Hyperparameter Tuning for the Prediction of Customer Revenue

MSc Research Project
Data Analytics

Rutuja Dinesh Mehta
Student ID: x20129751

School of Computing
National College of Ireland

Supervisor: Dr Giovani Estrada

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Rutuja Dinesh Mehta
Student ID:	x20129751
Programme:	Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Dr Giovani Estrada
Submission Due Date:	15/08/2022
Project Title:	Hyperparameter Tuning for the Prediction of Customer Revenue
Word Count:	XXX
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	19th September 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Hyperparameter Tuning for the Prediction of Customer Revenue

Rutuja Dinesh Mehta
x20129751

Abstract

Owning to cutthroat competition, the business ecosystem is paying increasing attention to customer relationship management (CRM). To build successful customer retention strategies, it is essential to optimize their lifetime value. Building an analytical data model that integrates financial, marketing, and advertising data to forecast the return on advertising across many scenarios is vital. The model should depict the revenue flow resulting from customer interactions at the top of the funnel by including the early marketing expenditure. Inspired by a recent Kaggle competition in 2019, we set out to identify profitable customers and predict their revenue. A range of Machine Learning and Deep Learning models were implemented and evaluated their performance based on the RMSE. We will show how the proposed methodology allows us to obtain better results than the winners of the Kaggle competition. Light Gradient Boost Model (LGBM) performed better than the rest giving the RMSE value of 0.95 after tuning the hyperparameter using the Random Search Cross-Validation technique. The model gives excellent results for classifying low and high-revenue generating customers, making a guideline for sale and marketing strategies.

1 Introduction

In recent years, there has been a massive pressure amongst companies to thrive in the business environment. This leads to a difficulty in knowing what actual contribution of a customer is to a company. Every organisation needs to recognize the true value of the customer from the customer's transaction data for establishing an apt customer relationship management (CRM). The concept of Customer Life-time value (CLV), established by Kotlar in 1974, is more widespread than CRM which is defined as "the as the recent value of the customers purchase over the life". As said by Wang and Liu, Customer Lifetime Value is an integral part of Strategic Customer Relationship Management. According to Pareto principle, only 20% of the customers are responsible for generating the 80% profit. After knowing the customer's value, it becomes easy for the company to implement the marketing strategies which would be more focused towards the homogeneous group of customers. According to Ohrimuk and Bezrukov, retention of the customers value adds more to the organisation than getting in new customers. In order to achieve the customer life-time value, it is essential to identify the profit generating customers and predict the revenue generated by them. Various Machine Learning and Deep Learning methodologies were used for the prediction by tuning the hyper-parameters. These data-driven tactics make suggestions based on user behavior, which assists in determining the marketing strategy and campaign planning. It becomes easier to achieve the long-term goals by implementing an intelligent customer success management model.

Research Questions:

- What impact do Machine Learning and Deep Learning techniques have in identifying the profit-generating customers and predicting their revenue?
- What is the most effective way to introduce hyper-parameter tuning into this research?

This study has been inspired by the Kaggle competition[1] organised in 2019, that asked participants to analyze a customer dataset from the Google Merchandise Store (commonly known as GStore, where Google swag is sold) and forecast revenue per customer. The study developed an intelligent customer success management model using machine learning and deep learning by tuning the hyper-parameters to generate maximum profit and retain customer lifetime value to survive in the cutthroat competition of the business. The research gave the solution to the presented question as mentioned earlier that focused on recognizing the valuable customers and predicting the revenue generated by them. This would help discover profitable customers, identify their niches, handle difficult customers and captivate the best ones. This would analyze the work of the salesperson to manage the operational cost and shape the focus of the organisation towards the customers.

The rest of the paper is organized as follows: Section 2 addresses the previous related work for understanding the domain, different methodologies, and techniques of Machine Learning and Deep Learning Models and their applications. The section also covers how hyper-parameter tuning helps in increasing the model performance. Section 3 describes the proposed methodology in depth. Further, Section 4 overviews the architecture and the various models used, proceeding with section 5, which gives a glimpse of the implementation of different Deep Learning and Machine Learning models. Section 6 highlights the findings and the results of the research. And lastly, section 7 gives a conclusion of the study and its future aspects.

2 Related Work

Many researchers have made significant contribution in predicting the revenue generated by the customers and optimizing their life-time value. Many surveys, studies, and articles employ various methodologies of machine learning, deep learning, neural networks, etc. in identifying profitable customers. This section would cover the reviewed work and compare against their advantages and disadvantages from the view of recognizing the targeted profit generating customers and their life-time value which would help to tackle these difficulties and make an improvement in the outcome of this research.

2.1 Identification Of Customer Behavior and Transaction Pattern

The research paper by Lv shows a study of a business that dealt with several customer relationship management issues. In order to strengthen its management and increase

[1] Google Analytics Customer Revenue Prediction https://www.kaggle.com/competitions/ga-customer-revenue-prediction/overview

the effectiveness of its operating system, a company must be familiar with large data sets and the CRM team's analysis of them. It also needs to know how to use information technology to achieve various platforms for the advancement of modern technology. Establishing the ideal management team, developing a perspective on data analysis, bolstering the information investment, and adhering to the "customer-centered" concept were among the steps taken to improve CRM.The study's goal was met, however the following articles indicate improvements that may have been made to the way data mining technologies could have been employed in CRM to acquire insightful business information.

Data mining in CRM and its application were the subject of study by Song and Liang with the aim of retaining current clients and attracting new ones to the company in order to boost sales. Large datasets were mined using data mining technologies to extract knowledge and information that might be utilized to convert a business from being product-centered to being customer-centered. For the purpose of delivering high-quality services to clients and lowering company costs to enhance business operations and profitability, the CRM was divided into operational, analytical, and collaborative CRM. Data mining was utilized to find new consumers, detect buying trends, and provide cross-marketing advice in order to increase sales. The study that is presented in the following research paper would have been improved by an investigation of the methods and algorithms that could be employed to attract niche markets and make money from them.

Predicting deal closing based on customer browsing behavior in sales content was presented by Nurbakova and Saumet. They used machine learning techniques to create accurate profit forecasts for automatically collected data and sent objective signals to his CRM system. Random forest and generic models were used with Double models forming two local groups of signable and unsignable models. Due to the class imbalance in the subset, the results for the signable items were higher than those for the unsignable items. The prediction accuracies of the random forest and generic models were similar, around 87%. Surprisingly, all five models performed much better predicting the positive class than the negative one. This research lags behind a detailed estimation of the traits that explain customer browsing patterns to improve the accuracy of optimistic predictions. These patterns are highlighted in the following research paper by generating user transaction sequences.

A dataset of 3700 records was used by Predicting Customer Behavior Using a Random Forest Algorithm Ghosh and Banerjee. Factors that influence purchasing behavior include transaction history, geographic location, ads viewed, and many other social and personal decision-making factors. A random forest algorithm was used to predict purchasing behavior and provide suggestions to customers based on past transactions. The model achieved 87% accuracy in predicting customer purchasing behavior. Larger datasets with more methods helped to obtain higher precision used in subsequent studies.

