Forecasting sales and traffic in-store using decision tree algorithms (XGBoost and Random Forest)

Melvin Shongwe
melvin.shongwe@gmail.com

October 31, 2022

Abstract

Estimating a store's revenues and losses for a given period is made easier by forecasting total sales and customer numbers in-store. Manual estimation may be challenging because certain days may differ owing to weather conditions and other factors that may have an effect on the retail industry. This project presents Extreme Gradient Boosting (XGBoost) and Random Forest regressor decision tree methods for projecting the number of customers in-store at a certain hour and the sales made during that hour. The two models presented in this study show that the XGBoost is the best model for forecasting sales, while the Random Forest regressor is the best model for forecasting store traffic. These forecasting models can be used interchangeably by the store to forecast future sales and the number of customers in the store for various management reasons, such as estimating monthly profit or forecasting the number of people for holidays to hire enough staff, while keeping in mind that for a better forecast of sales, XGBoost is the preferred model, and for traffic, Random Forest is the preferred model.

Contents

1. Introduction

[Parsons, 2016] discusses sales forecast as the foundation of the financial story that you are creating for your business. Once you have your sales forecast complete, you'll be able to easily create your profit and loss statement, cash flow statement, and balance sheet. Similarly, to the number of people in the store, if we can predict how many people were in the store for a specific hour and how much money was made in sales, we could also be able to compute the rate of people purchasing in the store this way.

On the other hand, counting the number of people in the store could also assist in optimizing costs by the management team deciding how many workers must occupy the shop for specific days/hours. [Wadsworth, 2019] discusses the benefits of counting the number of people in the store, which include assisting your business in planning ahead of time, understanding factors that impact your business, improving energy efficiency, and so on.

When forecasting any time-based quantity, using previous data to estimate the future value is useful. Depending on the nature of the problem, various time series models can be used to forecast future values based on existing data that the model trained on. Figure 1.1 shows various forecasting models and examples based on the nature of the problem. In this project, Extreme gradiente boosting (XGBoost) and Random Forest regressor , machine learning models are used to forecast sales and traffic in-store respectively for a given day and hour while taking into account several factors that may influence sales and traffic forecasting.

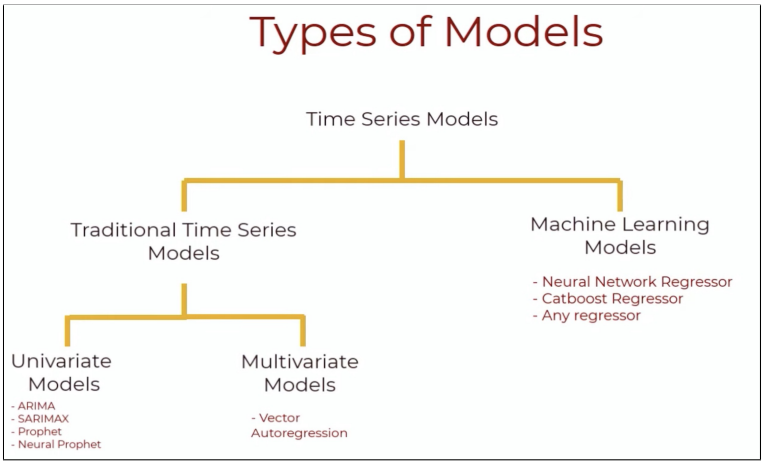


Figure 1.1: Types of time series models

2. Methodology

2.1 Overview

This chapter will focus over the methods used to develop an optimal model(s) for forecasting sales and traffic in-store. The first method of this project is data collection and preparation, which is the foundation for all of the project's subsequent steps when modelling data. The second method is to understand the data and make sense of the data in order to extract some hidden patterns and turn the data into information then knowledge after applying analytics where applicable. Once patterns and information have been extracted from the data, the data is modelled, which is then followed by model selection, in which optimal models are selected from the models built. Normally, after the models are selected and validated, metrics are used in this project to evaluate each model to determine its predictive ability, and the best models are then chosen to forecast sales and traffic in store.

2.2 Data Collection, Preparation and Analysis

Since the datasets used in this study were provided, the collection phase of this project is not necessary; however, the preparation phase is required because missing values and other flaws in the data, such as detection of outliers using box-plot/scatter plot and other flaws the data may show, must be controlled before performing analysis on the data. This phase also includes feature engineering, in which the addition of features that influence the retail industry are considered in this project in terms of having an impact on sales made in store and traffic experienced, which includes events, seasons, holidays, and other events. The objective of this study is to develop a model(s) that will forecast sales and store traffic for the following month hourly. As a result, the hours the store is open are considered, and working with hours is advantageous to working in 15-minute intervals because the data is in 15-minute intervals initially. Analysis is performed throughout the data preparation process to assist and guide which step is reasonable to execute as plots and other analytics are useful for data preparation as stated that other visualizations are used for outlier detection.

Time series based models require the data to be stationary prior to modelling although some models do not require the data to be stationary prior to modelling. [Radecic, 2020] puts forward that majority of models for time series require data to be stationary though some models like LSTM does not necessarily require data to be stationary but when made stationary it improves the model. Tests for stationarity are performed using the two popular tests for stationarity in time series which are Augmented Dickey–Fuller (ADF) and Kwiatkowski–Phillips–Schmidt–Shin (KPSS). The null hypothesis is rejected or fails to be rejected by the tests. [Perktold, 2019] lists the possible outcomes of the tests as well as the methods that should be used to make the time series stationary in each case.

