Machine Learning Engineer Nanodegree

Capstone Project- Google Analytics Customer Revenue Prediction

Udacity Machine Learning Engineer Nanodegree
Adithya Kamaraj, March 2019

I. Definition

Project Overview

The project is based on E-commerce customer behavior analysis. As Internet usage increases worldwide, one can expect to see a concomitant rise in the use of this interactive tool in consumers' shopping decisions [1]. Customer analysis plays an integral part in making marketing decisions and strategic planning of any E-commerce business. Internet marketing is the process of building and maintaining customer relationships through online activities to facilitate the exchange of ideas, products, and services that satisfy the goals of both parties [2]. Customer analysis is conventionally done from demography, behavioral, attitude, physiological and social data of the customer.

Web analytics is an approach that involves collecting, measuring, monitoring, analysing and reporting web usage data to understand visitors' experiences. Analytics can help to optimise web sites in order to accomplish business goals and/or to improve customer satisfaction and loyalty [3,4,5]. Google Analytics is an web analytics service which offers basic analytics tools to help in marketing decisions. Google Analytics allows users to export report data in MS Excel format, which when transformed can be analyzed with time series statistical programs[6]. E-commerce sites need to have customer behavior analysis for the retainment of the customers and reduce churn. There is very less research in the prediction of revenue of a customer using the Google analytics platform.

Problem Statement

The problem statement is “Google Analytics Customer Revenue Prediction” for Google Merchandise Store a Kaggle competition [7]. The goal of the project is to predict the revenue generated by the customer based on the data obtained. The revenue generated is based on 80/20 rule for most business that is most of the revenue generated by a small percentage of customers. Customer analysis can help make strategic marketing decisions by figuring out the pattern of high revenue customers.

The solution to the “Google Analytics Customer Revenue Prediction” problem is discussed in the following sections. The first step in the solution is the “**Analysis”** data set is first explored and visualized to gain insights about the data and then the benchmark model and algorithms for the solution is chosen. The next and most important step which is followed to provide a solution is the **“Methodology”** section where the data preprocessing and the building of the model is carried out. The final sections of the project are the **“Results”** and **“Conclusion”** where the final solutions with refinement and then summary of the project is delivered.

Metrics

RMSE is chosen as the evaluation metric to measure the performance of the model for the predicted. RMSE value can range from 0 to ∞ and are indifferent to the direction of errors and it is defined as negatively-oriented scores, therefore lower RMSE values are desirable. Since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors and penalizing large errors through the defined least-square terms proves to be very effective in improving model performance [8]. RMSE is used to measure the performance of the model predicting the natural log of the sum of all the transactions per user.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

II. Analysis

Data Exploration

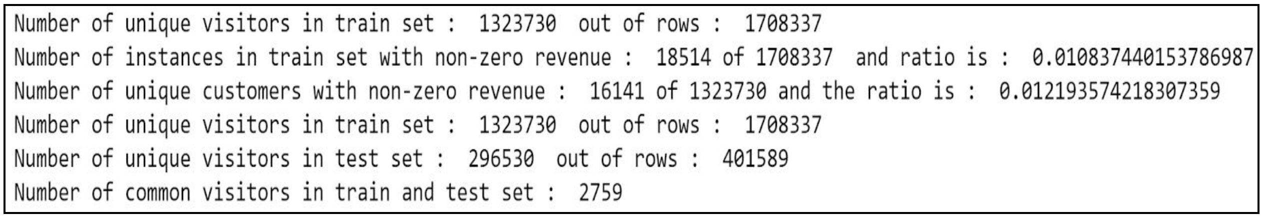
Google Merchandise Store dataset has a total of 12 attributes (some are JSON collection set). The data include attributes of 2 Dates, 4 numeric identifiers and 7 categorical. The 4 categorical attributes ('device', 'geoNetwork', 'totals', 'trafficSource) are in JSON set and can be divided into multiple attributes for analysis. Totals JSON attribute have TransactionRevenue values which will be the Response/Predictive attribute.

Each of the twelve attributes which are given as part of the Google Merchandise Store dataset is explained in the table below and the dataset has user data from the year 2016 to 2017.

Attribute	**Description**
fullVisitorId	A unique identifier for each user of the Google Merchandise Store.
channelGrouping	The channel via which the user came to the Store.
date	The date on which the user visited the Store.
device	The specifications for the device used to access the Store.
geoNetwork	This section contains information about the geography of the user.
socialEngagementType	Engagement type, either "Socially Engaged" or "Not Socially Engaged".
totals	This section contains aggregate values across the session.
trafficSource	This section contains information about the Traffic Source from which the session originated.

visitId	An identifier for this session. This is part of the value usually stored as the _utmb cookie. This is only unique to the user. For a completely unique ID, you should use a combination of fullVisitorId and visitId.
visitNumber	The session number for this user. If this is the first session, then this is set to 1.
visitStartTime	The timestamp (expressed as POSIX time).

