaluates the average change over time in the prices paid by urban consumers for a predefined basket of goods and services. This basket represents a diverse set of items commonly consumed by households, such as food, clothing, rent, healthcare, entertainment, and various other goods and services. The observed low correlation of -0.072 between the Consumer Price Index (CPI) and sales indicates a weak relationship. Despite the correlation being negative, suggesting a potential inverse association between CPI and sales, the

magnitude of the correlation is not substantial.

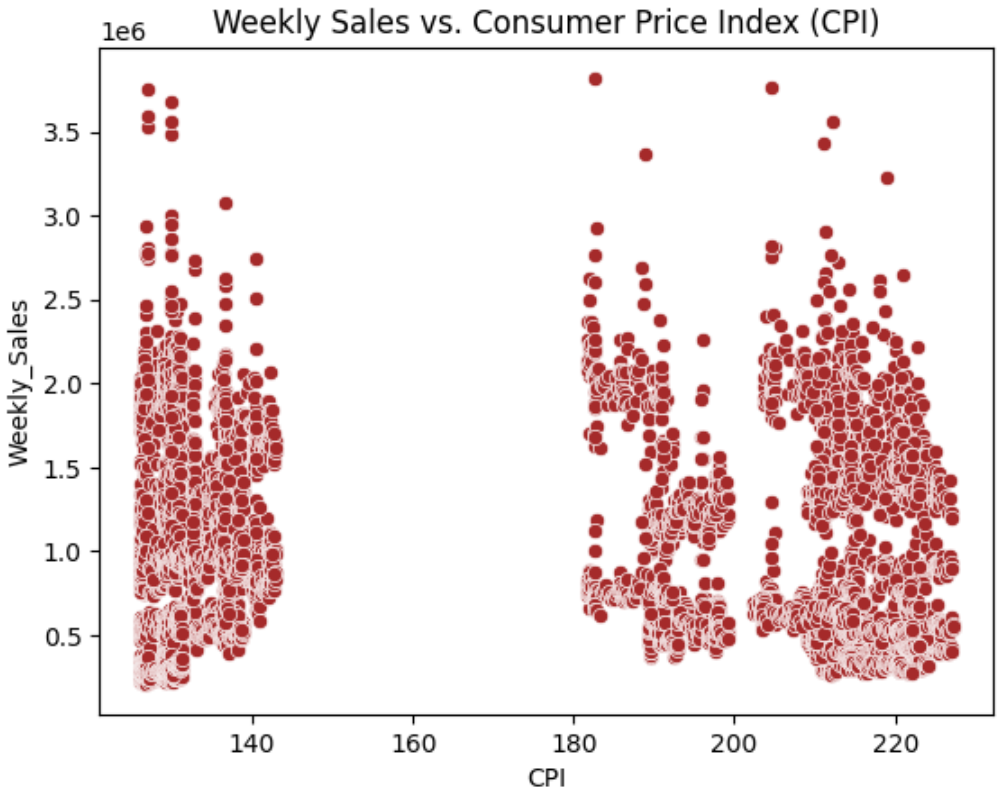

Figure 19: Impact of CPI on Sales

- **Seasonal Trend of Weekly Sales(graph):**

