Walmart Project

Mayur N Sastry

July 1, 2025

Goal

The Walmart project is dedicated to applying advanced data analysis techniques to build a predictive model for weekly sales and then deploy the model with the strategic objective of improving inventory turnover and reducing waste through precise optimization.

Dataset Description

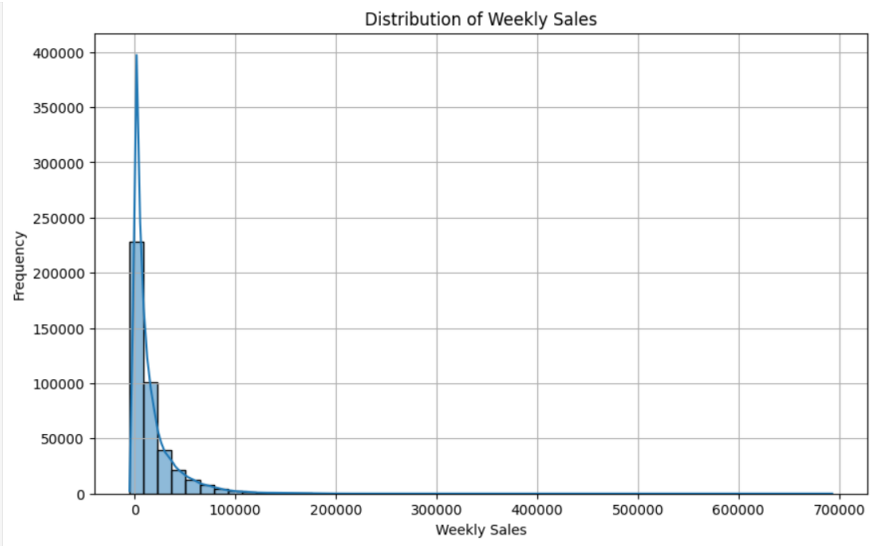
stores.csv
This file contains anonymized information about the 45 stores, indicating the type and size of store.

train.csv
This is the historical training data, which covers the dates from 2010-02-05 to 2012-11-01 and contains the following fields:
Store - the store number
Dept - the department number
Date - the week
Weekly Sales - sales for the given department in the given store
IsHoliday - whether the week is a special holiday week

features.csv
This file contains additional data related to the store, department, and regional activity for the given dates. It contains the following fields:
Store - the store number
Date - the week
Temperature - average temperature in the region Fuel Price - cost of fuel in the region
MarkDown1-5 - anonymized data related to promotional markdowns that Walmart is running. MarkDown data is only available after Nov 2011, and is not available for all stores all the time. Any missing value is marked with an NA.
CPI - the consumer price index
Unemployment - the unemployment rate
IsHoliday - whether the week is a special holiday week

