**MALIHE HASANIAREFI**

Student Number: 501203989

Supervisor: Prof. Ceni Babaoglu

30 April 2023

**Abstract**:

The Rossmann Store Sales problem is a supervised machine learning regression problem that aims to predict daily sales for Rossmann stores in Germany. The dataset contains historical sales data from January 2013 to December 2015, along with information on store attributes and external factors such as promotions, school, and state holidays. This information can be used as features for the model.

The dataset includes information on over 1,000 Rossmann stores, including store type, size, location, and the presence of specific products or services. Additionally, the dataset contains data on external factors such as school and state holidays, promotions, and competitor activity.

Solving this problem can help the company in various ways, including identifying stores with potential sales growth, products with high sales potential, and external factors and competitors that affect sales. By using historical sales data and other information in the dataset, the model can predict future sales to make data-driven decisions.

The structured dataset contains multiple tables with information on stores, sales, and external factors. The main table includes daily sales data for each store, including store ID, date, sales, number of customers, and whether the store was open or closed on a particular day. Other tables contain data on store attributes, promotions, and competitors.

This problem is challenging because it involves time series data, external factors' impact on sales, and dealing with data quality issues. Various regression techniques, including Linear Regression, Random Forest Regression, Gradient Boosting Regression, and Neural Network Regression, can be used to tackle this problem.

To evaluate the model's performance, the most commonly used metrics are Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), and the model's performance on unseen data should be tested. Cleaning or preprocessing the data may be necessary before analysis.

Various techniques and tools, such as time series forecasting, regression techniques, ensemble methods, feature engineering, cross-validation, hyperparameter tuning, Python libraries and frameworks such as scikit-learn and XGBoost, and data visualization tools such as Matplotlib and Seaborn, can be used to solve this problem.

**Introduction:**

The demand for a product or service sometimes changes. No business can improve its financial performance without accurately estimating customer demand and future sales of products/services. Sales forecasting refers to the process of estimating demand for a particular product or selling a specific product in a particular period. In this research, we show you how machine learning can be used to predict sales in a real-world business problem.

**Problem Statement:**

Rossmann has over 3000 pharmacies in 7 European countries. Currently, Rossmann store managers are tasked with predicting their daily sales up to six weeks in advance. Store sales are influenced by many factors, such as advertising, competition, school and state holidays, seasonality, and location. With thousands of managers predicting sales based on their unique circumstances, the accuracy of results can vary greatly.

**Objectives:**

Predict sales for the next 6 weeks based on the data.

Minimize the given metric as much as possible.

**Data:**

The data can be downloaded from the Kaggle website. The files provided are:

train.csv

test.csv

store.csv

The data fields available in the dataset are:

Text

Description automatically generated

**Exploratory Data Analysis (EDA)**

Exploratory data analysis is the most important part of data analysis/machine learning. This operation accounts for about 70 to 80% of the life cycle of any data science project. Exploring, preparing, and understanding data is part of the standard methodology in this field.

Identifying types of data is an important part of the exploratory data analysis process because the statistical method we use for in-depth analysis depends on the type of data.

The details of the train.csv dataset are as follows. As we can see here, we have about 1 million data points. Also, since this is a time series prediction problem, we need to sort the data based on the date.

Text

Description automatically generated

The details of the store.csv dataset are as follows. We have a total of 1115 unique stores. Many of the columns here have missing values, which we will discuss later.