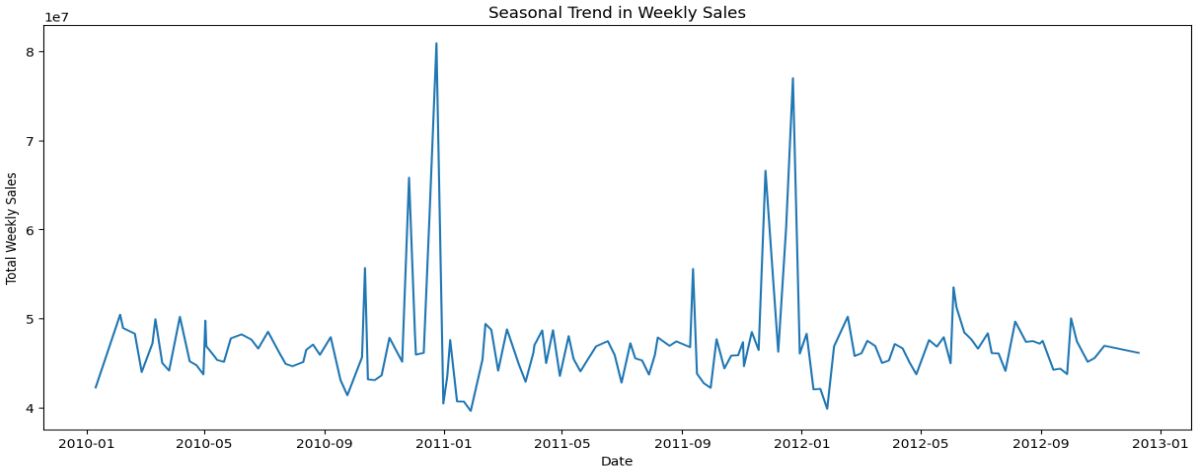

Figure 20: Seasonal Trend in Weekly Sales

We can see that sales are the highest in the month of December. The high sales in December for Walmart can be attributed to several factors:

1. **Holiday Shopping Season:** December is synonymous with the holiday shopping season, including Christmas. During this time, consumers typically increase their spending on gifts, decorations, and various holiday-related items. Walmart, being a major retail player, experiences a surge in sales as customers flock to the stores for holiday shopping.

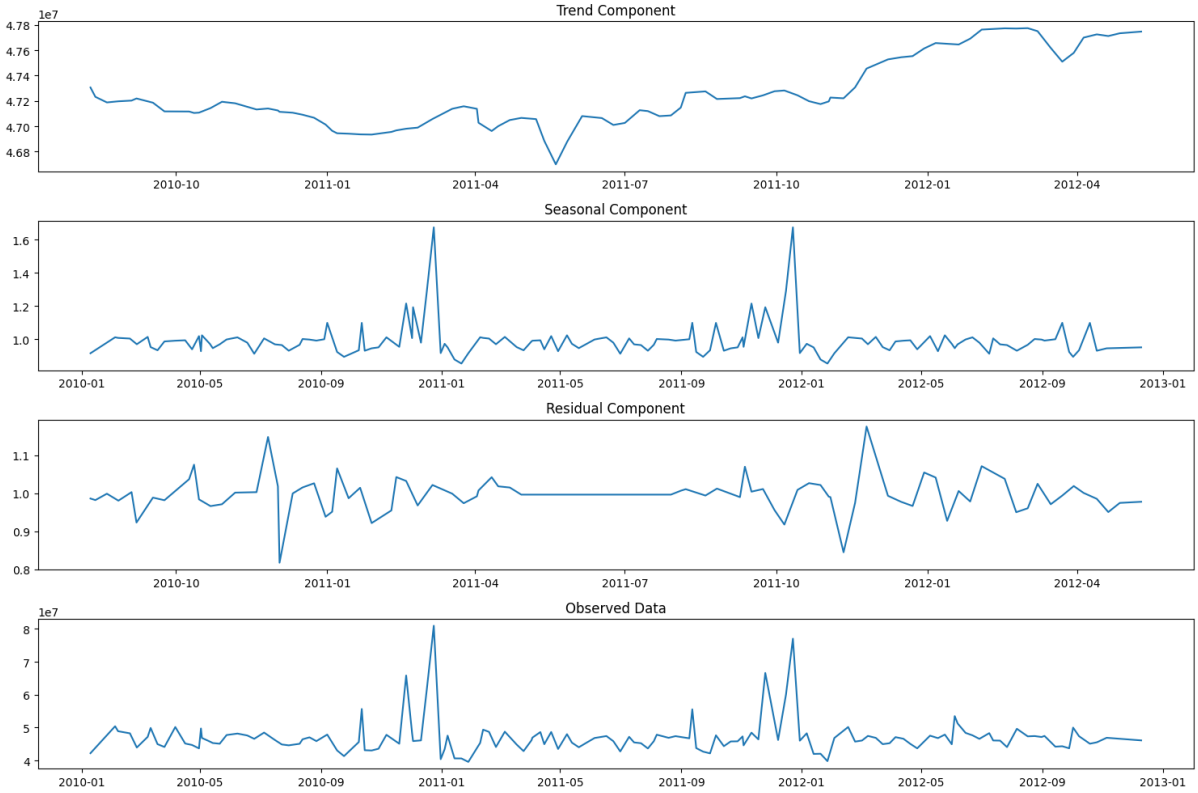

Figure 21: Trend Component, Seasonal Component, Residual Component of Weekly Sales