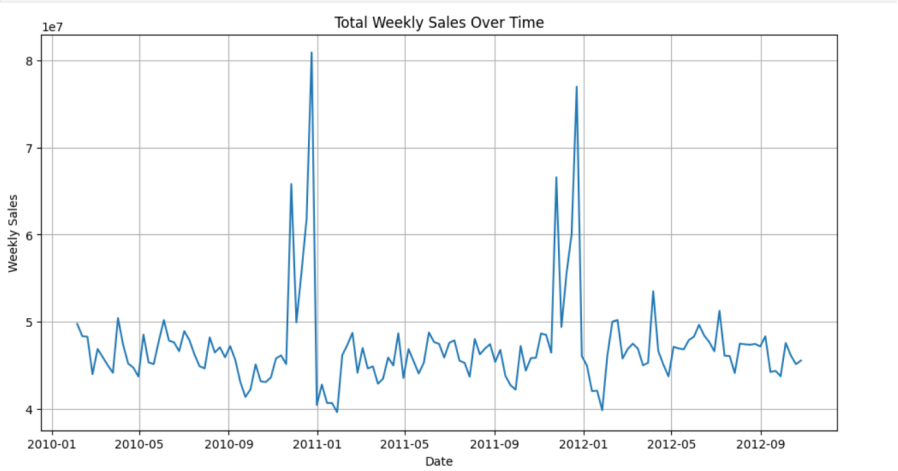
Exploratory Data Analysis

1. Distribution of Weekly Sales



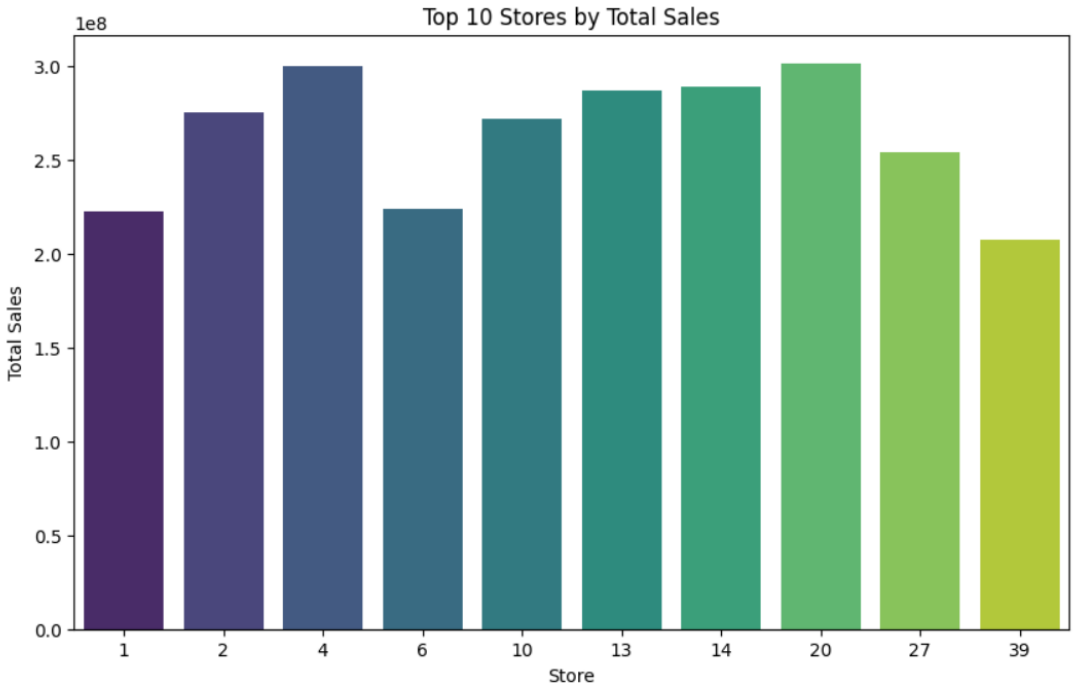
The plot reveals that the majority of weekly sales transactions are concentrated at the lower end of the spectrum, heavily skewed towards values between 0 and 10000. There's a long tail extending to higher sales figures, indicating a few instances of very large weekly sales, corresponding to high-performing stores or specific departments. This highly skewed distribution suggests that the data is not normally distributed and contains outliers on the higher side.