To better comprehend consumer behavior and trade patterns, Doan and Keng applied a cutting-edge methodology. Customer modeling can be challenging because it is a multifaceted and time-varying problem. A supervised learning technique with a recurrent neural network that considered transactional data sequences was used to predict the likelihood of a customer adding to their shopping cart in the next week using a Generative Adversarial Network (GAN). A Long-Short Term Memory (LSTM) was trained to predict whether it is the final product in the shopping cart and to predict the product category and price of the next product. For each customer he generated a shopping cart for 5 weeks. The resulting method replicated the statistics generated from the actual data distribution.

Researchers Parikh and Abdelfattah aimed at how to use RFM analysis to predict customer buying patterns and evaluate high-value customers. Basically, three aspects were considered to identify purchasing patterns and high-value customers for higher profits. This includes when, how often, and how much the customer has spent most recently transacting. RFM analysis with four clustering algorithms (DBSCAN, Mean-Shift, K-Means, and agglomerative clustering) was used to gain insight into the segments of customers ranging from highest to lowest valuable customers.

Text mining was used by Rotovei and Negru to manage difficult commercial sales decisions using models including artificial neural networks, random forests, and support vector machines that incorporated a lot of emotional elements. They also discuss the possibility of using data from B2B CRM systems to predict sales. The accuracy rose from 85% to 89% with the use of RF methis. It was discovered that the emotive potential that were recorded here aid in more precisely anticipating complicated transactions. Based on an intelligent study of consumer behavior patterns, Monastyrskaya and Soloviev conducted research on advances in customer relationship management. In order to develop better marketing campaigns that take into account user considerations and commercially viable KPIs, the article examined the regression and clustering approaches for looking for patterns in consumer behavior. They were able to anticipate the customer cost of acquisition with accuracy (cost-per-action, CPA). As a consequence, the budget allocation guidelines and marketing strategies were adjusted to better draw in the intended audience.

2.2 Proposed Techniques for Forecasting Sales and Retaining Customers

Walmart product sales were forecasted for approximately 3000 products in ten different Walmart stores by **?**. For effective mining of attributes across dimensions, he proposed using the XGBoost algorithm. Along with feature engineering, this paper also implements memory compression, feature extraction, and statistical feature selection. Comparing XGBoost's results with Logistic regression and Ridge method, which had 79 and 76 percent accuracy, respectively, also showed that the latter was much better than the former. Compared to the other two, XGBoost had a less accurate accuracy of 55%, but it performed better with feature engineering. The following paper illustrates that, even though the models provided accurate results, they could have given higher accuracy.

Classical forecasting models were compared with advanced technology forecasting models in a study by Wang and Liu for specific product categories, resulting in higher accuracy. Using thousands of historical data, they compared typical methods to various machine learning methods based on criteria such as execution time, cost, prediction accuracy, ability to generalize, and overall performance evaluation. The above five score indices were used to compare the results of ARIMA models, Support Vector Machine (SVM) algorithms, Long Short-Term Memory (LSTM) algorithms, and Recurrent Neural

Networks (RNN). The best accuracy was 99% for SVM for perishable products and 99% for LSTM for non-perishable products. Overall, SVM and LSTM proved to be superior in task prediction.

The authors of the research Mou and Tian offer time series prediction utilizing EMD and Deep Learning approaches. Using the "split and conquer" strategy, the method converts a real-time issue into a signal analysis. It can decompose source time series for Deep Learning models using the time series feature. For the purpose of forecasting two factors—consumption frequency and amount—a number of models, including KNN, RT, GDBT, XGBoost, MLP, EMD-GBT, EMD-M, and EMD-D were compared to the RSME and MAE indicators. It was determined that the EMD decomposition does not combine with all machine learning algorithms when forecasting a time series problem and that the performance of GBDT and MLP is superior to that of EMD-MLP and EMD-GBDT.

According to Chen, a neural network model was used to estimate the sales volume of an online retailer. The LSTM neural network's adaptive genetic algorithm (AGA) was applied to optimize network parameters including time step, number of hidden layers, and number of training cycles in order to improve forecasting accuracy for items type and overall sales volume. The MSE value of the AGA-LSTM model is 47% and 43% lower, respectively, when compared to the average MSE values of the BP and LSTM models. AGA-LSTM exhibited a higher prediction accuracy for consumer items, but as the number of prediction steps rises, performance accuracy falls. Additional strategies, like sentimental analysis or a single-step prediction model, would be beneficial.

Three different approaches; Holt-Winters exponential smoothing, ARIMA (Autoregressive integrated moving average), and neural network auto regression model—were used to predict the sales pattern in the sales forecast for Amazon based on historical data by Singh and Sharma. The information provides financial performance information from 2005 to 2018. The ARIMA model had the highest accuracy, with a MAPE value of 3.46, according to the evaluation's use of MAPE. Using a variety of ML and Deep Learning approaches, the study by Zhu provided the probability of purchasing things using historical data. The ensemble learning model is proposed employing various segments that utilize CNN, Random Forests, and XGBoost after pre-processing and cleaning the data. Evaluation was conducted by assessing the F1 score, specificity, sensitivity, precision, and other metrics. Later, examination of efficiency and complexity was also reviewed. The ensemble technique was used with XGBoosting using Base estimator RFC, XGBoosting using Base estimator CNN, and XGBoosting using Base estimator CNN + RFC. When the output from each block was fed into the ensemble classifier, the highest accuracy of 88 percent was reached.

Finding the Customer Lifetime Value (CLV), which aids in determining which customers need to be invested in to acquire a long CRM, is essential for developing a successful CRM strategy. This research was conducted by Win and Bo. The CLV model makes it possible to calculate the dollar value of consumers who made purchases throughout the course of their relationship with the company and to determine their worth. It also aids in problem-solving when it comes to choices about segmentation, customer acquisition, customer retention, and marketing tactics. Around ten thousand transactions' worth of data were utilized for training with the Random Forest approach, and random search tweaking was done to acquire the best accuracy. When compared to the other three RF models, Random Search's ideal hyperparameter had the best projected accuracy of 84 percent. As seen in the next example, other strategies might be employed to obtain more accurate findings.

2.3 Summary of Related Work

This study work, which demonstrates an original and improvised approach was motivated by the shortcomings in existing implementation and the limited amount of research on the current issue. Below is the summary of the related work.

Table 1: Summary of Related Work

Paper Name	Year	Models	Dataset	Evaluation	Improvements
Walmart Sales Forecasting using XGBoost algorithm and Features	2020	XGBoost Model	Kaggle	RMSE: 0.652	Use of larger dataset since this consisted of only 3000 entries
Study of Supervised Algorithms for Solve the Forecasting Retail Dynamics Problem	2020	Linear Regression,KNN Regressor & Random Forest	Kaggle	Accuracy: 94%	Prediction of the target variable more Accurately
Predicting purchase probability of retail items using an ensemble learning approach and historical data	2020	CNN+XGB +RFC, CNN+XGB, RFC+XGB	Web scraped	Accuracy: 88.84%	Use of feature engineering or hyperparameter tuning to archive more accuracy
Data-drivenn sales prediction using communication sentiment analysis in B2B CRM systems	2019	Support Vector Machines and Random Forest	Database extraction	Accuracy: 89%	Additional B2B data sets, use incremental learning and deploy the findings in a live CRM system

3 Methodology

The aim of the research is to identify the profit generating customers and forecast the revenue generated by them for analysing and establishing the marketing strategies so as to retain the customers for the sustainable growth of the business. The description to the steps being used for this research question has been briefed in this section. The methodology used here combines KDD and CRISP-DM, the advantage being that the iterative nature of KDD and the agility to move the phases back and forth of CRISP-DM helps build a more effective model. The key principles of the research methodology are illustrated in Figure 2.

