Predicting Walmart’s Weekly Revenue

Shangshang (Sophia) Song

Mentor: Thyago Porpino

Springboard Data Science Career Track 2019

Introduction

Walmart is an American retail operation that operates a chain of grocery stores by the same name. Since its opening in 1962, it has expanded its ventures to many other countries, including Mexico, China, and Germany. As of 2019, Walmart is the world's largest company by both revenue and number of employees, at $514.405 billion per year and 2.2 million people, respectively.

Like all companies, Walmart's revenue is impacted by a variety of factors, some of which are included in the Walmart Recruiting dataset on Kaggle https://www.kaggle.com/bletchley/course-material-walmart-challenge#test.csv). It contains information about the weekly revenue of 45 Walmart stores over 2.5 years, promotion events, and other information about the area in which the stores operate, such as the average temperature and gas price. Using this dataset, I would like to fit a regression model to predict the weekly sales made given the factors.

Data Description and Wrangling:

The data comes in two .csv files, one for testing and one for training. The training dataset contains 282,000 rows and 16 columns, one of which is the weekly revenue column that's the object of our prediction. The other 15 features are:

- Store ID (number)
- Department ID (number)
- Date
- Whether the week contains a holiday (boolean)
- The temperature of the area in Fahrenheit (number)
- The gas price of the area in dollars (number)
- 5 markdown columns with information about promotional offers (number).
- The Consumer Price Index (number)
- The unemployment rate (number)
- The Type of the store (strings)
- The size of the store (number)

The Testing dataset has identical features with the training dataset but does not have the weekly revenue column, which we would have to predict ourselves.

To clean the data, I looked at the summary of the dataframe and found that 1) there were several null values in the MarkDown columns, 2) the date column is not in DateTime format, but in object, and 3) the type column is shown to be type object, when type category would both consume less memory and be more convenient for OneHotEncoding later.

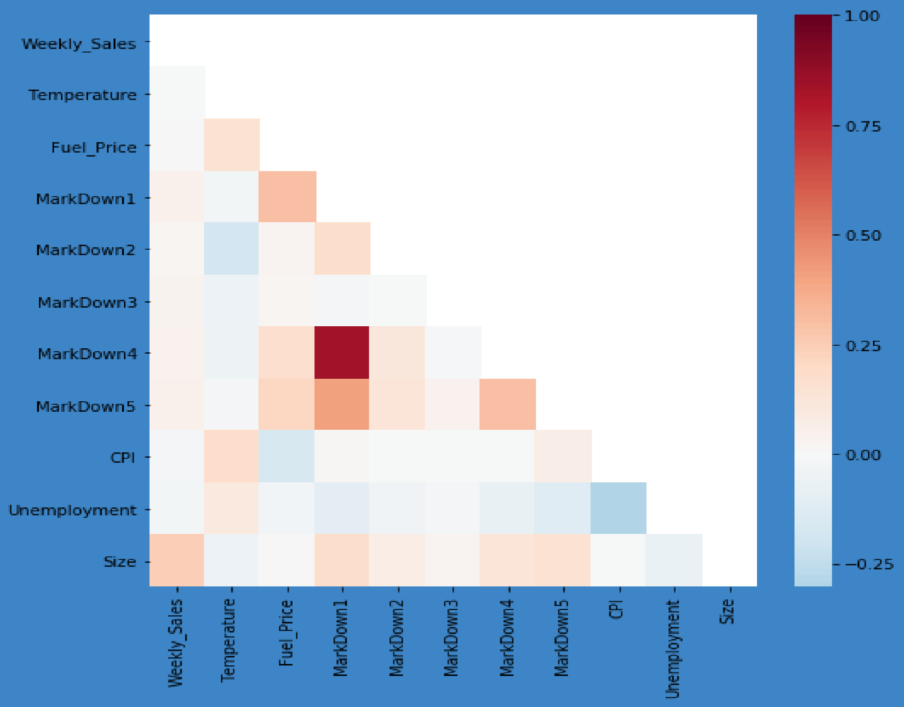
To make this dataset easier for analysis, I first filled in all the null values in the MarkDown columns with 0. The description of the dataset said that the MarkDown columns “contain information about promotional events”, so it makes sense to conclude that null values in MarkDown columns mean that there were no promotional events on that week in that particular store and department. Then, I changed the Date column to datetime format and the type column to category format for easier analysis.

