INTELLIPAAT

Walmart Sales Forecasting Project

Sales Forecasting and Demand Planning Using ARIMA Model

Anuj Kumar

28/20/2024

This project focuses on forecasting Walmart's sales using the ARIMA (AutoRegressive Integrated Moving Average) model, a widely used technique for time series forecasting. The primary objective is to predict future sales for Walmart based on historical data, helping the company optimize inventory management, demand planning, and promotional strategies. The dataset consists of daily sales data, including variables such as sales figures, store locations, product categories, and promotional events.

Table of Contents

Problem Statement:

A retail store that has multiple outlets across the country are facing issues in managing the inventory - to match the demand with respect to supply. You are a data scientist, who has to come up with useful insights using the data and make prediction models to forecast the sales for X number of months/years.

1) Using the above data, come up with useful insights that can be used by each of the stores to improve in various areas.
2) Forecast the sales for each store for the next 12 weeks

Project Objective:

The objective of this project is to predict Sales of store for the next 12 weeks. As in dataset, size and time related data are given as feature, so analyze if sales are impacted by time-based factors and space- based factor. Most importantly how inclusion of holidays in a week soars the sales in store?

Data Description:

1) **Source**: The **original data sources** of walmart.csv vary but are often gathered from Walmart's own historical records, publicly available datasets, or curated collections on platforms like **Kaggle** and **UCI Machine Learning Repository**. Each of these sources may include different features, so the exact composition of walmart.csv can vary based on the version used.

2) **Variables**: Walmart runs several promotional markdown events throughout the year. These markdowns precede prominent holidays, the four largest of all, which are the Super Bowl, Labour Day, Thanksgiving, and Christmas. The weeks including these holidays are weighted five times higher in the evaluation than non-holiday weeks. Part of the challenge presented by this competition is modeling the effects of markdowns on these holiday weeks in the absence of complete/ideal historical data. Historical sales data for 45 Walmart stores located in different regions are available.

   a. Store - the store number
   b. Date - the week of sales
   c. Weekly_Sales - sales for the given store
   d. Holiday_Flag - whether the week is a special holiday week 1 – Holiday week 0 – Non-holiday week
   e. Temperature - Temperature on the day of sale
   f. Fuel_Price - Cost of fuel in the region
   g. CPI – Prevailing consumer price index
   h. Unemployment - Prevailing unemployment rate

3) **Data Size and Structure**:

When describing the data size and structure for a Walmart sales dataset, focus on these details:

a. **Data Size**:

   i. **Number of Rows**: The walmart.csv contains 6435 rows

   ii. **Number of Columns**: The walmart.csv contains 8 columns.

   iii. **Time Span**: The data spans from 5-02-2010 to 26-10-2012.

   iv. **Frequency**: Note the frequency of data collection, whether it's daily, weekly, or monthly (e.g., "The sales data is recorded on a weekly basis.").

b. **Data Structure**:

   i. **Data Types**:

```
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)
```

Data Pre-processing Steps and Inspiration :

1. Cleaning

a. **Handling Missing Values**:

   i. **Identify Missing Data**: Use functions like isnull() or isna() to find columns with missing values.

b. **Removing Outliers**:

   i. **Identify Outliers**: For numerical fields like Sales, calculate outliers using statistical methods (e.g., values more than 1.5 times the IQR or outside Z-score limits).

c. **Addressing Anomalies**:

   i. Checking for any sudden sales drops or peaks that don’t follow expected trends (e.g., errors during data entry).

   ii. Flag or remove such anomalies if they result from known events or reporting errors, as they can distort seasonal trends.

2. Transformation

a. **Normalization or Log Transformations**:

   i. Apply log transformation on Sales. Log transformation can stabilize variance, making the data more normal.

b. **Differencing**

   i. For time series data with trends, apply differencing to make the data stationary, a requirement for models like ARIMA.
   ii. First-order differencing (subtracting current from previous value) is commonly used for eliminating trends.

3. Time Series Decomposition

a. **Seasonal-Trend Decomposition Using LOESS (STL)**:

   i. Decompose the time series into **Trend**, **Seasonality**, and **Residuals** using functions like seasonal_decompose (in Python).
   ii. **Trend Component**: Shows the overall direction in sales, indicating growth or decline over time.
   iii. **Seasonality Component**: Reveals repeating patterns, such as weekly, monthly, or holiday sales trends.
   iv. **Residuals Component**: Indicates irregular components or noise, helping evaluate the quality of the decomposition.

b. **Insights from Decomposition**:

   i. **Trend Insights**: Identifying long-term sales growth or decline can inform inventory and supply chain decisions.
   ii. **Seasonal Patterns**: Recognize recurring periods of high or low sales, which is valuable for planning promotions or staffing.