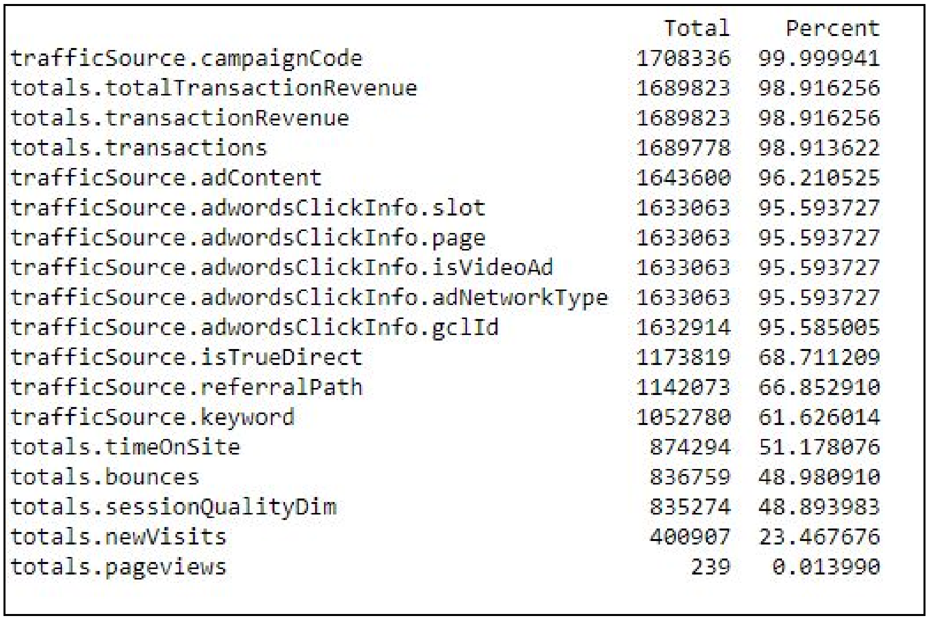
Users data Exploration :



```
Number of unique visitors in train set :  1323730  out of rows :  1708337
Number of instances in train set with non-zero revenue :  18514 of 1708337  and ratio is :  0.010837440153786987
Number of unique customers with non-zero revenue :  16141 of 1323730 and the ratio is :  0.012193574218307359
Number of unique visitors in train set :  1323730  out of rows :  1708337
Number of unique visitors in test set :  296530  out of rows :  401589
Number of common visitors in train and test set :  2759
```

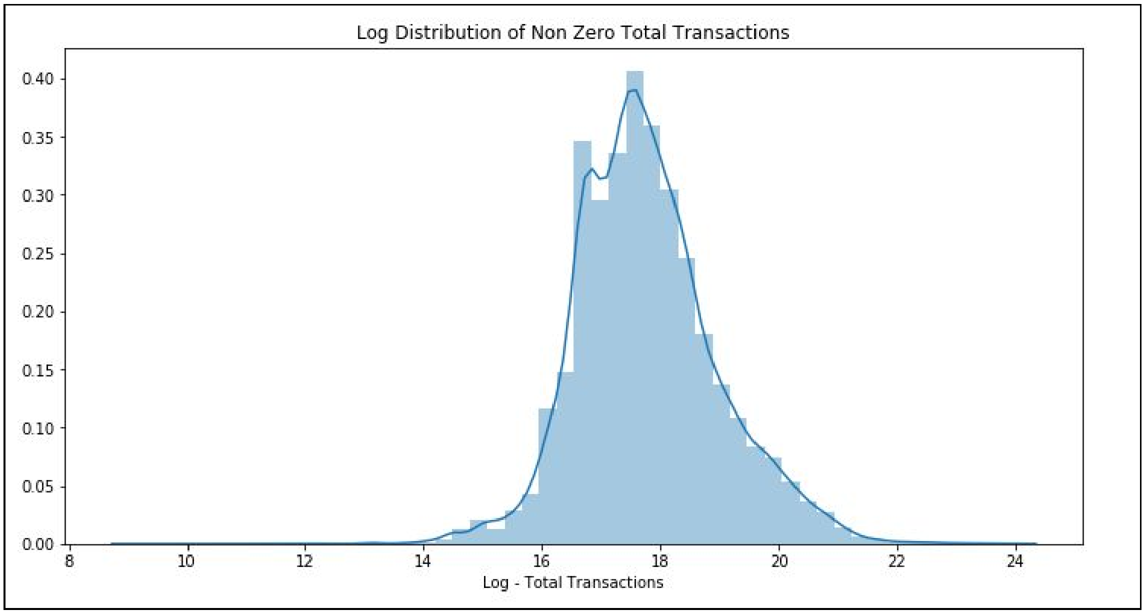
Abnormalities:

The dataset has columns with missing values and some of the columns have almost 98% of missing values, The columns with missing values have to be preprocessed or dropped to be used in the model.



```
                                              Total    Percent
trafficSource.campaignCode                  1708336  99.999941
totals.totalTransactionRevenue              1689823  98.916256
totals.transactionRevenue                   1689823  98.916256
totals.transactions                         1689778  98.913622
trafficSource.adContent                     1643600  96.210525
trafficSource.adwordsClickInfo.slot         1633063  95.593727
trafficSource.adwordsClickInfo.page         1633063  95.593727
trafficSource.adwordsClickInfo.isVideoAd    1633063  95.593727
trafficSource.adwordsClickInfo.adNetworkType  1633063  95.593727
trafficSource.adwordsClickInfo.gclId        1632914  95.585005
trafficSource.isTrueDirect                  1173819  68.711209
trafficSource.referralPath                  1142073  66.852910
trafficSource.keyword                       1052780  61.626014
totals.timeOnSite                            874294  51.178076
totals.bounces                               836759  48.980910
totals.sessionQualityDim                     835274  48.893983
totals.newVisits                             400907  23.467676
totals.pageviews                                239   0.013990
```

Statistics :



Kurtosis and Skewness of Transaction Revenue

I. Skewness

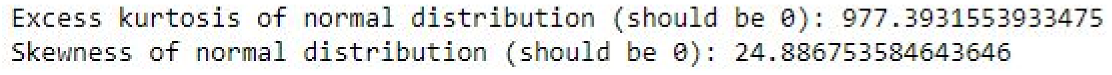
It is the degree of distortion from the symmetrical bell curve or the normal distribution. It measures the lack of symmetry in data distribution. A symmetrical distribution will have a skewness of 0.

Positive Skewness means when the tail on the right side of the distribution is longer or fatter. The mean and median will be greater than the mode. Negative Skewness is when the tail of the left side of the distribution is longer or fatter than the tail on the right side. The mean and median will be less than the mode.

II. Kurtosis

Kurtosis is all about the tails of the distribution — not the peakedness or flatness. It is used to describe the extreme values in one versus the other tail. It is actually the measure of outliers present in the distribution.