- Case 1: Both tests conclude that the series is not stationary - The series is not stationary
- Case 2: Both tests conclude that the series is stationary - The series is stationary.
- Case 3: KPSS indicates stationarity and ADF indicates non-stationarity - The series is trend stationary. Trend needs to be removed to make series strict stationary. The detrended series is checked for stationarity.
- Case 4: KPSS indicates non-stationarity and ADF indicates stationarity - The series is difference stationary. Differencing is to be used to make series stationary. The differenced series is checked for stationarity.

Following the completion of the stationarity tests, making the series stationary is performed in cases where the tests indicate that the series is non-stationary from at least one of the tests. Differencing, time-series decomposition, log transform, box-cox transform, and other transformations that could work to make a series stationary are common techniques.

2.3 Modelling

This section entails creating models to be evaluated using metrics for their predictive ability and comparing their metrics to obtain the best one from the models developed. The models proposed in this project are Linear regression, Extreme Gradient Boosting (XGBoost), Random Forest Regressor, Facebook Prophet, and Seasonal Auto-Regressive Integrated Moving Average with exogenous factors (SARIMAX). The method for validating these models consists of using sets of data (Data Splitting), a training set and a testing set, where the training set is the data on which the model will be trained on and the testing set is the set on which the trained model will be tested and metrics applied. As a result, an optimal model(s) are obtained. Figure 2.2 depicts some metrics used in this project to compare models, which are classified as scale-dependent metrics. [Cote, 2022] and [Rink, 2021] also discussed the benefits of scale-dependent metrics when evaluating time series models.

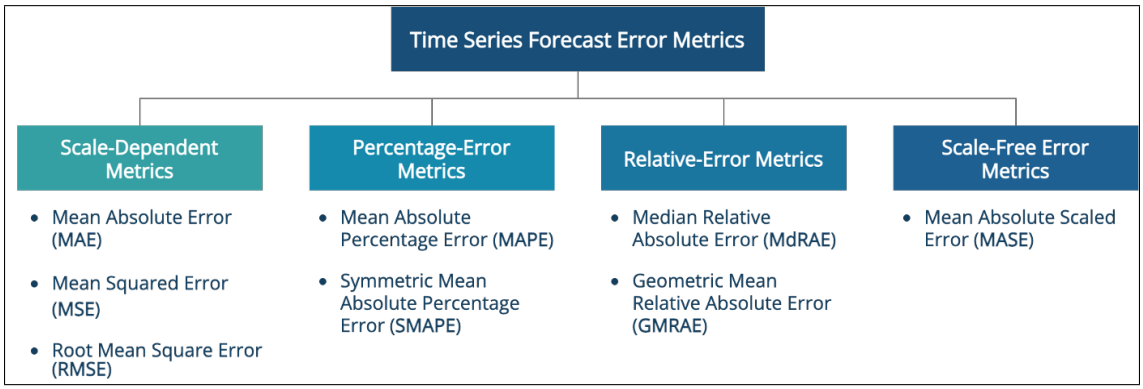


Figure 2.2: Time Series Forecast Error Metrics [Rink, 2021]

2.4 Forecasting

After the models have been evaluated using the metrics, the best-performing model(s) among the models is used to forecast the store's sales and traffic. This phase entails generating times for the model to forecast the store's sales and traffic for the upcoming month. After forecasting sales and traffic with the best model(s), graphical analysis is used to see if the predicted sales and traffic peaked during the seasons when the original data peaked. This allows us to determine whether the optimal model(s) are optimal in action not only optimal in metrics.

3. Results and Discussion

3.1 Overview

This chapter presents and discusses the findings obtained by following the methodology of this project.

3.2 Data Collection, Preparation and Analysis

The first computation considered for this phase was to rename the sales and traffic columns to make the visualizations easier to understand. The dataset interval is 15 minutes, and the goal is to forecast monthly sales and traffic per hour. Converting the timestamps to an hour interval before cleaning the datasets to carry out the analysis, as the goal of this project is to forecast the store's sales and traffic hourly for the next month. Initially, the datasets did not contain any null values because they only recorded sales and traffic for 15-minute intervals. However, after converting the intervals to an hour, the datasets showed null values in the store's traffic and sales.

The datasets are time series, considering the business hours of the United States, and adding a day of the week column in the data will help to identify the times when the store was closed and when the store was not making sales, because removing all the null values implies that the store does not experience hours with no sales, which is likely false. It may appear that removing the hours when the business is closed is removing information, but domain knowledge also assists in data exploration. [official travel site of the USA,] provides the following information about business hours in the United States:

- Shopping Malls
  - Monday-Saturday: 10 a.m.-9 p.m
  - Sunday: 11 a.m.-6 p.m
- Supermarket
  - Daily: 8 a.m.-8 p.m.
  - Some open 24 hours

	Date	Sales_cents
0	2013-11-07 13:00:00	3457.0
1	2013-11-07 14:00:00	2250.0
2	2013-11-07 15:00:00	0.0
3	2013-11-07 16:00:00	729.0
4	2013-11-07 17:00:00	0.0
...	...	...
39386	2018-05-06 15:00:00	3645.0
39387	2018-05-06 16:00:00	3372.0
39388	2018-05-06 17:00:00	3077.0
39389	2018-05-06 18:00:00	3190.0
39390	2018-05-06 19:00:00	759.0