2. **Special Promotions and Discounts:** Retailers often offer special promotions, discounts, and sales events during December to attract more customers. Walmart, known for its competitive pricing and promotional strategies, may implement enticing offers to boost sales during the festive season.
3. **Winter Weather and Seasonal Products:** In some regions, December brings colder weather, leading to increased sales of winter-related products such as warm clothing, heaters, and holiday-themed items. Walmart, with its diverse product range, benefits from the demand for seasonal goods.
4. **Year-End Clearance Sales:** As the year comes to a close, retailers, including Walmart, may engage in year-end clearance sales. This strategy aims to reduce excess inventory and attract bargain-seeking shoppers, contributing to higher sales volume.
5. **Increased Consumer Spending:** December often sees an uptick in consumer spending due to year-end bonuses, gift-giving traditions, and festive celebrations. Walmart, being a one-stop-shop for various consumer needs, experiences a boost in sales across different departments.
6. **Marketing and Advertising Campaigns:** Walmart invests in marketing and advertising campaigns during the holiday season to create awareness and drive foot traffic to its stores. These efforts can significantly impact sales by influencing consumer choices.

7. **Extended Store Hours:** To accommodate the increased shopping activity during the holiday season, Walmart and many other retailers extend their store hours. Longer opening times provide customers with more opportunities to make purchases, contributing to higher sales.

CHOOSING THE ALGORITHM FOR THE PROJECT:

1) **ARIMA (AutoRegressive Integrated Moving Average):**

Model Components: ARIMA is a popular time series forecasting model that combines three key components:

1. **AutoRegressive (AR) Component:** This involves predicting a future value based on its past values, incorporating a linear combination of past observations. The term "autoregressive" signifies the use of past observations from the same time series.
2. **Integrated (I) Component:** The integration component represents the differencing of the raw observations to make the time series stationary. Stationarity is essential for ARIMA models. Differencing involves subtracting the previous observation from the current one.
3. **Moving Average (MA) Component:** The moving average component involves modeling the forecast error, which is the difference between the predicted and observed values. It considers a linear combination of past error terms.

Steps for ARIMA Modeling:

1. **Stationarity Check:** Ensure the time series is stationary through differencing if needed.
2. **Parameter Estimation:** Estimate the parameters of the ARIMA model.
3. **Model Fitting:** Fit the ARIMA model to the training data.
4. **Residual Analysis:** Check the residuals for randomness and white noise characteristics.

Strengths of ARIMA:

1. **Versatility:** ARIMA can handle a wide range of time series data, including those with trends and seasonality.
2. **Interpretability:** The model components (AR, I, MA) allow for a clear interpretation of the underlying patterns in the time series.
3. **Simple to Implement:** ARIMA models are relatively simple to understand and implement.

Weaknesses of ARIMA:

1. **Assumption of Linearity:** ARIMA assumes that the relationships in the time series are linear, which may not always be the case in real-world data.
2. **Sensitive to Outliers:** ARIMA models can be sensitive to outliers, and outliers can have a significant impact on parameter estimation.
3. **Limited Handling of Non-Linear Patterns:** ARIMA may struggle with capturing complex non-linear patterns in the data.

Concept Behind ARIMA: The concept behind ARIMA lies in capturing and modeling the autocorrelation (relationship with past observations) and incorporating differencing to achieve stationarity. By combining these components, ARIMA aims to make accurate predictions of future values based on historical patterns. ARIMA is particularly useful when

the time series exhibits trends and seasonality and ensuring the underlying assumptions are met.