Text

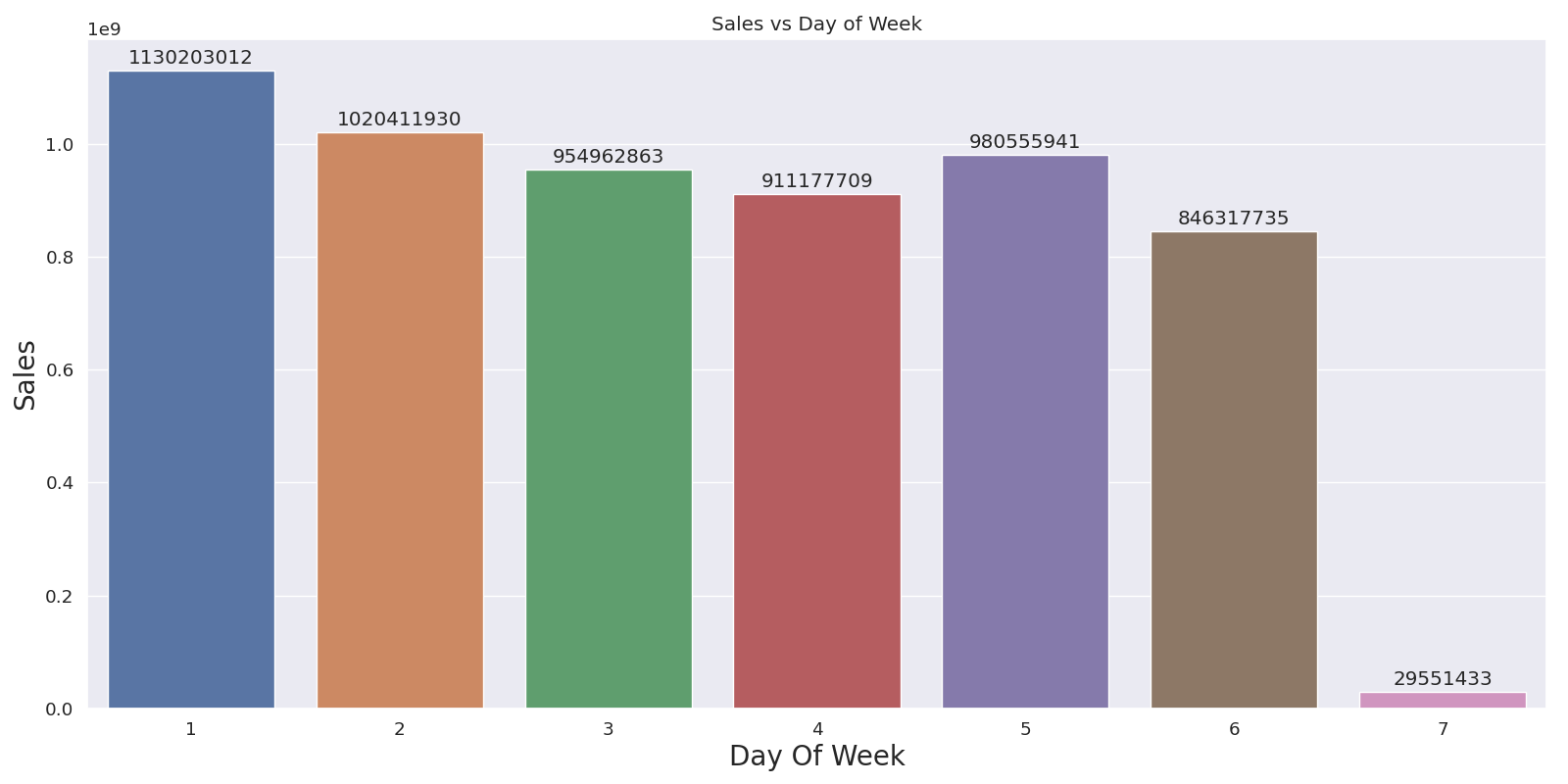
Description automatically generated

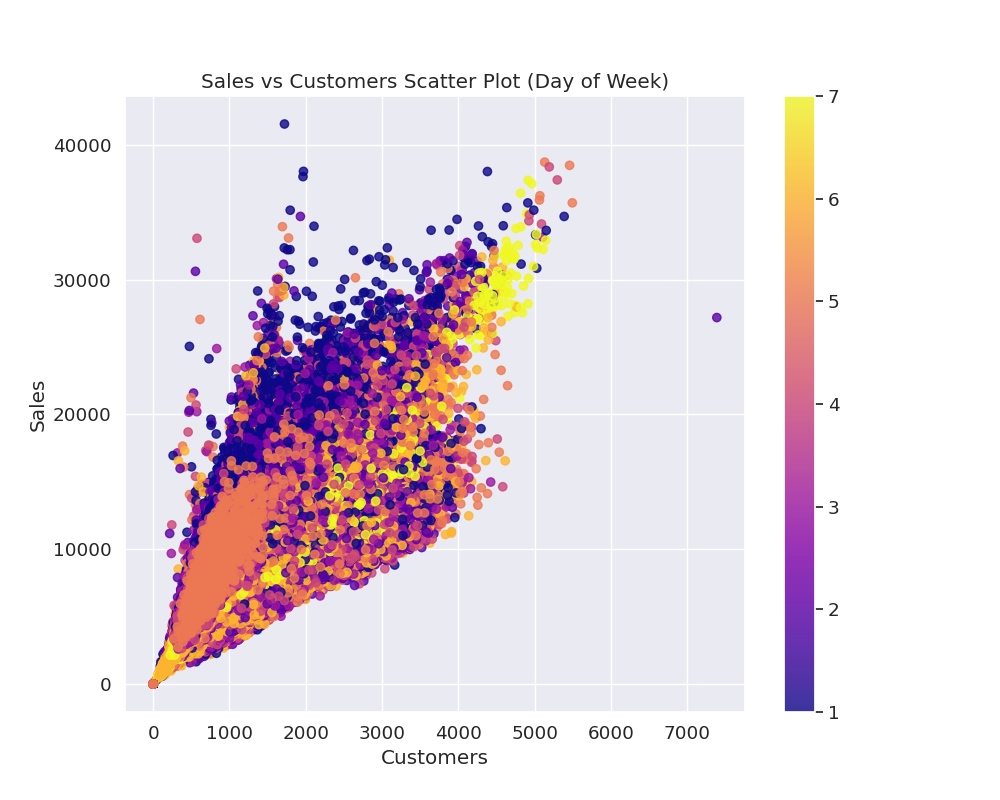
Details of the test.csv dataset are as follows.

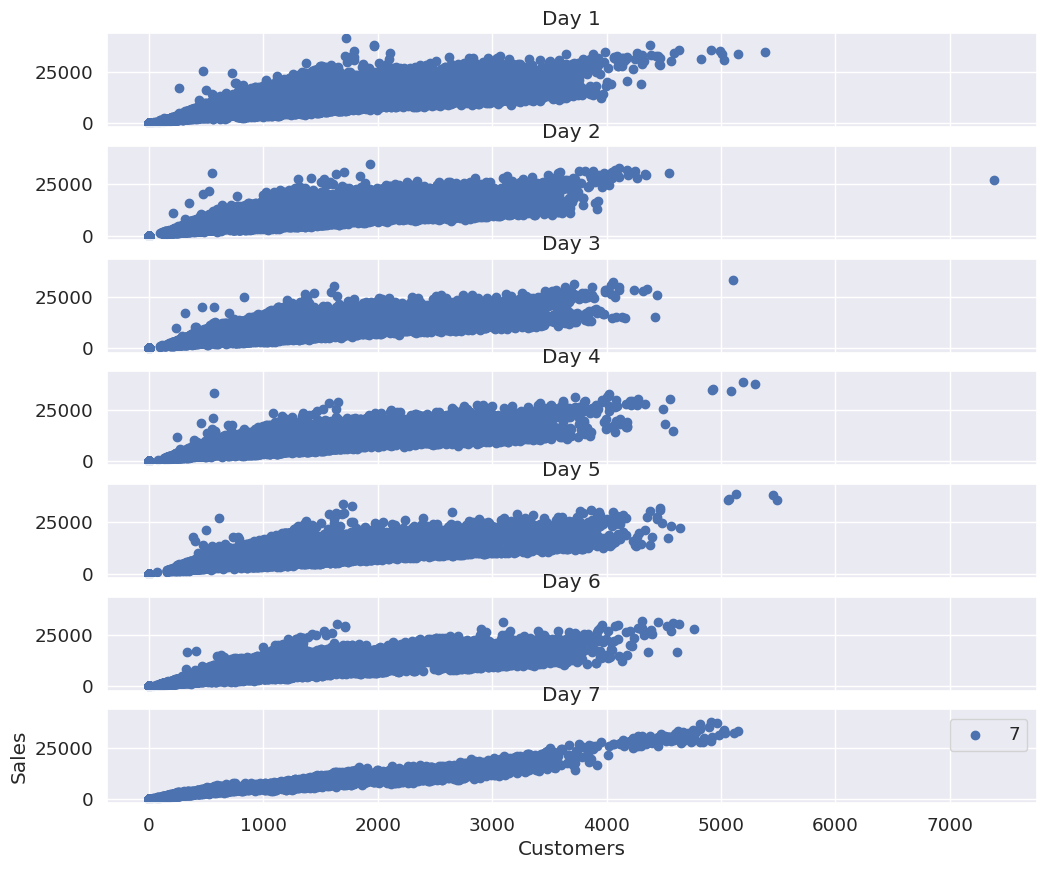
Text, table

Description automatically generated

We are now examining the information related to the columns available in the data. The figures below show the total sales on weekdays. It is clear that the highest sales are on the first day of the week or Monday, and the lowest sales are on the seventh day or Sunday.





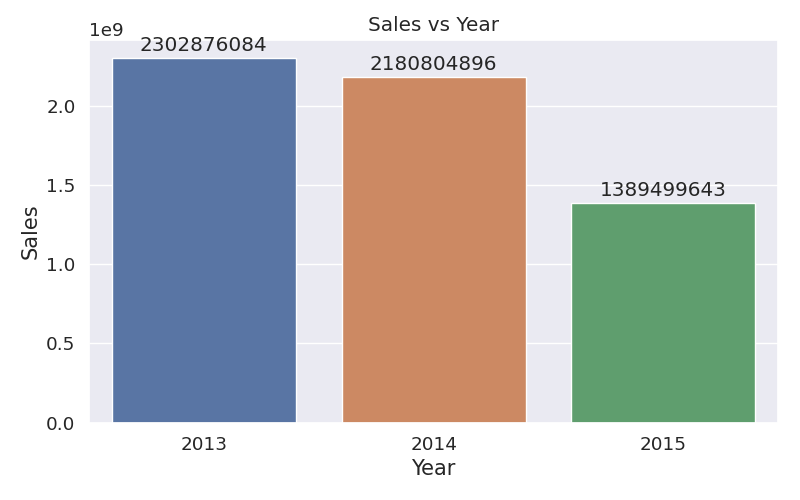


In the following graphs, we can see that both sales and customers increase significantly during promotions. This indicates that promotions have a positive effect for a store.

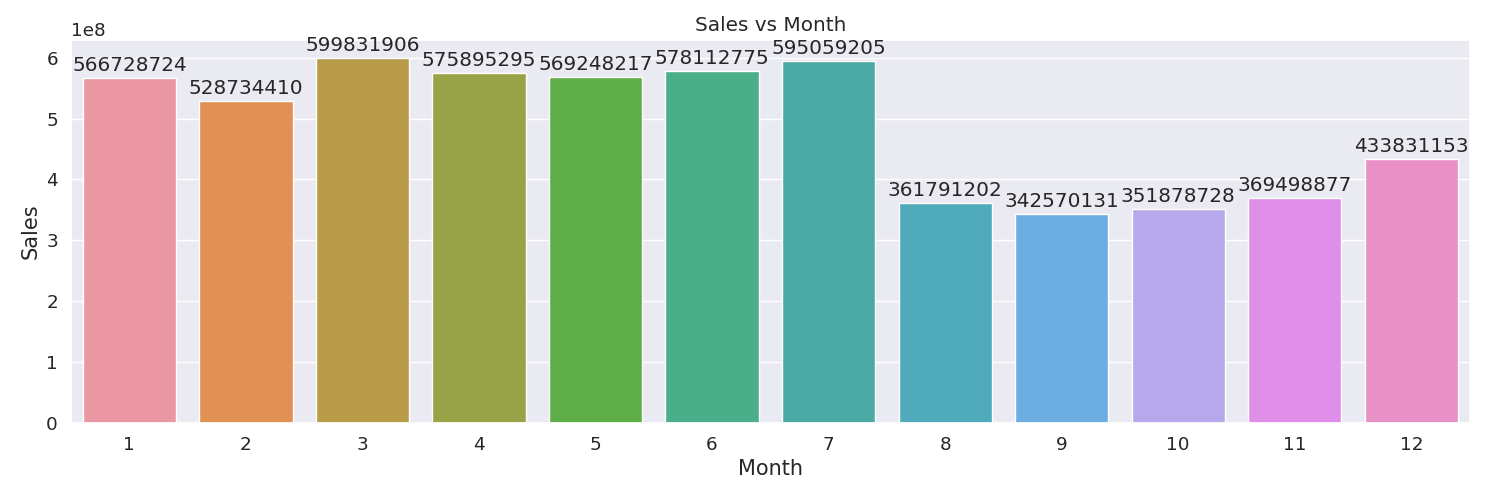
|  |  |
| --- | --- |
| C:\Users\MohammadReza\Downloads\download.png | D:\Pa\Pars207\Python NoteBooks\Store_Sales_Predictions_EDA\Sales vs Promo Plot.png |