23215 rows × 2 columns

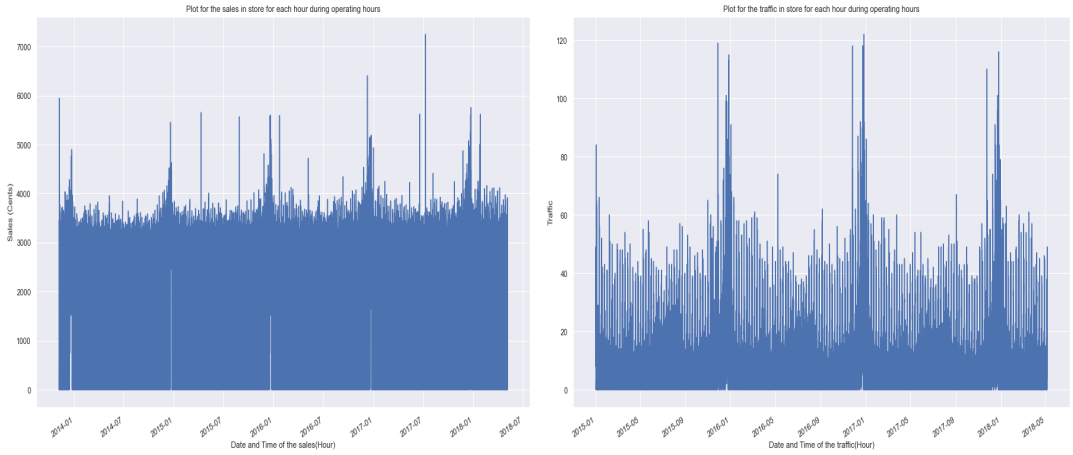
(a)

	Date	No_people
0	2015-01-01 10:00:00	8.0
1	2015-01-01 11:00:00	14.0
2	2015-01-01 12:00:00	28.0
3	2015-01-01 13:00:00	33.0
4	2015-01-01 14:00:00	47.0
...	...	...
29309	2018-05-06 15:00:00	41.0
29310	2018-05-06 16:00:00	33.0
29311	2018-05-06 17:00:00	27.0
29312	2018-05-06 18:00:00	14.0
29313	2018-05-06 19:00:00	4.0

17278 rows × 2 columns

(b)

Figure 3.3: Overview dataframe of the store's sales (a) and traffic (b) after renaming columns, converting intervals and handling missing values.



(a) (b)

Figure 3.4: Hourly sales (a) and traffic (b) of store in operating hours plots

After the datasets were cleaned and there were no missing values, feature engineering was performed by considering factors that influence the retail industry. A new binary feature was developed that determines if a certain date is a holiday or an event or not by considering all federal national days in the United States, where the column represents national holidays and events such as Black Friday by 1 and 0 otherwise. The plots in figure 3.4 show that every year-end there is a spike in sales and traffic in-store, indicating that adding a meteorological seasons element to the data would be beneficial. The only evident outlier using domain knowledge and the plots in figure 3.4 are the one on 2017-07-09 at 19:00:00, which is not a holiday because the only period when sales are high is at the end of the year and the beginning of each year. Furthermore, considering the traffic on this day, it demonstrates that traffic was low on this day and hour, implying that this is an outlier caused by an error or that more sales were made at random than typical. Finally, the season on this date was not summer, when sales typically surge, making it an exception.

As indicated in the methodology, ADF and KPSS test the stationarity of the datasets using the hypotheses listed below.

- **ADF**
  Null Hypothesis (H_0): The series has a unit root (Not Stationary)
  Alternate Hypothesis (H_a): The series has no unit root (Stationary)
- **KPSS**
  Null Hypothesis (H_0): The process is trend stationary.
  Alternate Hypothesis (H_a): The series has a unit root (series is not stationary)

ADF Sales results	
Test Statistic	-1.494710e+01
p-value	1.296729e-27
#Lags Used	4.600000e+01
Number of Observations Used	2.309800e+04
Critical Value (1%)	-3.430633e+00
Critical Value (5%)	-2.861665e+00
Critical Value (10%)	-2.566837e+00

(a)

KPSS Sales results	
Test Statistic	0.428168
p-value	0.065014
Lags Used	1704.000000
Critical Value (10%)	0.347000
Critical Value (5%)	0.463000
Critical Value (2.5%)	0.574000
Critical Value (1%)	0.739000

(b)

Figure 3.5: Stationarity tests (a) ADF and (b) KPSS for Sales

ADF Traffic results	
Test Statistic	-1.218724e+01
p-value	1.310226e-22
#Lags Used	4.400000e+01
Number of Observations Used	1.719300e+04
Critical Value (1%)	-3.430730e+00
Critical Value (5%)	-2.861708e+00
Critical Value (10%)	-2.566859e+00

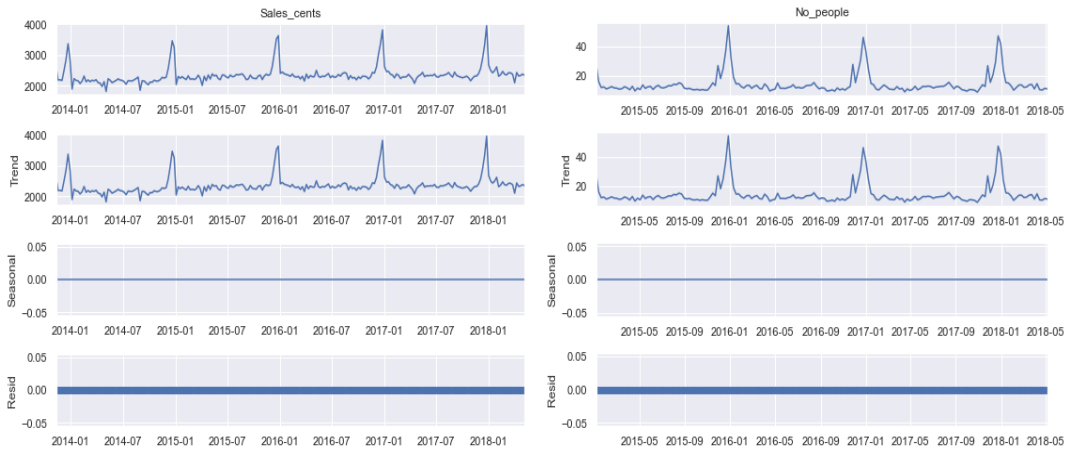
(a)