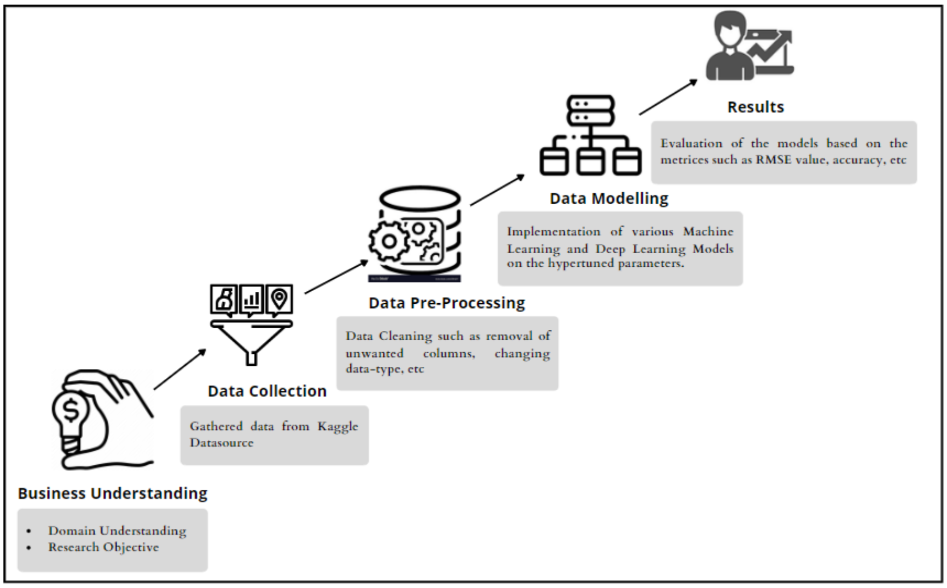

Results
Evaluation of the models based on the metrices such as RMSE value, accuracy, etc
Data Modelling
Implementation of various Machine Learning and Deep Learning Models on the hypertuned parameters.
Data Pre-Processing
Data Cleaning such as removal of unwanted columns, changing data-type, etc
Data Collection
Gathered data from Kaggle Datasource
Business Understanding
• Domain Understanding
• Research Objective

Figure 1: Methodology

• **Business Understanding :** It is essential to comprehend the research's scope and it's domain at first. The literature study helped to acquire the business knowledge that would assist the businesses in generating maximum profit and enforce good marketing strategies.

• **Data Collection :** The dataset was collected from the competition organized by Kaggle. Two variations of the ”train” and ”test” dataset were included; one had 25.5GB, while the other had 1.5GB. The characteristics that did not impact the predictions were deleted, resulting in a dataset size of 1.5 GB.

• **Data Pre-Processing :** The features of the raw dataset were of the json format, using the json_normalize function the json columns were flattened into a normal dataframe at first. Then, the dataset was treated for the missing values, irrelevant features were dropped and the all transaction records of a single user were grouped into one.

• **Data Modelling :** For establishing the better model performance, the hyperparameters were tuned using Random Search technique and various Machine Learning methods such as Random Forest Regressor, Gradient Boosting Regressor, Light Gradient Boosting Regressor Method, XGBoost Regressor and Stacked Regressor fed to the Meta Regressor were used for prediction.

• **Results :** The dataset includes time series data, and as regression models are used to make predictions, performance will be assessed using the Root Mean Square Error (RMSE).

4 Design Specification

The proposed framework has the architecture divided into three segments; the Data Tier, Application Tier and the Presentation Tier. As depicted in the architecture, data tier has the data sourced from Kaggle and converted the json columns into the tabular format. In the same data tier, the data was treated with the missing values, irrelevant features were removed and the datatypes were changed which is a part of cleaning and pre-processing step. The application tier performed the exploratory data analysis of various features for deriving the notable conclusions. Hyper-parameters were tuned using the Random Search Cross-Validation method. And, the tuned parameters were trained by various Machine Learning algorithms such as LGBM Regressor, Gradient Boosting Regressor, XGBoost Regressor, Stacking Regressor, etc. These models were evaluated on the basis on Root Mean Square Error (RMSE) value in the presentation layer.

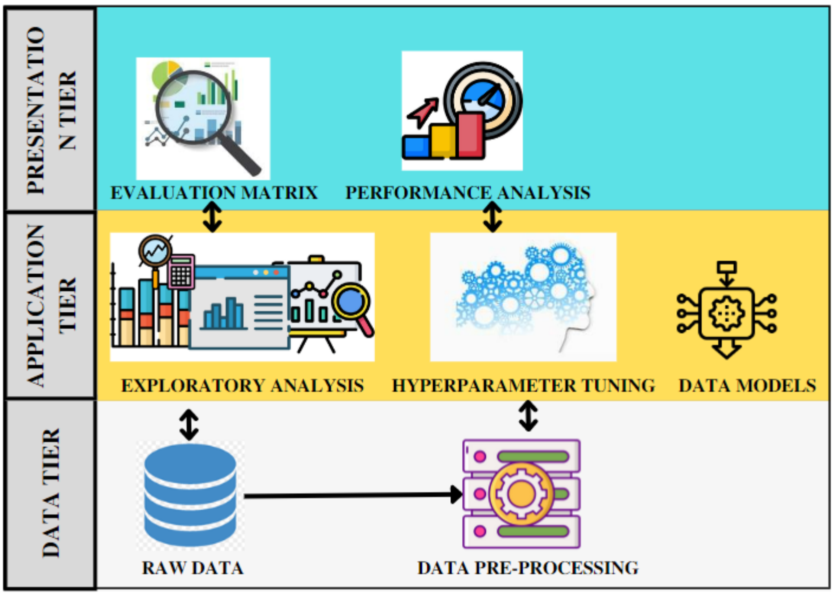


Figure 2: 3-Tier Architecture Implementation

The research also centered on implementing the Stacking Regressor, an ensemble learning technique for combining different regression problems through a meta-regressor as shown in figure 4. In this case, LGBM Regressor, Gradient Boosting Regressor, and XGB Regressor are used, and the LGBM Regressor acts as the meta-regressor. These models are trained individually and then fed to the meta-regressor based on their output for the final prediction.

5 Implementation

The project aims to create an intelligent customer success management model that uses deep learning and machine learning to maximize profit and maintain client lifetime value. This section of the research provides a detailed analysis of implementing the Machine Learning and Deep Learning models used to achieve customer success under the Python

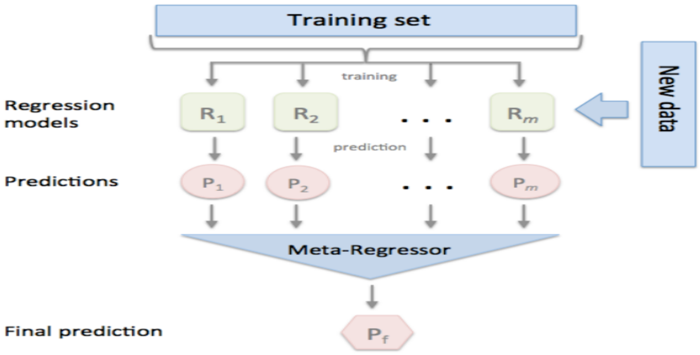


Figure 3: Stacked Regressor Architecture

environment. The local computer is powered by a DELL Vostro 3500 with 8GB of RAM, an 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz, and 2GB of graphics.