Exploratory Data Analysis and Inferential Statistics -- Numerical:

The numerical variables are: Temperature, Fuel_Price, MarkDowns 1-5, CPI, Unemployment, and Size. Both the weekly revenue and the markdown columns have an exponential distribution, with most of the values on the left side of the graph. CPI and fuel prices are bimodal, which is related to the year, and the size column is trimodal which corresponds with the three categories in the type column. The remaining columns, unemployment and temperature, are both approximately normal.

There are also some collinearity among the variables (Most notably between the Markdown columns), which is visualized in a heatmap (Figure 1)

Figure 1. Heatmap of the numerical variables.



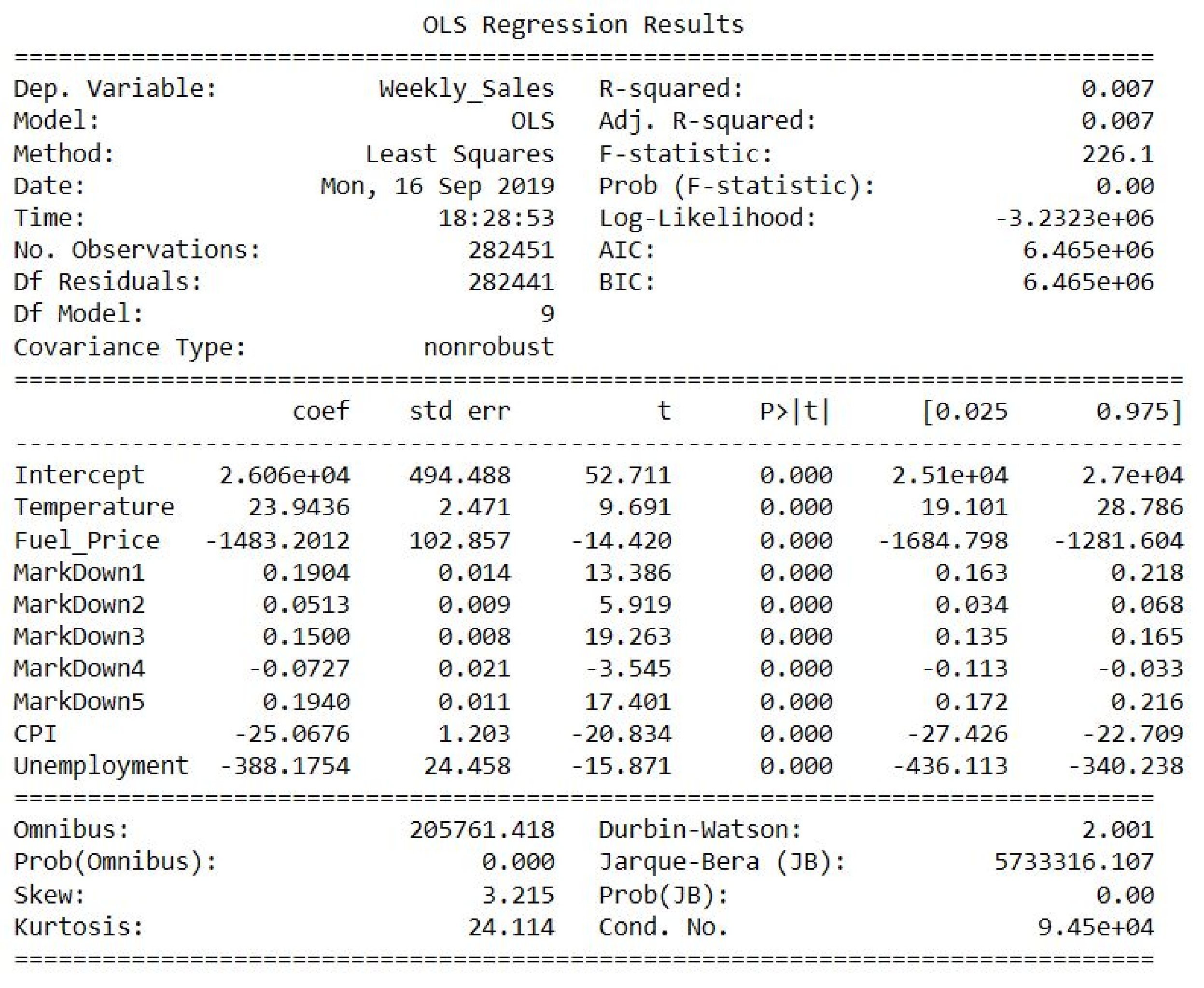
Some of them make sense: For example, the markdown columns, especially MarkDown1, MarkDown4, and MarkDown5, are very positively correlated with each other. Perhaps it'd be helpful to condense all of the markdown columns into one.