KPSS Traffic results	
Test Statistic	0.31247
p-value	0.10000
Lags Used	44.00000
Critical Value (10%)	0.34700
Critical Value (5%)	0.46300
Critical Value (2.5%)	0.57400
Critical Value (1%)	0.73900

(b)

Figure 3.6: Stationarity tests (a) ADF and (b) KPSS for Traffic

The stationarity tests, both the ADF Tests, $p-value < 0.05$ which implies that the null hypothesis is rejected. Thus, the series is stationary according to the ADF. Similarly, to the KPSS Tests, they suggest that we fail to reject the null hypothesis which implies that their series is stationary. The results show that both series are stationary, despite the fact that their plots exhibit some yearly seasonality. The ADF and Kwiatkowski, Phillips, Schmidt, and Shin (KPSS) tests are both designed to detect nonstationary in the form of a process unit root. However, they are not intended to identify other types of stationarity. As a result, it is unsurprising that they fail to detect seasonal non stationarity.



(a) (b)

Figure 3.7: Time series decomposition for (a) sales and (b) traffic in store

Time series decomposition is used to examine the series' seasonality and trend. Since seasonality does not change over time, the model used is additive for decomposition, resulting in a cycle. Seasonality does not appear in the findings 3.7 of the time series decomposition of traffic and sales and removing it from the additive model has no effect. After applying different transformations (Log and Square root) to the series, it is evident that seasonality cannot be removed from the data since it is continuous and will contribute to the model(s) Seasonal Cycle predictions. As a result, data peaks may not always indicate that a series is non-stationary, but these will help when forecasting as the model will be able to forecast peak sales and traffic. The plots for time series decomposition of the sales and traffic datasets are shown in the figure above (Figure 3.7).

3.3 Modelling

Linear regression, Extreme Gradient Boosting (XGBoost), Random Forest Regressor, Facebook Prophet, and Seasonal Auto-Regressive Integrated Moving Average with exogenous factors (SARIMAX) are the models presented in this research, as described in the previous chapter. This section will introduce some features that were optional to include in the project's data preparation phase. These new features are derived from the date of each data point and include the hour, day of the week, quarter, month, year, day of year, day of month, and week of the year. These features were not included in feature in feature engineering since they will be compared to a series with only two features: events/holidays and seasons. Thus, for each machine learning model proposed in this project, two models (many features and two features) are presented. As stated in the previous chapter, the datasets were partitioned for validation in order to compare the metrics of the models. The figure below shows an overview of the dataframes to be compared in each proposed machine learning model.

	Date	Sales_cents	Events_holidays	Season
0	2013-11-07 13:00:00	3457.0	0	1
1	2013-11-07 14:00:00	2250.0	0	1
2	2013-11-07 15:00:00	0.0	0	1
3	2013-11-07 16:00:00	729.0	0	1
4	2013-11-07 17:00:00	0.0	0	1
...	...	...	...	...
39386	2018-05-06 15:00:00	3645.0	0	1
39387	2018-05-06 16:00:00	3372.0	0	1
39388	2018-05-06 17:00:00	3077.0	0	1
39389	2018-05-06 18:00:00	3190.0	0	1
39390	2018-05-06 19:00:00	759.0	0	1

(a)

Date	Sales_cents	Events_holidays	Season	hour	dayofweek	quarter	month	year	dayofyear	dayofmonth	weekofyear
2013-11-07 13:00:00	3457.0	0	1	13	3	4	11	2013	311	7	45
2013-11-07 14:00:00	2250.0	0	1	14	3	4	11	2013	311	7	45
2013-11-07 15:00:00	0.0	0	1	15	3	4	11	2013	311	7	45
2013-11-07 16:00:00	729.0	0	1	16	3	4	11	2013	311	7	45
2013-11-07 17:00:00	0.0	0	1	17	3	4	11	2013	311	7	45
...	...	...	...	...	...	...	...	...	...	...	...
2018-05-06 15:00:00	3645.0	0	1	15	6	2	5	2018	126	6	18
2018-05-06 16:00:00	3372.0	0	1	16	6	2	5	2018	126	6	18
2018-05-06 17:00:00	3077.0	0	1	17	6	2	5	2018	126	6	18
2018-05-06 18:00:00	3190.0	0	1	18	6	2	5	2018	126	6	18
2018-05-06 19:00:00	759.0	0	1	19	6	2	5	2018	126	6	18

(b)

Figure 3.8: An overview of the sales datasets will used for training the different models (a) few features and (b) multiple features

The models were developed, and their metrics are shown in the tables below. The results show that the XGBoost and Random Forest are optimal models because their metrics indicate that they are. The results show that adding more features to the series improves the model. However, for the Facebook prophet, there isn't much of a difference between the models with few and many features because FB Prophet doesn't look for a casual relationship between the past and the future; instead, it simply finds the best curve that fits the data well while taking seasonality into account. Thus, adding more features is not necessary for FB Prophet as it learns seasonality through training. In this situation, FB Prophet does not appear to be as robust as XGBoost and Random Forest.

