Black Friday Sales Prediction

Parth P. Shah
Master of Science in Computer Science
Northeastern University
shah.parth2@northeastern.edu

Abstract

Nowadays, a successful business is developed by taking into consideration a lot of factors, one of which is the purchasing patterns and habits of the customers. Hence, sales prediction based on this factor is rapidly gaining importance. Sales forecasting is a key element in the financial planning for any retail business. It helps the owners to decide where they can invest more to get better returns and where they are losing capital. Forecasting is usually done on the sales data collected in the past for the store. An accurate sales prediction can prove to be fruitful in the long run for a business. In this paper, sales prediction is done using machine learning approaches on historical time-series data for Black Friday sales at a grocery store. The forecasting is performed based on the analysis of various fields and the correlations between them. Further, to predict the purchase amounts of various products, I have experimented with different supervised machine learning models such as Linear, Random Forest, and Extreme Gradient Boosting (XGBoost) Regression models. Some of the traditional prediction models that may have performed poorly led to the development and use of newer enhanced techniques. On the basis of the performance of these techniques, the best approach is suggested for sales prediction. The performance is measured by the degree of prediction errors of the models. Experiments have shown that the XGBoost Regression approach is the better one among the rest with the highest accuracy for sales prediction.

Introduction

The sale forecasting techniques play an important role in modern businesses. But, the lack or incompleteness of data makes it a sought-after problem. Currently, many companies maintain huge data storage mechanisms and try to extrapolate customer trends to develop sales-boosting strategies based on data analysis. Small businesses are gradually getting into this act. The business leaders are trying to come up with strategies that can use such methods for formulating their annual budgets and other finance-related aspects.

When it comes to stores in our neighborhood, we often see various kinds of sales, deals, or offers that are put up to attract customers. These need to be put up very cautiously as they have a huge impact (either positive or negative) on the business. For instance, if the store places offer on a high-selling item, it may incur a loss as the item was already selling more without the offer. Similarly, if the offer (keeping in mind the cost price and selling price) is made on a less selling item, it may boost its sales which in turn will lead to the increased overall business. Hence such decisions can only be made after analyzing the sales trends for the various items in the store. One such sale is the Black Friday sale, which is one of the biggest sales of the year. Hence, at this time, the customers and the stores together try to get the most out of the opportunity. These predictive models can help boost their sales by providing personalized offers to the customers based on their purchasing habits.

In this study, a dummy data-set provided by Analytics Vidhya is used. A grocery store has shared transaction details of various customers with respect to the products. The data-set has a large number of entries and contains several fields such as customer details like age, gender, marital status, category of city, etc., and details of products such as id and category, and the purchase amount. Here, goal is to accurately predict the purchase amount (target field) of various products in the store. Also, there are many invalid values for some of the fields, which are dealt with using different approaches. Extensive data analysis is performed on the fields to extract patterns among the fields. Several relations are made among the various fields for feature selection, as a result of which a new feature is generated by combining a few fields which have a higher correlation with the target field. Many visualization techniques like bar graphs, box-plots, and heat-maps have been implemented. The predictive modes namely, Linear Regression, Random Forest Regression, and the XGBoost Regression have been employed and compared to give the best prediction results. The evaluation of their performances is done using the RMSE values. Finally, analysis and results have been summarized in the conclusion.

In the following sections of the paper, Section 2 deals with the conceptual background of the various forecasting techniques used. Section 3 is about the various works that have already been published concerning the same problem. Section 4 describes the various methods for modeling in detail. Section 5 talks about the experimentation for the exploratory data analysis methods and the performance comparisons. Finally, Section 6 summarizes, concludes, and suggests future

possibilities for this problem.

Background

This paper evaluates three Supervised Machine Learning approaches that are used for the sales forecasting task. These are Linear Regression, Random Forest, and XGBoost. The basic concept of each of these algorithms is explained in detail in the following sections.

Linear Regression

In machine learning based approaches, Linear Regression refers to a statistical regression method that is used for predictive analysis. It depicts the linearity between the independent and the dependent (target) variables.