Some of the correlations,

however, seem to make little sense, for example the negative correlation between MarkDown2 and Temperature.

Let’s see if there are significant correlations between these variables and the weekly revenue by fitting a model using the statsmodel ordinary least square linear regression model.

Figure 2. OLS report of numerical variables.



```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:           Weekly_Sales   R-squared:                       0.007
Model:                            OLS   Adj. R-squared:                  0.007
Method:                 Least Squares   F-statistic:                     226.1
Date:                Mon, 16 Sep 2019   Prob (F-statistic):               0.00
Time:                        18:28:53   Log-Likelihood:            -3.2323e+06
No. Observations:              282451   AIC:                         6.465e+06
Df Residuals:                  282441   BIC:                         6.465e+06
Df Model:                           9                                         
Covariance Type:            nonrobust                                         
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept     2.606e+04    494.488     52.711      0.000    2.51e+04     2.7e+04
Temperature     23.9436      2.471      9.691      0.000      19.101      28.786
Fuel_Price   -1483.2012    102.857    -14.420      0.000   -1684.798   -1281.604
MarkDown1        0.1904      0.014     13.386      0.000       0.163       0.218
MarkDown2        0.0513      0.009      5.919      0.000       0.034       0.068
MarkDown3        0.1500      0.008     19.263      0.000       0.135       0.165
MarkDown4       -0.0727      0.021     -3.545      0.000      -0.113      -0.033
MarkDown5        0.1940      0.011     17.401      0.000       0.172       0.216
CPI            -25.0676      1.203    -20.834      0.000     -27.426     -22.709
Unemployment  -388.1754     24.458    -15.871      0.000    -436.113    -340.238
==============================================================================
Omnibus:                   205761.418   Durbin-Watson:                   2.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):          5733316.107
Skew:                           3.215   Prob(JB):                         0.00
Kurtosis:                      24.114   Cond. No.                     9.45e+04
==============================================================================
```

All of the variables are significantly correlated with the weekly sales, as seen in the P-values. There, however, are significant intercorrelations as seen in the large condition number.

The correlations are mostly as we expected: Fuel price, CPI, unemployment all have a negative correlation with the weekly sales while temperature has a positive correlation.

The effect of markdowns are a bit more ambiguous, and to see if grouping by the presence of any markdowns would help provide a clearer picture, I created a new boolean IsMarkDown column and analyzed the new grouping in the next section.

Exploratory Data Analysis and Inferential Statistics -- Categorical:

For the categorical variables, I performed a frequentist hypothesis test to see if the weekly sales are different for variables with two categories (IsHoliday and IsMarkDown). For the Type column, which contains 3 options, I did an ANOVA (Analysis of Variance) test to see if the type of store is relevant for the revenue prediction.

There are three requirements that must be met for frequentist hypothesis testing:

- First, the samples must be independent.

This assumption may be questionable, because the dataset has a time-series element to it and each week's revenue may have an impact on the revenue next week. For our purposes, though, we can assume that that each row is an independent observation based only on the features.

- Second, the samples must be random.

This is an assumption, but since the description of the dataset doesn’t mention any biases in selection we can assume that there is none.