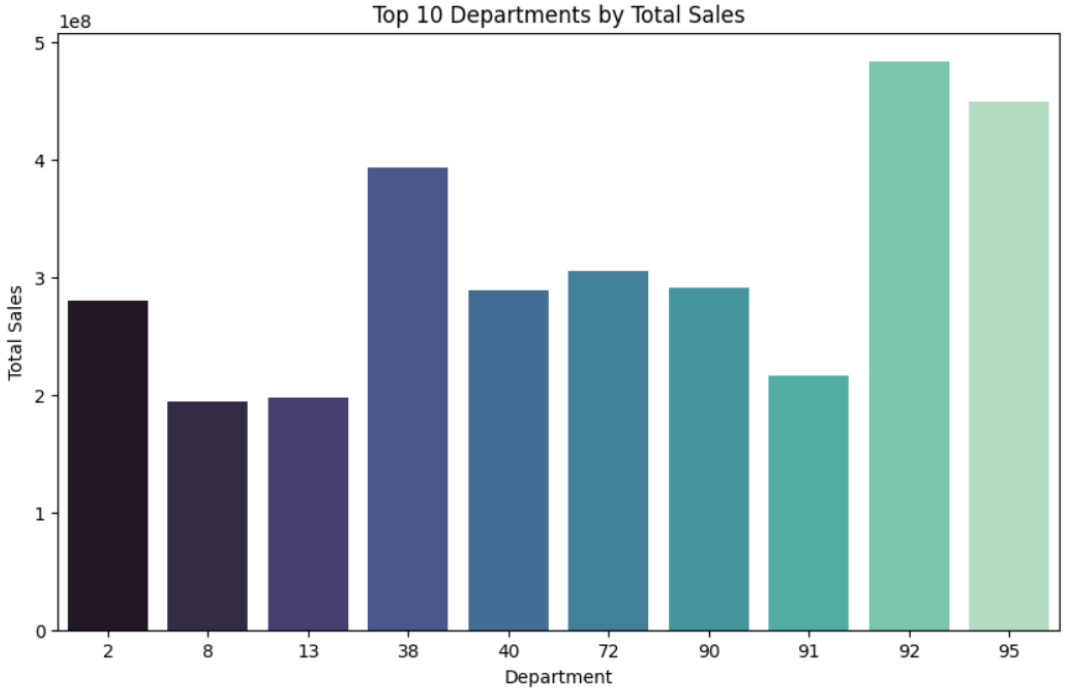
2. Total Weekly Sales time series plot



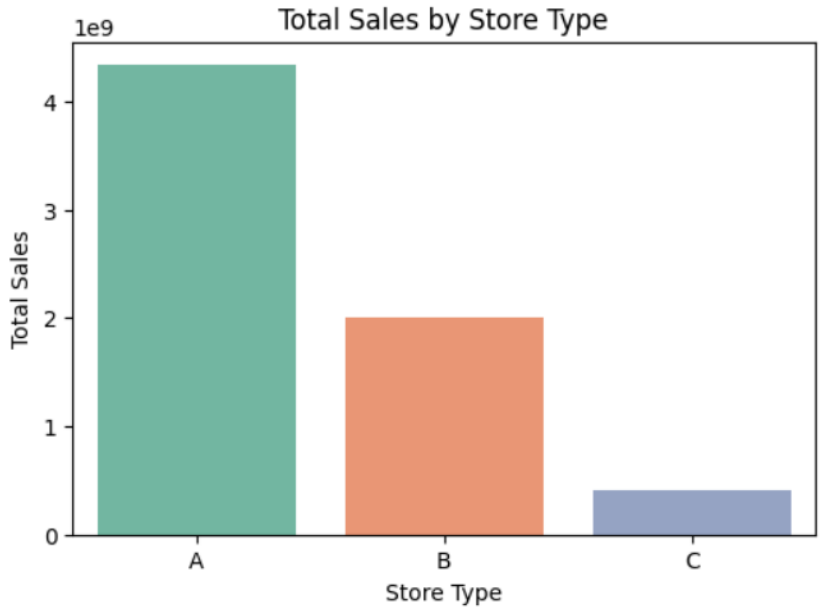
The plot clearly shows strong annual seasonality, with significant sales spikes occurring consistently around the end of each year (late 2010, late 2011). Beyond these sharp peaks, total weekly sales generally fluctuate within a relatively stable range, exhibiting some minor week-to-week variability and no pronounced long-term trend during the observed period.

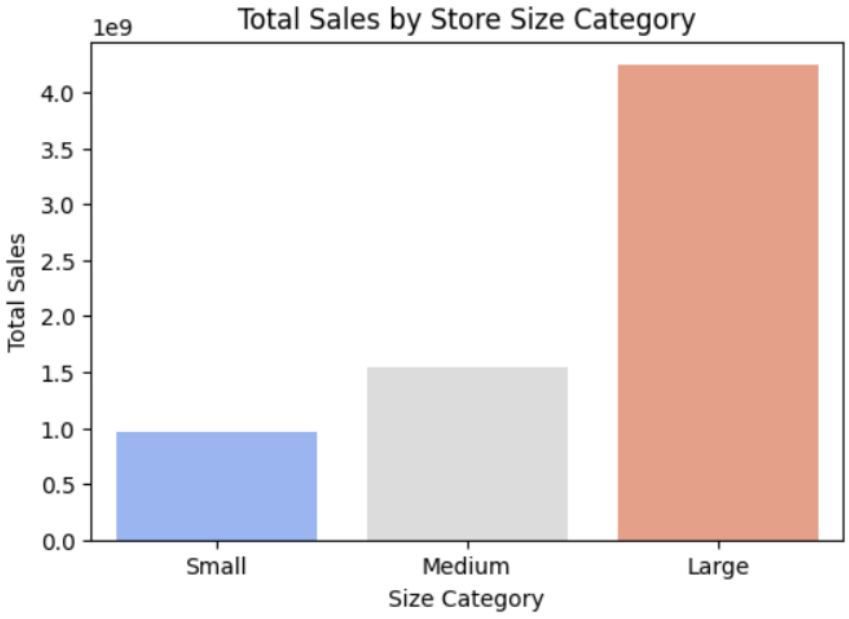
3. Top Stores and Departments (with respect to total sales)


Top 10 Stores by Total Sales
1e8
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Total Sales
1
2
4
6
10
13
14
20
27
39
Store