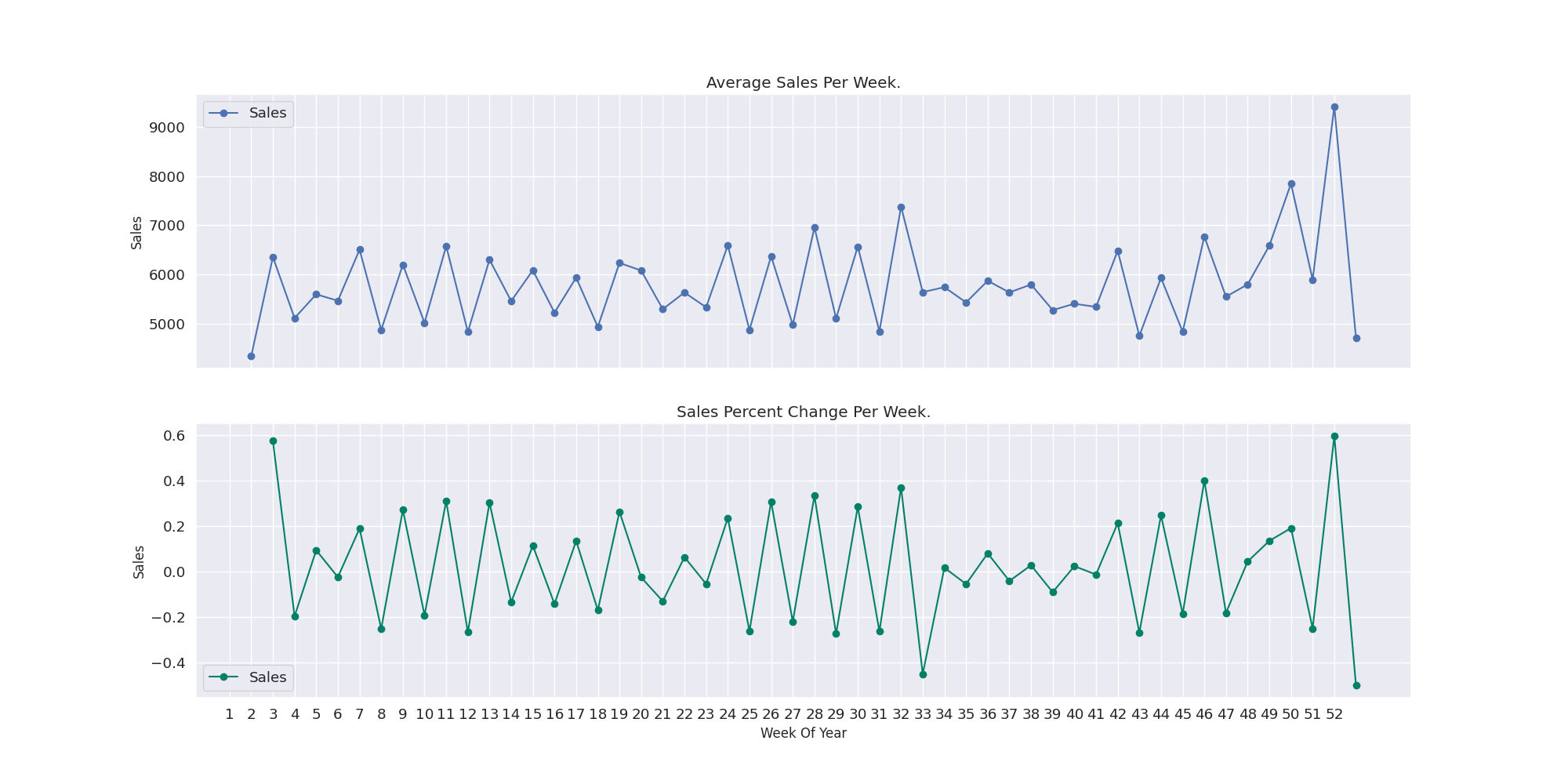


In the following figure, we see the total sales for each year. It is clear that the highest sales were in 2013 and the lowest sales were in 2015.



The figure below shows the total sales amount for each month of the year. It is clear that the highest sales occurred in the months of July and March, while the lowest sales occurred in September and October.

The figure below shows the average sales and percentage change in sales for each week of the year. The interesting point is that Christmas and New Year (weeks close to 52) lead to an increase in sales. Since Rossmann Stores sells health and beauty products, it is possible that during Christmas and New Year, people buy beauty products when they go out to celebrate, and this may suddenly increase sales.



In the following figures, we can see that both sales and customers are much lower on Mondays, as most stores are closed on Mondays. Additionally, sales are highest on Tuesdays throughout the week. This may be because stores are closed on Mondays.

|  |  |
| --- | --- |
| C:\Users\MohammadReza\Downloads\download.png | C:\Users\MohammadReza\Downloads\download (1).png |

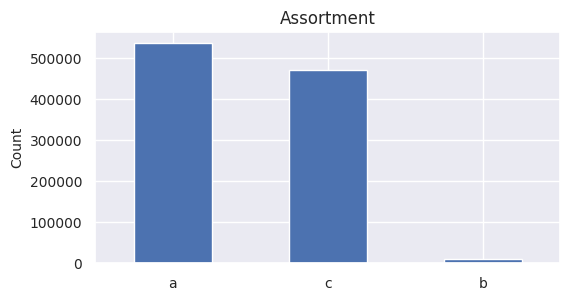
In the figure below, the number of each type of store is displayed. This figure shows that most stores are of type a and the least number of stores are of type b. Also, the scatter plot shows the amount of sales per customer for each type of store. In the figure below, stores a, b, c, and d are replaced by numbers 1, 2, 3, and 4, respectively. This figure shows that stores number 2 have the highest sales and customers.



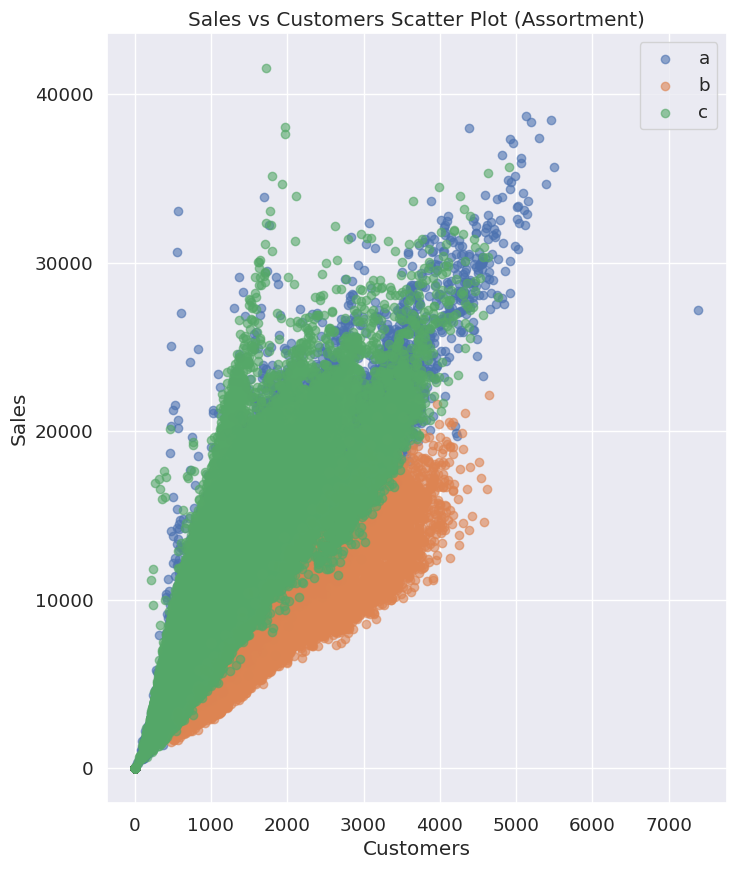


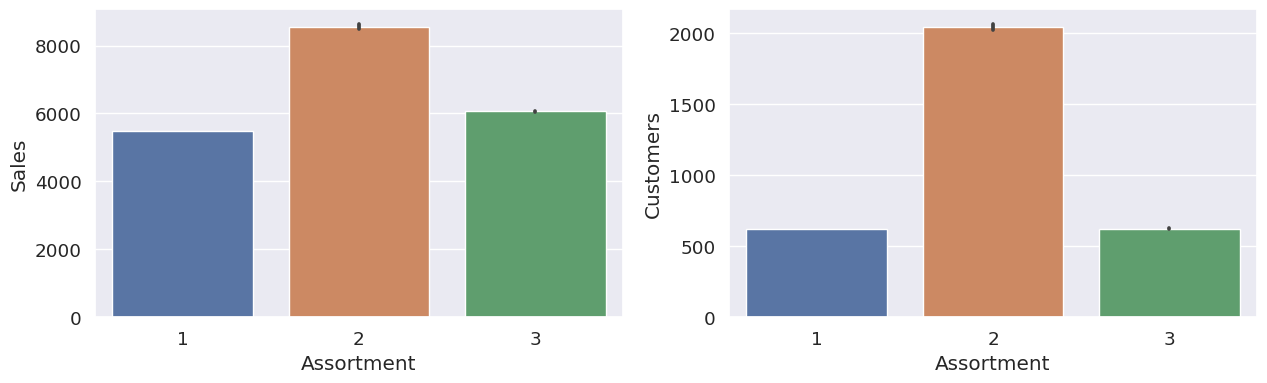


The following figure shows the number of stores for each category: a = base, b = extra, c = extended. It is clear that the majority of stores are of type a and the fewest are of type b.