- Third, the distribution must be normal.

By “distribution” here, we are referring to the distribution of the mean of the data according to the central limit theorem, which means that the sample data should either be 1) normal, or 2) greater than 30. As both the test and training data have more than 30 entries, this condition is satisfied.

Holiday

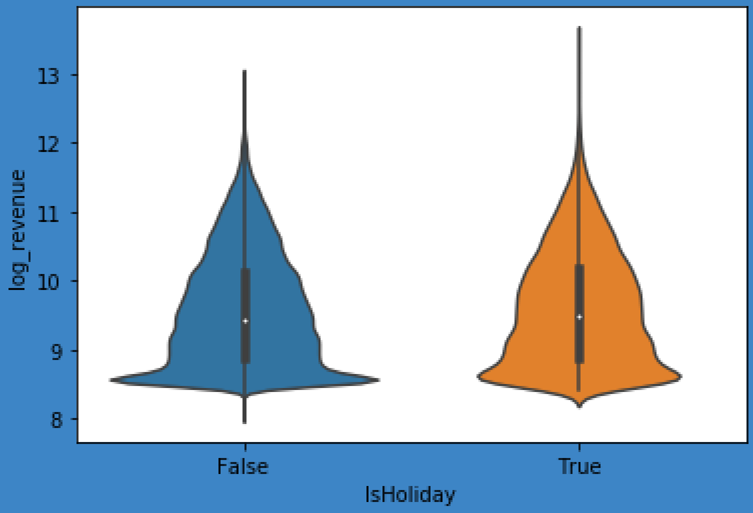
There are 19819 rows with IsHoliday==True included out of the total 282450 rows. Next I looked at the unique values included to see if these dates are holidays, and if there are any holidays missing. Below are the holidays Walmart gave:

Super Bowl: 12-Feb-10, 11-Feb-11, 10-Feb-12, 8-Feb-13
Labor Day: 10-Sep-10, 9-Sep-11, 7-Sep-12, 6-Sep-13
Thanksgiving: 26-Nov-10, 25-Nov-11, 23-Nov-12, 29-Nov-13
Christmas: 31-Dec-10, 30-Dec-11, 28-Dec-12, 27-Dec-13

Given that the range of dates included go from 2010-02-05 to 2012-10-26, we should have all of the holidays except Thanksgiving and Christmas of the years 2012 and 2013. A look at the unique date values for IsHoliday==True says we do.

Next, I compared the distribution of the revenue for holiday and non-holiday weeks in Figure 3. As can be seen, holiday weeks generated more revenue compared to non-holiday weeks. As the distribution of weekly revenue is very skewed to the right, I took the log of the revenue in the violin plots for clearer comparison

Figure 3. Distribution of the log of revenues for Holiday and Non-Holiday weeks



Descriptive statistics and a hypothesis test confirms what we see: the average revenue for holiday weeks is 16986.67 dollars with a standard deviation of 26955.16 dollars, while non-holiday weeks have a mean revenue of 15907.72 dollars with a standard deviation of 22301.76 dollars, and holiday weeks see significantly higher weekly revenue compared to non-holiday weeks.