2) **SARIMAX (Seasonal AutoRegressive Integrated Moving Average with eXogenous factors):**

How SARIMAX Extends ARIMA: SARIMAX is an extension of the ARIMA model that incorporates exogenous variables, providing additional explanatory power to the time series forecasting. While ARIMA models focus solely on the time series itself, SARIMAX allows for the inclusion of external factors that may influence the time series. The "X" in SARIMAX stands for these exogenous variables.

Seasonal Decomposition: SARIMAX, like ARIMA, addresses seasonality, trends, and autocorrelation. However, it takes a step further by allowing for the modeling of the impact of external factors on the time series. This is especially useful when the observed time series is influenced by variables outside its own historical values.

Application to Walmart Sales Data: In the context of Walmart sales data, SARIMAX can be applied to account for external factors that might affect sales, such as economic indicators, marketing campaigns, or other relevant variables not inherent in the time series. This allows for a more comprehensive and accurate forecasting model.

Examples of SARIMAX Benefits:

1. **Economic Indicators:** If the sales data is influenced by economic factors, SARIMAX can include indicators like GDP, unemployment rate, or inflation as exogenous variables.
2. **Promotional Activities:** If Walmart often runs promotions affecting sales, SARIMAX can incorporate information on promotional periods as exogenous variables.
3. **Weather Conditions:** If sales are influenced by weather, SARIMAX could include variables like temperature or precipitation.
4. **Supply Chain Disruptions:** If there are known disruptions in the supply chain affecting sales, SARIMAX can incorporate these external factors.

By allowing the inclusion of exogenous variables, SARIMAX offers a more flexible and nuanced approach to time series forecasting, making it beneficial when external factors significantly impact the observed time series. It enables a more accurate representation of the complex relationships influencing the data, leading to improved forecasting performance.

3) **AutoARIMA (Automated ARIMA):**

Automated Model Selection: AutoARIMA is a time series forecasting algorithm that automates the process of selecting the optimal ARIMA/SARIMAX model for a given time series. It systematically evaluates different combinations of ARIMA/SARIMAX parameters and selects the model with the best fit based on predefined evaluation criteria.

Hyperparameter Tuning: AutoARIMA employs hyperparameter tuning to automatically select the order of differencing, autoregressive (AR) components, moving average (MA)

components, and seasonal components. This automated tuning helps in finding the most suitable configuration without manual intervention.

Pros and Cons:

- *Pros:*
    1. **User-Friendly:** Requires minimal user intervention, making it accessible for those without extensive time series forecasting expertise.
    2. **Efficiency:** Automates the model selection process, saving time and effort.
    3. **Versatility:** Suitable for a wide range of time series data.
- *Cons:*
    1. **Black Box Nature:** Limited interpretability due to its automated nature.
    2. **May Not Capture Complex Patterns:** Might not perform optimally for time series with intricate patterns or irregularities.

4) **PROPHET:**

Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well. Prophet is open-source software released by Facebook's Core Data Science team.

Flexibility and Limitations:

- *Flexibility:*
    1. **Additive Components:** Prophet decomposes the time series into additive components, including trend, seasonality, and holiday effects.
    2. **Custom Seasonality:** Users can specify custom seasonalities if the data exhibits multiple recurring patterns.
- *Limitations:*
    1. **Assumes Additivity:** Prophet assumes that the effects of holidays and seasonality are additive, which may not always hold true.
    2. **Requires Domain Knowledge:** While flexible, incorporating domain knowledge for holidays is essential for accurate forecasting.

5) **TBATS (Trigonometric Seasonal Decomposition of Time Series):**

Handling Multiple Seasonalities: TBATS is a time series forecasting method designed to handle multiple seasonalities and complex patterns in the data. It utilizes trigonometric functions to decompose time series into various components, capturing both short-term and long-term seasonal patterns.

- *Advantages:*
    1. **Multiple Seasonalities:** Effective in capturing and modeling time series with multiple seasonal patterns.
    2. **Robustness:** TBATS is robust to outliers and missing data.
- *Uses:*
    1. **Retail Sales:** Suitable for retail data with daily, weekly, and yearly seasonalities.