4. Feature Engineering

a. **Moving Averages**:

   i. Create rolling averages (e.g., 7-day, 30-day) on Sales to smooth out fluctuations and capture underlying trends.

b. **Lagged Variables**:

   i. Introduce lag features by shifting the sales data by one or more time periods (e.g., Sales_Lag_1, Sales_Lag_7) to capture temporal dependencies.

c. **Date-Based Features**:

   i. Extract features from Date such as Month, Day, Day of Week, Quarter, and Year to help identify seasonal and cyclic patterns.

Choosing the Algorithm for the Project

Key Reasons for Choosing ARIMA

1. **Seasonality and Trend Analysis**:

   a. ARIMA is highly effective in capturing underlying trends in time series data, especially when the dataset becomes stationary through differencing.

   b. Although ARIMA itself doesn't handle seasonality explicitly, seasonality can be incorporated by transforming the data. For more complex seasonal patterns, SARIMA (Seasonal ARIMA) could be considered as an extension, but initial tests showed that ARIMA performs well with basic seasonal trends.

2. **Model Simplicity**:

   a. ARIMA is straightforward, requiring only three main parameters (p, d, q), which makes it easier to tune compared to more complex models.

   b. For time series data with a regular pattern and limited external factors, ARIMA can produce accurate forecasts without overcomplicating the model.

3. **Interpretability**:

   a. ARIMA's parameters—autoregressive (p), differencing (d), and moving average (q)—are easy to understand and interpret. This transparency allows for better insights into how past sales values are influencing future predictions.

   b. In business scenarios like sales forecasting, interpretability is essential because it provides clarity to stakeholders about how forecasts are derived.

4. **Relevance to Dataset Characteristics**:

   a. The Walmart sales dataset likely contains consistent weekly or monthly observations with potential trends over time. ARIMA's reliance on past data points makes it a natural choice for this structure.

   b. By differencing, ARIMA is effective at transforming non-stationary sales data into stationary form, which helps to model and predict future sales more accurately.

Alternative Models Considered

1. **SARIMA (Seasonal ARIMA)**:
    a. **Pros**: SARIMA is an extension of ARIMA that explicitly accounts for seasonality, which can be helpful if there are distinct seasonal patterns.
    b. **Why Not Chosen**: While SARIMA is a good alternative, ARIMA was initially preferred due to its simplicity and lower complexity. If seasonal patterns emerge, SARIMA could be revisited as a potential enhancement.
2. **Prophet**:

    a. **Pros**: Prophet, developed by Facebook, is effective for handling missing data, incorporating seasonality, and working with irregular time series intervals.
    b. **Why Not Chosen**: Prophet is robust and flexible but may be less interpretable than ARIMA for this dataset. Additionally, ARIMA often provides comparable or superior accuracy in situations with regular time series intervals and limited external factors.

Conclusion

ARIMA was selected for its simplicity, effectiveness in capturing sales trends, and ability to work with stationary data, making it ideal for this structured sales dataset. Should more complex seasonality or irregular patterns become apparent, models like SARIMA or Prophet could be explored as secondary options.

Assumptions

1. Stationarity of the Data:
2. No Significant Impact from External Factors Beyond the Dataset:
3. Consistent Sales Patterns Over Time:
4. Sufficient Data for Reliable Pattern Recognition:
5. Minimal Data Noise:

Model Evaluation and Techniques

To assess the performance of the ARIMA model, several metrics are considered:

1. **Root Mean Squared Error (RMSE)**: RMSE measures the average magnitude of the error between predicted and actual values, penalizing larger errors more heavily. It gives insight into how much deviation exists between the model's predictions and real sales values.
2. **Mean Absolute Error (MAE)**: MAE calculates the average absolute differences between predicted and actual sales values. It is helpful for understanding the average size of errors without disproportionately emphasizing large errors.

Inferences from the Model Evaluation

1. Key Findings

a. **Trends**: The ARIMA model revealed a general upward trend in sales, indicating a consistent growth pattern over time. This trend may be influenced by factors such as expanding customer base, successful promotions, or seasonal demand shifts.

b. **Seasonality**: The data exhibits distinct seasonal patterns, with noticeable peaks during certain times of the year, especially around holidays. For Walmart, this typically includes the holiday shopping season (November-December) and back-to-school periods. Such patterns reinforce the importance of seasonal adjustments in sales forecasting.

c. **Peak Sales Periods**: The model identified predictable peak periods corresponding with major holidays and special promotions, indicating these periods are critical for inventory and staffing planning.

2. Model Strengths and Weaknesses

a. **Strengths**:

   i. **Trend and Seasonality Capture**: The ARIMA model effectively captured overall sales trends and simple seasonal patterns, demonstrating reliable forecasting performance for general sales forecasting.

   ii. **Interpretability**: The ARIMA model's straightforward parameters allowed for clear insights into how past sales data influences future predictions, making it valuable for strategic planning and management understanding.

b. **Weaknesses**:

   i. **Handling of Complex Seasonality**: Although ARIMA performed well with basic seasonality, it may struggle with more complex or irregular seasonal patterns. A model like SARIMA or Prophet could potentially handle these cases more effectively.

   ii. **Spikes and Outliers**: ARIMA had some difficulty predicting sudden sales spikes accurately, as it primarily relies on past data points. External factors like promotions, holidays, or economic events can introduce spikes that ARIMA may not fully account for, resulting in underestimations or overestimations during such periods.