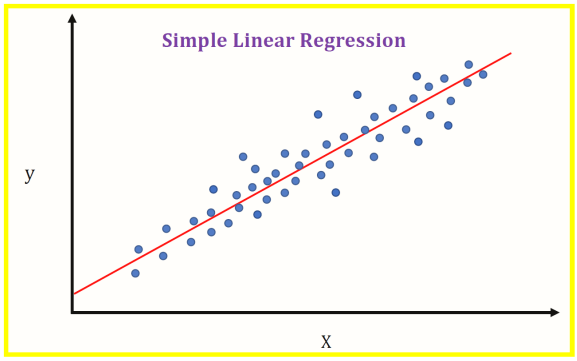

Figure 1: Simple Linear Regression

From the graph we can see linearity between the output and the input variables. The red line is called the best-fit line. From the given data points, we attempt to plot a line that best fits the points.
The equation for linear regression is given as follows:

$$Y_i = \beta_0 + \beta_1 X_i$$

where Y_i = dependent variable, β_0 = intercept, β_1 = slope, X_i = independent variable.
The linearity between the dependent and the independent variables can be understood by a straight line as follows:

$$Y = B_0 + B_1 X.$$

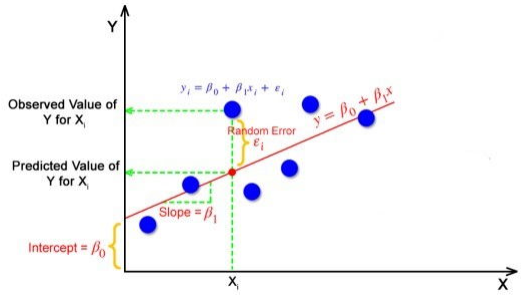

Figure 2: Linear Regression

The linear regression algorithm tries to get the best values for B_0 and B_1 in the search of the regression line. The best-fit line is the one that has the lowest amount of error between predicted and actual values. In other words, it is a line which can fit a given scatter plot in the best way possible.

Random Forest

Random forest is widely used in regression problems. In this, decision trees are built for different samples and the average vote is counted for regression. Random Forest Algorithm can handle the data set that contains continuous variables.

Random forest works on the Bagging principle. In Bagging, a random sample is chosen from the given data set. Thus, every model is built from the samples given by the original data with replacement. Now, each model is independently trained to generate results. The final output is based on average votes after combining the results of all the models.

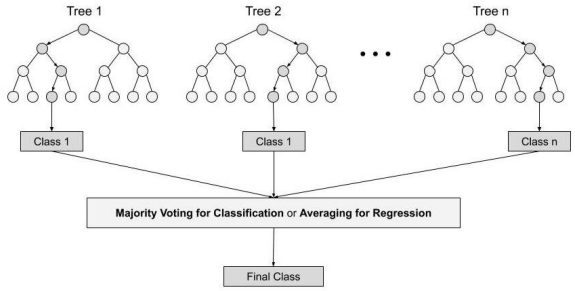

Figure 3: Random Forest

Following steps are involved in the random forest algorithm:

1. Some of the samples are taken at random from the data set.
2. For every sample, an individual decision tree is constructed.
3. An output is generated by every decision tree.
4. For the final regression prediction, the average of the outputs of all the individual decision tress is considered.

XGBoost Regression

Extreme Gradient Boosting is an ensemble learning method. In an ensemble learning model, a combination of multiple models is used for prediction task. The newly formed model is a unique model that gives the final output by accumulating the outputs from the models it consists. Two most widely used ensemble approaches are bagging and boosting. Bagging reduces the variance in any algorithm. In this approach, the decision trees are built one after the other in a sequence, in such a way that every new tree attempts to decrease previous tree's error. Every tree learns from its predecessor and updates the errors. Therefore, the newly added tree in the sequence will learn from an updated version of the errors. In contrast to Random Forest, in which trees are grown to their

maximum extent, boosting makes use of trees with fewer splits.