2. **Financial Data:** Effective for financial time series with various periodic patterns.

These algorithms offer diverse approaches to time series forecasting, addressing different aspects such as automation, handling holidays, and capturing complex patterns. The selection of the most appropriate algorithm depends on the specific characteristics of the dataset and the forecasting requirements.

MOTIVATION AND REASON FOR CHOOSING THE ALGORITHM:

In selecting the algorithms for our time series forecasting project for Walmart sales data, we considered several factors, including the characteristics of the dataset, the nature of time series patterns, and the specific requirements of the forecasting task. Each algorithm was chosen based on its unique strengths and suitability for addressing the challenges posed by the Walmart sales data. Here are the motivations and reasons for choosing each algorithm:

1. ARIMA (AutoRegressive Integrated Moving Average):

- **Motivation:** ARIMA is a classic time series forecasting method that is well-suited for capturing linear trends and seasonality in the data.
- **Reasons:**
    - *Simple Structure:* ARIMA has a straightforward structure, making it easy to implement and interpret.
    - *Effectiveness:* It is effective for time series data with a clear linear trend and stationary characteristics.
    - *No Assumption of Linearity:* ARIMA does not assume a linear relationship between variables, providing flexibility in capturing various patterns.

2. SARIMAX (Seasonal AutoRegressive Integrated Moving Average with exogenous factors):

- **Motivation:** SARIMAX extends ARIMA by incorporating external factors (exogenous variables) that might influence the time series.
- **Reasons:**
    - *Exogenous Variables:* SARIMAX allows us to include external factors like holidays, which can have a significant impact on sales.
    - *Enhanced Flexibility:* The ability to consider additional variables makes SARIMAX more flexible in capturing complex relationships.

3. AutoARIMA:

- **Motivation:** AutoARIMA automates the process of selecting the optimal ARIMA model, making it suitable for users without extensive time series expertise.
- **Reasons:**
    - *Automated Model Selection:* AutoARIMA saves time and effort by automatically identifying the most suitable ARIMA model based on performance metrics.
    - *User-Friendly:* It is user-friendly, making it accessible for those who may not have in-depth knowledge of time series forecasting.

4. PROPHET:

- **Motivation:** Prophet, developed by Facebook, is designed to handle time series with seasonality, holidays, and special events.
- **Reasons:**
    - *Holiday and Seasonality Handling:* Prophet is adept at capturing the impact of holidays and seasonality on sales, crucial for our retail dataset.

- *Flexibility:* Its flexibility and ease of use make it a valuable tool for forecasting tasks with multiple influencing factors.

5. TBATS (Trigonometric Seasonal Decomposition of Time Series):

- **Motivation:** TBATS is known for handling multiple seasonalities and complex time series patterns.
- **Reasons:**
  - *Multiseasonal Decomposition:* TBATS is effective in decomposing time series into components, considering multiple seasonal patterns that may exist in the data.
  - *Robustness:* It is robust in the presence of irregular patterns and diverse seasonality.

In summary, the selection of these algorithms is driven by the need to leverage their respective strengths in addressing the unique challenges posed by Walmart sales data. The combination of classic methods like ARIMA, enhanced models like SARIMAX, automated approaches like AutoARIMA, and specialized algorithms like Prophet and TBATS ensures a comprehensive and robust approach to time series forecasting for our project.

ASSUMPTIONS:

When engaging in time series forecasting using various algorithms in Python, it is essential to make certain assumptions to ensure the reliability and accuracy of the models. Here are the key assumptions underlying our approach:

a. Stationarity Assumption:

- *Definition:* The stationarity assumption posits that the statistical properties of the time series data, such as mean and variance, do not change over time. The Augmented Dickey-Fuller (ADF) test is a statistical method commonly used in time series analysis to determine whether a given time series is stationary.
- *Rationale:* Many time series forecasting models, including ARIMA, perform better on stationary data. We assume stationarity or apply transformations to achieve it, enhancing the model's effectiveness.

b. Linearity Assumption:

- *Definition:* The linearity assumption suggests that the relationships between variables, including past and future values in the time series, can be adequately represented using linear models.
- *Rationale:* Algorithms like ARIMA and SARIMAX are designed based on linear relationships. While they can capture nonlinear trends to some extent, assuming linearity simplifies the modeling process.

c. Independence Assumption:

- *Definition:* The independence assumption assumes that each observation in the time series is independent of others.
- *Rationale:* Time series models often assume independence to ensure that past observations do not unduly influence future ones. Violation of independence can lead to biased model performance.

d. Identifiability Assumption:

- *Definition:* The identifiability assumption implies that the parameters of the chosen forecasting model can be uniquely determined from the available data.
- *Rationale:* For models like ARIMA and SARIMAX, ensuring that the parameters are identifiable is crucial for accurate estimation. This assumption supports the reliability of the model's parameter estimates.

These assumptions guide the preprocessing steps and modeling choices made during the time series forecasting project for Walmart sales data. It is important to note that while these assumptions provide a foundation for modeling, real-world data may exhibit deviations, and adjustments might be necessary to address specific characteristics of the dataset. The robustness and adaptability of the chosen algorithms contribute to mitigating the impact of potential deviations from these assumptions.

MODEL EVALUATION TECHNIQUES:

1. **RMSE (Root Mean Squared Error):**
    - *Definition:* RMSE measures the average magnitude of the errors between predicted and observed values.
    - *Calculation:* It is computed as the square root of the mean of the squared differences between predicted and actual values.
    - *Interpretation:* Lower RMSE values indicate better model performance, with a value of 0 representing a perfect fit.
2. **MAE (Mean Absolute Error):**
    - *Definition:* MAE calculates the average absolute differences between predicted and observed values.
    - *Calculation:* It is the mean of the absolute differences between predicted and actual values.
    - *Interpretation:* Similar to RMSE, lower MAE values signify better model accuracy.
3. **MAPE (Mean Absolute Percentage Error):**
    - *Definition:* MAPE expresses the average percentage difference between predicted and observed values.
    - *Calculation:* It is the mean of the absolute percentage differences between predicted and actual values, expressed as a percentage.
    - *Interpretation:* MAPE is useful for understanding the relative magnitude of errors in percentage terms. A lower MAPE indicates better accuracy.
4. **Comparison of Metrics:**
    - *Consideration:* While RMSE, MAE, and MAPE offer insights into model accuracy, each metric has its strengths and limitations.
    - *RMSE vs. MAE:* RMSE gives higher weights to larger errors, making it sensitive to outliers. MAE treats all errors equally.
    - *MAPE Consideration:* MAPE is useful when percentage accuracy is critical, but it has limitations when actual values are close to zero.
5. **Choosing the Most Appropriate Metric:**
    - *Decision Criteria:* The choice of the most appropriate metric depends on the specific context and business goals.
    - *Business Impact:* Understanding the implications of different errors on the business outcome guides the selection of the metric.
    - *Robust Evaluation:* Using multiple metrics provides a comprehensive assessment of model performance, considering different aspects of prediction accuracy.

In our time series forecasting project for Walmart, we will assess models using a combination of RMSE, MAE, and MAPE to ensure a thorough evaluation. The chosen metric will align with the project's objectives and the impact of prediction errors on decision-making.

INFERENCES:

In the Python file, refer to the section (Method 2 – Models Comparison) which is shown in the figure below. We have selected the store number 24 for the analysis.