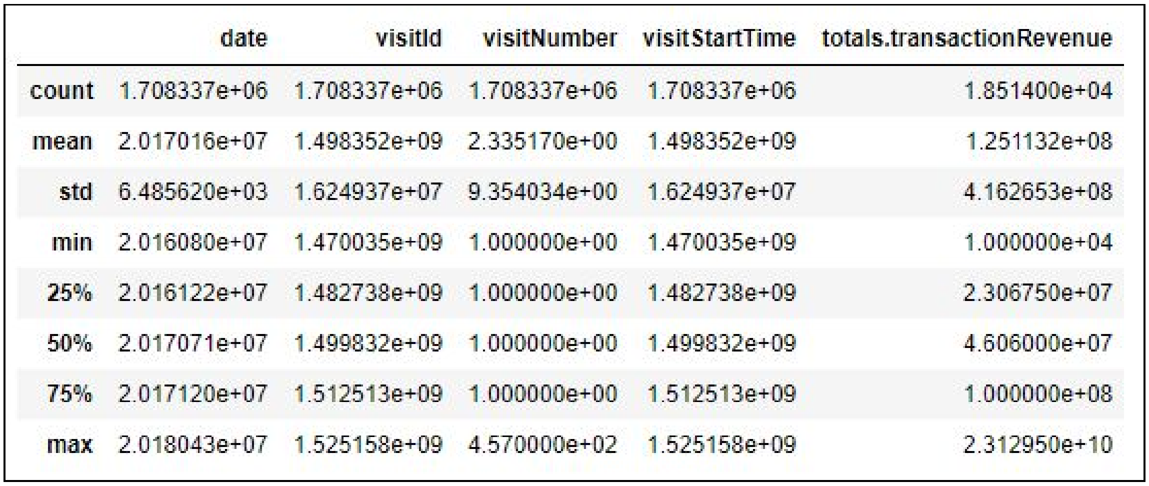
High kurtosis in a data set is an indicator that data has heavy tails or outliers. If there is a high kurtosis, then, we need to investigate why do we have so many outliers. Low kurtosis in a data set is an indicator that data has light tails or lack of outliers.



```
Excess kurtosis of normal distribution (should be 0): 977.3931553933475
Skewness of normal distribution (should be 0): 24.886753584643646
```

The dataset is fairly **symmetrical skewed** and has a **High Kurtosis**.

The following table was obtained by using the **data.describe** function to get the statistical data of the numerical columns in the dataframe before preprocessing.

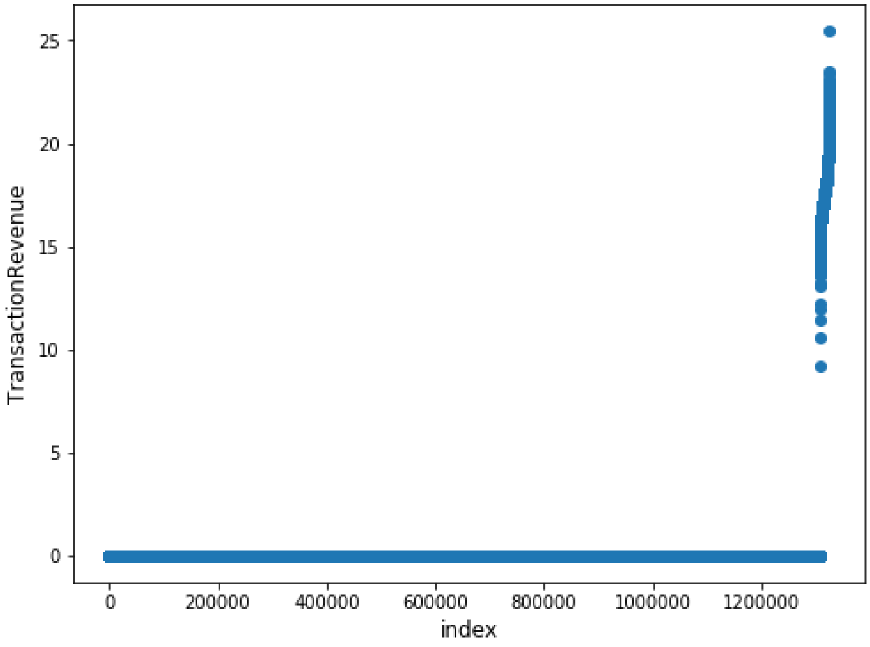


	date	visitId	visitNumber	visitStartTime	totals.transactionRevenue
count	1.708337e+06	1.708337e+06	1.708337e+06	1.708337e+06	1.851400e+04
mean	2.017016e+07	1.498352e+09	2.335170e+00	1.498352e+09	1.251132e+08
std	6.485620e+03	1.624937e+07	9.354034e+00	1.624937e+07	4.162653e+08
min	2.016080e+07	1.470035e+09	1.000000e+00	1.470035e+09	1.000000e+04
25%	2.016122e+07	1.482738e+09	1.000000e+00	1.482738e+09	2.306750e+07
50%	2.017071e+07	1.499832e+09	1.000000e+00	1.499832e+09	4.606000e+07
75%	2.017120e+07	1.512513e+09	1.000000e+00	1.512513e+09	1.000000e+08
max	2.018043e+07	1.525158e+09	4.570000e+02	1.525158e+09	2.312950e+10