5.1 Data Collection and Data Transformation

The dataset for this research has been taken from the competition organized by Kaggle. The data had 9,03,653 entries and 12 columns, out of which four columns ('device', 'geoNetwork', 'totals', 'trafficSource') were of the JSON format. The JSON columns were converted into the tabular format using the json_normalize function. After the JSON columns were flattened, the dataset consisted of 55 columns where the user's information about the device, geography, transactions, etc., was obtained.

5.2 Exploratory Data Analysis

The fundamental purpose of the exploratory data analysis (EDA) is to help glance at the data before any assumptions are made. It identifies the manifest errors, understands the pattern of the data, and detects the outliers, anomalous events, or any interesting relations amongst the dependent and independent variables. The exploratory data analysis (EDA) performed on the target variable, revenue-generating customers, device category, geo_network category, traffic_source category, totals category, and the date is depicted below.

- **EDA of the target variable**

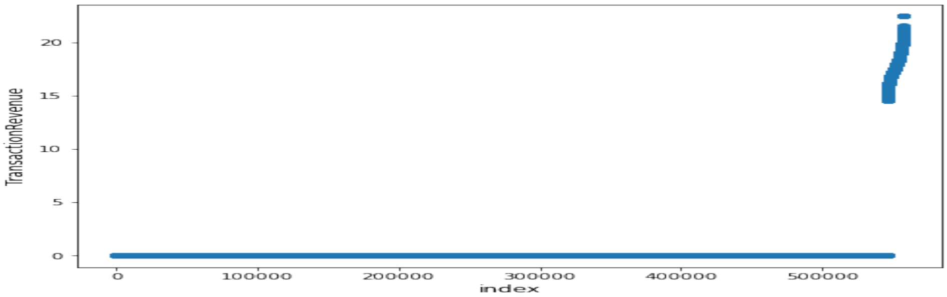


Figure 4: Distribution of Target Variable

The above distribution of the target variable "totals.transactionrevenue" was to check whether the Pareto principle was followed. The graph depicts the imbalance in the dataset where 98% of the records belong to single class.

- **EDA of the Device Category**

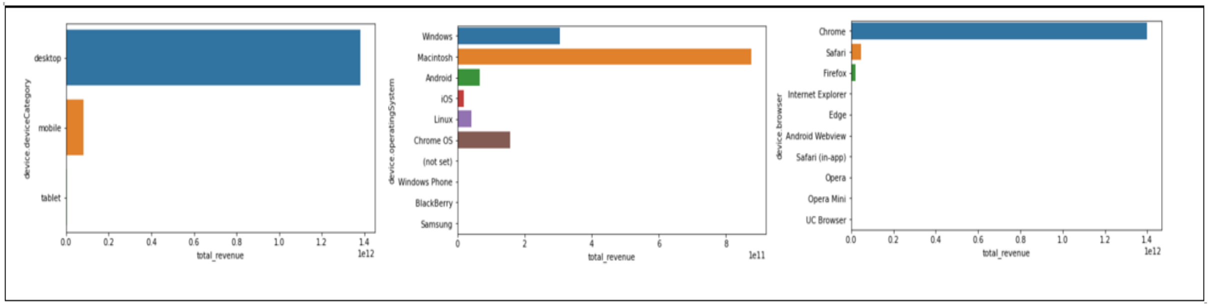


Figure 5: EDA of the Device Category

Above is the exploratory analysis of the device category, including the browser, device type, and operating system. Chrome users are the primary source of revenue generation. Though Firefox users have more mean value, revenue generated is less as compared to Chrome. For the type of device, desktop users generate significant revenue compared to mobile and tablet users. The Windows users are more, but the revenue is obtained from Macintosh users. To sum up, Macintosh, Windows, and Chrome OS contribute to generating the maximum revenue.

- **EDA of the Geonetwork Category**

The "geoNetwork" category includes the exploratory analysis of the continent, subcontinent, country, and network domain. Asia and Europe have a decent number of users, but the revenue generated is almost NIL. Northern America is the clear winner in terms of revenue. The maximum revenue is obtained in America, where the number of users is also high. The countries Canada and Germany have the users, but no revenue is generated from them.

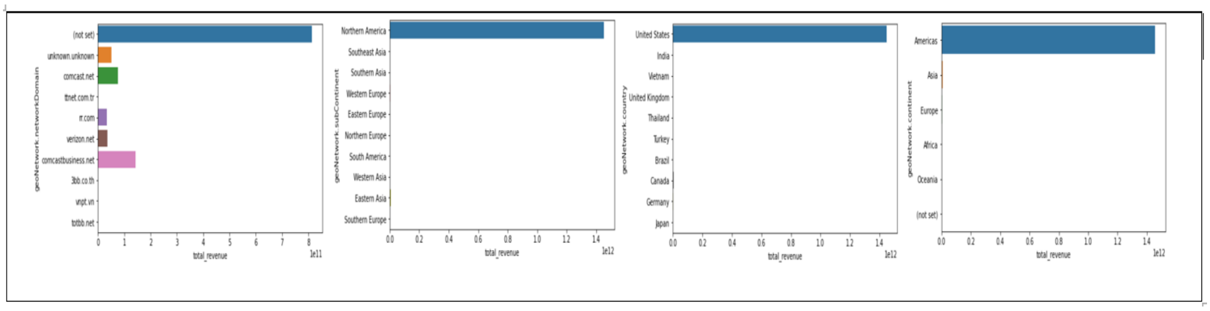


Figure 6: EDA of the Geonetwork Category

- **EDA of the Traffic Source**

Though the "direct" source has the highest number of users, maximum revenue is generated by "mall.googleplex.com". On the traffic source medium, "referral" has more non-zero revenue count than the "organic" medium.