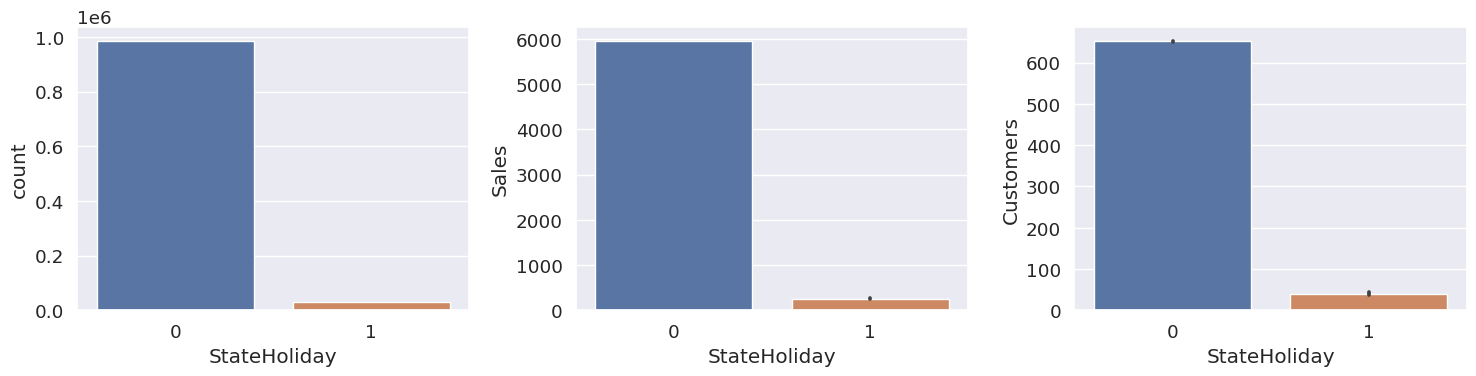


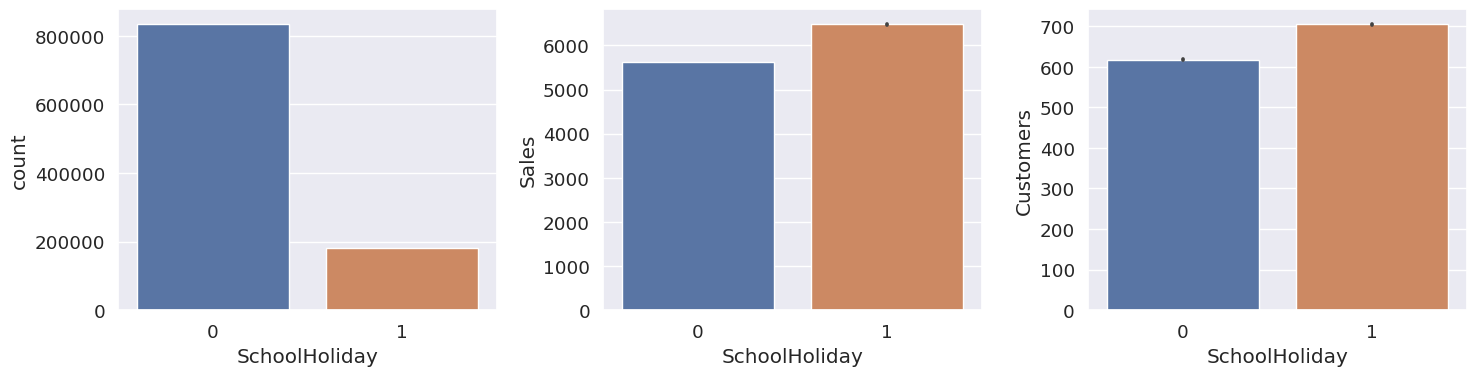
The following figures show that type b or 2 stores have the highest sales and customers. In these figures, a, b, c are replaced by numbers 1, 2, 3 respectively.



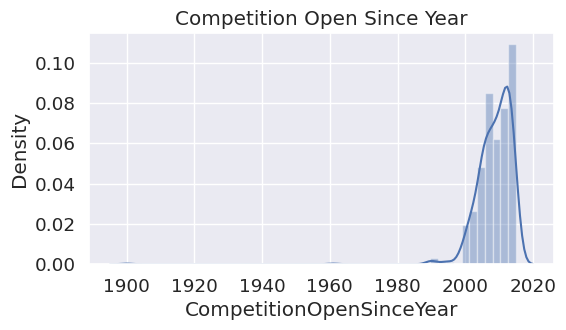


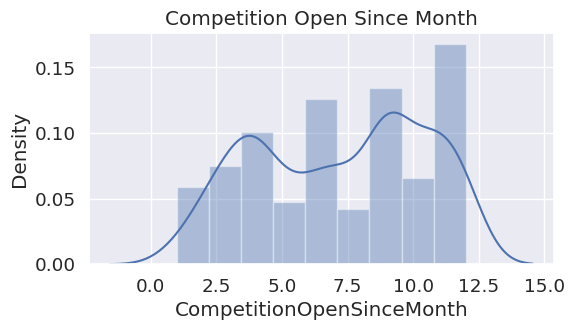
The figures below show the number of government holidays and school closures, as well as sales and customers for those days. It can be seen that most stores are closed on government holidays and school closures. However, it is interesting to note that the number of open and active stores during school closures is greater than the number of stores that are open during government holidays. Another important point to mention is that stores that were open during school closures had higher sales than normal.



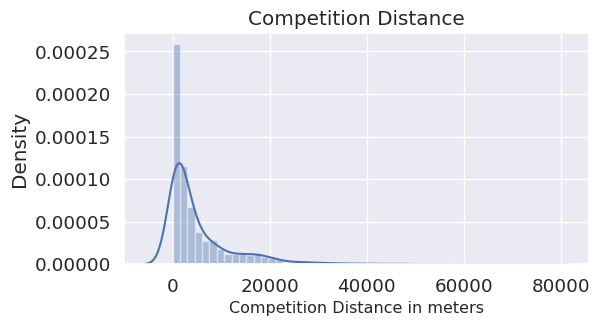


The following chart shows the approximate year and month of the nearest competitor store opening. It is clear that most stores started their competition after the year 2000.





The figure below shows the distance in meters to the nearest competitor store. It is clear that most stores have their competition within a range of 2 kilometers.



Descriptive statistics for the variables in the datasets used in this study are presented in the following tables.

