

# Walmart Sales Predictions

Group 6

Raisha Rawal  
Kerek Spinney  
Silas Phillips

# Problem Statement

- We used historical sales data for 45 Walmart stores located in different regions. Each store contains a number of departments, and our task was to predict the weekly sales for each store. In addition, Walmart runs several promotional markdown events throughout the year. These markdowns precede prominent holidays, the four largest of which are the Super Bowl, Labor Day, Thanksgiving, and Christmas.
- More specifically, the nature of the Weekly Sales variable classifies this as a regression problem.
- Regression techniques are employed when the objective is to predict a continuous quantitative value, such as sales figures, based on a set of input features or predictors.





# Data Sources and Info

The project utilized data from four primary sources:

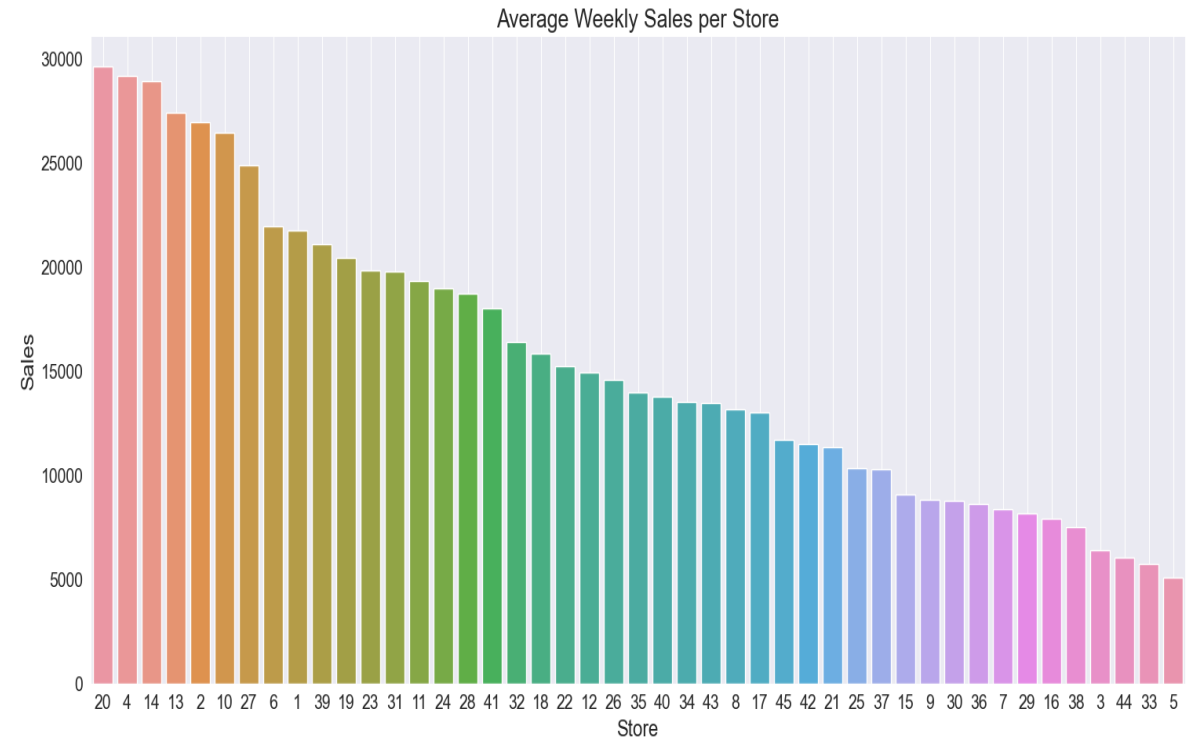
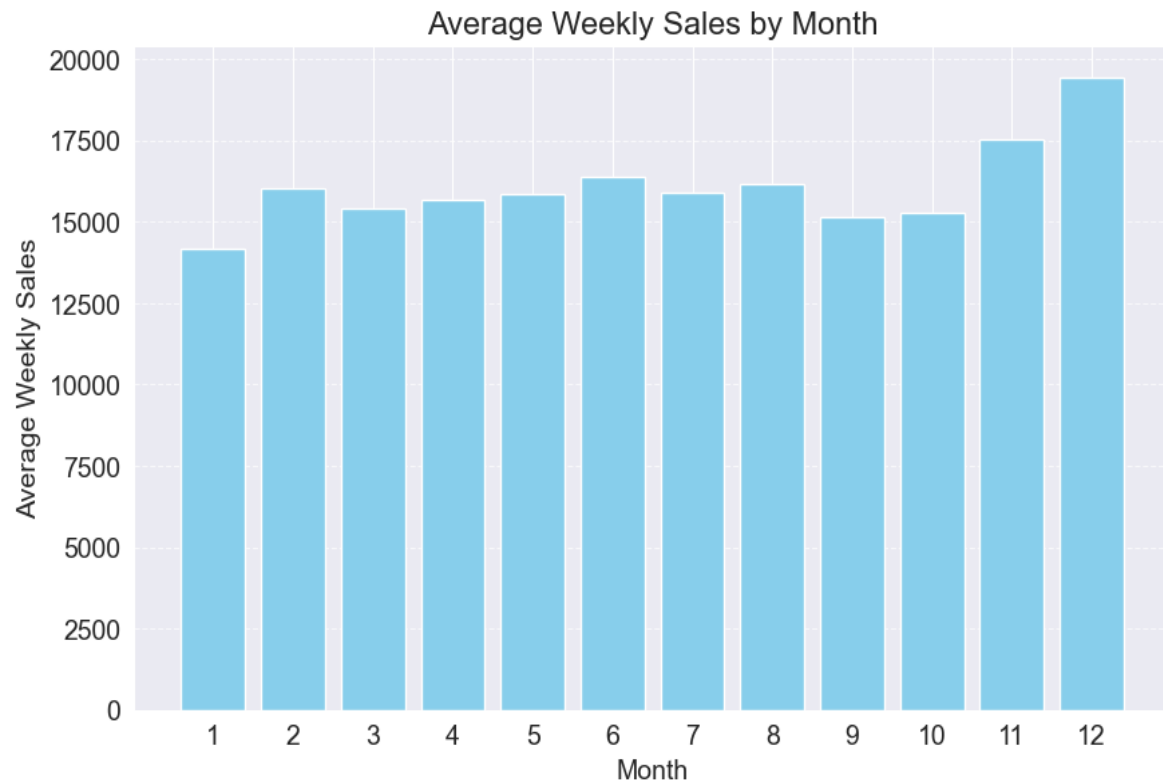
stores.csv (store, Dept, Date, Weekly Sales and IsHoliday) with a shape (45, 3)