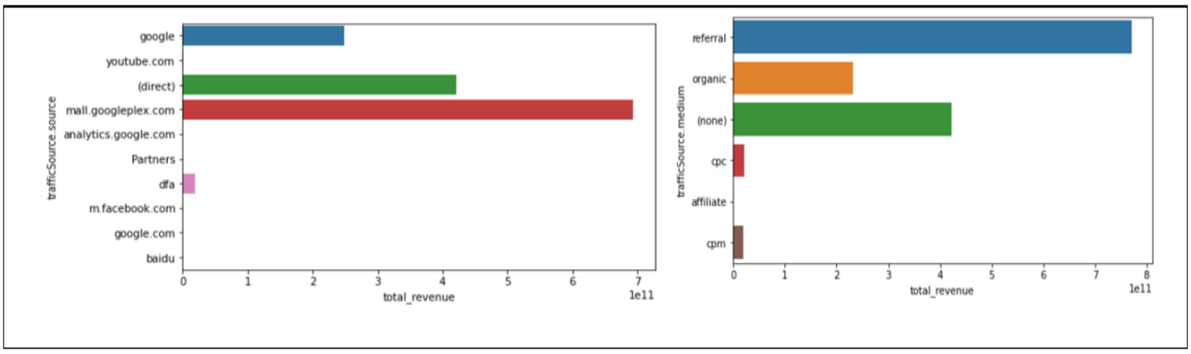


Figure 7: EDA of the Traffic Source

- **EDA of the Total Hits and Pageviews**

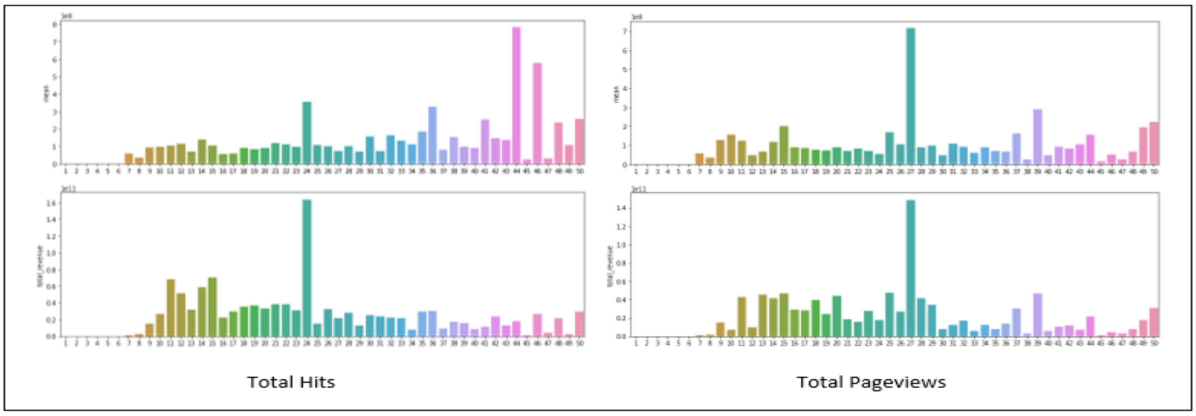
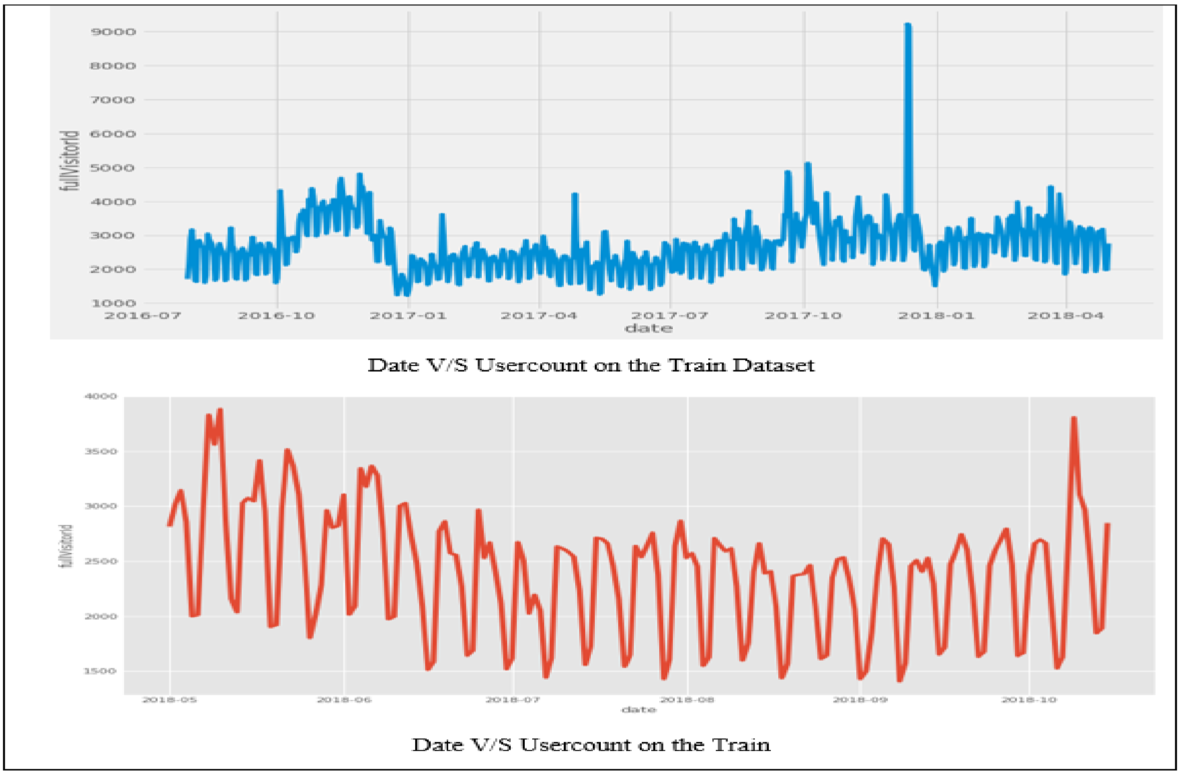


Figure 8: EDA of the Total Hits and Pageviews

The graphs, "totals.hits" and "totals.pageviews" look similar. From the plot, it can be derived that the chances of generating revenue are on the higher side if the visitor sees more number of pages. The same applies to the number of "hits". Less the "hits", less the revenue. It shows a clear trend that the revenue increases when the "hits" and "pageviews" increase.

- **EDA of the Date versus Usercount on the Train and Test Dataset**

Various features were extracted from the date feature, such as day, week, month, and year. The training dataset consists of the data from 1st Aug'16 to 30th Apr'18, and that of test data consists from 1st May'18 to 15th Oc'18. A considerable spike in the customers was seen from Nov'17 to Dec'17, and the number of customers reduced over the period.



Date V/S Usercount on the Train Dataset

Date V/S Usercount on the Train

Figure 9: EDA of the Date versus Usercount on the Train and Test Dataset

5.3 Data Cleaning and Pre-processing

- **Treating the missing values :** The missing values in the dataset were dealt separately for the categorical and continuous features. Target encoding was used for the categorical feature and each value was replaced with the mean value of the target variable.

- **Removal of unnecessary columns :** Out of the 55 features in the dataset, 12 features has the missing values more than 50% which were removed from data. Along with this, 18 single feature variables were also dropped from the dataset.

- **Grouping the records :** For the given time period, a single user had the multiple visits so, there are multiple records of a given user. The aggregation operation such as mean, mode, maximum and minimum was applied with respect to the ”fullVisitorId” which resulted in the single record per visitor.

5.4 Model Building

The research would be classified into two problems: phase one for identifying the customers that generate profit and phase two predicting their revenue. The models implemented for the solutions to the above-stated problems are discussed below.

