WALMART RETAIL SALES FORECASTING
A MACHINE LEARNING PROJECT

PRESENTED BY:
NATALIA DOMÍNGUEZ JIMÉNEZ

Overview

Project Overview

🎯 Objective

Predict weekly sales across Walmart stores & departments.

💡 Why It Matters

Supports better inventory, logistics, and staffing decisions.
Critical during holidays, promotions, and seasonal peaks.

⚡ Key Challenge

Uncovering complex sales patterns shaped by time trends and holiday effects.

Dataset Description

📍 Source

- Kaggle – Walmart Sales Forecast Dataset

📂 Main Files

- train.csv → Weekly sales history (Store, Dept, Date, Weekly_Sales, IsHoliday)
- test.csv → Same structure without Weekly_Sales (to predict)
- stores.csv → Store information (Type, Size)
- features.csv → Extra variables (Temperature, Fuel_Price, CPI, Unemployment, MarkDowns, IsHoliday)

📐 Scale

- Data from ~45 stores, 81 departments, and several years of weekly records (2010-2012)

Data Preprocessing & Feature Engineering

Before training the models, the raw data required several preprocessing steps and the creation of new features. This process ensured data consistency, handled missing values, and generated more informative variables to improve model performance.

📂 Approach

- Worked only with train dataset for modeling
- Merged: train + stores + features

🧹 Preprocessing

On train and features separately
- Converted Date → datetime

On merged table
- Joined train + stores on Store
- Joined with features on Store & Date
- Found duplicate IsHoliday_x / IsHoliday_y → kept one (IsHoliday)
- Filled missing values in MarkDown1–5 with 0

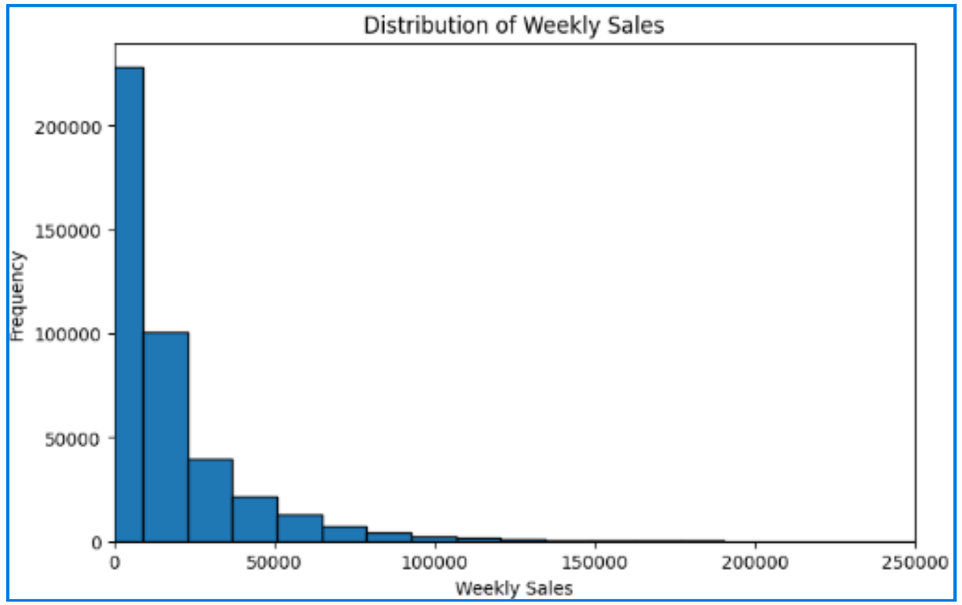
✨ Feature Engineering

On Train Table
- Date → datetime format
- IsHoliday → boolean
- One-hot encoding: Store, Dept
- Defined target: Weekly_Sales