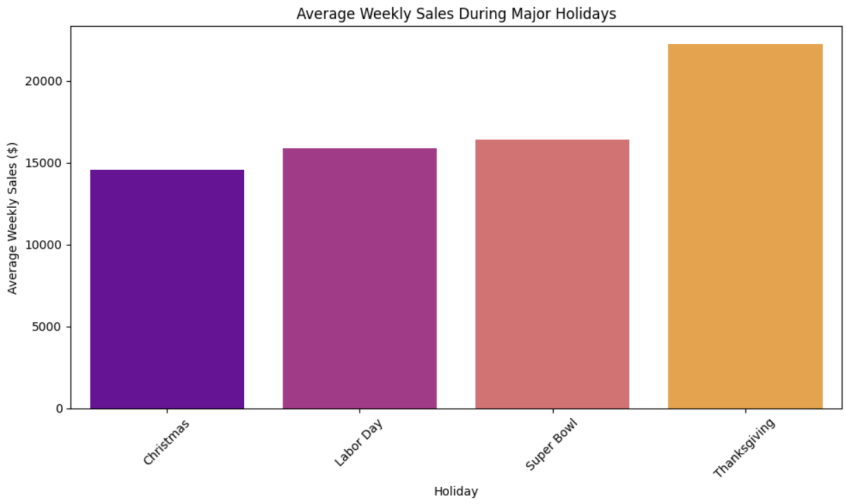

Top 10 Departments by Total Sales
1e8
5
4
3
2
1
0
Total Sales
2
8
13
38
40
72
90
91
92
95
Department

4. Total Sales by Store Type and Size



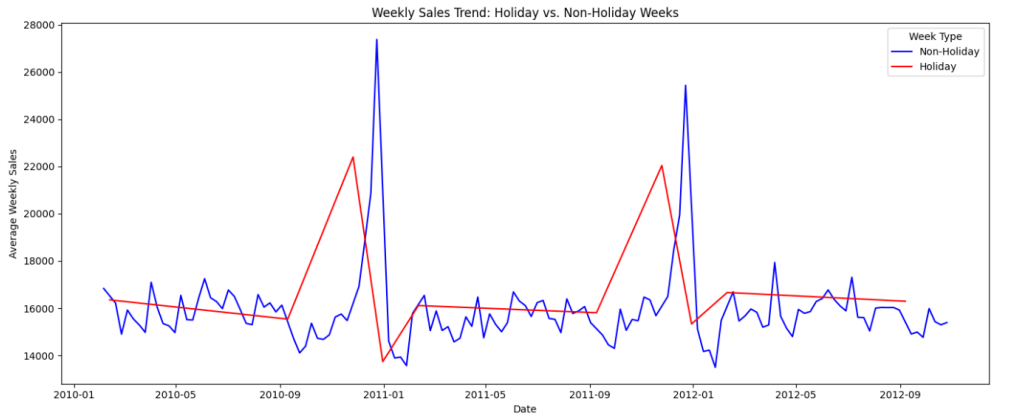


5. Average Weekly Sales during holidays



This bar chart illustrates the significant impact of major holidays on average weekly sales, with Thanksgiving clearly standing out as the period with the highest average sales. While other holidays like Super Bowl, Labor Day, and Christmas also show increased sales, their impact is notably less pronounced compared to Thanksgiving.

6. Weekly sales trend - holiday vs non-holiday



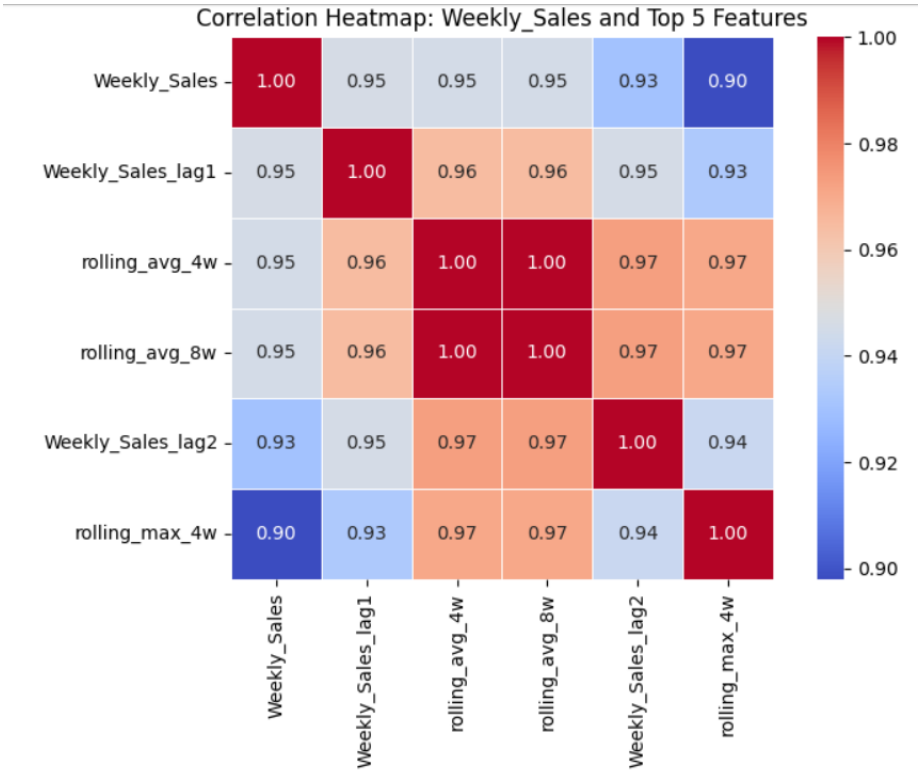
This plot effectively highlights the distinct average weekly sales patterns between holiday and non-holiday weeks. Holiday weeks (red line) consistently show pronounced spikes, indicating their significant positive impact, while non-holiday weeks (blue line) exhibit more stable, albeit fluctuating, sales within a lower range.