The boosting ensemble technique consists of three steps:

1. Initially, a model F_0 is defined to predict the target variable y. This model will have a residual $(y - F_0)$.
2. A new model h_1 is fit to the errors from the previous step.
3. Now, F_0 and h_1 are combined to give F_1, the boosted version of F_0. The mean squared error from F_1 will be lower than that from F_0:

$$F_1(x) = F_1(x) + h_1(x)$$

For improving the performance of F_1, it is modeled after the errors of F_1 and create a new model F_2:

$$F_2(x) = F_1(x) + h_2(x)$$

This procedure is repeated for 'm' iterations until the errors become minimal:

$$F_m(x) = F_{m-1}(x) + h_m(x)$$

In this, the newly added models do not disturb the functions created in the previous steps. Instead, they try to reduce the errors by imparting their own information.

Gradient descent minimizes any differential function. In gradient boosting, the average gradient component is computed. For each node, there is a factor γ with which $h_m(x)$ is multiplied. This is the difference in the impact of each branch of the split. Gradient boosting predicts the optimal gradient for the new model, whereas classical gradient descent techniques reduce errors in the output of each iteration. The following steps are involved in gradient boosting:

- $F_0(x)$ – initializing the boosting algorithm:

$$F_0(x) = \text{argmin}_\gamma \Sigma\, L(y_i, \gamma)$$

- The loss function gradient is computed iteratively as follows:

$$r_{im} = -\alpha \left[\frac{\partial (L(y_i, F(x_i)))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)}, \text{ where } \alpha \text{ is the learning rate}$$

- Each $h_m(x)$ is fit on the gradient obtained at each step.
- The multiplicative factor γm for each terminal node is derived and the boosted model $F_m(x)$ is defined:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$$

Related Work

Scanning through the works of various researches concerning similar problems has shown that many strategies are used for sales forecasting. Regression techniques, Box Jenkins model, Holt-Winters model, Exponential smoothing and ARIMA are examples of well-known mathematical techniques. The effectiveness of these algorithms depends on the task at hand.

In one of the approaches, an insightful sales forecasting framework is created. The framework combines nine different techniques such as ARIMA, exponential smoothing, moving average, Holt-Winters, Deep Learning models including multi-layer feed-forward artificial neural network, Regression Models and SVR algorithm. These are combined using combination methodology such as boosting ensemble procedure. The results showed that this framework produces recognizable precision enhancements for the sales forecasting process in contrast with single prediction models.

Researchers have also performed sales forecasting by employing intelligent algorithms like Random forest, K-Nearest Neighbour and Gradient Boosting. While comparing the results, it was evident that the Random Forest algorithm outperformed the other two models. The Gradient Boosted model resulted in over-fitting the data-set, while the K-Nearest Neighbour performs the worst among the three.

It can be seen that many researches recommend that hybrid forecasting models perform well.

In this project sales forecasting is done using Linear Regression, Random Forest, and XGBoost, as these are the most commonly used algorithms for forecasting with proven performances.

Project Description

The entire process for the task of sales prediction is described below.

Data-set

The data-set has been acquired from the Analytics Vidhya website. It consists of 12 fields and more than 550,000 entries in total.

```
RangeIndex: 550068 entries, 0 to 550067
Data columns (total 12 columns):
 #   Column                      Non-Null Count   Dtype
---  ------                      --------------   -----
 0   User_ID                     550068 non-null  int64
 1   Product_ID                  550068 non-null  object
 2   Gender                      550068 non-null  object
 3   Age                         550068 non-null  object
 4   Occupation                  550068 non-null  int64
 5   City_Category               550068 non-null  object
 6   Stay_In_Current_City_Years  550068 non-null  object
 7   Marital_Status              550068 non-null  int64
 8   Product_Category_1          550068 non-null  int64
 9   Product_Category_2          376430 non-null  float64
 10  Product_Category_3          166821 non-null  float64
 11  Purchase                    550068 non-null  int64
dtypes: float64(2), int64(5), object(5)
memory usage: 50.4+ MB
```

Figure 4: Description of Data Set

From the above description of the data-set, it is clear that there are many missing values for some of the fields. Here, the *Purchase* field is the target variable.

Environment and Libraries

The project has been carried out in the Google Colab environment. Various Python libraries have been used in the process, like pandas, numpy, matplotlib, seaborn and sklearn.