Exploratory Visualization

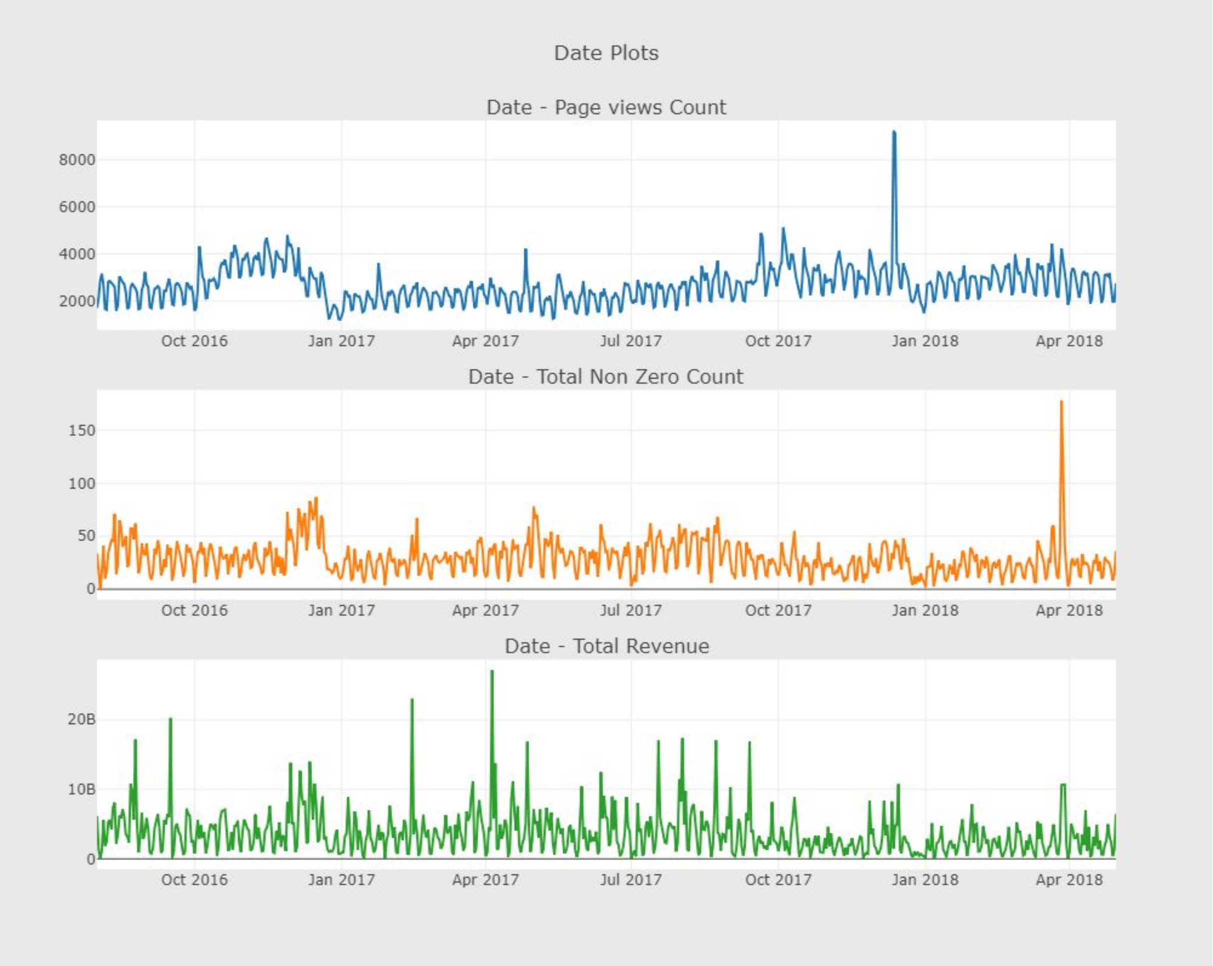
The complete exploratory visualization is discussed in this section for the features considered in the dataset. Out of the 57 columns and one target variable **totals.transactionRevenue** the visualization is done to obtain a correlation between the target variable and the columns in the dataset. In the given dataset by intuition, we can correlate the revenue generated to specific columns and this can be visualized to understand the dataset and relation between the features and the target variable.

Revenue Generated with respect to users :



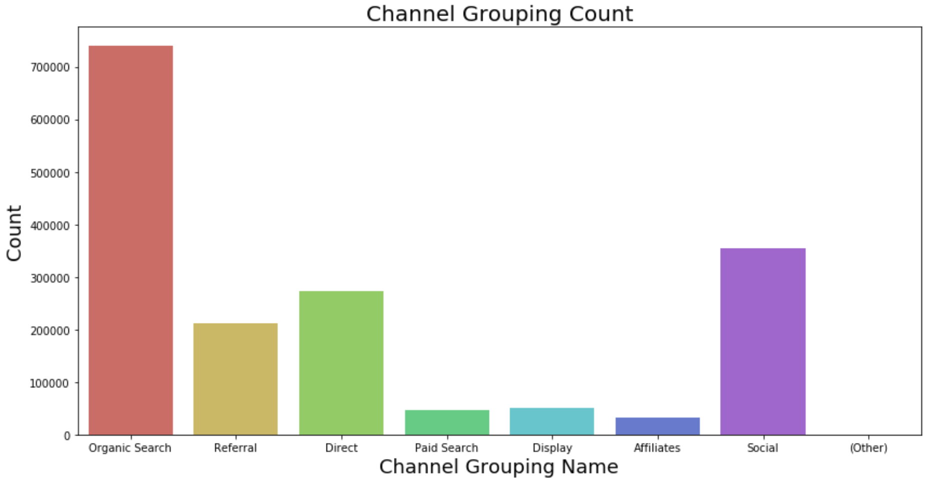
The revenue generated is from a small group of users and this proves the 80/20 rule where most revenue generated is by 20% of the users, in this case, it is even less. Approximately 1% of the users contribute to the most revenue generated. This is a valuable insight as we can identify potential high paying customers easily.

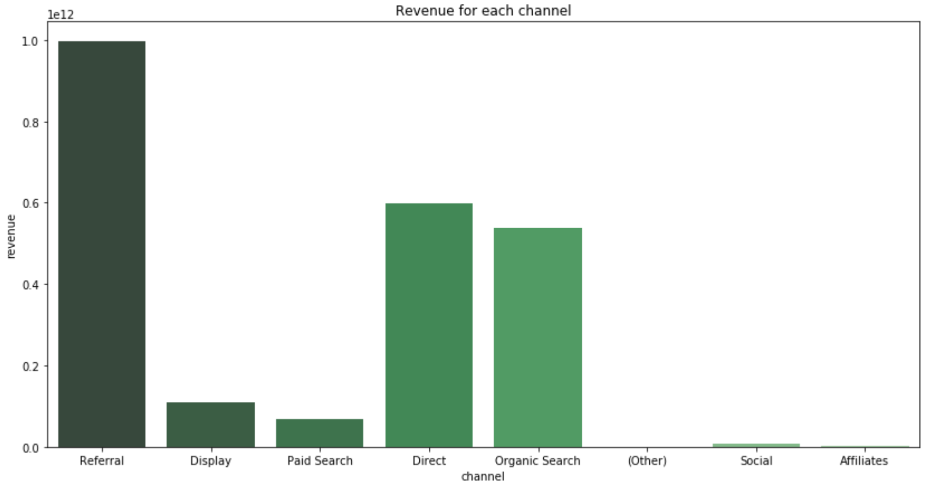
Time series data:



Looking at the time data, there are increased visits and revenue generated during different seasons for example during December end and January there is a constant rise in page visits and revenue generated for both the years in the dataset and a significant drop just after the new years. This data proves that the number of visits increases during certain time periods which also affect the revenue generated during those time periods.