Feature Engineering

```
Data columns (total 26 columns):
 #   Column             Non-Null Count   Dtype
---  ------             --------------   -----
 0   Store              421570 non-null  int64
 1   Dept               421570 non-null  int64
 2   Date               421570 non-null  datetime64
 3   Weekly_Sales       421570 non-null  float64
 4   IsHoliday          421570 non-null  int64
 5   Temperature        421570 non-null  float64
 6   Fuel_Price         421570 non-null  float64
 7   MarkDown1          421570 non-null  float64
 8   MarkDown2          421570 non-null  float64
 9   MarkDown3          421570 non-null  float64
 10  MarkDown4          421570 non-null  float64
 11  MarkDown5          421570 non-null  float64
 12  CPI                421570 non-null  float64
 13  Unemployment       421570 non-null  float64
 14  Type               421570 non-null  object
 15  Size               421570 non-null  int64
 16  Week               421570 non-null  UInt32
 17  Month              421570 non-null  int32
 18  Year               421570 non-null  int32
 19  Weekly_Sales_lag1  421570 non-null  float64
 20  Weekly_Sales_lag2  421570 non-null  float64
 21  rolling_avg_4w     421570 non-null  float64
 22  rolling_max_4w     421570 non-null  float64
 23  rolling_avg_8w     421570 non-null  float64
 24  rolling_max_8w     421570 non-null  float64
 25  Size_Category      421570 non-null  category
```

Feature engineering significantly enriched the dataset, focusing on extracting more predictive signals. This involved creating temporal features (Week, Month, Year) to capture seasonality, and lagged sales and rolling window statistics (Sales of the previous week, average of sales over past 4 weeks, etc.) to reflect historical sales patterns. Additionally, Size was transformed into a Size Category to better categorize stores. These additions provide a more comprehensive view of the factors influencing weekly sales.

Correlation Matrix



This correlation heatmap reveals very strong positive correlations among Weekly Sales and its lagged and rolling average features, all exceeding 0.90. Notably, Weekly Sales lag1, rolling avg 4w, and rolling avg 8w show the

highest correlations with Weekly Sales itself (0.95), indicating their high predictive power. The strong inter-correlation among these engineered features also suggests potential multicollinearity.

Predicting Weekly Sales using time series and ML Models

The prediction strategy employs a two-stage hybrid modeling approach to leverage the strengths of both time series and machine learning techniques.

Data Splitting: The dataset was initially split into an 80% training set and a 20% testing set to ensure a robust evaluation of model performance on unseen data.
Time Series Baseline (ETS): To capture the inherent temporal patterns and overall market dynamics, an Exponential Smoothing (ETS) model was fitted to the total weekly sales summed across all stores and departments. This stage specifically accounts for trends and seasonality present at the aggregate level.
Individual Baseline Allocation: For each individual sales instance (i.e., each unique Store-Department-Date record), a naive baseline prediction was established by dividing the ETS model's predicted total sales for that date by the total number of active store-department pairs on that specific date.
Residual Modeling (Machine Learning): The core idea of this hybrid approach is to then model the residuals, defined as the difference between the actual individual weekly sales and this ETS-derived individual baseline. This allows the machine learning model to focus on explaining the granular, feature-driven deviations from the overarching temporal forecast.
Ensemble ML Model: To predict these residuals, an ensemble machine learning model was utilized. This ensemble consists of an average of two powerful regressors: a Random Forest Regressor with 20 estimators and an XGBoost Regressor with 30 estimators, chosen to enhance robustness and predictive accuracy.