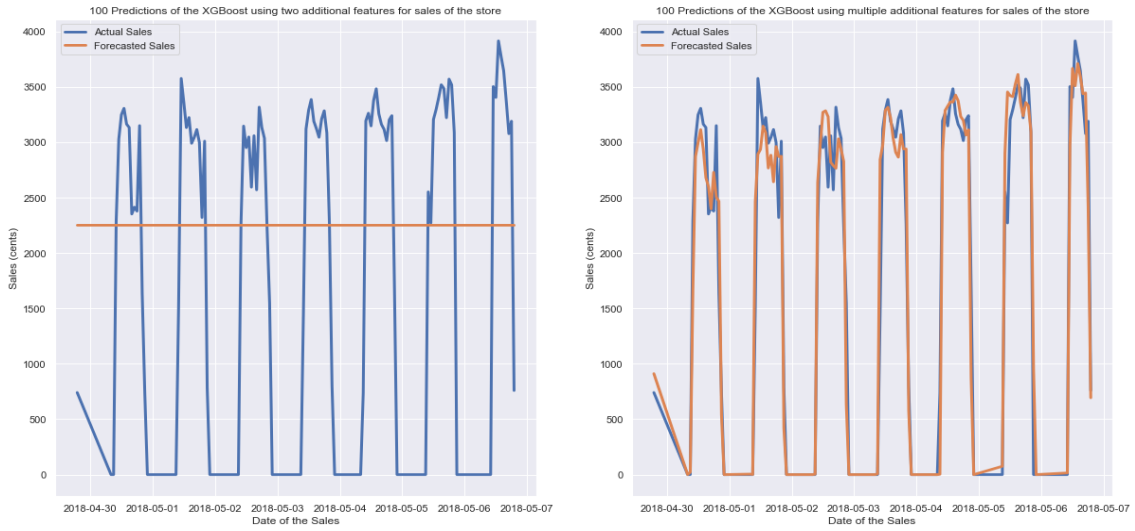
Sales Model	MSE	RMSE	MAE
Linear regression (Few Features)	1804533.0160	1343.3290	1102.7065
Linear Regression (Multiple Features)	1778306.98227	1333.5317	1035.3698
XGBoost (Few features)	1812860.7488	1346.425	1160.70855
XGBoost (Multiple Features)*	261834.61186	511.69777	333.68248
Random Forest (Few features)	1812833.1468	1346.414	1160.6282
Random Forest (Multiple features)*	270478.9426	520.07590	324.17303
Facebook Prophet (Few Features)	376038.0646	613.2194	416.70186
Facebook Prophet (Multiple Features)	370372.9694	608.5827	409.8470
SARIMAX (Few Features)	1820149.7854	1349.129	1146.60931
SARIMAX (Multiple Features)	2784417038.6584	52767.57563	45869.3951

Table 3.1: Metrics of the models proposed for the sales of the store

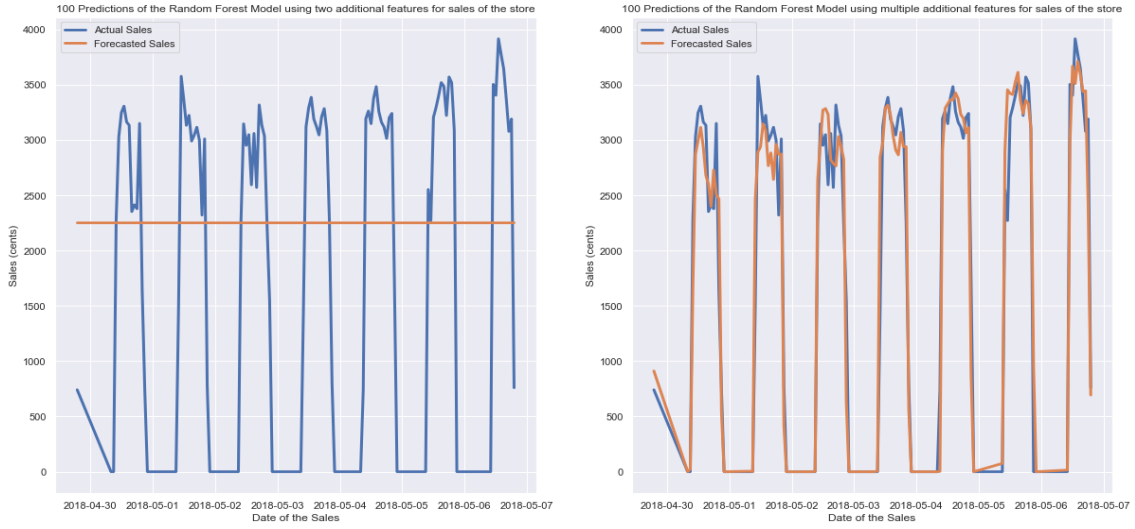
Traffic Model	MSE	RMSE	MAE
Linear regression (Few Features)	211.6781	14.54916	10.8720
Linear Regression (Multiple Features)	174.40386	13.2062	9.7976
XGBoost (Few features)	207.8491	14.41697	10.4168
XGBoost (Multiple Features)*	39.50328	6.2851	3.8699
Random Forest (Few features)	207.8445	14.41681	10.4180
Random Forest (Multiple features)*	29.6067	5.44120	3.29376
Facebook Prophet (Few Features)	78.3081	8.84918	6.25139
Facebook Prophet (Multiple Features)*	76.3179	8.7360	6.1940
SARIMAX (Few Features)	284.4899	16.8668	14.3657
SARIMAX (Multiple Features)	56968.2493	238.6802	208.7949