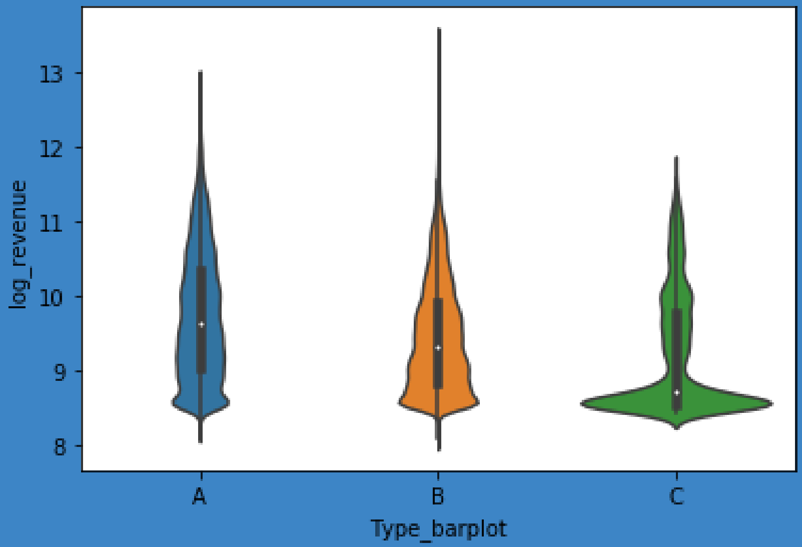
Type

Browsing the data, I noticed that type A stores tend to be larger than type B stores, which are in turn larger than type C stores. A boxplot (Figure 4a) confirms my observation

Figure 4a. A box-and-whiskers plot of the sizes for each type of store.

Next we look at the weekly revenue. Surprisingly, the highest revenue was generated by a type B store (Figure 4b). On average, however, the weekly revenue is as we expected with the largest type of store generating the most revenue.

Figure 4b. Types of Stores vs. Weekly Sales

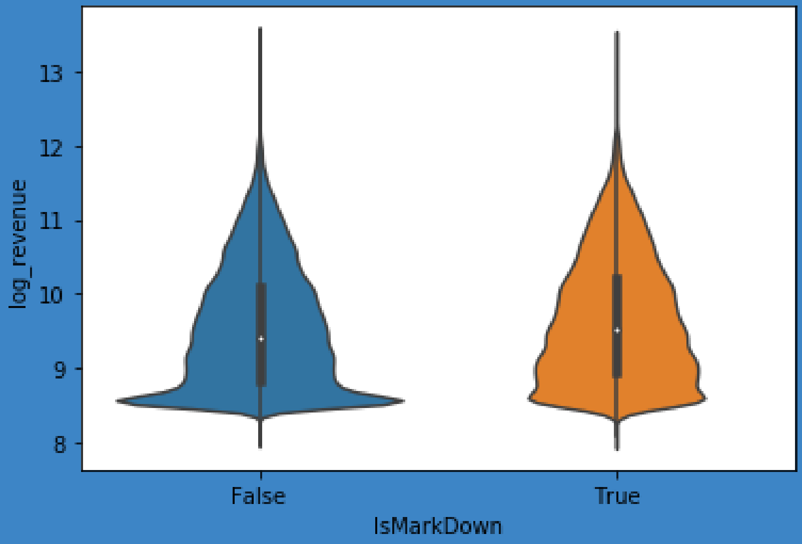


The ANOVA analysis yielded a F-value of 5290, which is larger than 3, or the threshold F-value for alpha = 0.05. This would say that the type of store significantly affects the weekly revenue.

IsMarkDown

There were 64818 rows with markdowns and 217633 rows without markdowns. First, we looked at violin plots of the logged weekly revenues made during weeks with markdowns vs. during weeks without markdowns (Figure 5).

Figure 5. IsMarkDown vs. Weekly_Sales



The highest number of sales were made on a week without markdowns. However, the average revenue for markdown weeks is 19867.04 dollars with a standard deviation of 24729.34 dollars, while non-holiday weeks have a mean revenue of 15422.34 dollars with a standard deviation of 21976.45 dollars. A hypothesis test yielded a p-value of 1.95 * 10^-8, so it appears that revenues made during markdown weeks are significantly higher than non-markdown weeks.

Feature Engineering

Before fitting a model with our training data, I deleted some features to reduce redundancy and engineered some new features to either restate the information in the deleted columns in a way more interpretable for the model, or to derive predictors that could be helpful for revenue prediction.

The features I engineered are:

1) Total Markdowns

As seen in Figure 2, the markdown columns are highly correlated with each other and contribute to multicollinearity among the features. Considering that the total amount of discounts is more likely to contribute to revenue than a specific promotional event, I combined them into one Total Markdowns column, which kept the relevant information about how much discounts there were and reduced redundancy.

2) Type A, B, and C store dummy variables.

In order to use categorical variables in regression models, they have to be converted to dummy variables.

3) **Year, Month, Week of Year, and Day**

The year, month, week, and day could have an impact on the revenue made. The original date format won't work with a lot of models and thus, I parsed the date into year, month, week of year, and day columns.