features.csv (store, date, , temperature, fuel prices, and 5 Markdown columns, CPI, Unemployment, IsHoliday with a shape (8190, 12)

Test and train.csv have the same feature(historical sales data). The target variable was identified as Weekly Sales, with Store, Dept, Date, and IsHoliday serving as key predictors.

Train Shape (421570, 5),  
Test shape (115064, 4)

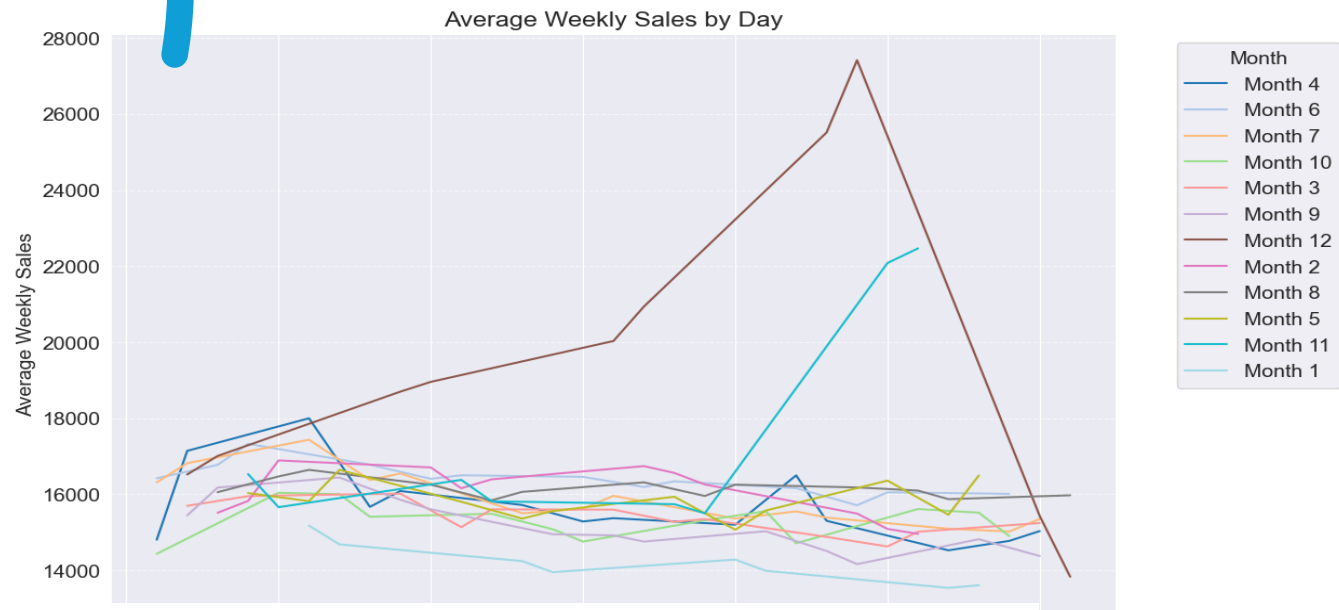
# Exploratory Data Analysis



# EDA continued

From this heatmap, Size .24 and Dept .15 have the highest correlation with Weekly Sales.

	Weekly_Sales
Store	-0.09
Dept	0.15
IsHoliday	0.01
Temperature	-0.00
Fuel_Price	-0.00
CPI	-0.02
Unemployment	-0.03
Size	0.24
Day	-0.01
Month	0.03
Year	-0.01
WeekOfYear	0.03
Quarter	0.02
Markdown	0.07
Weekly_Sales	1.00



The weekly sales are consistent with spike around the<sup>0</sup> November and December Holidays.

# Data Preprocessing and feature engineering

- We merged the train and test datasets with the features. Doing this adding additional information that can be useful for training and testing machine learning models.
- Converting IsHoliday boolean data to 0 if False and 1 if True
- The only features with missing values are Markdown(1–5) Any missing value is marked with an NA. We will replace the NaN values in these columns with 0.
- Breaking down the Date column into Day, Month, Year, Week of Year and Quarter into separate columns.

# Model Development and Evaluation

---

## Baseline ML model —

Linear Regression-Linear  $R^2 = .088$  (very poor)