Results:

(i) Root Mean Squared Error (RMSE) = 2867
(ii) Mean Absolute Error = 1386
(iii) Normalized RMSE = RMSE / Mean of Weekly Sales = 2867 / 15982 = 0.18

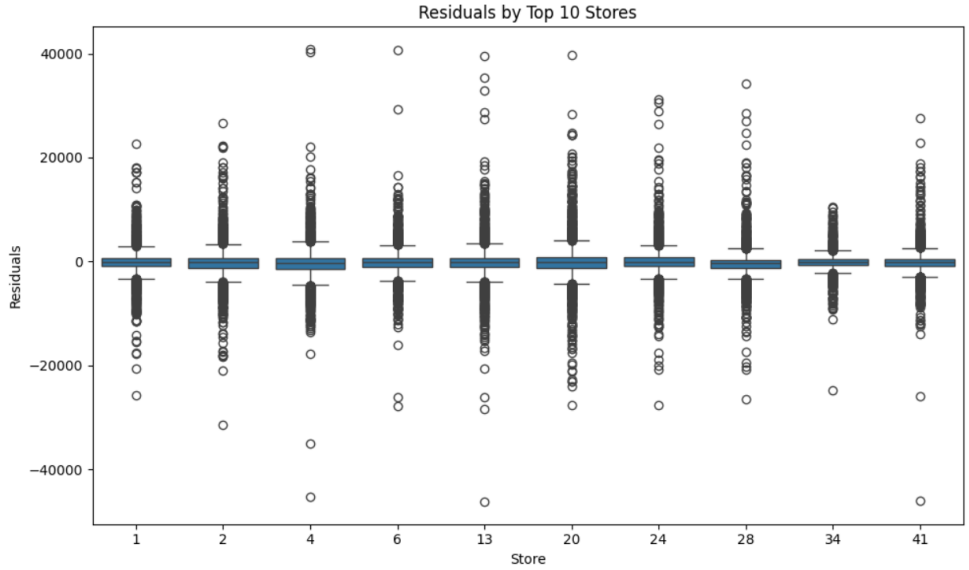
The model demonstrates a Root Mean Squared Error (RMSE) of 2867, indicating that, on average, predictions deviate by approximately $2867 from actual weekly sales, with larger errors being penalized more heavily. A Mean Absolute Error (MAE) of 1386 suggests that the typical magnitude of prediction error, without considering direction, is around $1386. The larger RMSE compared to MAE implies the presence of some larger individual prediction errors that disproportionately affect the RMSE metric.
The normalized RMSE value indicates that our model, on average, makes an error of about 18% of the average sales, which is good in retail forecasting scenarios because of the high volatility.

Residual Analysis

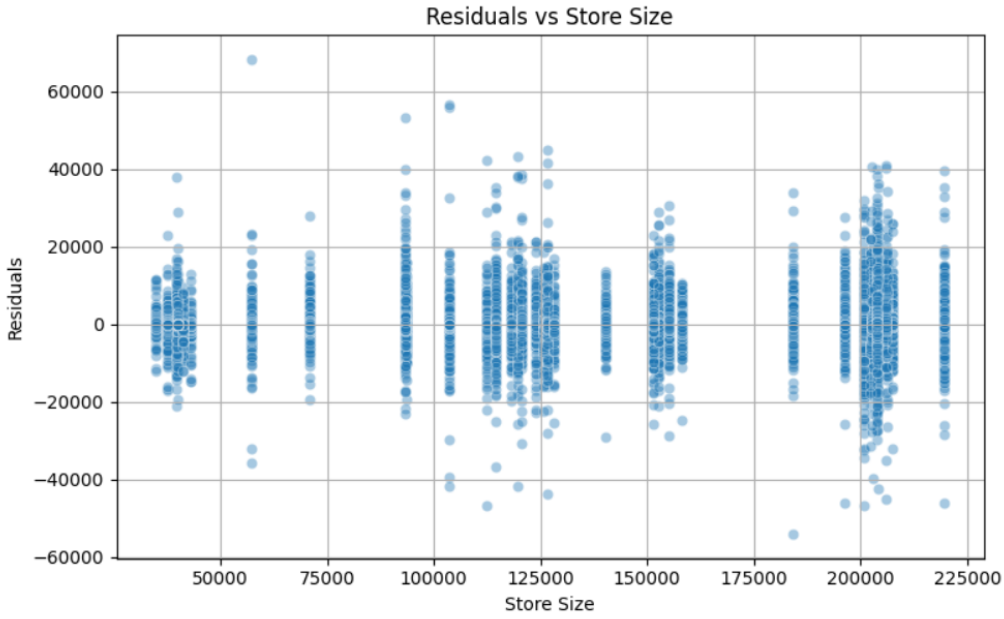
Residual analysis helps us understand Where the model is under/over-performing, and if errors vary by store, product type, holiday, or store characteristics.

1. Residual analysis by Store:

The box plot illustrates that residuals are generally centered around zero for all top 10 stores, indicating no systematic bias in predictions across these stores, though some stores exhibit a wider spread of residuals and more extreme outliers.