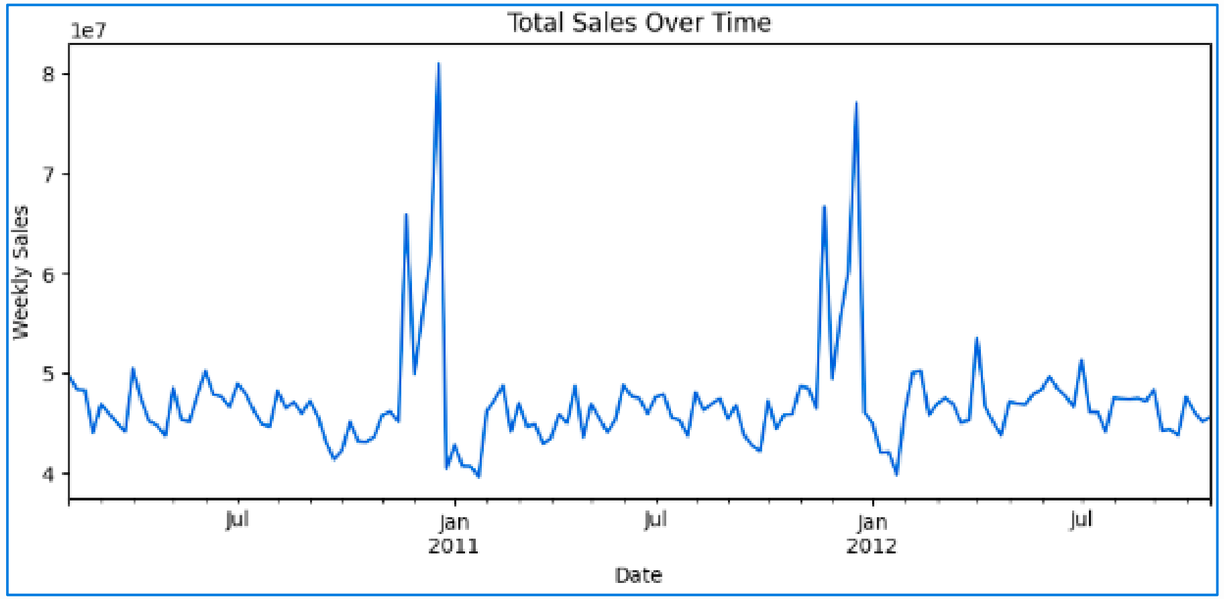
On Merged Table
- Added extra features: Type, Size, Temperature, Fuel_Price, MarkDowns, CPI, Unemployment
- Generated time features: Year, Month, Quarter
- Correlation heatmap for feature insights

Exploratory Data Analysis & Insights

Train Table



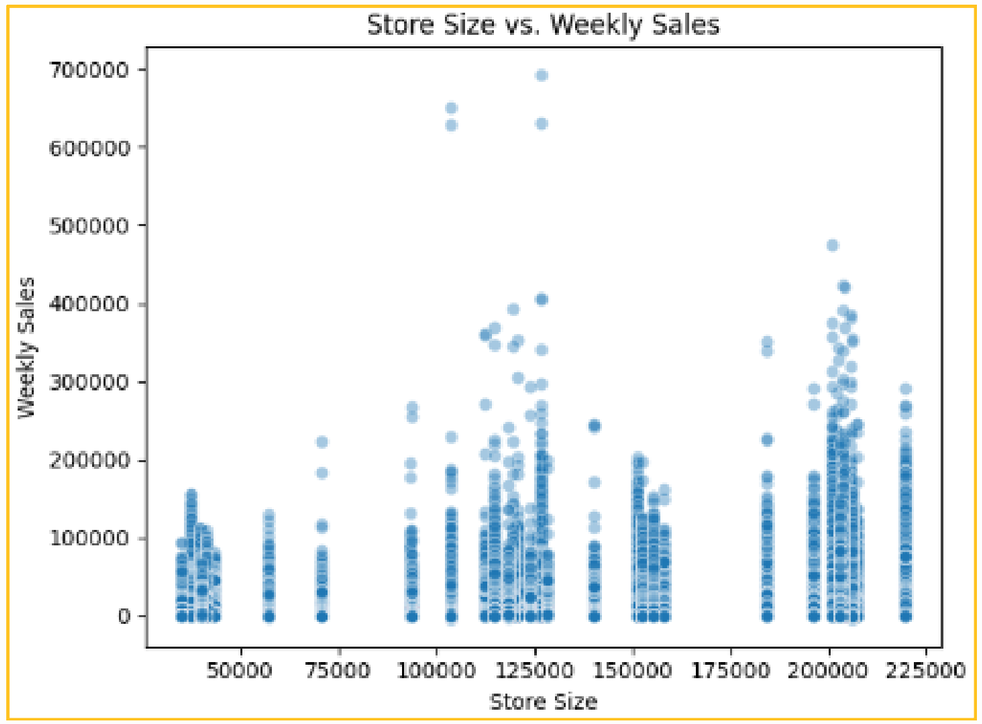
The histogram shows the distribution of weekly sales, which is highly right-skewed. Most sales values are concentrated at lower levels (below 25,000), while higher sales are less frequent and gradually decrease as the amount increases.