Descriptive statistics values of the features in the train dataset

Table

Description automatically generated

Descriptive statistics values of the features in the store dataset

Table

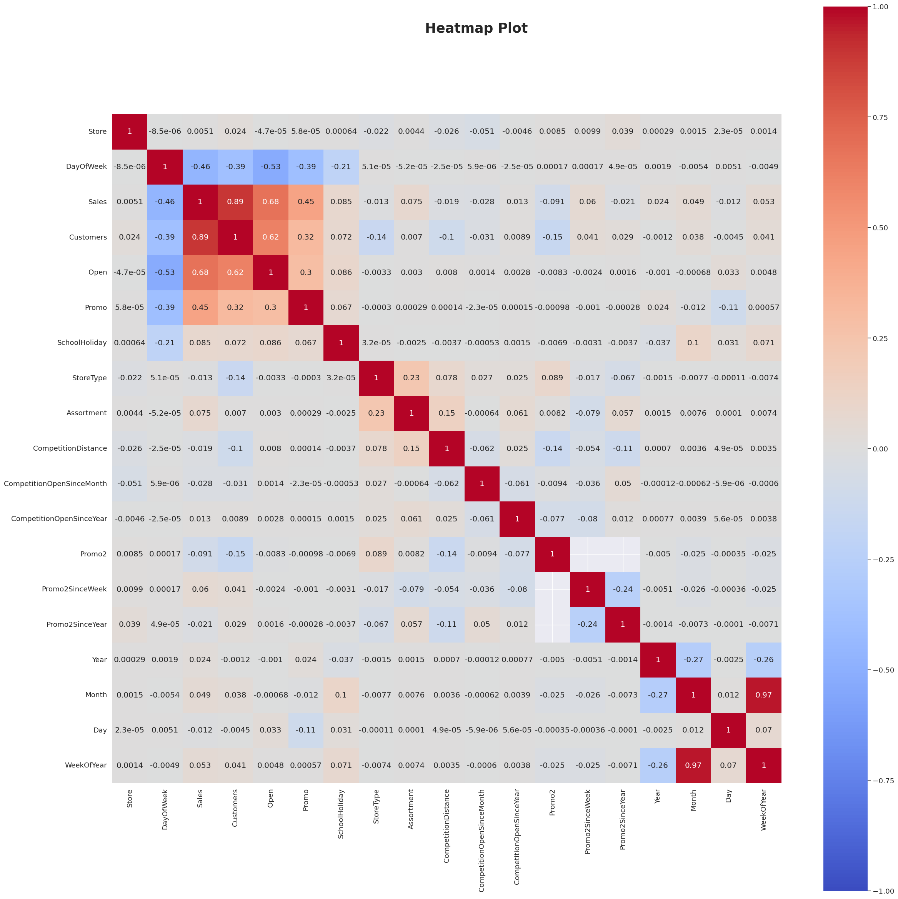
Description automatically generated

Descriptive statistics values of the features in the test dataset

Table

Description automatically generated

The figure below illustrates the correlation between variables in the dataset.

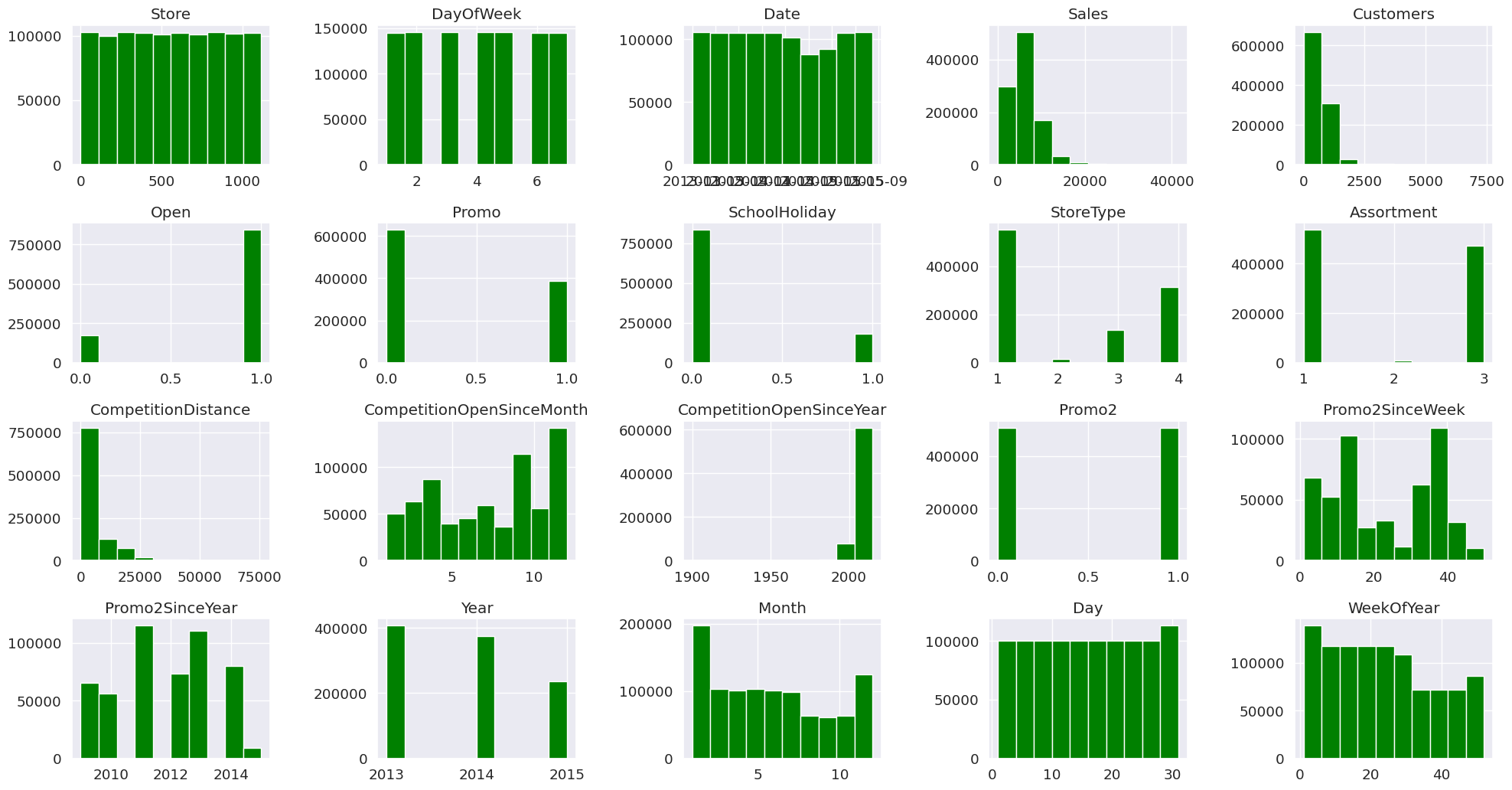


The correlation between two features X and Y is calculated according to the following equation.

Text

Description automatically generated

In this equation, Corr: correlation of two features, Cov: covariance of two features, ρ: standard deviation, E: mathematical expectation and µ: mean of a feature are. The numerical correlation coefficient is between 1 and -1. And it shows the degree of relationship and the type of relationship between two features (direct or inverse). If two features are not related to each other, the correlation will be zero. Also, the histogram of the numerical variables in the dataset used in this research is shown in below figure



**Steps of proposed method for predicting Sales**

In the first stage, data preprocessing is performed, which includes data selection, data cleaning, and data transformation and normalization. In the next stage, after performing the data preprocessing steps, we present the preprocessed dataset to machine learning algorithms XGBoost, Random Forest, Decision Tree, and Linear Regression to predict the Sales. Finally, the evaluation parameters of accuracy and performance of the machine learning algorithms are obtained in the training and testing stages and compared with each other. First, we merge the store data with the train and test data based on the store number. Then, we proceed to preprocess the data.

**Missing Data**

In some cases, one or more features of a sample may be missing valid values. This can have various reasons, including noisy recorded data, lack of recording, or invalid value. These data are called missing data. To process datasets that face the challenge of missing data, a solution must be found to determine the values of missing data. Various methods exist for managing missing data, including deleting samples, manual filling, using a constant value, using the average feature value, and so on. In this study, in the train and test datasets, we replaced the missing values of the CompetitionDistance variable with its maximum value. Other variables have their own preprocessing and feature engineering operations.

Table

Description automatically generated

Table

Description automatically generated

We use data from stores that were open to train machine learning models for predicting the Sales variable.

Text

Description automatically generated

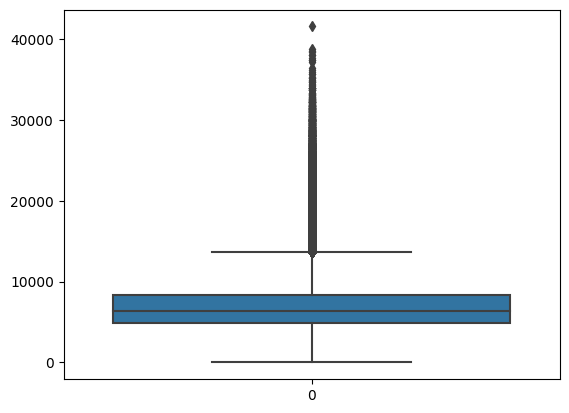
We can use the CompetitionOpenSince[Month/Year] columns from the store dataset to calculate the number of months that competitors have been open near each store.