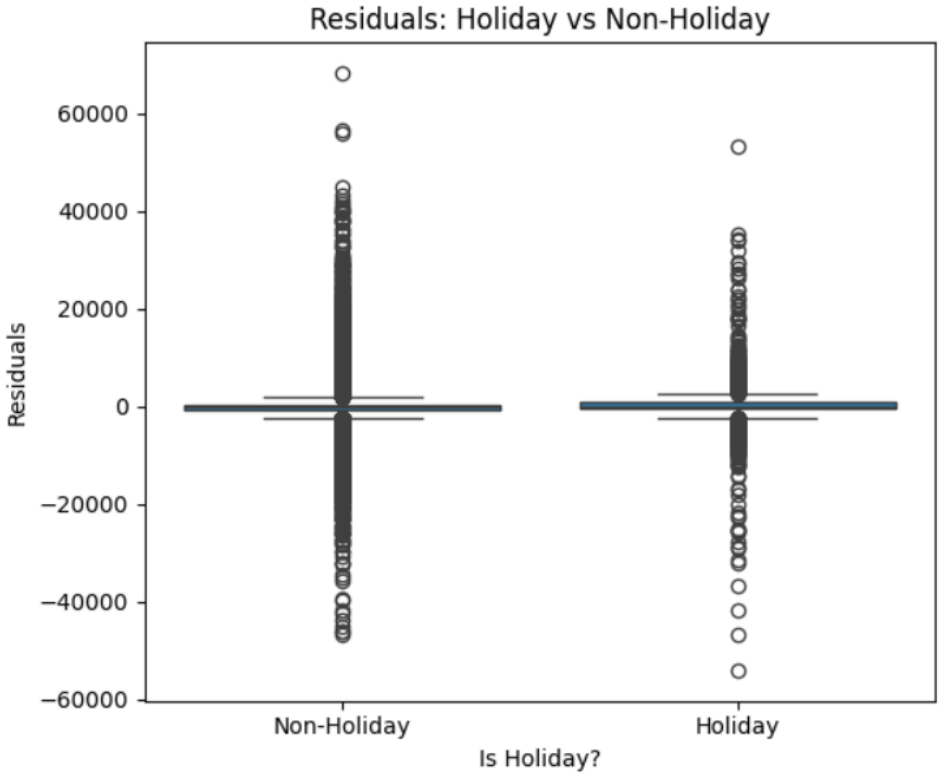
2. Residual Analysis by Size:



The scatter plot of residuals against store size indicates no apparent linear

relationship or clear pattern between the model's errors and the size of the store, suggesting that store size is adequately captured by the model or does not systematically influence prediction errors.

3. Residual Analysis : Holiday vs Non-Holiday



The box plot shows that residuals are centered around zero for both holiday and non-holiday weeks, indicating that the model, on average, predicts well for both types of weeks.

Deployment Strategy

For real-world deployment, the trained hybrid forecasting model (Exponential Smoothing + Machine Learning) will predict future weekly sales for all stores and departments based on provided input data. The process will involve ingesting store and department details for future dates, typically supplied in a CSV format.

A critical aspect of the deployment strategy is its iterative, one-date-at-a-time prediction methodology. This sequential approach is necessitated by the model's reliance on lagged and rolling features, which require the actual or predicted Weekly Sales from preceding periods. By predicting one week at a time, we ensure that these time-dependent features are accurately computed using the most recently available (or predicted) sales data, maintaining the integrity of the model's inputs and forecast accuracy over time. This approach allows the system to continuously update its internal state as new predictions become available, mimicking the flow of actual sales data.

Inventory Optimization

Following the successful deployment of the weekly sales forecasting model, in the next phase we focus on optimizing Walmart's inventory management. The primary objective was to maximize profit while minimizing stockouts and overstock situations using the predicted weekly demand.

Objective: To determine the optimal weekly order quantity per Store–Department pair based on: Predicted weekly sales (in monetary terms), Inventory holding and stockout costs and Uncertainty in demand.

Methodology:

1. Unit Level Simulation

We assumed each Store–Dept pair sells a single product with a randomly assigned unit cost between 100 and 500. Using this, we derived:
Predicted Units = Predicted Weekly Sales / Unit Cost
Actual Units was simulated using a normal distribution centered around predicted sales (to account for uncertainty in prediction)

2. Cost Structure

To evaluate profitability, the following per-unit costs were defined:
Unit Selling Price = 1.4 × Unit Cost
Unit Holding Cost = 0.35 × Unit Cost
Unit Stockout Cost = (Selling Price - Cost) + 0.5 × Unit Cost
These reflect realistic penalties for overstocking (holding cost) and understocking (stockout cost).
The unit holding cost accounts for expenses associated with storing unsold inventory, such as warehousing, insurance, and depreciation, thereby penalizing overstocking.
Conversely, the unit stockout cost reflects the loss incurred due to unmet customer demand, including lost sales, customer dissatisfaction, and potential long-term brand damage, thus penalizing stockouts