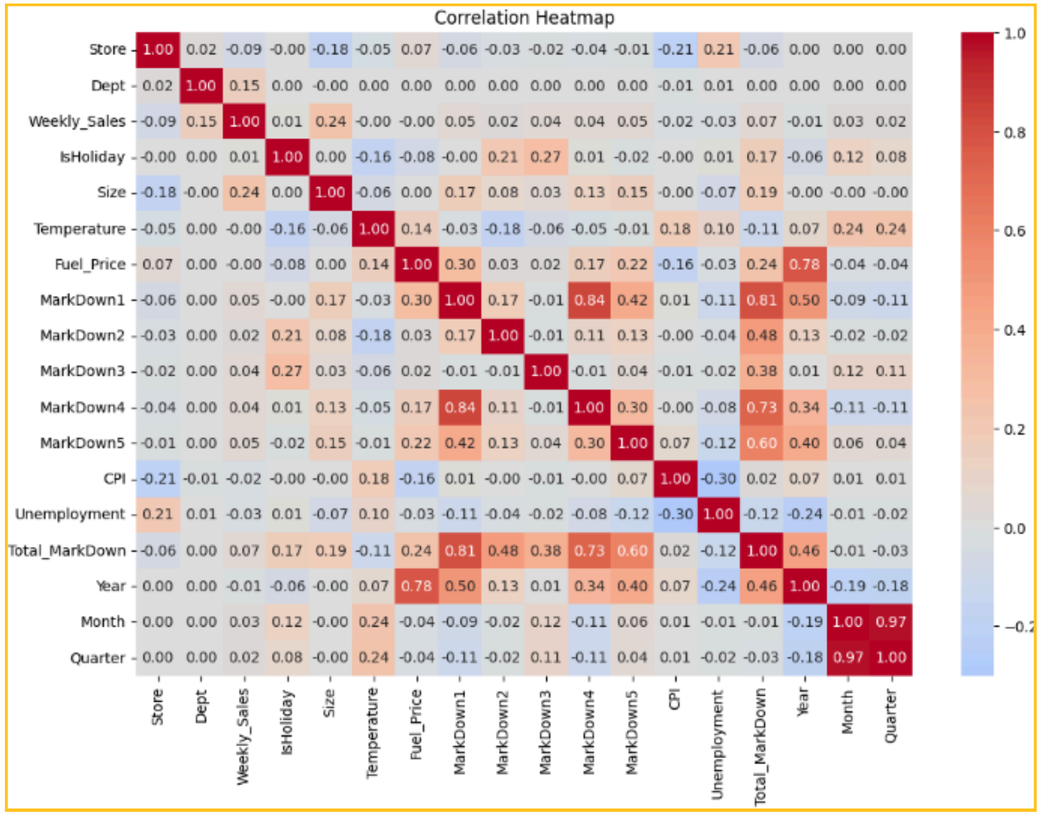
The time series plot shows weekly sales over time, with overall stable values between 40 and 50 million. However, there are noticeable spikes around the end of 2010 and 2011, suggesting seasonal peaks, likely related to holiday shopping periods.