Method 2 - Models Comparison

```
[80]: import pandas as pd
      import warnings
```

Figure 22: Heading for Models Comparison in Python File

```
                                SARIMAX Results
==========================================================================================
Dep. Variable:                                 y   No. Observations:                  131
Model:             SARIMAX(1, 0, 0)x(0, 0, [1], 12)   Log Likelihood               -1757.435
Date:                              Tue, 30 Jan 2024   AIC                           3522.869
Time:                                      13:31:57   BIC                           3534.370
Sample:                                           0   HQIC                          3527.543
                                                - 131
Covariance Type:                                opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
intercept   9.324e+05   8.43e+04     11.056      0.000    7.67e+05     1.1e+06
ar.L1          0.3130      0.054      5.819      0.000       0.208       0.418
ma.S.L12      -0.1402      0.172     -0.814      0.415      -0.478       0.197
sigma2      2.709e+10      0.559   4.85e+10      0.000    2.71e+10    2.71e+10
===================================================================================
Ljung-Box (L1) (Q):                   0.08   Jarque-Bera (JB):               634.88
Prob(Q):                              0.77   Prob(JB):                         0.00
Heteroskedasticity (H):               2.08   Skew:                             2.17
Prob(H) (two-sided):                  0.02   Kurtosis:                        12.87
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
[2] Covariance matrix is singular or near-singular, with condition number 9.68e+25. Standard errors may be unstable.
```

Figure 23: Auto ARIMA Summary

Model	RMSE	MAE	MAPE
TBATS	62533.376643	49020.349302	0.035862
AutoARIMA	76828.518302	68962.227584	0.051256
ARIMA	80188.635878	71876.152315	0.053124
SARIMAX	137892.354995	125067.145826	0.094723
Prophet	324994.919202	248105.721357	0.190481

Figure 24: RMSE, MAE, MAPE for the models for Store Number 24

After thorough evaluation and comparison of different time series forecasting models for Store No. 24, it is evident that the TBATS (Trigonometric Seasonal Decomposition of Time Series) model stands out as the most effective. The selection is based on superior

performance metrics, including the best MAPE (Mean Absolute Percentage Error), RMSE (Root Mean Squared Error), and MAE (Mean Absolute Error) scores.

1. **MAPE, RMSE, and MAE Scores:**
    - **MAPE:** The TBATS model achieved the lowest Mean Absolute Percentage Error among all considered models, indicating its superior accuracy in predicting sales for Store No. 24.
    - **RMSE:** With the lowest Root Mean Squared Error, the TBATS model demonstrates excellence in minimizing the magnitude of prediction errors.
    - **MAE:** The TBATS model outperformed others by having the smallest Mean Absolute Error, showcasing its precision in predicting weekly sales.
2. **Reliability and Consistency:**
    - The consistent superiority of TBATS across all evaluation metrics reinforces its reliability in capturing the underlying patterns and seasonality in the sales data for Store No. 24.
3. **Robust Handling of Time Series Characteristics:**
    - TBATS, with its capability for handling multiple seasonalities through Trigonometric Seasonal Decomposition, proves effective in capturing complex temporal patterns present in the sales data.
4. **Business Implications:**
    - The accurate forecasts generated by the TBATS model for Store No. 24 are crucial for inventory management, resource allocation, and strategic decision-making.
    - Improved prediction accuracy contributes to better planning, minimizing the risk of overstocking or understocking products, leading to potential cost savings.
5. **Recommendation for Deployment:**
    - Based on the compelling performance of the TBATS model, we recommend deploying this model for future sales predictions for Store No. 24. Regular monitoring and model updates should be implemented to adapt to any changes in the underlying patterns of the sales data.