4) **Yearly Median value of each Store-Dept combination**

This provides a baseline value for each store-department combination that the models could modify for each week given the other attributes. I'm using the median instead of the mean because the mean can be biased by extreme values, and thus won't give as clear an indicator of each store-department's average performance.

5) **Lagged Revenue and its difference from Median**

The median provides a baseline for our predictions, but doesn't tell us anything about how well the specific department-store combination is doing at a specific period in time. To account for this, I added a column with the weekly revenue of the department-store combination during the week before. I'm also adding a difference from median to give information on how well the particular combination did last week relative to its median.

6) **Big Holidays**

It appears that a significant portion of weekly revenues > $300,000 are made in the Thanksgiving, Christmas weeks, so I made some dummy variables for them. The week before Christmas also sees higher sales, so I made a dummy variable for that, too.

7) **Department 72 dummy variable**

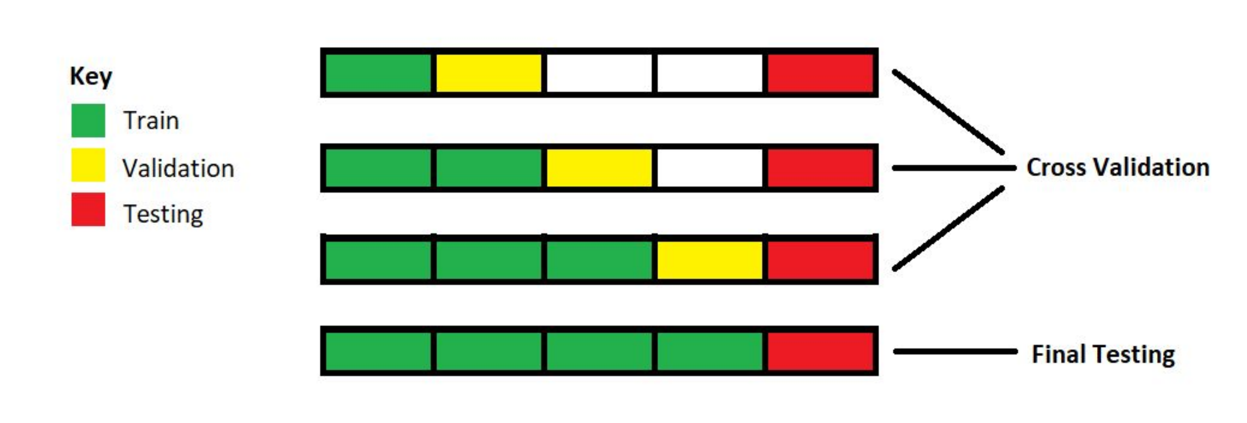
Looking at the EDA, department 72 sees higher sales compared to other departments, and so I created a dummy variable for that.

Machine Learning

After adding and deleting the features mentioned in the “Feature Engineering” section, I split the data in the “training” csv file into the training and testing dataset, with the chronologically latest 20% as the test dataset and the remaining 80% as the training dataset. I split the data this way instead of the usual random split because our data has a time series element to it, and since we’re trying to predict a store’s weekly revenue, we cannot randomly split the dataset without consideration of the time series element.

In the same logic I cross-validated my results not by splitting and testing on 4 equal segments, but on 4 incrementally larger parts and predicting each quarter with all the data that came before. (See Figure 6 for more explanation), which is standard for Time Series Forecasting.

Figure 6. Explanation of cross-validation.



The models I used in this capstone are:

- Linear Regression
- Ridge Regression
- Lasso Regression
- RandomForestRegressor
- XGBoost
- Support Vector Regressor

For Linear, Ridge, and Lasso Regression, I used sklearn’s RFE (Recursive Feature Elimination), which is a feature selector that recursively selects a smaller and smaller subset of features until the optimal subset of features is reached. Unlike the tree-based regressors, Linear Regression and its related algorithms do not have a built in function measure each feature’s contribution to variance in the target column. This process should help to reduce the distraction from unimportant features and better accuracy of prediction.