Table 3.2: Metrics of the models proposed for the traffic in the store

According to the metrics, the XGBoost model with several features is the best model for forecasting in-store sales. The Mean Square Error and Root Mean Square Error indicate how well the model fits the data, and the lower the value of RSME and MSE, the better, as it suggests that the model makes few errors in fitting the data. According to the results, the XGBoost appears to be the best model for forecasting sales because it has the lowest MSE and RMSE. Furthermore, the MAE represents the mean absolute error, which regards the model's error equally, and in this case, it is low for the XGBoost, which enhances the model's accuracy since MAE assesses accuracy. Similarly, to the Traffic of the store, the Random forest model with multiple functions seems to be the optimal model for forecasting the traffic in the store considering the metrics of the model which indicate that it is optimal. Overall, the two models will be used to forecast the store's sales and traffic since they appear to be optimum. Although both are optimal, the XGBoost regressor will be favoured for sales forecasting over the Random Forest regressor. In contrast, the Random Forest regressor results will be chosen for projecting sales above the XGBoost results. Figures 3.9 and 3.10 show plots for the two best models proposed in this project, and the results clearly show that models with multiple features outperform models with only two features . Unsurprisingly, the forecasted values of the two models (XGBoost and Random Forest) are close since they use the same model representation and inference but train differently.

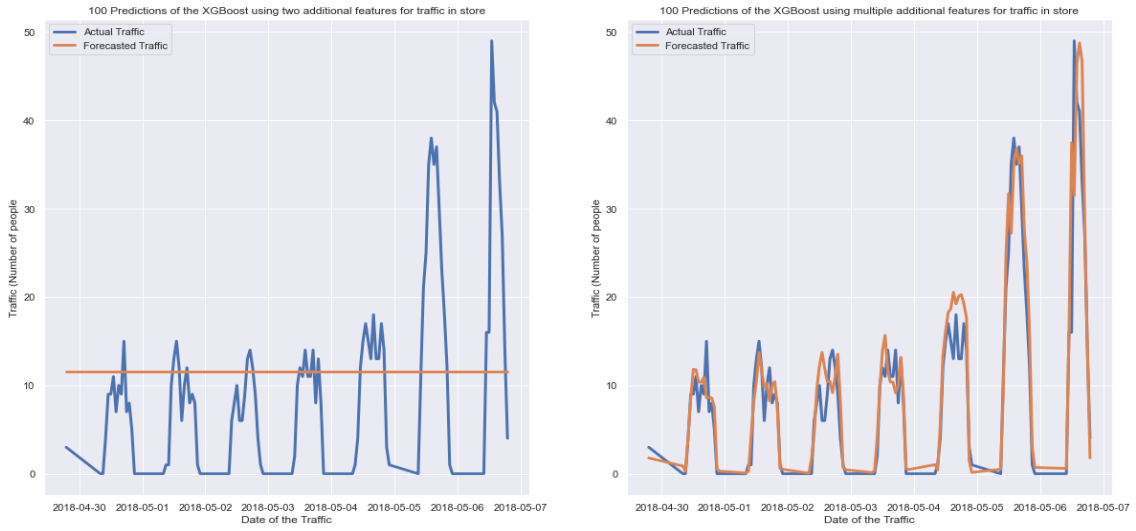


(a)

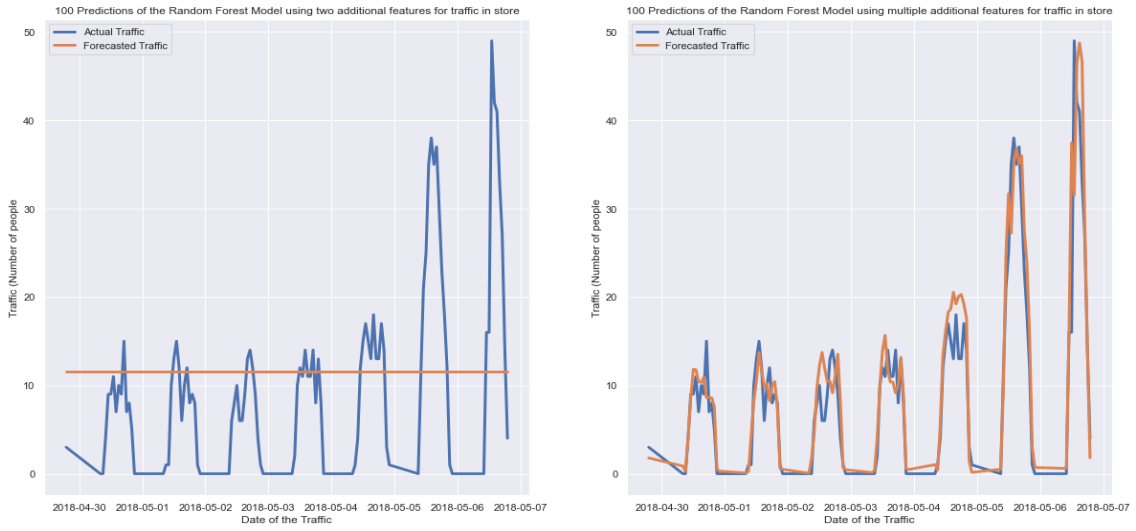


(b)

Figure 3.9: Plots for store's sales forecasted using (a) Random Forest and (b) XGBoost



(a)



(b)

Figure 3.10: Plots for store's traffic forecatsed using (a) Random Forest and (b) XG-Boost

3.4 Forecasting

The primary purpose of this study is to forecast the store's hourly sales and traffic for the month following the last day. The prediction requires an input that will be the dates from the final data points of the data provided, as well as dates generated from that date, to have extra data points for estimating store sales and traffic. Since the latest data point was on May 6, 2018, at 19:00:00, forecasting will begin with that data point and continue until a month period is met. Below is a data frame with the generated datapoints for forecasting.