Channel Grouping data:

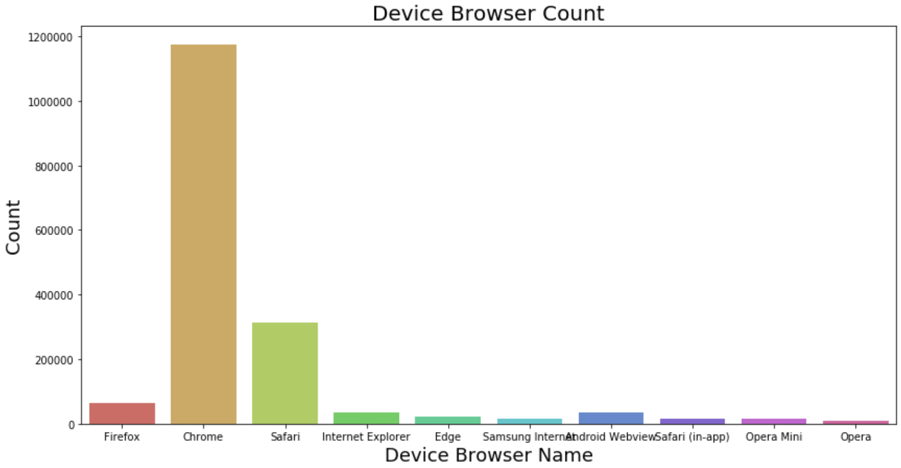


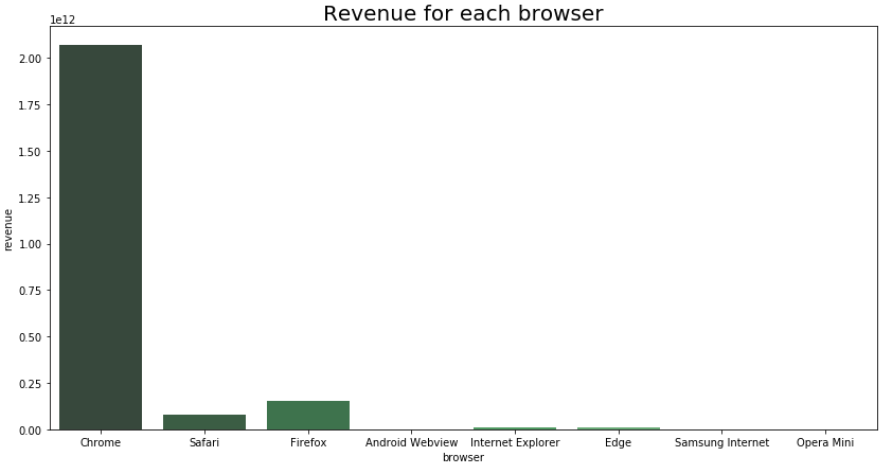


Considering the channel grouping column the most amount of users are through organic search followed by social almost half less than that of organic search. Direct and Referral are the nest the most number of users with 300000 and 200000 each approximately. Paid search, Display, and affiliates have the lowest amount of users.

Looking at the revenue generated by each group Referral has the most revenue generation even though it has lesser users compared to the organic search and social. The next most revenue generating groups are direct and organic search. Social has the second lowest revenue even though it has the second most user count. Display and paid search have relatively justifiable revenue comparing the users they have.

Device Browser data:

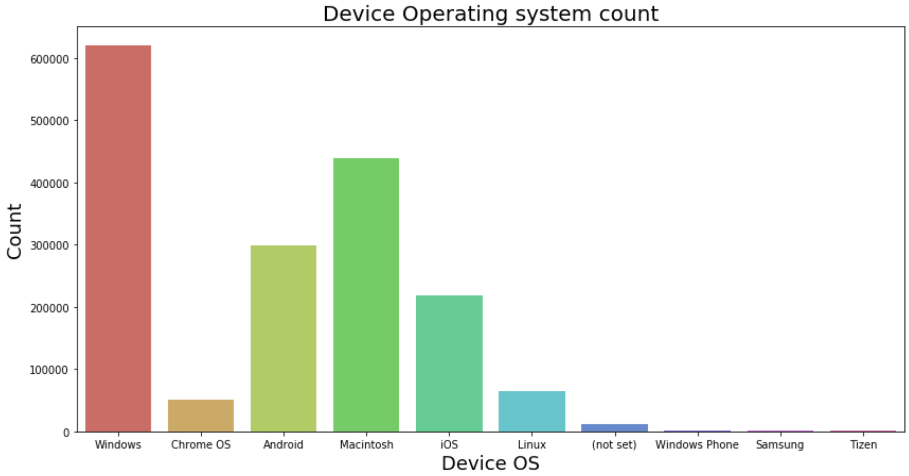


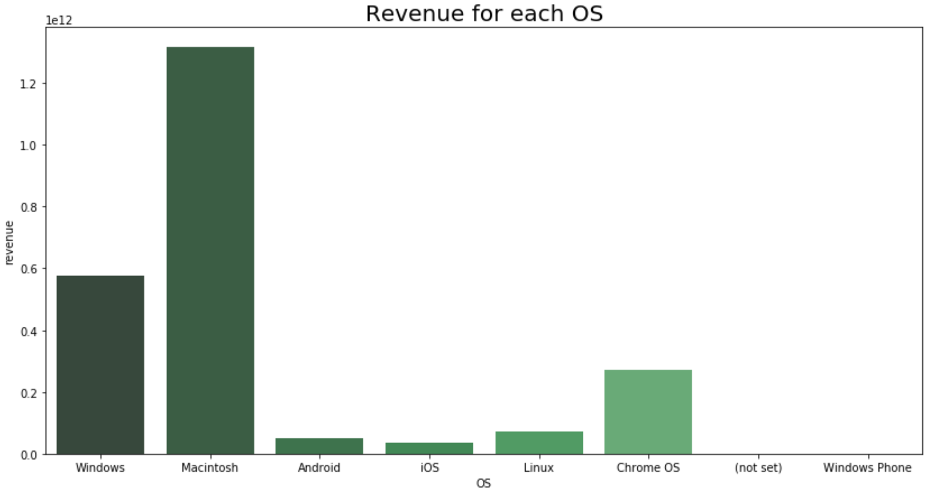


Chrome has the most amount of users which is approximately three times more than Safari and ten times more than Firefox and Internet Explorer combined. All the other browsers except Chrome and Safari has users less than 100000. Internet Explorer and Android Webview have an equal number of users sane was Samsung internet, opera mini and Safari (in-app) have a relatively equal number of users.

The revenue generated is ninety percent from chrome browser and Firefox has more revenue generated even though it has lesser users than Safari. Safari has half the revenue of Firefox generated and stands in the third position. The revenue contributed from users of other browsers are very less in comparison.

Device Operating System data :

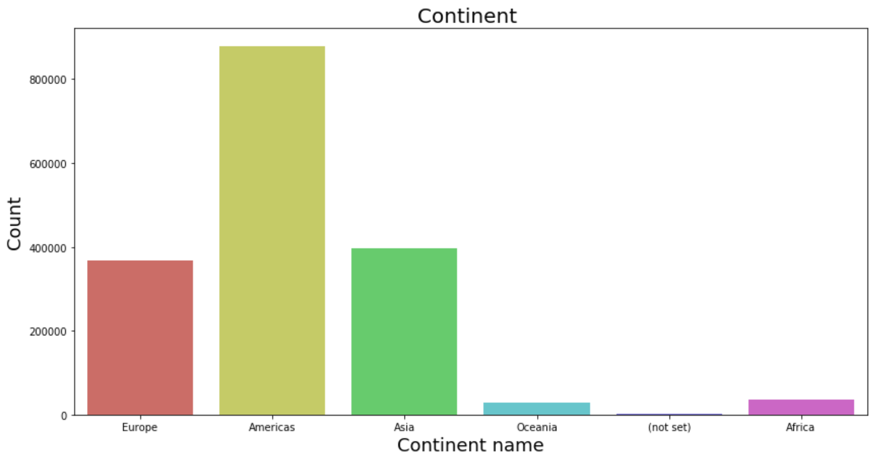


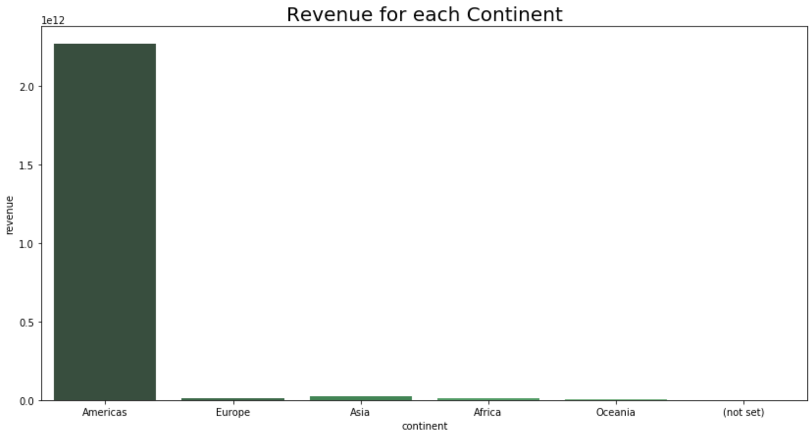


Windows operating system has the most amount of users visiting the page with about 600000 users and Macintosh has second most users count with about 450000 users. In the mobile application, Android has the highest users count than iOS. Linux and Chrome OS both have less than 50000 users.

The revenue generated by users of Macintosh OS is the highest almost twice as that of Windows users, even though the count of overall users are less in Macintosh compared to Windows OS users. Chrome OS has the third highest revenue generation.Looking at the data the Android and iOS users have very less contribution to the revenue, therefore, desktop OS users have more revenue generation than mobile users.

Continent data:

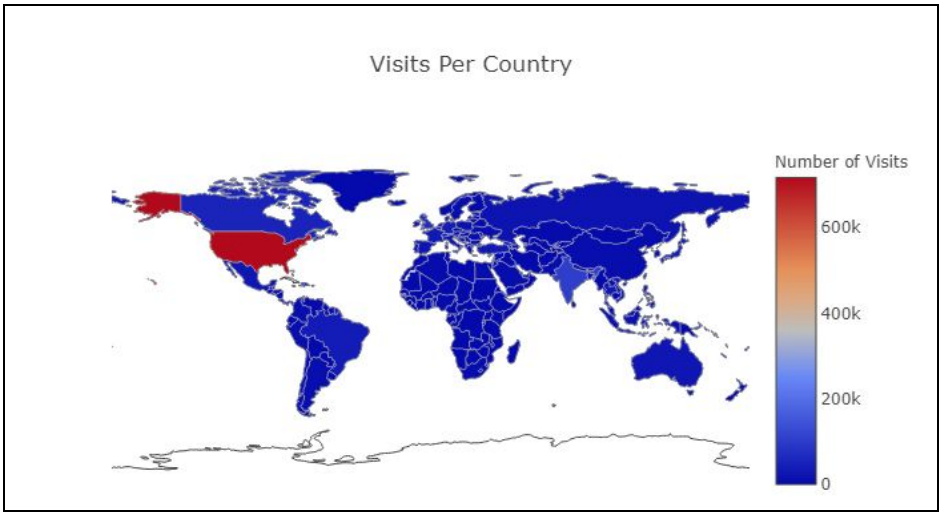


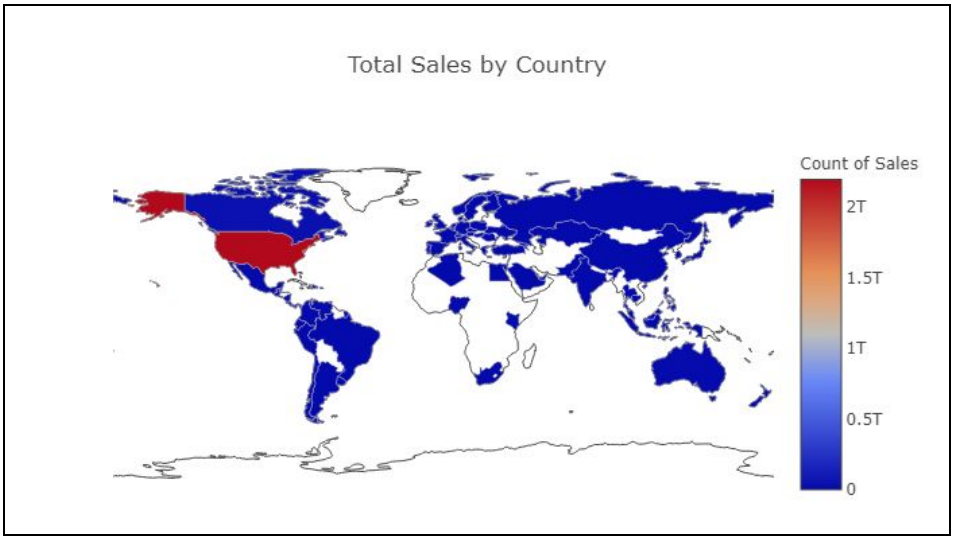


The number of users is the highest in the Americas continent with almost 900000 users which is equal to Europe and Asia combined. Asia and Europe continents have almost an equal number of users while Oceania and Africa have almost an equal number of users.

The revenue generated from the Americas continent is more than 95% of the total revenue generated. Asia has the second most revenue generated while its almost 2% of the total revenue generated. Europe, Africa, and Oceania have the rest of the contribution. In the next part, each countries visits and revenue is calculated

Country Data :





The total number of visits is highest in the United States of America with 700k visits in the given time period by the users. The second highest is India which has approximately 100k user visits. The United Kingdom has 70k user visits and Canada with 50k.

The revenue generated is highest in the United States of America with about 2.19 Trillion and the rest of all the countries have their revenue in Billions. Therefore the most significant country for the Google Merchandise Store.

Algorithms and Techniques

Algorithm :

“Google Analytics Customer Revenue Prediction” is a supervised learning problem. Supervised machine learning problems provide the learning algorithms with known quantities to support future judgments. Both the input and the output variable is given and the algorithms train on the known data to predict on the test data.

The Algorithm used in this project is LightGBM. **LightGBM** is a gradient boosting framework that uses tree-based learning algorithms. **LightGBM** produces much more complex trees by following leaf wise split approach rather than a level-wise approach which is the main factor in achieving higher accuracy [4]. Comparing to other boosting frameworks like XGBOOST, LightGBM is faster and has high performance.

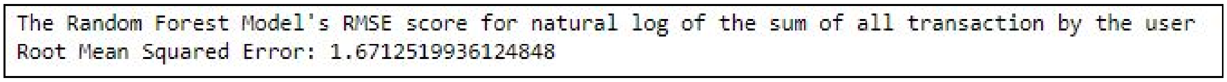
Techniques :

Label Encoding a technique to encode categorical values, which allows you to convert each value in a column to a number. Numerical labels are always between 0 and n - 1 of categories. In this dataset, more than half the columns are categorical values.

Data split was done based on the time, data till 31-5-2017 was taken as the training data and data from dates after 31-5-2017 was taken as the validation dataset.

Benchmark

The benchmark model chosen is random forest tree regressor as continuous values are predicted. Random forest is one of the Ensemble methods as they build many individual trees used for both classification and regression applications that are built using some fraction of the data. Random forest is taken as the benchmark model as they can predict continuous values and can perform better than simple regression models and most non-ensemble models The evaluation metric used for the benchmark is RMSE.



```
The Random Forest Model's RMSE score for natural log of the sum of all transaction by the user
Root Mean Squared Error: 1.6712519936124848
```

III. Methodology

Data Preprocessing

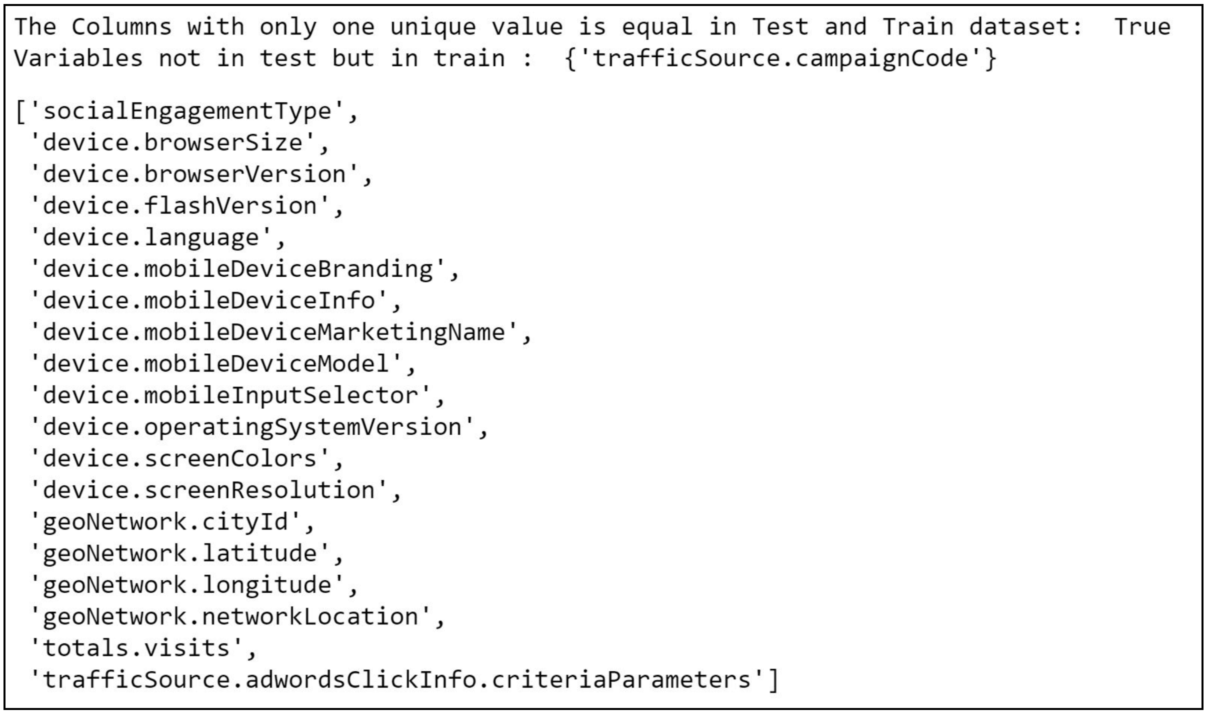
JSON objects :