Future Possibilities of the Project (ARIMA Model)

1. Further Model Improvements

1. **Testing More Complex Models**:
   a. **Seasonal ARIMA (SARIMA)**: Since ARIMA might struggle with pronounced seasonal patterns (such as sales surges during holidays), SARIMA could be a better choice. SARIMA accounts for both trend and seasonality, providing better accuracy for datasets with strong seasonal effects, like retail sales data.

   b. **Prophet Model**: Facebook's Prophet model is highly effective for time series data with strong seasonal effects, missing data, and irregular intervals. It's more flexible than ARIMA and can handle holidays and special events better by explicitly modeling these components.
   c. **Machine Learning Models**: For more complex relationships between sales data and influencing factors, machine learning models such as Random Forests, XGBoost, or even deep learning techniques like Long Short-Term Memory (LSTM) networks can be explored. These models can capture non-linear relationships and provide potentially more accurate predictions for larger datasets.

2. **Incorporating External Data**:
   a. **Weather Data**: Integrating weather conditions (temperature, rainfall, etc.) can help explain fluctuations in sales for weather-dependent products. For instance, outdoor products or seasonal clothing sales might be impacted by temperature trends.
   b. **Economic Indicators**: Including macroeconomic indicators, such as GDP growth, consumer confidence, or unemployment rates, can help model external factors that influence consumer spending. These variables could improve the model's ability to capture changes in sales trends during economic shifts.
   c. **Promotion and Event Data**: Incorporating data on specific promotions, marketing campaigns, or major events (such as Black Friday sales) could help the model account for sudden spikes in demand. These data points can be treated as external regressors in ARIMA or its advanced versions (like SARIMAX).

3. Scaling

   a. **Expanding to Multiple Stores and Regions**:
      i. **Regional Models**: The ARIMA model could be scaled by building separate models for different stores or regions, as sales patterns can vary significantly by location. Localized models can capture specific regional trends, consumer behavior, and local events that affect sales.
      ii. **Hierarchical Forecasting**: If regional models are built, hierarchical forecasting techniques can be used to aggregate forecasts from store-specific models to get broader insights at the national level. This ensures that inventory and supply chain decisions are made with a clear understanding of both local and overall demand.
   b. **Product Line and Category Forecasting**:
      i. **Product-Specific Models**: ARIMA can be adapted to forecast sales for individual product categories, such as electronics, groceries, or apparel. Product-specific sales forecasting can allow Walmart to tailor inventory management, pricing strategies, and promotions for each category.
      ii. **Multi-Product Forecasting**: By scaling the ARIMA approach across different product lines, Walmart can better manage its diverse inventory and improve its ability to predict demand for various product segments over time.

4. Potential Business Applications

   a. **Demand Planning**:
      i. **Inventory Management**: With more accurate sales forecasts from ARIMA, Walmart can optimize inventory levels, ensuring sufficient stock for predicted demand while avoiding overstocking and reducing carrying costs.
      ii. **Supply Chain Efficiency**: By aligning the supply chain with accurate demand predictions, Walmart can reduce lead times and transportation costs, improve warehouse operations, and optimize product availability across its network of stores.
   b. **Targeted Promotions**:

        i. **Sales Campaigns**: ARIMA's ability to predict peak sales periods can help Walmart plan its marketing campaigns around these times. This can increase the effectiveness of promotions and help ensure that products are marketed at the most opportune times.
        ii. **Dynamic Pricing**: With accurate demand predictions, Walmart could implement dynamic pricing strategies that adjust prices based on forecasted demand, ensuring competitive pricing while maximizing revenue during high-demand periods.

c. **Supply Chain Optimization**:
    i. **Production and Restocking**: With reliable forecasts, Walmart can synchronize its production and restocking efforts more effectively. For example, anticipating a spike in demand due to a promotional event or holiday can ensure that the right products are delivered to the right locations on time.
    ii. **Logistics Planning**: The insights from the ARIMA model can improve logistics decisions by aligning the supply of goods to stores with forecasted demand. This reduces transportation costs, optimizes route planning, and ensures stock levels are maintained without overburdening warehouses.

Conclusion

Incorporating more advanced models, adding external data sources, and scaling the ARIMA model to individual regions and product categories would significantly enhance Walmart's forecasting capabilities. By applying these improvements, Walmart can refine its demand planning, optimize supply chains, and leverage targeted promotions to improve operational efficiency, reduce costs, and increase customer satisfaction.