We can also add some extra columns to indicate how long a store has been running Promo2 and whether a new round of Promo2 will start in the current month or not.

**Outliers**

In some cases, values may appear in a dataset that have a significant and unusual difference from other values in the dataset. These are called outliers. In fact, outliers are data or samples that do not resemble the general behavior or pattern of the overall dataset and do not conform to it. Analyzing and managing outliers is one of the most important stages in data preprocessing, as they can cause problems for the performance of algorithms and classifiers used. Therefore, the impact of outliers should be reduced as much as possible. There are various methods for identifying and managing these data, including removal, replacement with the mean, and so on.

The Box Plot graph effectively and efficiently summarizes the data using a simple box. The Boxplot summarizes sample data using quartiles of 25, 50, and 75. By simply looking at the box plot of the data, you can obtain quartiles, median, and outliers for each data set. In the Box Plot graph, which is related to the train data shown below, you can clearly see that values above 13611.5 in sales act as outliers. Therefore, these points are removed from the dataset.



**Data Normalization**

Normalization methods such as Min-Max normalization, Z-score normalization, decimal scaling, etc. are among the most common methods of normalization. In order to reduce the effects of data scaling on model results, data normalization was performed in the range of 0 to 1. Before starting the modeling process, both inputs and outputs should be normalized, as entering raw data can reduce the speed and accuracy of the model. The following formula is used to normalize input data within the range of a and b.

Diagram, schematic

Description automatically generated

In this regard, the minimum and maximum value of input data andnormalized data, respectively. Also, a and b are equal to the lower and upper limits of the desired range for normalization, which are equal to 0 and 1, respectively.

**K-Fold Cross Validation for Model Validation**

The cross-validation method used in this study for training machine learning models for customer classification is K-fold cross-validation. If we randomly divide the training dataset into k equal sub-samples or folds, at each stage of the CV process, k-1 of these folds can be considered as the training set and one as the validation set. By choosing k=10, for example, the number of repetitions of the CV process will be 10, and obtaining a suitable model will be quickly possible. In this study, the number of folds was set to 5 (K=5).

**Evaluation and Validation Metrics**

In this research, in order to evaluate and compare the performance of different machine learning models in predicting Sales, the following valid metrics have been used:

* Determination Coefficient

Text

Description automatically generated with medium confidence

* Root Mean Squared Error

A picture containing text, clock

Description automatically generated

* Mean Absolute Percentage Error

Diagram

Description automatically generated

* Mean Absolute Error

Diagram

Description automatically generated

**Hyperparameter tuning of machine learning methods using grid search**

Hyperparameters of machine learning algorithms need to be determined by the user. Tuning the hyperparameters of machine learning algorithms means finding the best combination of these parameters so that the model performance improves. For this reason, in this research, the grid search method has been used to tune and select the best hyperparameters of XGBoost, Random Forest, Decision Tree, and Linear Regression machine learning methods. To do this, we define a list of hyperparameters and ranges of values for each hyperparameter. Then we create a network of these hyperparameters to test the available samples in this network during the training phase. We do this using the GridSearchCV method, which evaluates all combinations in this network. Finally, the best values for the hyperparameters of the machine learning algorithm are determined using this method.

**Results of Sales Prediction using XGBoost**

The best hyperparameters for the XGBoost model obtained through GridSearchCV method are as follows. The total running time for this model was 5 minutes and 37 seconds.



Sales prediction was performed using the XGBoost model, and the evaluation metrics and results in both training and testing stages are as follows:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Train Results** | **Test Results** |
|  | **0.9** | **0.64** |
| **RMSE** | **988.7** | **1813.55** |
| **MAPE** | **0.04** | **0.07** |
| **MAE** | **695.35** | **1241.69** |
| **Time** | **5 min 37 s** | **-** |

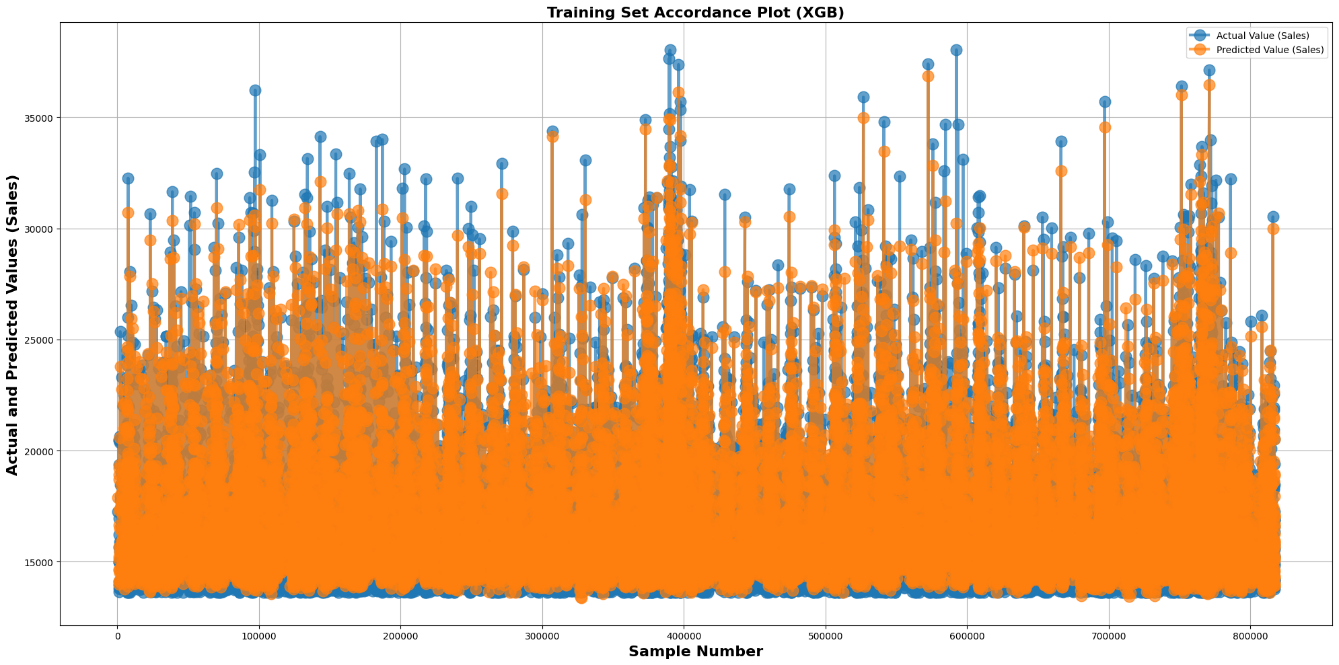
**Scatter plot of actual and predicted values of Sales variable by XGBoost model**

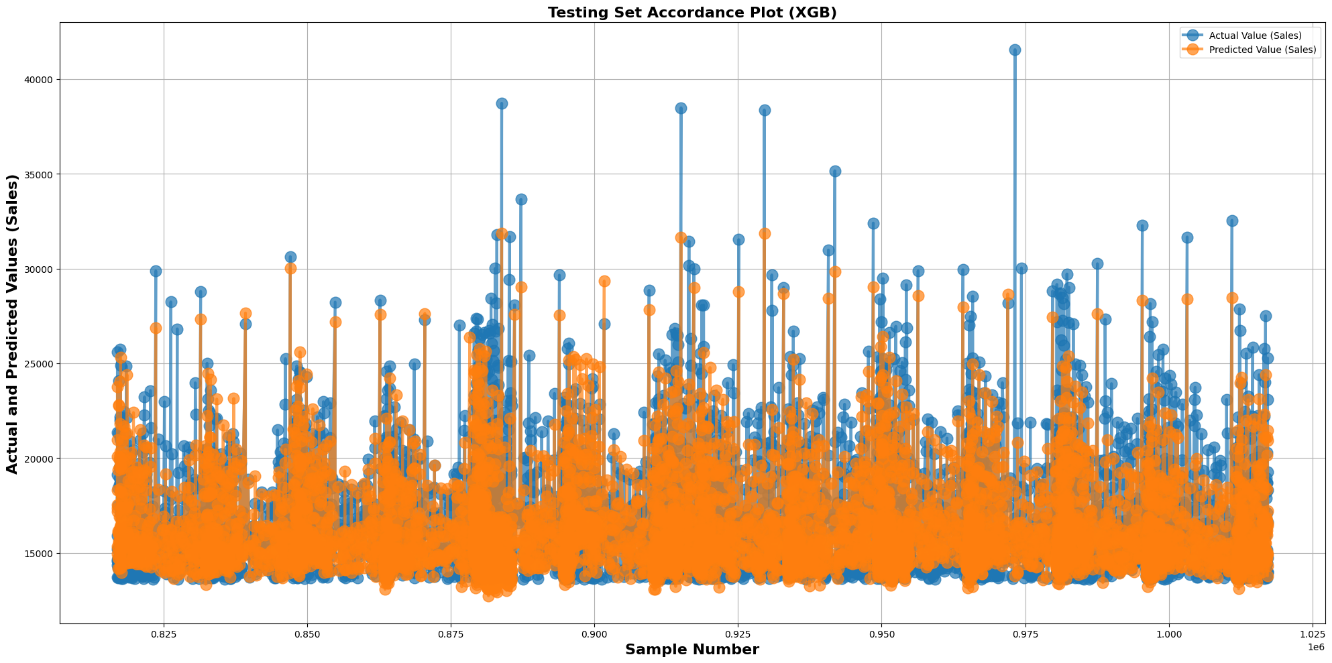
In the following figures, the scatter plot related to the prediction of Sales variable using XGBoost model is shown. As can be seen from this plot, the dark black diagonal line represents the first bisector in the evaluation metric and performance assessment. The actual values of Sales variable are represented on the X-axis and the predicted values of Sales variable are represented on the Y-axis. These values are clustered around the first quadrant (dark black diagonal line) and closer to it, indicating better correlation, accuracy and more precise prediction of Sales variable. It is well evident from this plot that a high percentage of Sales variable values are close to the first bisector line.