## Model Evaluation-

Random Forest Regressor Model Scores

- Training  $R^2 = 1.00$
- Validation  $R^2 = .98$

XG Boost Scores

- Training  $R^2 = .95$
- Validation  $R^2 = .94$

Gradient Boosting Regressor Scores

- Training  $R^2 = .74$
- Validation  $R^2 = .74$



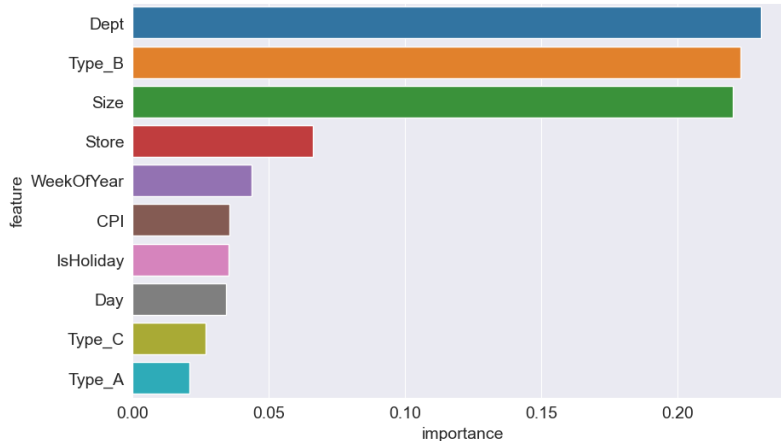


# Feature Engineering and Selection

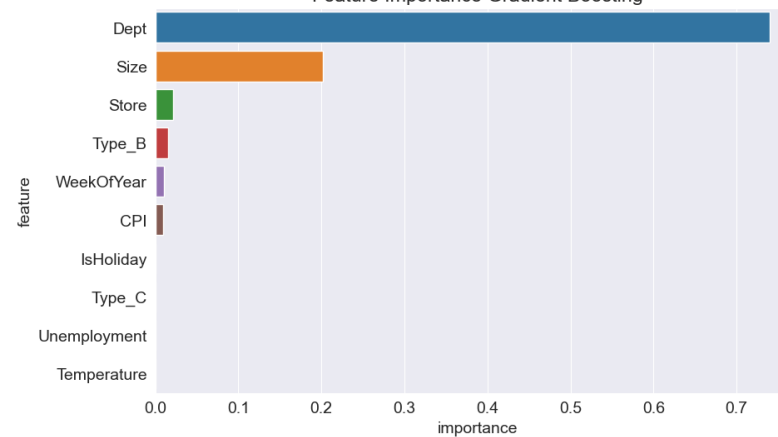
Department and size are the two most important features across the 3 models.

We also know from the EDA that the Holidays have a higher sales count and we wanted to account for this. We did this by assigning higher weights to holiday samples, we emphasized the importance of accurately predicting sales during holidays. Holidays may have significantly different sales patterns compared to regular days.