Exploratory Data Analysis

In order to understand the relationship between different fields, many count-plots, box-plots and heat-maps have been created, all of which are listed below.

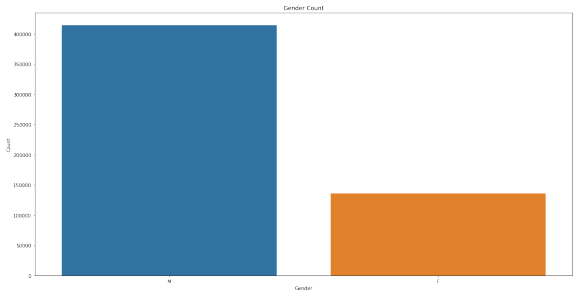

Figure 5: Count-Plot of Purchase vs Gender

This graphs shows that males have spent more than females.

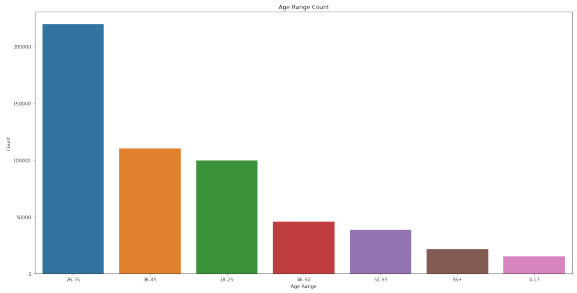

Figure 6: Count-Plot of Purchase vs Age

This graphs shows that customers belonging to the age group of 26-35 have spent the most.

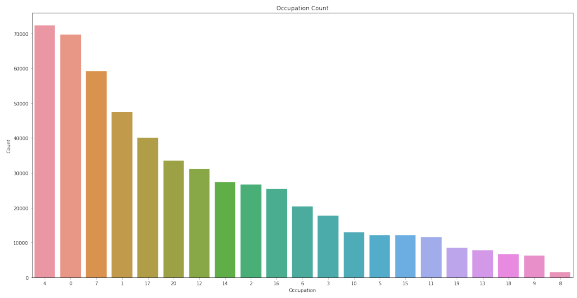

Figure 7: Count-Plot of Purchase vs Occupation

This graphs shows that customers belonging to the Occupation category 4 have spent the most.

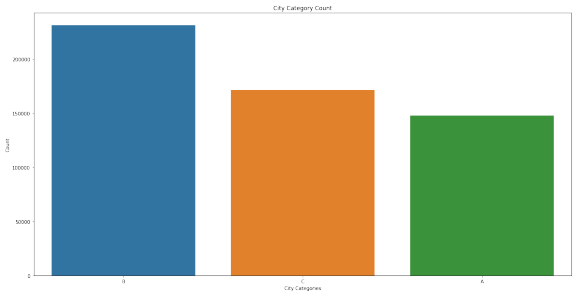

Figure 8: Count-Plot of Purchase vs City Category

This graphs shows that customers belonging to the City Category B have spent the most.

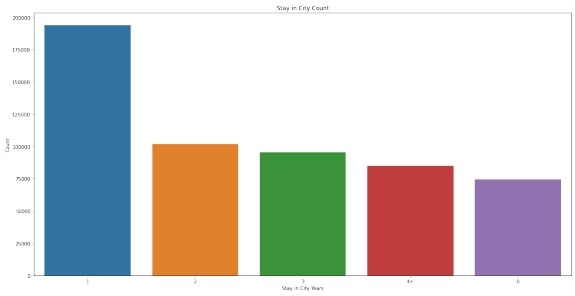

Figure 9: Count-Plot of Purchase vs Years Of Stay In Current City

This graphs shows that customers residing in their current city for 1 year have spent the most.

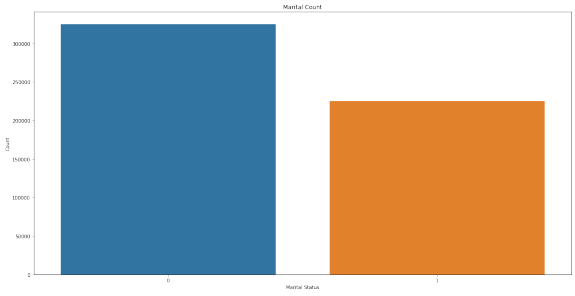

Figure 10: Count-Plot of Purchase vs Marital Status

This graphs shows that unmarried customers have spent the most.

