Sales Forecasting for Retail- Final Report

Sales forecasting is the process of estimating future sales. Accurate sales forecasts enable companies to make informed business decisions and predict short-term and long-term performance. Companies can base their forecasts on past sales data, industry-wide comparisons, and economic trends.

Our analysis for the following reasons:

1. To predict and plan the inventory throughout the year

An accurate sales forecast allows you to properly plan for impending sales. If you have developed an understanding of how your business' sales naturally increase and decrease over the year, you can plan by -Keeping appropriate levels of stock.

2. Helps to manage staff and other resources

In retail, the holiday season typically influences a sharp increase in sales and to prepare for that store ensure they have more stock than normal, and they hire a seasonal workforce to support the demand. Predicting sales beforehand gives a detailed idea for the retail stores as to how much extra workforce is required and how to manage resources efficiently.

3. Predict achievable sales revenue and in turn make wise business investments

Accurate sales forecasting allows you to predict the funds you have coming in against your anticipated costs. These forecasts allow you to understand when you will have the funds available to wisely invest in growth without sacrificing much needed capital for your day-to-day business expenses.

Problem Statement

The task here is to predict department-wide sales for a large-scale retail chain for the following year.

Source of Data:

The data contains historic sales data from the client's database. It was found on Kaggle. We have data ranging from the year 2010 to the year 2013.

This data contains anonymized information for 45 stores in different locations. It contains important data columns such as sales, discounts, size of the stores, CPI, fuel price etc. The data can be found at the below link.
https://www.kaggle.com/manjeetsingh/retaildataset

Data Cleaning and Wrangling

The raw data that we obtained had 3 .csv files – stores.csv, features.csv and sales.csv.

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

Features.csv contains additional data related to the store, department, and regional activity for the given dates like the Store number, Date - the week, Temperature in the region, Fuel price, Markdown (Discounts), CPI, Unemployment and isHoliday (Holiday week or not).

Sales.csv contains the following data – Store num, Dept num, department number, Date- the week, Weekly Sales -Sales per week, isHoliday (holiday week or not).

The sales dataset contained 421,570 rows and 5 columns. Features dataset contained 8,190 rows and 12 columns. The stores dataset contained 45 rows and 3 columns.

The sales and stores dataset contained no missing values. The features dataset had missing values in Markdown1,2,3,4,5, CPI and Unemployment. Markdown id the discounted price of an item. 50% percent of the Markdown values were missing. In spite of 50 percent of the data missing, I decided not to drop those columns because the data description information said that 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. However, 7% of CPI and Unemployment data were also missing.

The NaN values in MarkDown 1,2,3,4 and 5 were filled with 0. The NaN columns in CPI and Unemployment were filled with their respective mean values. Once the data was clean and free of null values, the 3 datasets were merged into a single dataframe. The resultant dataframe contained 421570 rows and 17 columns.

Exploratory Data Analysis

A single dataframe named as retail_data contains all necessary retail sales data from 2010-02-05(2010Feb) to 012-11-01(2012Nov). I performed detailed analysis on this dataframe. The first step was to obtain summary statistics for the data – that involves mostly calculating the mean, median and mode.

```
In [7]: #Let's calculate the average sales throughout the year
        retail_data.Weekly_Sales.mean()

Out[7]: 15981.258123467327

In [8]: #Calculate the median for Weekly sales
        retail_data.Weekly_Sales.median()

Out[8]: 7612.03

In [9]: retail_data.Weekly_Sales.mode()

Out[9]: 0    10.0
        dtype: float64
```

Since the end goal is to predict the sales for the year 2013, I was particularly interested to analyze Weekly Sales for each department, each store and each month. We discovered some very interesting things from our analysis. Here is a snippet of our analysis-

```
[29]: #Find out the top 3 departments with highest average weekly_sales
      AvgWeekly_sales_perDept.nlargest(3)
```

```
[29]: Dept
      92    75204.870531
      95    69824.423080
      38    61090.619568
      Name: Weekly_Sales, dtype: float64
```

From the calculations we can see that Dept 92, 95 and 38 have highest average sales for the period from Feb 2010 to Nov 2012.

```
]: #Find out which stores has the highest average weekly_sales
   AvgWeeklySales_perStore.nlargest(3)
   #https://www.geeksforgeeks.org/get-n-largest-values-from-a-particul
```

```
]: Store
   20    29508.301592
   4     29161.210415
   14    28784.851727
   Name: Weekly_Sales, dtype: float64
```