|  |  |
| --- | --- |
| C:\Users\MohammadReza\Downloads\download.png | C:\Users\MohammadReza\Downloads\download (1).png |

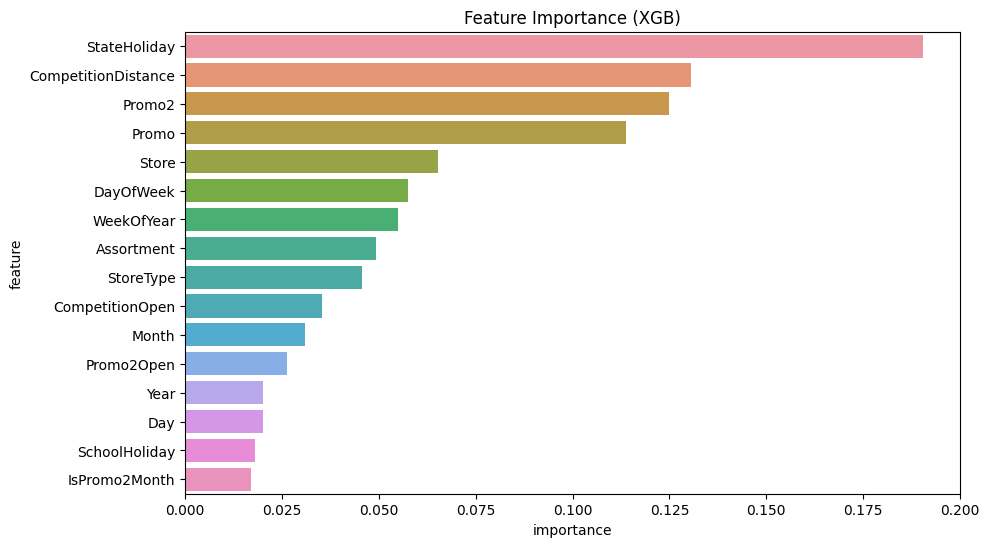
**Scatterplot of the agreement between actual and predicted values of the Sales variable by the XGBoost model**

The scatterplots of the agreement between actual and predicted values of the Sales variable by the XGBoost model in the training and testing stages are shown in the following figures. As it can be seen from these plots, the closer the two plots are to each other, the more accurate the Sales variable prediction with less error is indicated.





The following figure shows the importance score of each feature for predicting the Sales variable using the XGBoost method on the input dataset.



**Results of Sales prediction using Random Forest**

The best hyperparameters of the Random Forest model obtained by GridSearchCV are as follows. The total running time of this model was 11 minutes and 23 seconds.

Text

Description automatically generated

Sales prediction was performed by the Random Forest model and the evaluation metrics for the training and testing phases are as follows:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Train Results** | **Test Results** |
|  | **0.92** | **0.66** |
| **RMSE** | **903.26** | **1763.21** |
| **MAPE** | **0.04** | **0.07** |
| **MAE** | **608.3** | **1239.6** |
| **Time** | **11 min 23 s** | **-** |

**Scatter plot between actual and predicted values of Sales variable by Random Forest model**

The scatter plot related to Sales variable prediction using the Random Forest model is shown in the following figures. As it can be seen from this plot, the dark black diagonal line represents the first bisector in the perfect prediction and fit. The actual values of the Sales variable are shown on the X-axis, and the predicted values of the Sales variable are shown on the Y-axis. The closer these values are to the first bisector (dark black line), the better the fit and more accurate the Sales variable prediction. It is clear from this plot that a high percentage of Sales variable values are close to the first bisector line.

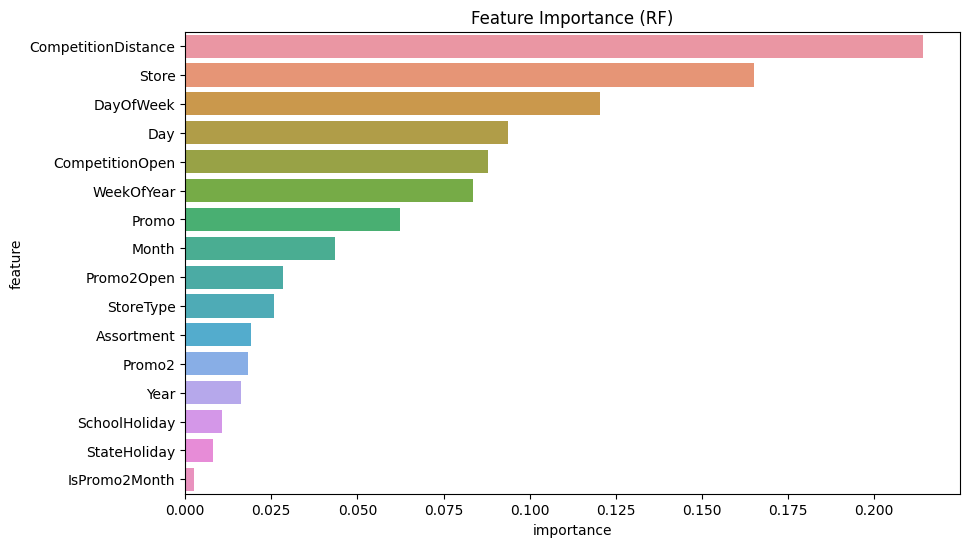
|  |  |
| --- | --- |
| C:\Users\MohammadReza\Downloads\download (5).png | C:\Users\MohammadReza\Downloads\download (6).png |

**Scatterplot of actual and predicted values of Sales variable by Random Forest model**

In the following figures, the match plot for predicting the Sales variable using the Random Forest model in two stages of training and testing is shown. As it can be seen from this plot, the more similar the two plots are, the more accurate the prediction of the Sales variable with less error is. The actual values of the Sales variable are represented on the X-axis and the predicted values of the Sales variable are represented on the Y-axis. The closer the two plots are to the 45-degree line, the more accurate the predictions are.

|  |
| --- |
| C:\Users\MohammadReza\Downloads\download (7).png |
| C:\Users\MohammadReza\Downloads\download (8).png |

The following figure shows the importance score of each feature in the input dataset for predicting the Sales variable using the Random Forest method.

 Results of Sales Prediction Using Decision Tree

The best hyperparameters of the Decision Tree model obtained by the GridSearchCV method are as follows. The total execution time of this model was 2 minutes and 28 seconds.



The Sales prediction was performed using the Decision Tree model, and the evaluation metrics and results in the training and testing stages are as follows.

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Train Results** | **Test Results** |
|  | **0.61** | **0.34** |
| **RMSE** | **1932.81** | **2450.5** |
| **MAPE** | **0.08** | **0.09** |
| **MAE** | **1354.73** | **1640.66** |
| **Time** | **2 min 28 s** | **-** |

**Scatter plot of the actual and predicted values of Sales variable by Decision Tree model**

In the following figures, the scatter plot of the Sales variable prediction using the Decision Tree model is shown. As can be seen from this plot, the dark black diagonal line indicates the first-degree line of accuracy and equality. The actual values of the Sales variable are represented on the X-axis, and the predicted values of the Sales variable are represented on the Y-axis. The closer these values are to the first-degree line, the better the estimation accuracy of the Sales variable is. It is well evident from this plot that a high percentage of Sales variable values are close to the first-degree line.