Exploratory Data Analysis & Insights

Merged Table



While larger store sizes generally show higher weekly sales, the relationship is not strictly linear. Some smaller and mid-sized stores also achieve high sales, suggesting that additional factors such as location, store type, or promotions may play an important role in performance.



Weekly Sales show only weak correlations with most variables. The highest positive relationship is with Store Size, while Markdown features and external factors (like Fuel Price or CPI) have very low correlations, indicating that sales are likely influenced by a complex mix of factors rather than a single strong driver.

Models Tested

Train Table

Linear Regression

Random Forest

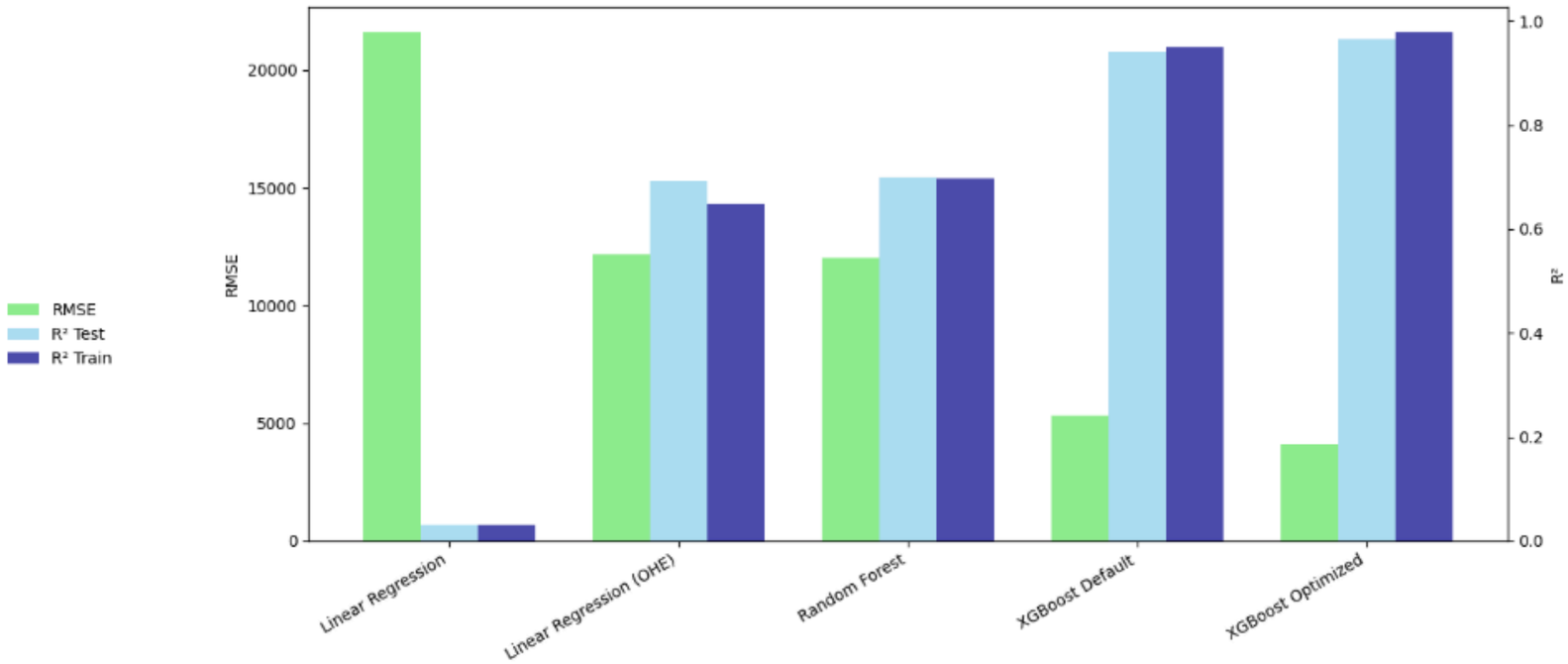
XGBoost

Metrics & Evaluation

Train Table

Train Table

Linear Regression	Linear Regression (OHE)	Random Forest	XGBoost Default	XGBoost Optimized
Linear Regression Results: RMSE: 21596.99 R² test: 0.0316 R² train: 0.0304	Linear Regression (OHE) Results: RMSE: 12151.61 R² test: 0.6934 R² train: 0.6497	Random Forest Regression Results: RMSE: 12013.69 R² test: 0.7004 R² train: 0.6963	XGBoost Regression Results: RMSE: 5312.81 R² test: 0.9414 R² train: 0.9492	XGBoost Regression (Optimized) Results: RMSE: 4088.19 R² test: 0.9653 R² train: 0.9781