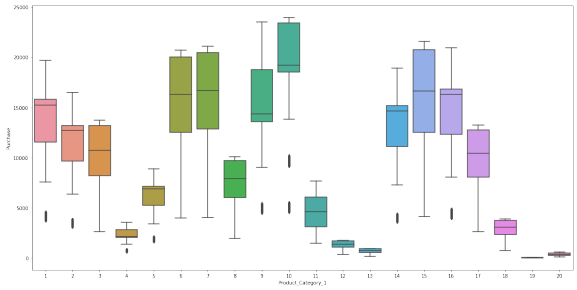

Figure 11: Box-Plot of Purchase vs Product Category 1

This graphs shows that products of category 1 has substantial impact on the purchase amount.

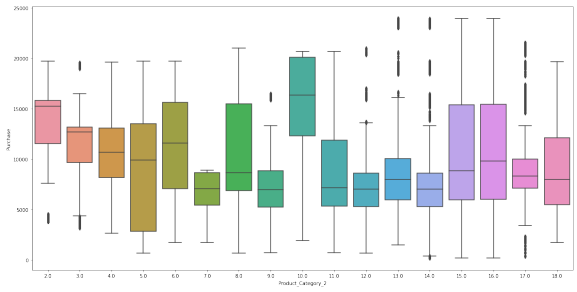

Figure 12: Box-Plot of Purchase vs Product Category 2

This graphs shows that products of category 2 has substantial impact on the purchase amount.

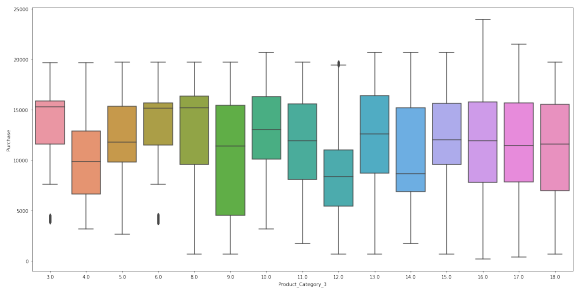

Figure 13: Box-Plot of Purchase vs Product Category 3

This graphs shows that products of category 3 has substantial impact on the purchase amount.

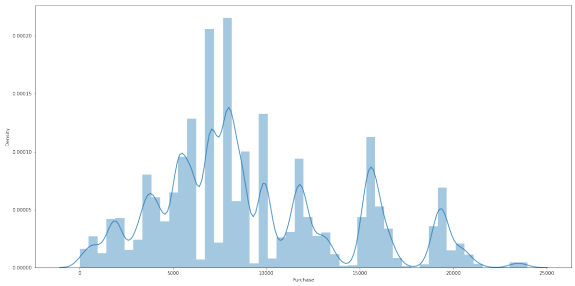

Figure 14: Purchasing Distribution

This graphs shows that the purchasing pattern is normally distributed.

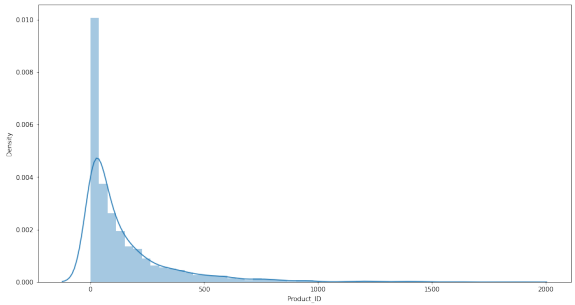

Figure 15: Product Purchase Frequency

This graphs shows that roughly 500 products are purchased more compared to the rest of the items.

Pre-Processing

The missing values have been primarily handled in three ways:

- Replace with 0
- Replace with mode of the categorical variable
- Replace with median of the categorical variable

As machine learning models do not work well with strings, for the model to work better, all the categorical variables have been label encoded by assigning a unique integer based on alphabetical ordering.