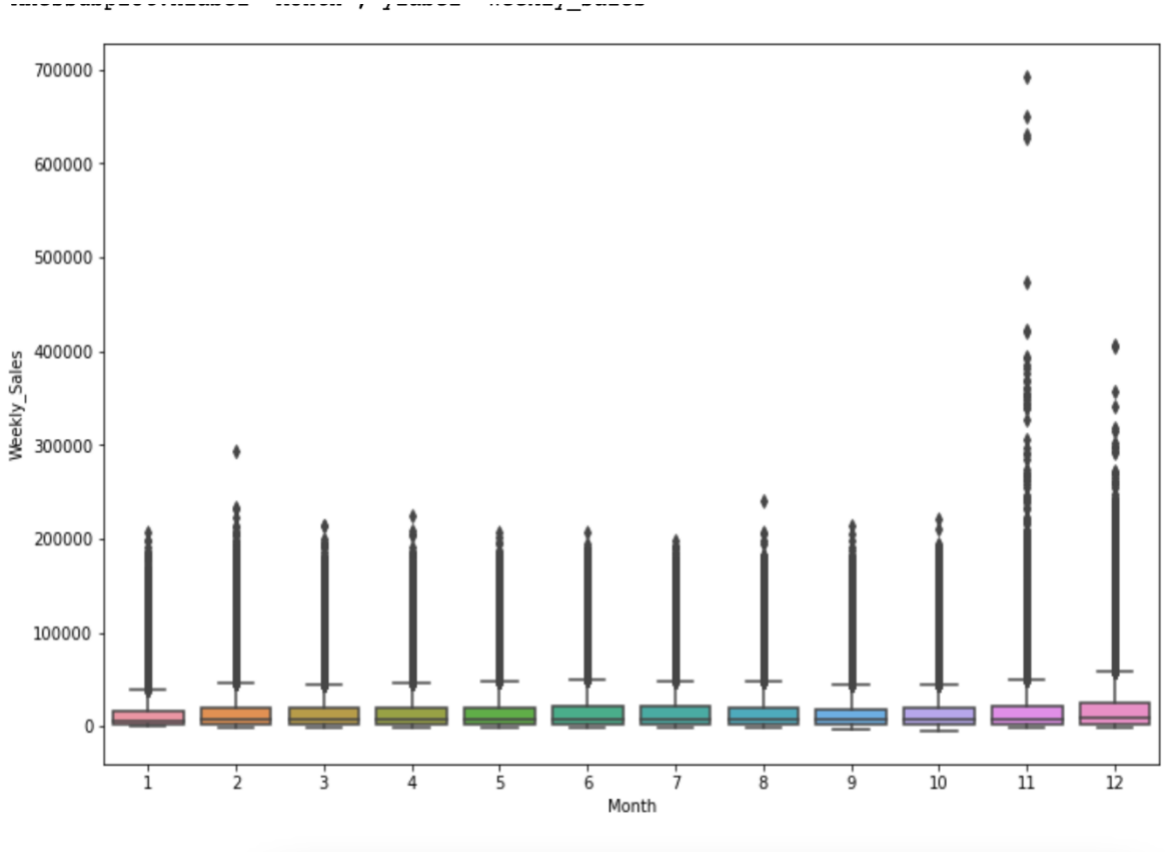
Store Number 20, 4 and 14 are the top 3 in average Sales per week from Feb 2010 to Nov 2012.

```
[43]: #Find out the top 3 Months with highest average_sales
      AverageMonthly_sales.nlargest(3)
```

```
t[43]: Month
       12    19355.702141
       11    17491.031424
       6     16326.137002
       Name: Weekly_Sales, dtype: float64
```

December, November and June months lead in Average sales per week.

We can clearly see some seasonality in the sales. We can see that holiday season and June (being summer vacation for kids) has higher average sales than other months. My analysis also suggests that sales are higher during last 2 weeks of November and December.



The boxplot here clearly shows that November and December months have lot of outliers. The average sales per month remains almost consistent throughout the year and sales shoots up rapidly during holiday season.

Pre-processing and Training

The goal of the preprocessing work is to prepare our data for Modeling. We will create dummy features for categorical values and standardize the numerical values.

I identified categorical and numerical columns in the dataframe and split them into separate lists. Dummy values were created for categorical columns and numerical columns were scaled.

Additionally, I split the dataset into train and test. We used 75% of the data for training and the rest 25% was used as a test set. I then saved the pre-processed version of the dataset as a CSV file for Model selection.