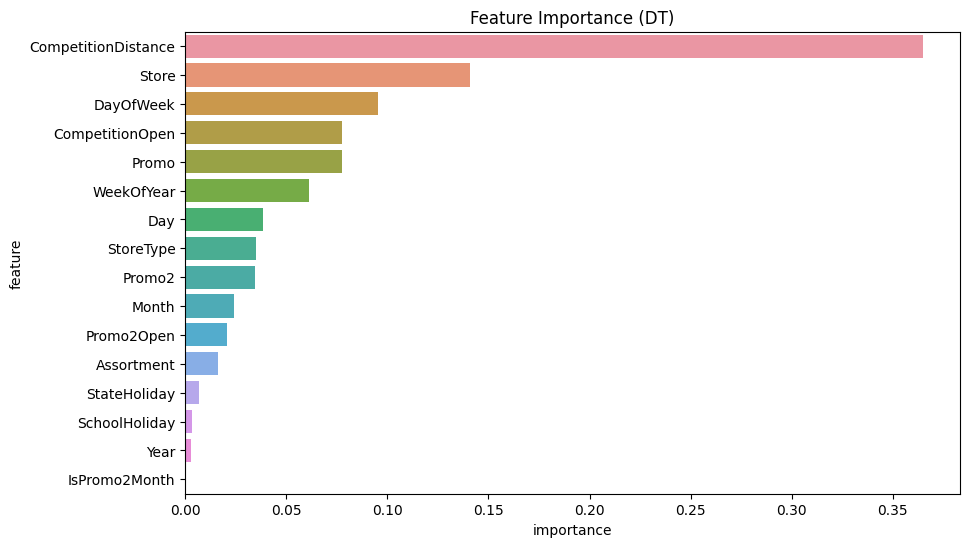
|  |  |
| --- | --- |
| C:\Users\MohammadReza\Downloads\download (10).png | C:\Users\MohammadReza\Downloads\download (11).png |

**Scatter plot of the match between actual and predicted values of the Sales variable by the Decision Tree model**

In the following figures, the matching plot for predicting the Sales variable using the Decision Tree model in both the training and testing stages is shown. As can be seen from these plots, the more the two plots are matched, the more accurate the prediction of the Sales variable with higher accuracy and lower error.

|  |
| --- |
| C:\Users\MohammadReza\Downloads\download (12).png |
| C:\Users\MohammadReza\Downloads\download (13).png |

The figure below shows the importance score of each feature in the input dataset for predicting the Sales variable using the Decision Tree method.



**Results of Sales prediction using Linear Regression**

The total execution time of this model was 172 milliseconds.

Sales prediction was performed using Linear Regression model, and the results and evaluation metrics in the training and testing stages are as follows:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Train Results** | **Test Results** |
|  | **0.05** | **0.07** |
| **RMSE** | **3022.24** | **2906.8** |
| **MAPE** | **0.13** | **0.12** |
| **MAE** | **2246.37** | **2102.19** |
| **Time** | **172 ms** | **-** |

**Scatter plot of the actual and predicted values of Sales variable by Linear Regression model**

The scatter plot related to the prediction of Sales variable using Linear Regression model is shown in the following figures. As the plot shows, the dark black diagonal line is the first bisector in the origin of coordinates. The actual values of Sales variable are plotted on the X-axis and the predicted values of Sales variable are plotted on the Y-axis. The values in the first quadrant (dark black diagonal line) represent the agreement between the actual and predicted values, indicating a better prediction of the Sales variable with less error. This figure shows that the values of the Sales variable are far away from the first bisector line, which indicates the weakness of this model in predicting the Sales variable accurately.

|  |  |
| --- | --- |
| C:\Users\MohammadReza\Downloads\download (15).png | C:\Users\MohammadReza\Downloads\download (16).png |

**Scatterplot of the match between actual and predicted values of the Sales variable by the Linear Regression model**

The scatterplot of the match between actual and predicted values of the Sales variable by the Linear Regression model in both training and testing stages is shown in the following figures. As can be seen from these plots, the more the two plots match each other, the more accurate the prediction of the Sales variable with less error. It can be inferred from these plots that the linear regression method is not capable of predicting the Sales variable accurately.

|  |
| --- |
| C:\Users\MohammadReza\Downloads\download (17).png |
| C:\Users\MohammadReza\Downloads\download (18).png |

The figure below shows the coefficient of the regression line for each feature for each variable in predicting Sales variable using the Linear Regression method from the input dataset. Also, the value of the intercept is equal to 306155.13.

Graphical user interface, application

Description automatically generated

**Overall Comparison of Models in Predicting Sales Variable**

As can be seen from the table below, the Random Forest model has the highest accuracy in predicting the Sales variable, followed by the XGBoost model. The results of the Decision Tree model indicate relatively low accuracy in predicting the Sales variable. The results of the Linear Regression model show that this model is not capable of predicting the Sales variable, probably because there is a non-linear relationship between the input variables and the Sales variable.

**Discussion and Conclusion**

After finding the best model and its corresponding hyperparameters, we used the XGBoost model to make predictions for all stores and calculated the mean of the evaluation metrics.

After evaluating all the machine learning models, the XGBoost model showed the highest accuracy and the lowest error. Hence, we used this model to predict the sales of all the shops, and the RMSE and Score values for each shop are presented in the figure below.

Although the XGBoost model demonstrated relatively good accuracy even without hyperparameter tuning, optimizing hyperparameters for all the shops was time-consuming due to the large number of shops. In many cases, classical models with tuned hyperparameters can provide good accuracy, and it is not always necessary to use new and complex models. For this project, we even attempted to use a deep learning model such as the Artificial Neural Network, but it did not perform well in learning the training data, and there was a very high error rate for new data.

Classical models like XGBoost, by improving their algorithms and processing speed, have met the modeling needs of researchers. Currently, a significant portion of data science competitions, including those on Kaggle, use boosting models such as XGBoost, which provides excellent results.

Finally, we used the best model to predict future sales and saved the predicted values in the test file.

Table

Description automatically generated

**GitHub**: <https://github.com/mallyarefi/TMU-Project>

**References**:

1. Russell, S.J. and P. Norvig, Artificial intelligence: a modern approach. Malaysia. 2016, Pearson Education Limited.
2. Friedman, J., T. Hastie, and R. Tibshirani, The elements of statistical learning. Vol. 1. 2001: Springer series in statistics New York.
3. Hertz, J., A. Krogh, and R. Palmer, Introduction to the theory of neural computation Addison Wesley. Redwood City, CA, 1991.
4. Lawrence, J., Introduction to Neural Networks: Design. Theory, and Applications (California Scientific Software, Nevada City, CA), 1994.
5. Sheela, K.G. and S.N. Deepa, Review on methods to fix number of hidden neurons in neural networks. Mathematical Problems in Engineering, 2013. 2013.
6. Hornik, K., M. Stinchcombe, and H. White, Multilayer feedforward networks are universal approximators. Neural networks, 1989. 2(5): p. 359-366.
7. Zhang, G., B.E. Patuwo, and M.Y. Hu, Forecasting with artificial neural networks: The state of the art. International journal of forecasting, 1998. 14(1): p. 35-62.
8. Lippmann, R., An Introduction to Computing with Neural Nets' IEEE ASSP Magazine. Proc. IEEE, 1987. 4: p. 4-22.
9. Wang, Z., et al., A procedure for determining the topology of multilayer feedforward neural networks. Neural Networks, 1994. 7(2): p. 291-300.
10. Tang, Z. and P.A. Fishwick, Feedforward neural nets as models for time series forecasting. ORSA journal on computing, 1993. 5(4): p. 374-385.
11. Kaastra, I. and M.S. Boyd, Forecasting futures trading volume using neural networks. Journal of Futures Markets, 1995. 15(8): p. 953-970.
12. Rumelhart, D.E., B. Widrow, and M.A. Lehr, The basic ideas in neural networks. Communications of the ACM, 1994. 37(3): p. 87-93.
13. Mohammadbagher, M., Foundations of neural networks: Computational intelligence. Vol. 0. 1396: Amirkabir University of Technology (Tehran Polytechnic).
14. Haykin, S., Neural networks: a comprehensive foundation. 1994: Prentice Hall PTR.