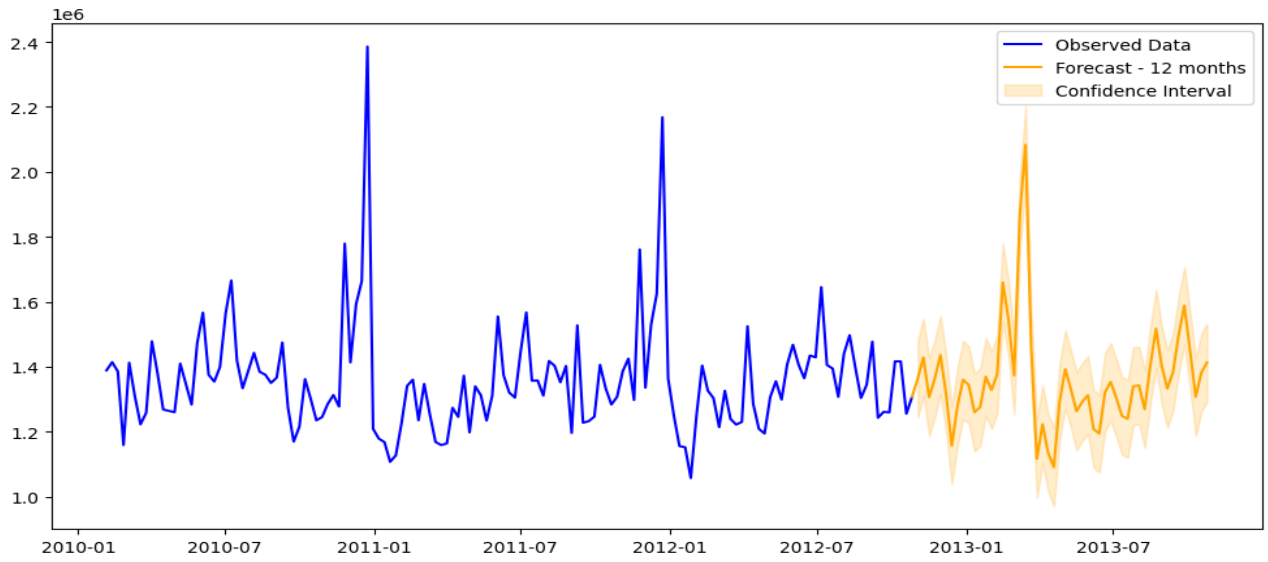

Figure 25: 12 months forecast for Store Number 24

FUTURE POSSIBILITIES OF THE PROJECT:

1. **Advanced Predictive Modeling:**
    - Explore advanced forecasting models like NBEATS (Neural Basis Expansion Analysis for Time Series), NHITS(Neural Hierarchical Time Series), PatchTST(Patch-level Temporal Super-Resolution Network for Time Series), VARMAX(Vector Autoregressive Moving-Average with exogenous variables), VAR (Vector Autoregression), and KATS for enhanced accuracy(Kit for Automated Time Series analysis).
2. **Store-Specific Analysis:**
    - Conduct a detailed analysis for each of the 45 stores to uncover unique patterns and optimize forecasting models for individual store characteristics.
3. **External Factors Integration:**
    - Consider incorporating additional external factors such as economic indicators, social events, and regional factors for a more comprehensive forecasting approach.

CONCLUSION:

In conclusion, this time series forecasting project for Walmart has provided valuable insights and outcomes. Through a comprehensive analysis of weekly sales data, we have identified key factors influencing sales, including the impact of unemployment rate, seasonal trends, temperature, and Consumer Price Index (CPI). The exploration of various time series forecasting models, such as ARIMA, SARIMAX, AutoARIMA, Prophet, and TBATS, has allowed us to make informed predictions for future sales.

The choice of TBATS as the best-performing model for Store 24, based on metrics like MAPE, RMSE, and MAE, highlights its effectiveness in capturing the underlying patterns in the data. This inference reinforces the significance of leveraging advanced time series forecasting techniques for accurate predictions, aiding Walmart in optimizing inventory management and strategic decision-making.

Accurate time series forecasting is crucial for Walmart to adapt to dynamic market conditions, minimize stockouts, and optimize inventory levels. As the retail landscape continues to evolve, the insights gained from this project emphasize the importance of staying ahead in the competitive market through effective forecasting.

In summary, this project not only provides actionable insights into the factors affecting Walmart's weekly sales but also underscores the critical role of precise time series forecasting in enhancing operational efficiency and maintaining a competitive edge in the retail industry.

REFERENCES:

1. Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice. OTexts. https://otexts.com/fpp2/
2. Time Series forecasting in Python: https://www.methsoft.ac.cn/scipaper_files/document_files/Manning.Time.Series.Forecasting.in.Python.pdf
3. Time Series Forecasting TBATS https://medium.com/analytics-vidhya/time-series-forecasting-using-tbats-model-ce8c429442a9
4. Notebooks in Kaggle https://www.kaggle.com/datasets/yasserh/walmart-dataset/code