Modeling

The end goal of this project is to predict sales for next 12 months. For predictions, regression models serve best. So, I decided to use Random Forest Regression and Linear Regression. Since the data also has some seasonality to it, I have implemented Time Series Forecasting as well.

Time series data is just any data displaying how a single variable changes over time. It comes as a collection of metrics typically taken at regular intervals. Common examples of time series data include weekly sales data and daily stock prices.

Random Forest Regression is a supervised learning algorithm that uses ensemble learning method for regression. An Ensemble method is a technique that combines the predictions from multiple machine learning algorithms together to make more accurate predictions than any individual model.

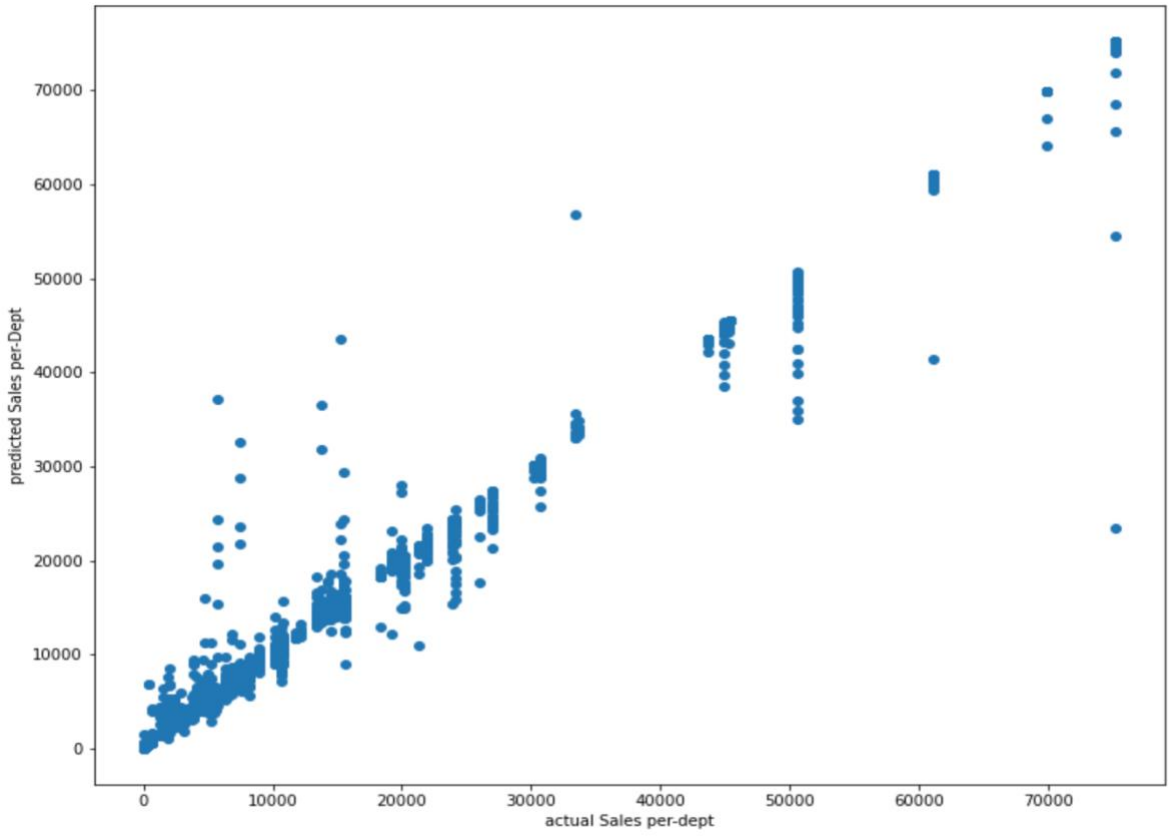
A Random Forest operates by constructing several decision trees during training time and outputting the mean of the classes as the prediction of all the trees.

Linear Regression It's a method to predict a target variable by fitting the best linear relationship between the dependent and independent variable.

Random Forest Regression

To perform Random Forest Regression, I split the data into train and test sets. 75% data was used for training and 25% for testing. After training the data our model was fitted on the test data and I used RMSE (Root Mean Squared error) to evaluate the model. The RMSE obtained for this model is 365.19, which is good considering the scale of the dependent variable.

A graph showing correlation between the actual values and predicted values of Sales per dept.