Scaling is important to bring the data in the same range when the data has a vast range. This makes it easier for the model to work with the data and find a relation between the target variable and dependent variables since data does not vary too much.

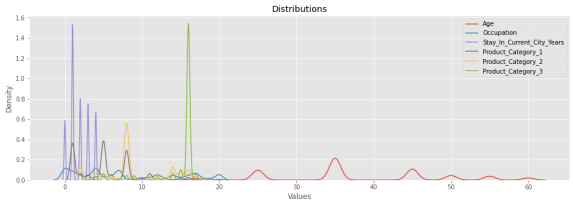

Figure 16: Before Scaling

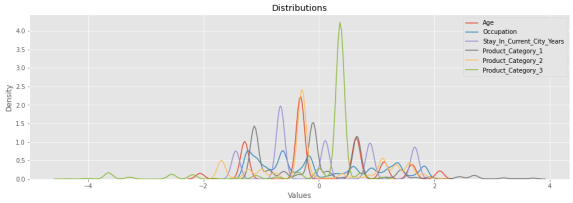

Figure 17: After Scaling

The above two graphs show the different values of the fields before and after scaling, respectively.

Based on an analysis of the fields and their relative impact on the target variable, two new fields were created in an attempt to get a better correlation to the target field. The field named *avg purchase per product* was added by averaging over the purchase amount for each product, whereas *avg purchase per user* was prepared by averaging over the purchase amount for each customer.

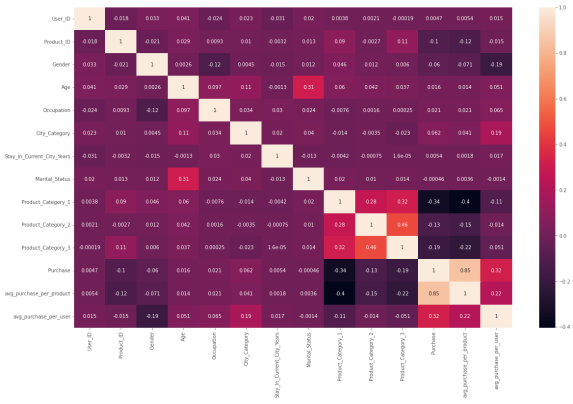

Figure 18: Heat Map

From the heat-map, it can be seen that *avg purchase per product* has a high correlation (0.85) to the target variable (*Purchase*).

Modeling

After experimenting with different train:test split ration, the best results were obtained with a split of 80:20. The three models are as follows:

- Linear Regression
- Random Forest Regression
- XGBoost Regression

are trained on the training set, and then their predictions are tested against the test set using the Root Mean Squared Error(RMSE) and the R2 score.

Experiments

Several experiments were performed for various attributes in the pursuit of better results.

- Three different ways have been experimented with for replacing the missing values (0, mode, median).
- Various combinations of existing fields were made in order to try and find better results.
- Different train:test split ratios have been experimented with.
- Different values for hyper-parameters (n estimators, random state) have been examined for the Random Forest regressor.
- Similarly, different values for hyper-parameters (n estimators, learning rate, gamma, min child weight=10, subsample, colsample bytree, max depth) have been examined for the XGBoost regressor.

Finally, the combination of values and approaches that have been selected produce the highest accuracy among different combinations. There are two performance measures that have been utilized in comparing and analyzing the outcome of each experiment. These measures are as follows:

- Root Mean Square Error(RMSE)
  RMSE is the standard deviation of the prediction errors. Error is the distance of the data points from the regression line. It is widely used in regression analysis to verify experimental results. There are many advantages of using RMSE over other measures. The prediction values that are generated, have the same unit as the required output variable, which makes interpretation of loss easy. Since the errors are squared before they are averaged, the RMSE penalises the larger errors heavily by assigning them relatively higher weights.
  Formula for the RMSE metric is given as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

  where, N is Total Number of Observations.
  When actual observations and forecasts are passed to the RMSE as inputs, there is a direct relationship with the correlation coefficient. For instance, if the correlation coefficient is 1, the RMSE will be 0, as all the points lie on the regression line resulting in no errors.
  Simply put, the less the RMSE value, the better is the performance of that model.
- R-squared (R2) Score R-squared is a statistical measure that represents the goodness of fit of a regression model.