5.4.1 Prediction of the Revenue Generated by the Customers

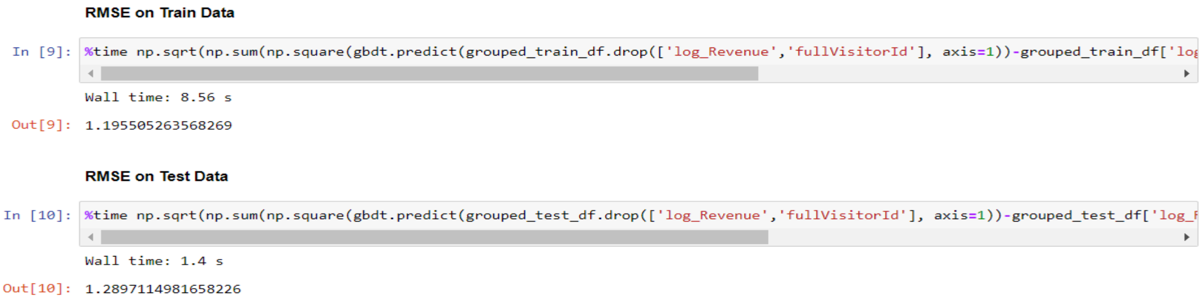
The problem was classified into a classification problem depending on whether the profit is generated from the customers or in other words, profitable and non-profitable customers contributing to the revenue generation. The data was reverted to the initial transaction level data for model training. The results of this problem were incorporated as an independent feature to the feature set. For capturing the seasonality and trend of the dataset, the time-stamp was splitted into day of the week, hour of the day and day of the month for capturing the seasonality in the data. Then, the models were trained using Machine Learning and Deep Learning Algorithms.

Hyper-parameter tuning, where a set of optimal hyper-parameters are chosen for the learning of the models. The combination of hyper-parameters helps maximize the model's performance, and reduces the predefined loss function for producing better results with minimal errors. Here, the Random Search method is used where the hyper-parameters are selected randomly, and a model has trained accordingly. Sci-kit learn (sklearn) library is used for its implementation. Cross-validation is done in a Random Search and each model is trained after obtaining the best suited hyper-parameter.

1) Gradient Boosting Regression Method :

"Sklearn.ensemble.GradientBoostingClassifier" is used in Python 3.6 to do gradient boosting. Boosting is a method for creating compositions in which basic algorithms are added one after the other in a sequential fashion. Every new algorithm that comes after that is built to fix the flaws of the previous composition. Due to the directed nature of composition creation in boosting, simple fundamental methods are sufficient.

Evaluation And Results :



RMSE on Train Data

```
In [9]: %time np.sqrt(np.sum(np.square(gbdt.predict(grouped_train_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_train_df['log

        Wall time: 8.56 s

Out[9]: 1.195505263568269
```

RMSE on Test Data

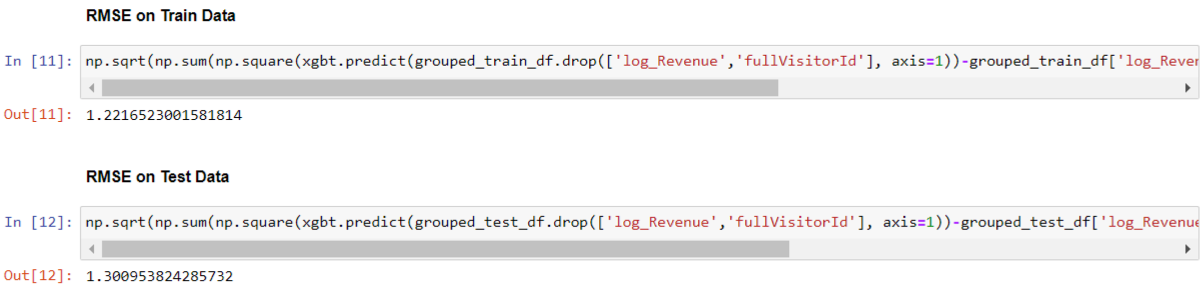
```
In [10]: %time np.sqrt(np.sum(np.square(gbdt.predict(grouped_test_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_test_df['log_F

         Wall time: 1.4 s

Out[10]: 1.2897114981658226
```

Figure 10: RMSE Value on Train and Test Data for Gradient Boosting Regressor

Using the Random Search technique, the Gradient Boosting Regressor model was trained with tuned hyper-parameters. For this model, the best hyper-parameters were subsample size of 0.7, number of estimators as 400, minimum samples of leaves as 3, learning rate of 0.1 with the maximum depth of 5. It gave the RMSE value of 1.19 on the train data and 1.28 on the test data with the overall computational time of 5hrs 48mins.

2) Extreme Gradient Boosting (XGBoost) Method :

XGBoost trains and combines the individual models for getting a single prediction. The validity of the XGBoost Regressor can be known after inferring the objective function



RMSE on Train Data

```
In [11]: np.sqrt(np.sum(np.square(xgbt.predict(grouped_train_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_train_df['log_Rever
Out[11]: 1.2216523001581814
```

RMSE on Test Data

```
In [12]: np.sqrt(np.sum(np.square(xgbt.predict(grouped_test_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_test_df['log_Revenue
Out[12]: 1.300953824285732
```

Figure 11: RMSE Value on Train and Test Data for XGBoost Regressor

and the base learners. Loss function and the regularization term is contained in the objective function, it tells us how far the real values are from the model results. The worst predictions are canceled out and the better one comes up with final good predictions.

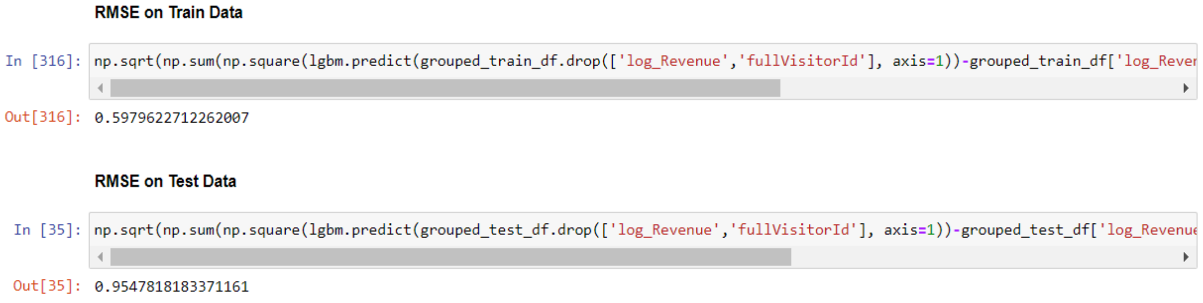
Evaluation And Results :

By using the Random Search technique, we trained the XGBoost Regressor model with the tuned hyperparameters. This model yielded the best results when the subsample size was 0.8, reg_lambda and reg_alpha were 0.5 and 1, the number of estimators was 200 with the learning rate of 0.1. The RMSE on the train data was 1.22 and on the test data, 1.30, with the least overall computational time of 1hr 50mins compared to other models.

3) Light Gradient Boosting Method (LGBM) :

Light Gradient Boosting Method adds automatic feature selection and focuses on boosting algorithms with larger gradients. This drastically speeds up the training of the model and improves the predictive performance. A synthetic regression problem is created with ”n” number of sample samples and input features. The performance is evaluated using the repeated k-fold cross-validation method.