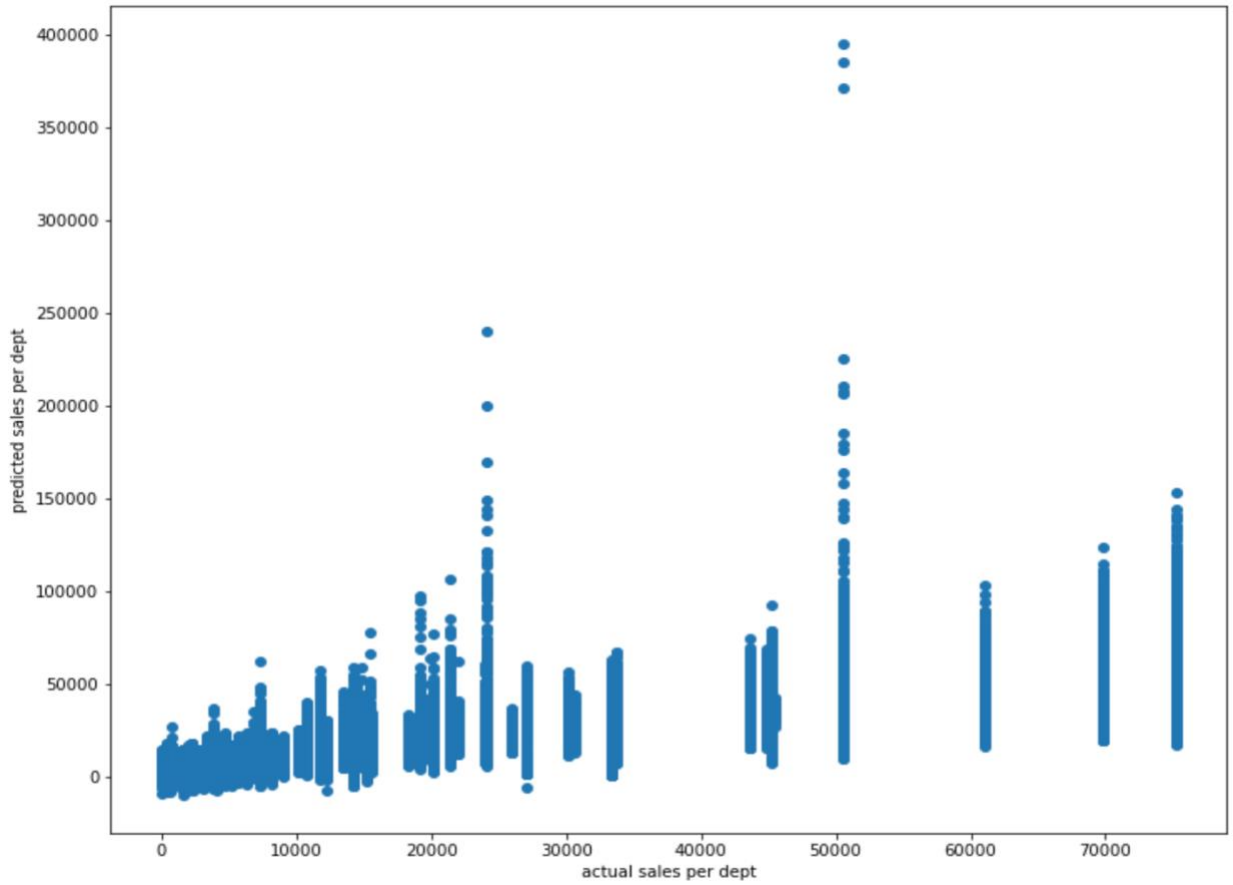
The graph shows a pretty good correlation between actual and predicted values.

Linear Regression

Another Regression model I used is Linear Regression. Just like Random Forest Regression I fitted the model on the data and used R2(Coefficient of determination) and RMSE to evaluate the model.

For our model, we got R2 of 0.6 which indicates that 40% of the variability in the outcome data cannot be explained by the model. But the RMSE of this model is extremely high at 10248.75

Here is a graph which analyses the correlation between actual and predicted values