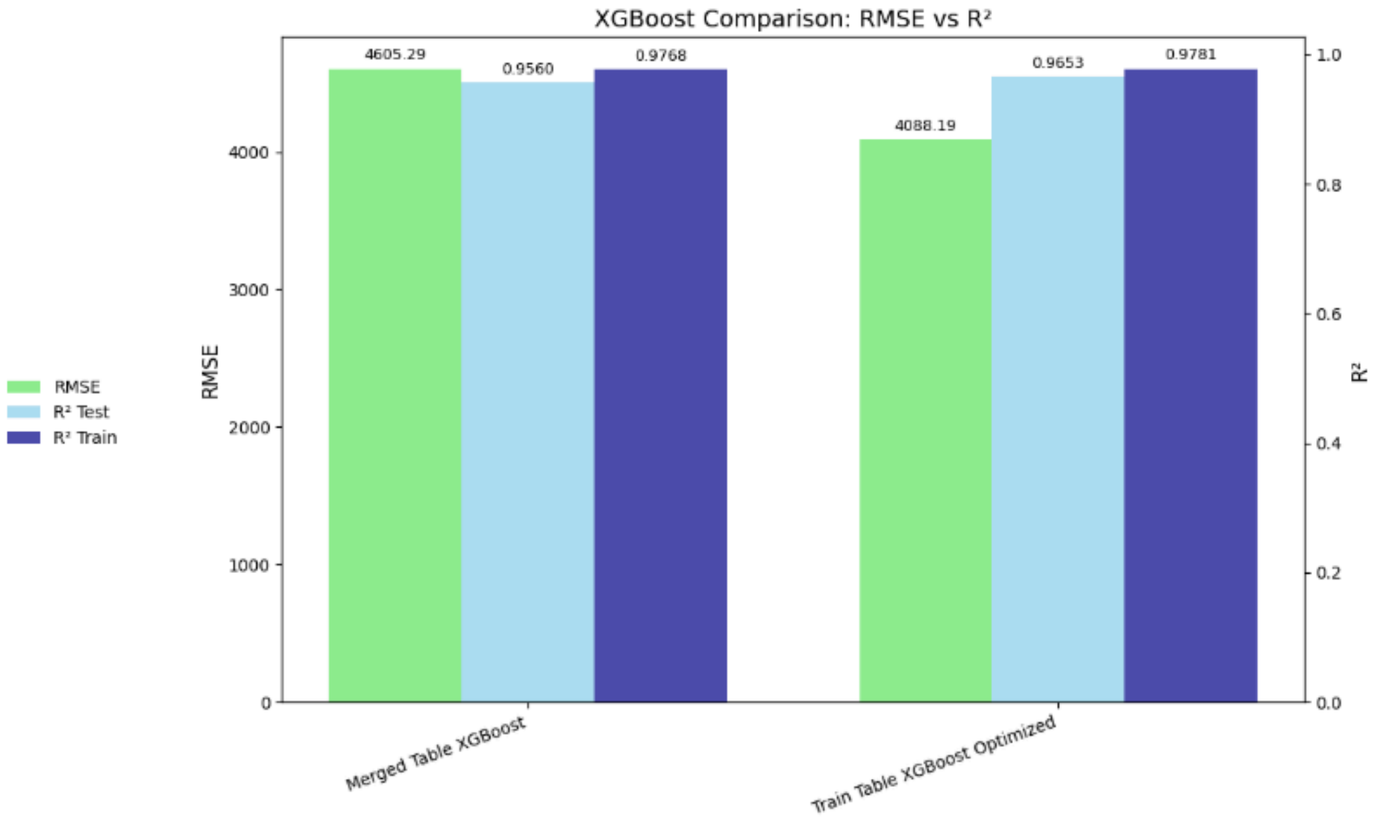
Metrics & Evaluation

Merged Table + Comparison

Merged Table

XGBoost Default

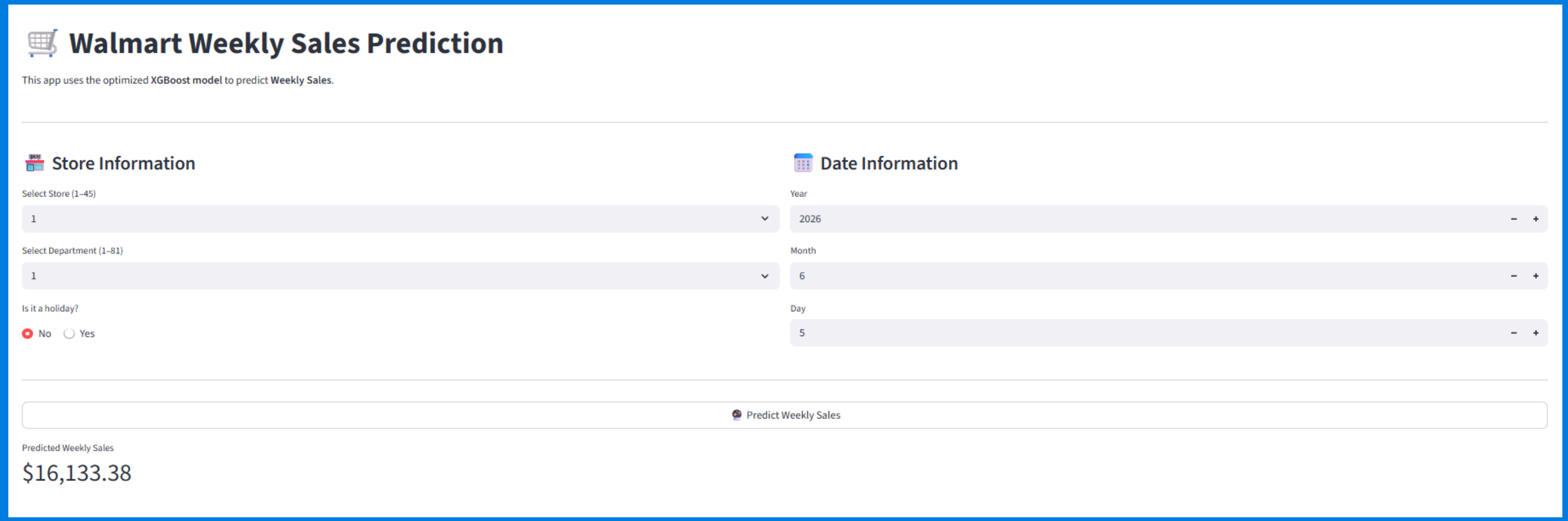
```
XGBoost Regression Results:
RMSE: 4605.29
R² test: 0.9560
R² train: 0.9768
```



Best Models

Train Table

XGBoost



Conclusions and Future Steps

01

Conclusions:

- XGBoost (optimized) performed best on the train table.
- Using the merged table confirmed the importance of seasonality, holidays, promotions, and store characteristics.
- Streamlit deployment made the model interactive and easy to use.

02

Future Steps:

- Test more advanced models (e.g., LightGBM, Prophet).
- Add external data sources (weather, regional events).
- Improve Streamlit app with more visualizations and simulations.
- Use Tableau for advanced business-oriented dashboards.
- Apply SQL for efficient data extraction and pipeline integration.
- Deploy on cloud for scalability and real-time predictions.

THANK YOU!