Evaluation And Results :



RMSE on Train Data

```
In [316]: np.sqrt(np.sum(np.square(lgbm.predict(grouped_train_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_train_df['log_Rever
Out[316]: 0.5979622712262007
```

RMSE on Test Data

```
 In [35]: np.sqrt(np.sum(np.square(lgbm.predict(grouped_test_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_test_df['log_Revenue
Out[35]: 0.9547818183371161
```

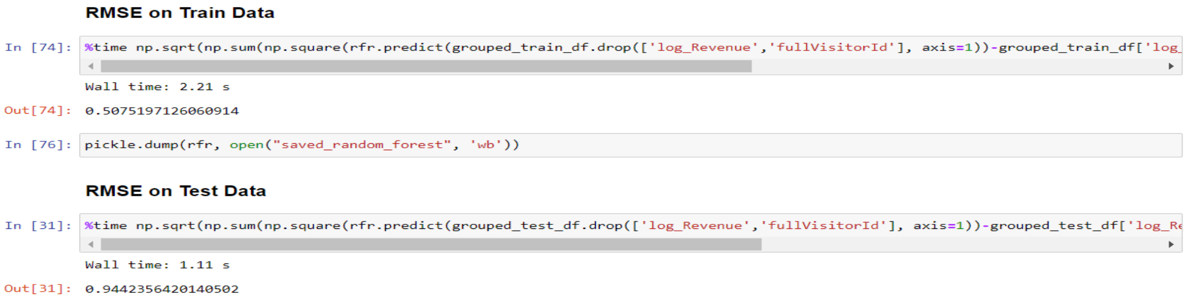
Figure 12: RMSE Value on Train and Test Data for LGBM Regressor

The LGBM Regressor model was trained with the tuned hyperparameters using the Random Search technique. The best hyper-parameter suited for this model came out to be when subsample size of 0.7, reg_lambda and reg_alpha of 1 and 0.5 respectively, number of estimators as 350, number of leaves as 100, with the learning rate of 0.1 having minimum 1 child was obtained. With the least overall computational time of 6mins 54s as compared to other models, it gave the RMSE value of 0.59 on the train data and 0.95 on the test data.

4) Random Forest Regressor Method :

Python 3.6 uses ”sklearn.ensemble.RandomForestRegress” to implement random forest. An arrangement of trees that have grown apart from one another is known as a random forest. The union of N different algorithms creates a composition. The algorithms are generally trained before the results they produce are averaged.

Evaluation And Results :



RMSE on Train Data

```
In [74]: %time np.sqrt(np.sum(np.square(rfr.predict(grouped_train_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_train_df['log_
         Wall time: 2.21 s
Out[74]: 0.5075197126060914

In [76]: pickle.dump(rfr, open("saved_random_forest", 'wb'))
```

RMSE on Test Data

```
In [31]: %time np.sqrt(np.sum(np.square(rfr.predict(grouped_test_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_test_df['log_Re
         Wall time: 1.11 s
Out[31]: 0.9442356420140502
```

Figure 13: RMSE Value on Train and Test Data for Random Forest Regressor

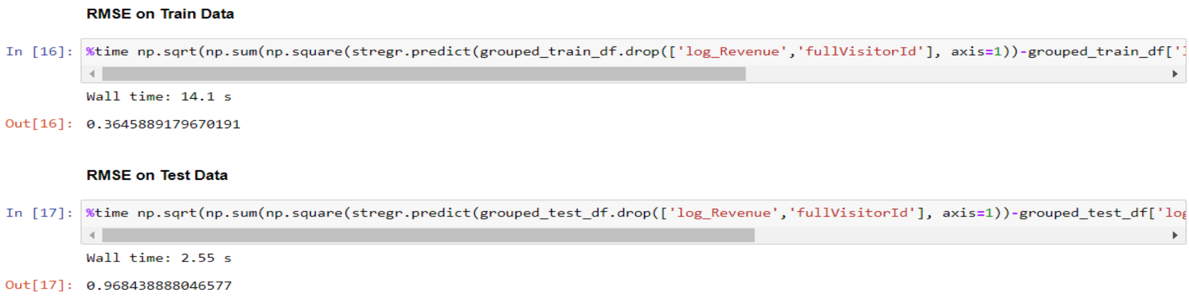
Random Forest Regressor model with tuned hyper-parameters was trained using Random Search. In this model, 100 estimators, 6 minimum sample split and 2 minimum samples of leaf are the best hyperparameters. Compared to other models, it took overall 4hrs 15mins to compute, resulting in a RMSE of 0.50 for train data and 0.96 for test data.

5) Stacking Regressor Method :

With stacking regression, multiple regression models are combined using a meta-regressor in order to improve its learning efficiency. Here, the Stacked model made up of LGBM Regressor, Gradient Boosting Regressor, XGB Regressor and Random Forest Regressor, and the meta regressor is LGBM Regressor. These individual regression models are trained using the complete training set, and then the meta-regressor is fitted using the individual model outputs, which are known as meta-features.

Evaluation And Results :

In order to train the Stacking Regressor model, the model was made of Gradient Boosting Regressor, XGB Regressor and Random Forest Regressor were the features were to the Meta Regressor (LGBM Regressor). On the train data, it gave RMSE values of 0.36 and 0.968 on the test data with the overall computational time of 39mins 44secs.



RMSE on Train Data

```
In [16]: %time np.sqrt(np.sum(np.square(stregr.predict(grouped_train_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_train_df['
         Wall time: 14.1 s
Out[16]: 0.3645889179670191
```

RMSE on Test Data

```
In [17]: %time np.sqrt(np.sum(np.square(stregr.predict(grouped_test_df.drop(['log_Revenue','fullVisitorId'], axis=1))-grouped_test_df['log
         Wall time: 2.55 s
Out[17]: 0.968438888046577
```

Figure 14: RMSE Value on Train and Test Data for Stacking Regressor

6 Evaluation

The hyperparameters were tuned using the RandomSearch technique. The search space in the case of the RandomSearch approach is a bounded domain of hyperparameter values where points are randomly sampled. The minimized error method is often used to evaluate the performance of regression models, with zero being a model with perfect ability. In this instance, a Root Mean Square Error (RMSE) was employed to evaluate how well the model performed. The depth of each individual tree has an effect on overfitting. Options for maximum depth and leaf count may have an impact. The maximum number of leaves that a tree may have is limited by number of leaves, but a tree's maximum depth is determined by its maximum depth.

Five separate experiments were used in this study where the maximum number of pageviews turned out to be the best predictors of customer revenue : the LGBM (Light Gradient Boosting Model) Regressor, the XGBoost Regressor, the Gradient Boosting Regressor, the Random Forest Regressor, and the Stacked Regressor. The LGBM Regressor was the model that suited the experiment the best. The winner of the competition got the RMSE value of 0.98 using the Gradient Boosting Algorithm.

The best performance was given by LGBM (Light Gradient Boosting Model) Regressor with the least computational time of 6mins 54secs. The model also gave the least Root Mean Square Value (RMSE) score of 0.95 on the test data and 0.59 on the train data. Random Forest Regressor followed LGBM Regressor regarding the RMSE score on the test data. Though the computational time (4hrs 15mins) is much higher than LGBM, it gave 0.964 as the RMSE value on the test dataset. Both these models were trained after the Random Search technique was applied to obtain the best hyper-parameters. The Stacked Regressor gave the exciting result even though the parameters were not tuned when the output was fed to the LGBM Regressor. The RMSE score was 0.968 on the test data. XGBoost Regressor and Gradient Boosting Regressor gave 1.30 and 1.29 on the test data, respectively.

6.1 Discussion

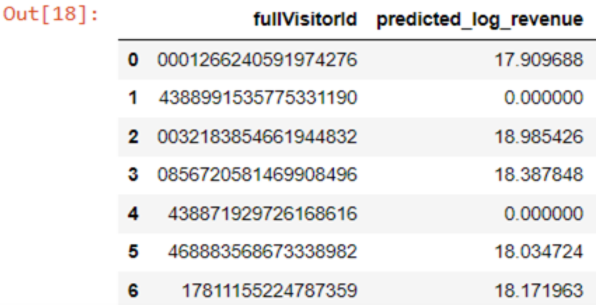
The dataset was pre-processed, and the models were trained after the best hyper-parameters were obtained with the Random Search technique. The results of the models being trained are summarized in table 2.