The ideal value for r-square is 1. The closer the value of r-square to 1, the better is the model fitted. It is a comparison of the residual sum of squares (SS_{res}) with the total sum of squares (SS_{tot}). SS_{tot} is calculated by summing over the squares of perpendicular distance between data points and the average line. SS_{res} is calculated by summing over the squares of perpendicular distance between data points and the best-fitted line.
The R2 score is calculated using the following formula:

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \overline{y})^2}$$

In other words, closer the r-square values is to 1, better is the model.

Performance Comparison and Analysis

The following table shows the RMSE performance matrix for each of the models.

Handling Missing Values	Linear Regression	Random Forest	XGBoost
Replacing with 0	2568.2658	2528.7132	2456.0697
Replacing with mode	2568.5666	2527.8529	**2453.2947**
Replacing with median	2568.6121	2528.5021	2454.5938

Figure 19: RMSE Performance Matrix

Here, we can see that the XGBoost performs better than the other two models overall. This is reflected by the fact that the RMSE value is the lowest for the XGBoost for all three cases. It performs relatively better when the missing values are replaced with the mode of the field.

The following table shows the R-squared performance matrix for each of the models.

Handling Missing Values	Linear Regression	Random Forest	XGBoost
Replacing with 0	0.74	0.75	**0.76**
Replacing with mode	0.74	0.75	**0.76**
Replacing with median	0.74	0.75	**0.76**

Figure 20: R-squared Score Performance Matrix

From the performance matrix, it is evident that XGBoost performs better than the other two models overall. This is reflected by the fact that the R-squared score is closest to 1 for the XGBoost for all three cases.

In general, it is safe to say that the XGBoost regressor outperforms the other two models on all occasions. Hence, it can be regarded as the best among the three for the task at hand.

Conclusion

The research concludes that machine-learning based intelligent sales prediction systems are the need of the hour for companies to handle large volumes of data for forecasting sales and customer purchase patterns. The Sales forecasting process is very complex given that a lot of factors need to be considered. The machine learning approaches mentioned in this research paper may prove to be effective in data tuning and decision making. In this study, more than 550,000 records are used for the comparison of algorithms. Since there were many missing values, different methods were employed to handle those missing values during the analysis phase. XGBoost performed better overall, especially when the missing values were replaced with the mode of the field, as it had a lower RMSE value and a higher R-squared score than the other two models. It is possible that getting more data would enable the models to learn even better and perform well, increasing the predictive power of the models. The number of fields and attributes used in the study may be restrictive, and the availability of more attributes may have been fruitful for further analysis. Moving forward, Big data can be employed for predictive analysis in sales forecasts to reduce the processing time and handle large data-sets. Big data analysis and forecasting are essential areas in today's economy.

References

[1] Analytics Vidhay
https://datahack.analyticsvidhya.com/

[2] K. Saraswathi, N. T. Renukadevi, S. Nandhinidevi, S. Gayathridevi, and P. Naveen, "Sales Prediction Using Machine Learning Approaches", *AIP Conference Proceedings Volume 2387, Issue 1*, 2021.

[3] Akshay Krishna, Akhilesh V, Animikh Aich, Chetana Hegde, "Sales Forecasting of Retail Stores using Machine Learning Techniques", *3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, 2018.

[4] Shreya Kohli, Gracia Tabitha Godwin, Siddhaling Urolagin, "Sales Prediction Using Linear and KNN Regression", *Advances in Machine Learning and Computational Intelligence*, pp 321-329, 2020.

[5] Wei-Yin Loh, "Classification and Regression Trees", *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 1, no. 1*, pp. 14-23, 2011.

[6] Sunitha Cheriyan, Shaniba Ibrahim, Saju Mohanan, Susan Treesa, "Intelligent Sales Prediction Using Machine Learning Techniques", *2018 International Conference on Computing, Electronics Communications Engineering (iCCECE)*, 2018.