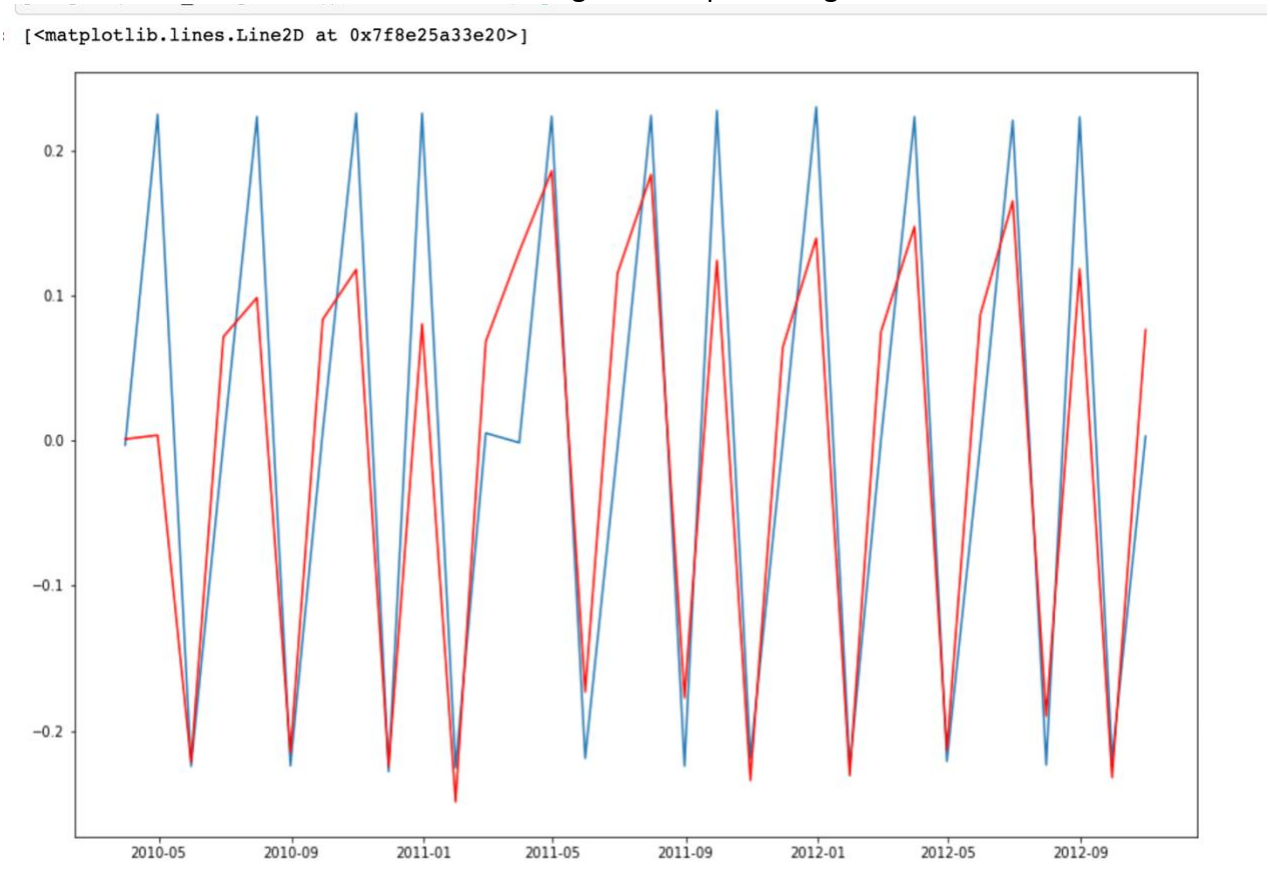


Judging by the graph, we can see that there is no correlation between the actual and predicted values. The RMSE is also very high. Typically, lower values of RMSE indicate a better fit. That means Linear Regression is not a good fit for our data.