Date	Events_holidays	Season	hour	dayofweek	quarter	month	year	dayofyear	dayofmonth	weekofyear
2018-05-07 08:00:00	0	1	8	0	2	5	2018	127	7	19
2018-05-07 09:00:00	0	1	9	0	2	5	2018	127	7	19
2018-05-07 10:00:00	0	1	10	0	2	5	2018	127	7	19
2018-05-07 11:00:00	0	1	11	0	2	5	2018	127	7	19
2018-05-07 12:00:00	0	1	12	0	2	5	2018	127	7	19
...	...	...	...	...	...	...	...	...	...	...
2018-06-07 22:00:00	0	1	22	3	2	6	2018	158	7	23
2018-06-08 08:00:00	0	1	8	4	2	6	2018	159	8	23
2018-06-08 09:00:00	0	1	9	4	2	6	2018	159	8	23
2018-06-08 10:00:00	0	1	10	4	2	6	2018	159	8	23
2018-06-08 11:00:00	0	1	11	4	2	6	2018	159	8	23

476 rows × 10 columns

Figure 3.11: An overview of the generated data frame for forecasting

The results show that using XGBoost for traffic forecasting gives negative values which makes the argument stronger of using Random Forest for forecasting traffic and using XGBoost for sales. The results in figure 3.12 demonstrate that the two best models predict store sales and traffic with very close values to one another.

Date	Predicted_sales_RF	Predicted_sales_XGB	Predicted_traffic_RF	Predicted_traffic_XGB
2018-05-07 08:00:00	17.066	0.693005	0.0	-0.0
2018-05-07 09:00:00	124.805	33.487144	0.0	-0.0
2018-05-07 10:00:00	2319.555	2300.225830	5.0	4.0
2018-05-07 11:00:00	3029.169	3051.654297	8.0	10.0
2018-05-07 12:00:00	3244.245	3103.032959	13.0	13.0
...	...	...	...	...
2018-06-07 22:00:00	0.000	55.358997	0.0	1.0
2018-06-08 08:00:00	5.936	117.680641	1.0	1.0
2018-06-08 09:00:00	117.362	150.474701	1.0	1.0
2018-06-08 10:00:00	2553.053	2442.248047	7.0	5.0
2018-06-08 11:00:00	3272.844	3237.556152	13.0	12.0

Figure 3.12: Forecasted results using the optimal models

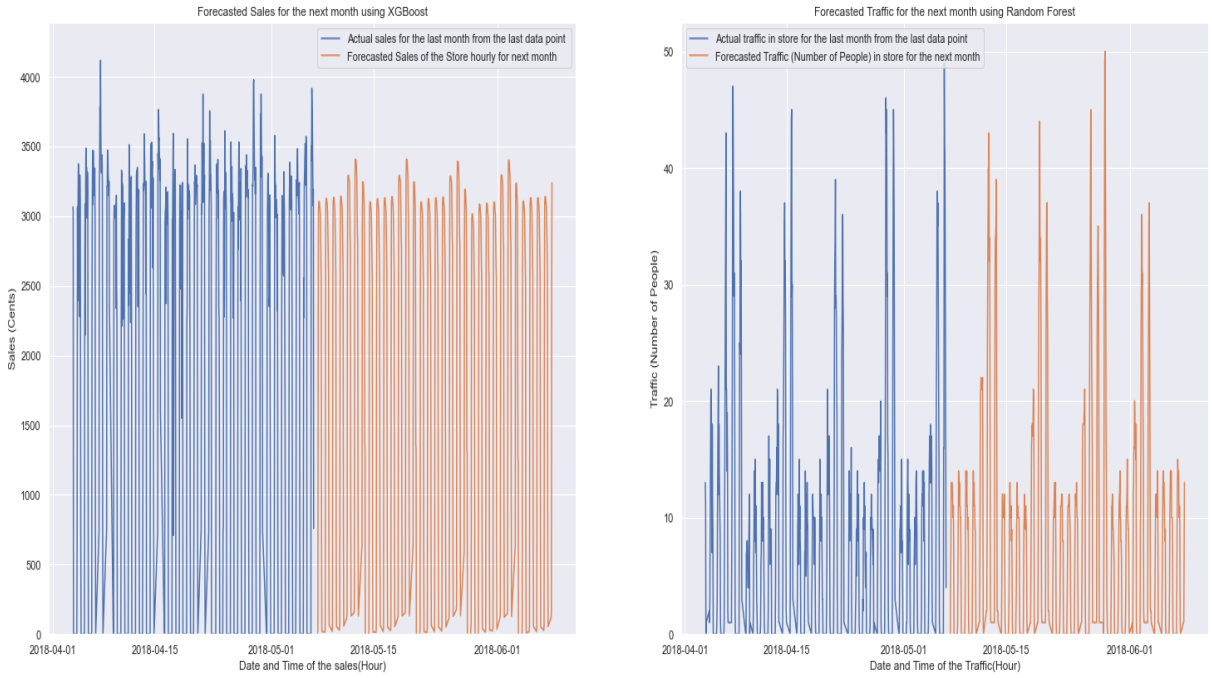


Figure 3.13: Forecasted results using the optimal models for both sales and traffic of the store

4. Conclusion

The models were able to forecast the store's hourly sales and traffic for the next month. I may conclude that employing machine algorithms and statistics models in the retail industry would assist in decision-making, as said previously, knowing an estimate of sales and the number of consumers would help in distributing items and in other cases. Furthermore, the company can target areas where sales were lowest by running promotions on low peak days and weeks, which could help with profits. The models can also be used to forecast the total daily, weekly, or monthly based on the target and period. As can be seen in the appendix, the forecasted sales and traffic are closer since the two models use the same algorithm but differ in how they are trained. A company could use any of these models for forecasting while keeping in mind that there is a better performance when forecasting sales when using the XGBoost and Random Forest for traffic.