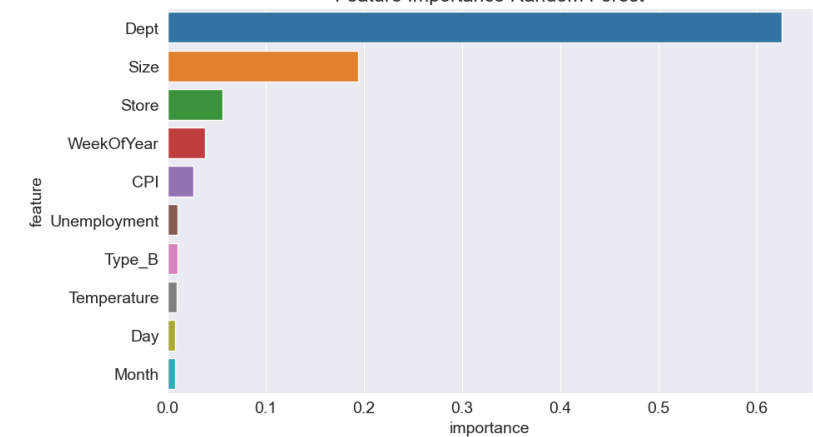
Feature Importance XGBoost



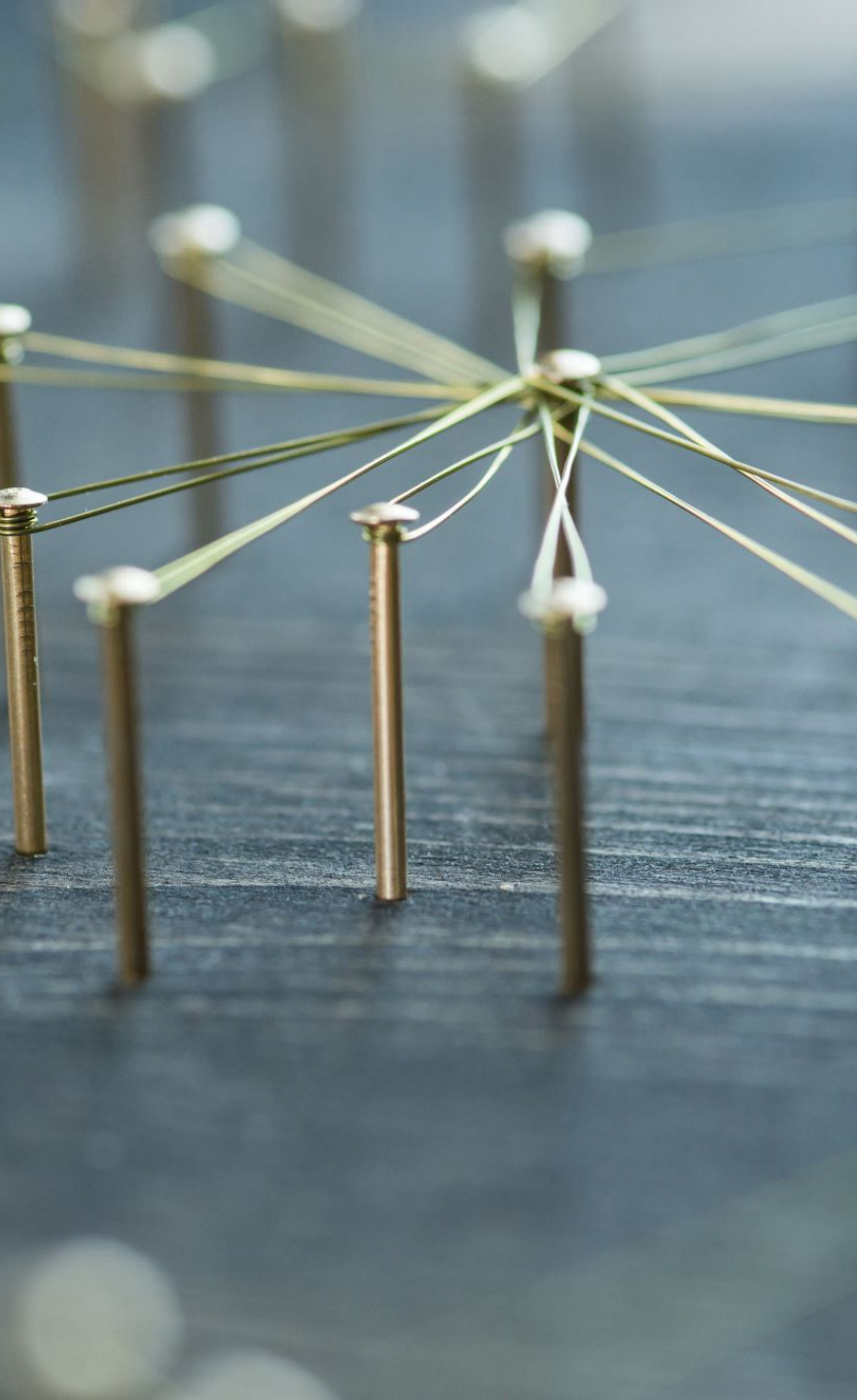
Feature Importance Gradient Boosting



Feature Importance Random Forest








# Hyperparameter Tuning

Initial results indicated that linear models performed poorly, while ensemble models showed promise. Hyperparameter tuning was conducted using cross-validation, with Random Forest emerging as the best-performing model after optimization.

We chose 2 of the 3 models using the following variables and based on the results we further tuned the Random forest model.

```
RandomForestRegressor,  
    max_depth: [5, 10, 15, 20, 25],  
    n_estimators: [20, 50, 200, 250],  
    min_samples_split: [2, 4, 5, 10],  
XGBRegressor,  
    max_depth: [3, 4, 6, 8, 9, 10],  
    n_estimators: [30, 50, 200, 250]  
    learning_rate: [0.3, 0.2, 0.1, 0.01, 0.001]
```



# Choosing the model and adjusting the settings

---

We picked the Random Forest Regressor because we think the first outputs were overfitted and wanted to test our theory. The `n_estimator`, max depth and min samples were picked based on the test parameters.

```
model = RandomForestRegressor(random_state=42,  
                              n_jobs=-1)
```

Based on the graphs we selected the following variables.

```
n_estimators: [200],  
max_depth: [8],  
min_samples_split: [5]
```

The results of the tuning.

Cross-validation  $R^2$  scores: [0.96626685 0.96708718 0.96505649 0.96797142 0.96504317] Mean  $R^2$ : 0.9662850228946687 Validation  $R^2$ : 0.9637673284267992.

Based on this result the Random Forest performed the best with a  $r^2$  score of 96%.

# Conclusions

The project successfully developed an end-to-end machine learning model for sales forecasting at Walmart. Key factors driving weekly sales were identified, providing valuable insights for business decision-making.

However, there is room for further improvement. Exploring time-series modeling techniques, such as ARIMA, could potentially enhance the accuracy of the forecasts. Additionally, incorporating additional features, such as promotional information and competitor data, may provide a more comprehensive understanding of the sales dynamics.

This project serves as a foundation for further advancements in predictive modeling in the class and after we leave and jump into the deep waters of the real world.