Table 2: Evaluation

Model	**Best Hyper-parameters**	**Computational Time**	**RMSE on Train Data**	**RMSE on Test Data**
LGBM Regressor	subsample:0.7, reg_lambda:1, reg_alpha:0.5, num_leaves:100, n_estimators:350, learning rate:0.1, min_child_samples: 1	6mins 54secs	0.955	0.955
XGB Regressor	subsample:0.8, reg_lambda:0.5, reg_alpha:1, n_estimators:200, learning rate:0.1	1hr 50mins	1.222	1.301
Gradient Boosting Regressor	subsample:0.7, n_estimators:400, min_samples_split:2, min_samples_leaf:3, max_depth:5	5hrs 48mins	1.196	1.29
Random Forest Regressor	n_estimators:100, min_samples_split:6, min_samples_leaf:2, max_depth:none	4hrs 15mins	0.506	0.964
Stacked Regressor	-	39mins 44secs	0.365	0.968

Root Mean Square Error (RMSE) is used here to evaluate the performance of the Machine Learning models. The Root Mean Square Error (RMSE) statistic calculates the error in our results when the target or response variable is a continuous number. A standard deviation of residuals or prediction errors is known as RMSE. It shows how dispersed the data is relative to the line of best fit. Five regressor models were trained to predict the revenue generated by the customers. It was found that LGBM (Light Gradient Boosting Model) outperformed the rest by giving the Root Mean Square Error (RMSE) value of 0.598 on the train data and 0.95 on the test data. LGBM model was followed by Random Forest Regressor with an RMSE value of 0.506 and 0.964 on the train and test data, respectively. The Stacking Regressor gave exciting results. The output was fed to the LGBM Regressor, which obtained the RMSE value of 0.96 on the test data without tuning its hyper-parameters. If the hyper-parameters are tuned, they will surpass the rest.

The figure 15 of the research has some of the visitors from the test data and the prediction of the revenue generated by them.



```
Out[18]:
             fullVisitorId  predicted_log_revenue
0      0001266240591974276              17.909688
1      4388991535775331190               0.000000
2      0032183854661944832              18.985426
3      0856720581469908496              18.387848
4       438871929726168616               0.000000
5       468883568673338982              18.034724
6        17811155224787359              18.171963
```

Figure 15: Predicted log value of the revenue of few customers

Table 3: Model Summary

Model	**Computational Time**	**RMSE on Train Data**	**RMSE on Test Data**
LGBM Regressor	6mins 54secs	0.598	0.955
XGB Regressor	1hr 50mins	1.22	1.301
Gradient Boosting Regressor	5hrs 48mins	1.196	1.29
Random Forest Regressor	4hrs 15mins	0.506	0.964
Stacked Regressor	39mins 44secs	0.365	0.968

7 Conclusion and Future Work

Every business is feeling the pressure as consumers become more recognizable and less devoted due to their increased reliance on one another's opinions, which has led to an increase in their demands and uncertainty. Customer Analytics, Customer Relationship Management, and enhancing Customer Lifetime Value are required to accomplish the goals of creating more earnings, keeping the current customers while bringing in new ones, and maintain existing consumers. This proposal tries to identify the customers that provide the most significant profit and forecast their sales based on their prior dealings. Random Forest and three different Boosting Regressors were trained along with the Stacked Regressor. LGBM (Light Gradient Boosting Model) outperformed the rest with the least computational time of just 6mins 54secs with 0.95 as the RMSE value of test data. The most exciting result was given by the Stacked Regressor fed to the Meta Regressor, obtaining the RMSE value of 0.968 without tuning the parameters. Definitely, the stacked Regressor will succeed if the hyper-parameters are tuned.

Both variables and historical data are used in the predictive analytics model. For identifying patterns and trends in the user's activity, historical data is essential. Finding important customers, learning about their skills, handling difficult clients, and attracting the best ones would all be much simpler as a result. This would assess the salesperson's attempts to reduce operational costs and focus the company's attention on its customers. This can be applied for product recommendations, targeted marketing, prioritizing quality of sales of likely users, etc after building an recommendation engine.

The future scope of the research would be imputing the missing values for two groups separately, one for the profit-generating customers and the second for the customers who didn't generate at all. Since the dataset had only 1.21% of the users who generated revenue, to counter this, an upsampling technique like Synthetic Minority Over-sampling Technique (SMOTE) can be given a try. Taking the research to a different level, Customer Lifetime Value could be found.

References

Chen, K. (2019). An online retail prediction model based on aga-lstm neural network.
URL: *https://ieeexplore.ieee.org/document/9360965*

Doan, T., V. N. and Keng, B. (2018). Generating realistic sequences of customer-level transactions for retail datasets.
URL: *https://ieeexplore.ieee.org/document/8637567*

Ghosh, S. and Banerjee, C. (2020). A predictive analysis model of customer purchase behavior using modified random forest algorithm in cloud environment.
URL: *https://ieeexplore.ieee.org/document/9290700*

Lv, Y. (2021). Research on customer relationship management of a company under the background of e-commerce.
URL: *https://ieeexplore.ieee.org/document/9406948*

Monastyrskaya, M. and Soloviev, V. (2020). Improving customer relationship management based on intelligent analysis of user behavior patterns.
URL: *https://ieeexplore.ieee.org/document/9247718*

Mou, S., J. Y. and Tian, C. (2021). Retail time series prediction based on emd and deep learning.
URL: *https://ieeexplore.ieee.org/document/8525707*

Nurbakova, D. and Saumet, T. (2020). Deal closure prediction based on user's browsing behaviour of sales content.
URL: *https://ieeexplore.ieee.org/document/9346295*

Ohrimuk, E.S., R. N. M. Y. and Bezrukov, A. (2020). Study of supervised algorithms for solve the forecasting retail dynamics problem.
URL: *https://ieeexplore.ieee.org/document/9039112*

Parikh, Y. and Abdelfattah, E. (n.d.). Clustering algorithms and rfm analysis performed on retail transactions, *11th IEEE Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)* .

Rotovei, D. and Negru, V. (2019).). data driven sales prediction using communication sentiment analysis in b2b crm systems.
URL: *https://ieeexplore.ieee.org/document/9049875*

Singh, B., K. P. S. N. and Sharma, K. (2020). Sales forecast for amazon sales with time series modeling, *First International Conference on Power, Control and Computing Technologies (ICPC2T)* .

Song, D. and Liang, C. (2021). Application research of data mining technology in customer relationship management., *4th International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE)* .

Wang, J., L. G. and Liu, L. (2019). A selection of advanced technologies for demand forecasting in the retail industry.
URL: *https://ieeexplore.ieee.org/document/8713196*

Win, T. and Bo, K. (2020). Predicting customer class using customer lifetime value with random forest algorithm.
URL: *https://ieeexplore.ieee.org/document/9261792*

Zhu, H. (2021). A deep learning based hybrid model for sales prediction of e-commerce with sentiment analysis.
URL: *https://ieeexplore.ieee.org/document/9463271*