Bibliography

[Cote, 2022] Cote, D. (2022). Rdr score metric for evaluating time series forecasting models.

[official travel site of the USA,] official travel site of the USA, T. Time & business hours.

[Parsons, 2016] Parsons, N. (2016). How to do a sales forecast for your business the right way-2021 guide.

[Perktold, 2019] Perktold, J. (2019). Stationarity and detrending (adf/kpss)¶.

[Radecic, 2020] Radecic, D. (2020). What is stationarity in time series and why should you care.

[Rink, 2021] Rink, K. (2021). Time series forecast error metrics you should know.

[Wadsworth, 2019] Wadsworth, C. (2019). Measuring retail store traffic: How people counting works.

Appendix

Date	Predicted_sales_RF	Predicted_sales_XGB	Predicted_traffic_RF	Predicted_traffic_XGB
2018-05-07 08:00:00	17.066	0.693005	0	0
2018-05-07 09:00:00	124.805	33.487144	0	0
2018-05-07 10:00:00	2319.555	2300.225830	5	4
2018-05-07 11:00:00	3029.169	3051.654297	8	10
2018-05-07 12:00:00	3244.245	3103.032959	13	13
2018-05-07 13:00:00	3164.316	3103.032959	13	14
2018-05-07 14:00:00	3157.094	3100.365723	13	14
2018-05-07 15:00:00	3060.547	3065.971924	13	14
2018-05-07 16:00:00	2916.576	3043.463379	12	13
2018-05-07 17:00:00	2807.641	3001.175537	11	11
2018-05-07 18:00:00	2958.498	2832.532715	10	10
2018-05-07 19:00:00	2742.005	2643.376953	11	9
2018-05-07 20:00:00	2190.919	2520.368896	7	6
2018-05-07 21:00:00	588.232	686.031677	0	0
2018-05-07 22:00:00	0.000	19.160870	0	0
2018-05-08 08:00:00	0.000	15.207832	0	0
2018-05-08 09:00:00	27.498	48.001987	1	0
2018-05-08 10:00:00	2423.576	2322.823242	5	4
2018-05-08 11:00:00	3217.019	3074.251709	8	10
2018-05-08 12:00:00	3248.490	3125.630371	14	13
2018-05-08 13:00:00	3123.187	3125.630371	13	14
2018-05-08 14:00:00	3143.331	3122.963135	11	14
2018-05-08 15:00:00	3065.235	3088.569336	13	14
2018-05-08 16:00:00	2772.338	3066.060791	11	13
2018-05-08 17:00:00	2815.976	3023.772949	11	11
2018-05-08 18:00:00	2944.394	2855.130127	10	10
2018-05-08 19:00:00	2847.018	2665.974365	10	9
2018-05-08 20:00:00	2673.802	2567.589844	7	6
2018-05-08 21:00:00	306.889	731.867676	0	0
2018-05-08 22:00:00	0.000	64.996964	0	0

Figure F.14: First 30 Forecasted results using the optimal models for both sales and traffic of the store

Date	Predicted_sales_RF	Predicted_sales_XGB	Predicted_traffic_RF	Predicted_traffic_XGB
2018-06-06 12:00:00	3071.134	3129.987549	12	14
2018-06-06 13:00:00	3157.014	3129.987549	14	15
2018-06-06 14:00:00	3106.434	3127.320312	13	15
2018-06-06 15:00:00	3033.895	3092.926514	14	15
2018-06-06 16:00:00	3027.330	3070.417969	14	14
2018-06-06 17:00:00	3070.046	3028.130127	10	12
2018-06-06 18:00:00	2625.822	2859.487305	10	11
2018-06-06 19:00:00	2638.657	2670.331543	10	10
2018-06-06 20:00:00	2989.216	2571.947021	9	7
2018-06-06 21:00:00	432.314	736.224609	1	1
2018-06-06 22:00:00	0.000	51.861706	0	1
2018-06-07 08:00:00	37.100	26.746828	0	1
2018-06-07 09:00:00	133.237	59.540916	0	1
2018-06-07 10:00:00	2543.064	2335.034180	4	4
2018-06-07 11:00:00	3127.462	3087.195557	10	11
2018-06-07 12:00:00	3174.792	3138.574219	13	14
2018-06-07 13:00:00	3177.761	3138.574219	15	15
2018-06-07 14:00:00	3126.096	3135.906982	14	15
2018-06-07 15:00:00	2969.080	3101.513184	14	15
2018-06-07 16:00:00	3108.165	3079.004639	14	14
2018-06-07 17:00:00	3062.838	3036.716797	11	12
2018-06-07 18:00:00	2699.716	2868.073975	10	11
2018-06-07 19:00:00	2576.104	2678.918213	11	10
2018-06-07 20:00:00	2953.224	2580.533691	9	7
2018-06-07 21:00:00	655.967	744.811096	1	1
2018-06-07 22:00:00	0.000	55.358997	0	1
2018-06-08 08:00:00	5.936	117.680641	1	1
2018-06-08 09:00:00	117.362	150.474701	1	1
2018-06-08 10:00:00	2553.053	2442.248047	7	5
2018-06-08 11:00:00	3272.844	3237.556152	13	12

Figure F.15: Last 30 Forecasted results using the optimal models for both sales and traffic of the store