Time Series Analysis

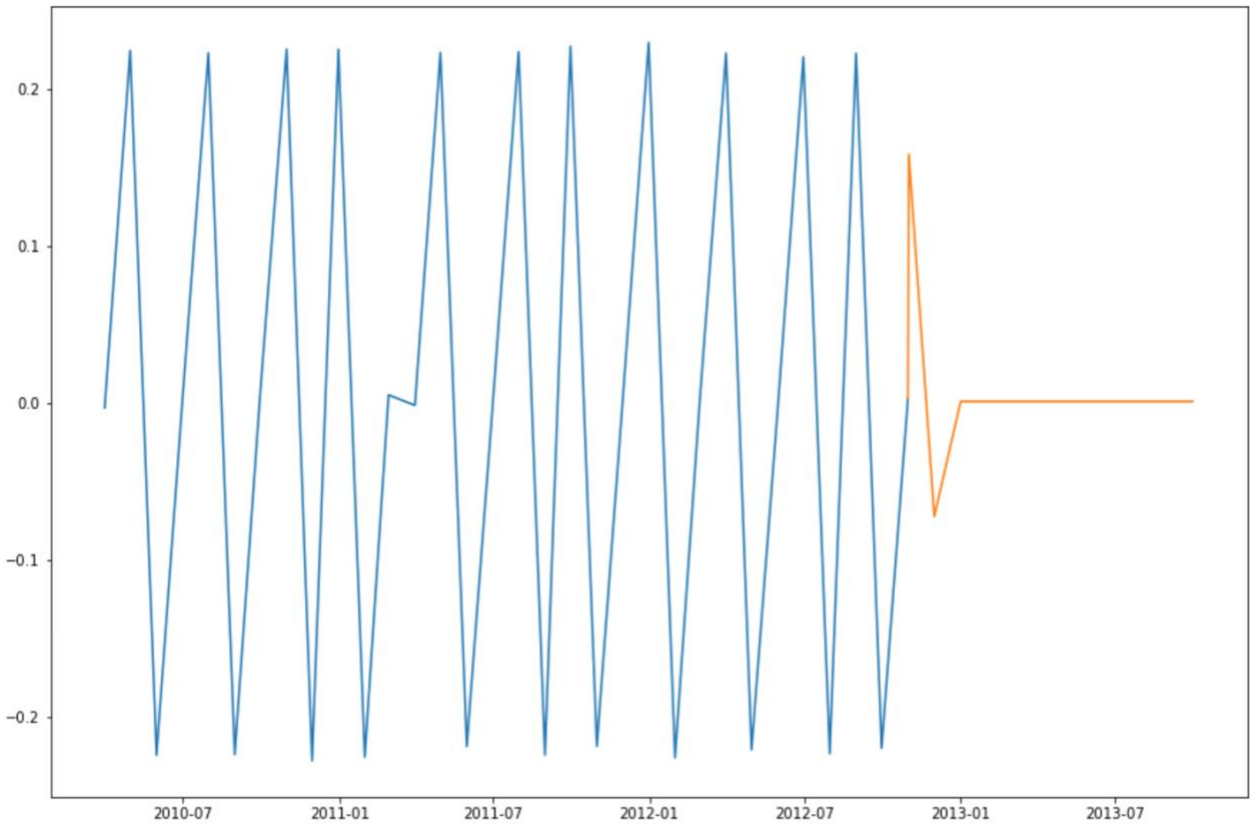
To perform time series analysis, first I had to index the data by 'Date', then had to check if the data is stationary. Since the data was not stationary, I had to make it stationary by taking a natural log of all the values in our dataset and by also differencing the data. Once the data was stationary, I applied ARIMA model and evaluated the model with different p, d, and q values and picked the model which had the least MSE (mean squared error). After picking the best p, d and q values I fitted the model for our data and forecasted sales for the next 12 months.

Here is a visualization of our original data plotted against our model



Our model fits pretty closely to our existing(original) dataset.

Here is a graph that visualizes our Time Series forecasting



The ARIMA model has made quite good predictions. Sales seem to increase during June, November and December of every year, perhaps due to holidays and good weather in June. We can see that Sales remains consistent during other times of the year.

Evaluation and Conclusion

Comparing the models – Random Forest Regression, Linear Regression and Time Series forecasting.

Random Forest Regression has predicted the future sales quite accurately. The model gave an accuracy of 99% for training data. The RMSE of 365 is quite good considering the scale of our target variable.

The correlation graph also shows strong correlation between actual and predicted values. The model is correctly able to predict the target value. All in all, this model is a great fit.

Linear Regression gave a 60% percent accuracy on training data. The R2(coefficient of determination) obtained is 0.62 which is not too good. The RMSE we got is extremely high 10248.75 which is too high even for a large-scale target variable. The correlation graph also showed no correlation between actual and predicted values. So, we can conclude that Linear Regression isn't suitable for our data.

The ARIMA model produced good predictions. The summary results for the model looks satisfactory. It has an AIC of -63.44 and BIC of -54.5 Lower AIC and BIC values are considered better (Smaller values in the number line are considered better) This means that our model is making better predictions. The line graph shows that our model fits quite closely to our original data. The forecasting for the next 1 year also seems to be pretty accurate. Remarkably, our ARIMA model made predictions. However, some sudden circumstances are not handled well by ARIMA.

I think the Time Series Forecasting produced great predictions and was able to identify the seasonality in the data, on the other hand Random Forest Regression gave accurate numbers for our target variable – Average Sales per department. So, I think both of these models are winners. Linear Regression performed poorly on our data.

Future work

I think it would be great to apply some variations of Time Series Forecasting models and experiment how those models perform on our data.