In the models I tried both the unscaled, unmodified dataset and a standardized version of it because for some models standardizing promises better results.

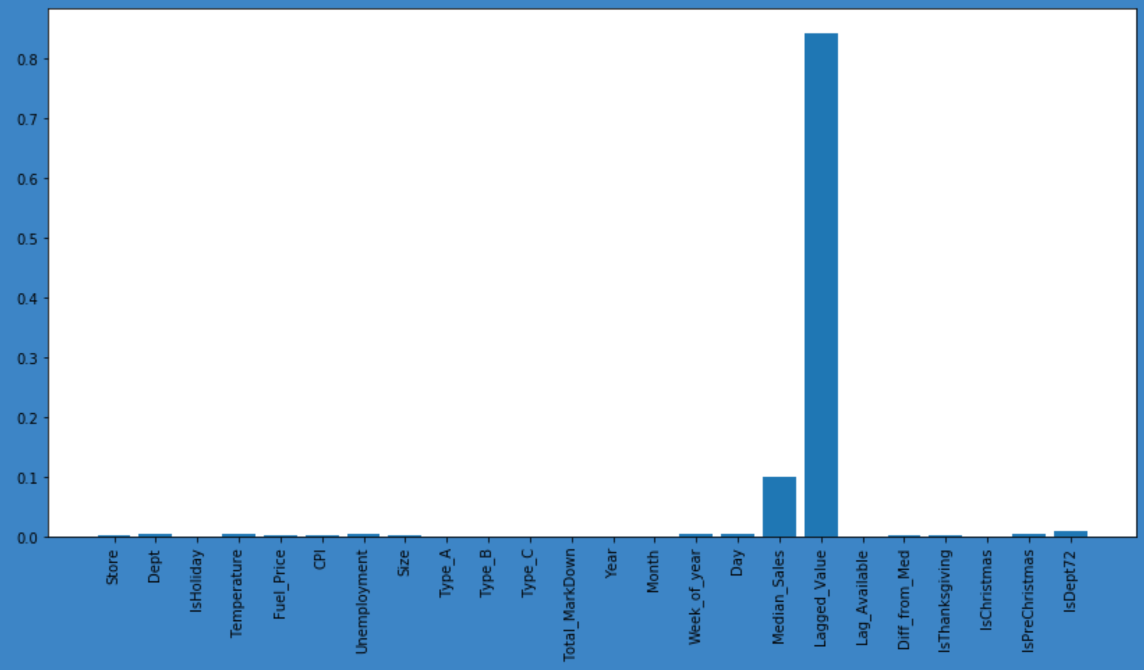
The validation results are shown in Table 1

Model	MAE(Scaled)	MAE(unscaled)
Linear Regression	16030153634809.34	2641.02
Ridge Regression	14979.07	2641.02
Lasso Regression	15045.89	2631.40
RandomForestRegressor	11766.21	2573.75
XGBoost	13107.88	2224.51
SVR	13345.28	13431.92

Generally, Ridge and Lasso Regression performed better than simple Linear Regression, and non-linear regressors like RandomForest and XGBoost performed better than Ridge and Lasso. The best model was XGBoost with the unscaled dataset, which gave us the lowest MAE at 2224.51 dollars.

The MAE achieved using the unscaled datasets were lower than those using scaled datasets. Perhaps this is because by normalizing all the features, we made it so that no one feature is closer in scale to the target column than another. If you look at RandomForestRegressor's feature importances on the unscaled dataset (Figure 7), the feature that contributed the most is Lagged Revenue, mostly likely due to its closeness in scale to the Weekly Revenue.

Figure 7. RandomForestRegressor Feature importance on unscaled dataset



Conclusion

As can be seen, the best model for this dataset is XGBoost using the unscaled dataset. Using this model to predict on our test dataset (as described in Figure 6) yields a final MAE of 1566.00 dollars.

This model can be used to predict a particular store's total revenue, but it can also be helpful for understanding other factors that influence sales. For example, department 72 seems to do better than other departments, and so do the holidays. This information can provide insight for future business, and drive more data-informed decisions.