3. Dynamic Ordering Strategy

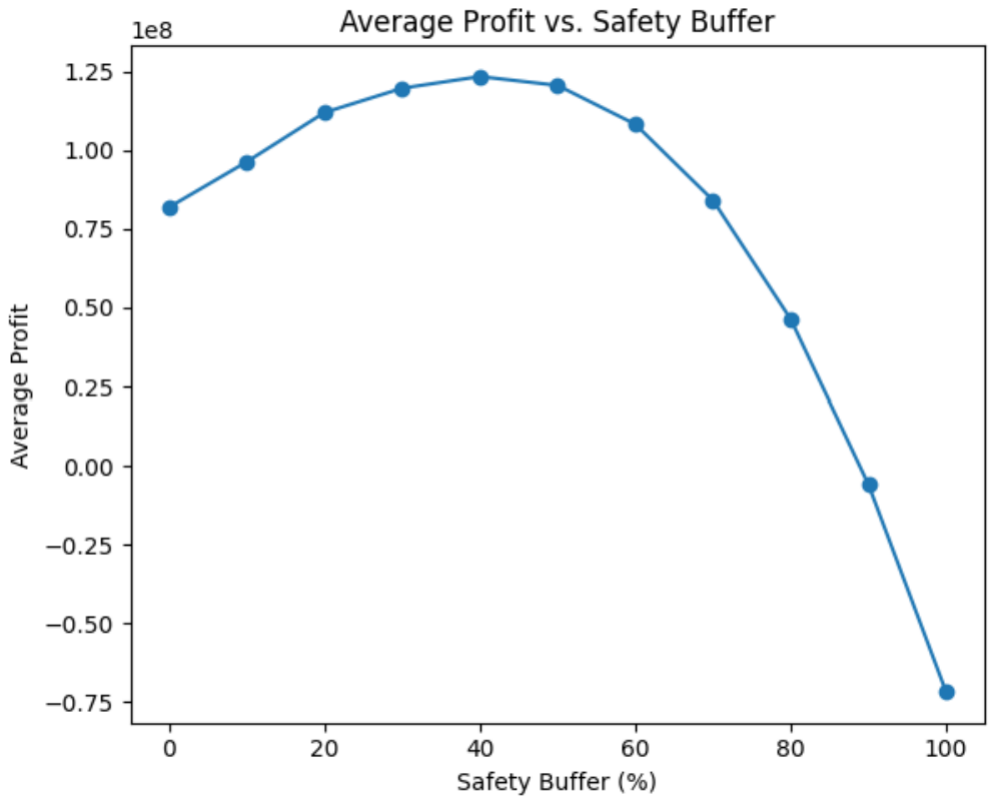
For each week, order quantity was dynamically calculated using the predicted sales for this week and next week, and the current inventory, assuming that it will take one week for the order to arrive in the store. The current inventory was updated every week based on the actual sales and the quantity that was ordered the previous week.

4. Safety Buffer

To protect against variability in demand, we introduced a safety buffer (0% to 100% in 10% increments). For each value:
A simulation of actual sales was run for all Store–Dept pairs five times.
Average Total profit and stockout rate were recorded.

Results:

Profit increased with buffer upto 40%, beyond which gains diminished.
Stockout rate consistently decreased as buffer increased, confirming its effectiveness.



40% gave a good balance between profit and service level.

Further Inventory Optimization - Dynamic Safety Buffer Strategy

To further refine the inventory ordering process, we introduced a dynamic safety buffer strategy based on the rolling standard deviation of predicted sales. The idea is to adjust the safety buffer based on the volatility of recent demand:
(i) For each (Store, Dept) pair, we calculated the rolling standard deviation over the past 4 weeks of predicted weekly sales.
(ii) Using the 30th and 70th percentiles of these rolling standard deviations across all data:
- If a week's rolling std was below the 30th percentile, it indicated stable demand, so we reduced the safety buffer to 35%.
- If it was above the 70th percentile, reflecting high variability, the safety buffer was increased to 45%.
- Otherwise, the default buffer of 40% was retained.
This adaptive strategy aims to balance profitability and service levels by tightening inventory when demand is predictable and loosening it when demand is erratic. Simulation results confirmed that this approach offers higher profit (an increase of 4.5%) while maintaining a lower stockout rate.

Conclusion

This project developed and deployed a robust predictive model for weekly sales at Walmart, utilizing a hybrid approach that combines time series analysis with machine learning, demonstrating good accuracy in a volatile retail environment. The project extended to practical inventory optimization, where a dynamic safety buffer strategy was implemented to increase profits. This comprehensive solution addresses key business objectives by enhancing inventory turnover and reducing waste through optimization.