The complete dataset is loaded which has about 1708337 entries. There are 12 original columns, but four of them have a JSON object that can be converted

- **device**: 16 new columns
- **geoNetwork**: 11 new columns
- **totals**: 6 new columns
- **trafficSource**: 14 new columns

Total there are 58 columns including the target variable **totals.transactionRevenue**.

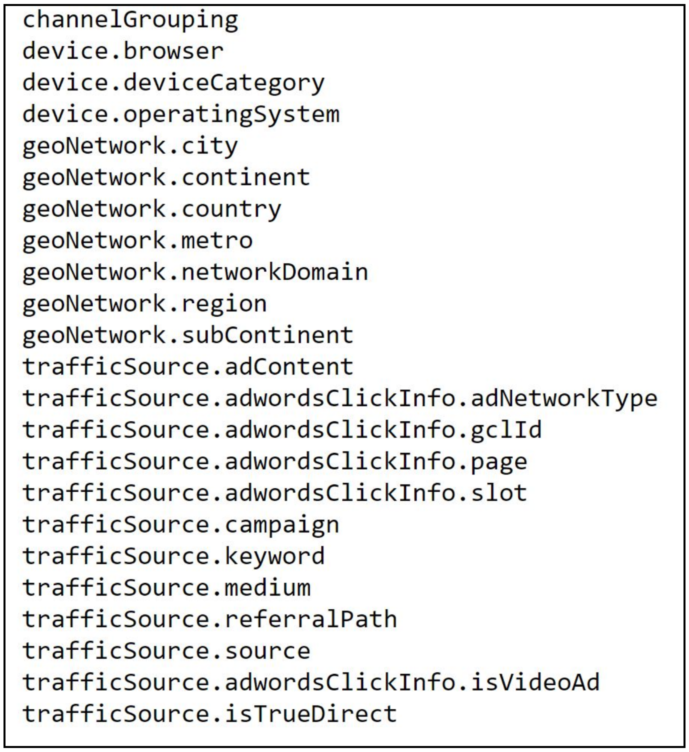
Columns with only one unique value :



```
The Columns with only one unique value is equal in Test and Train dataset:  True
Variables not in test but in train :  {'trafficSource.campaignCode'}

['socialEngagementType',
 'device.browserSize',
 'device.browserVersion',
 'device.flashVersion',
 'device.language',
 'device.mobileDeviceBranding',
 'device.mobileDeviceInfo',
 'device.mobileDeviceMarketingName',
 'device.mobileDeviceModel',
 'device.mobileInputSelector',
 'device.operatingSystemVersion',
 'device.screenColors',
 'device.screenResolution',
 'geoNetwork.cityId',
 'geoNetwork.latitude',
 'geoNetwork.longitude',
 'geoNetwork.networkLocation',
 'totals.visits',
 'trafficSource.adwordsClickInfo.criteriaParameters']
```

The above columns only have one unique variable like “not available in dataset”, ”not socially engaged”, ”not set”. They will not have any effect on our prediction as all the fields have the same data. Their columns are dropped from the dataset. And “trafficSource.campaignCode” is a column present only in the training dataset and not in the test dataset given in the kaggle competition. Therefore this will also be dropped from the dataset. Totally 20 columns are dropped during this pre-processing step.

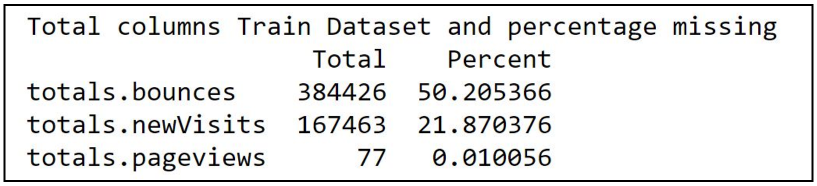
Label encoding categorical features :



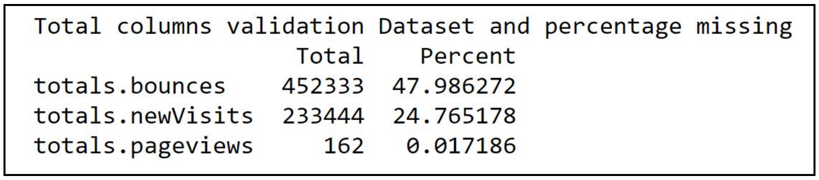
```
channelGrouping
device.browser
device.deviceCategory
device.operatingSystem
geoNetwork.city
geoNetwork.continent
geoNetwork.country
geoNetwork.metro
geoNetwork.networkDomain
geoNetwork.region
geoNetwork.subContinent
trafficSource.adContent
trafficSource.adwordsClickInfo.adNetworkType
trafficSource.adwordsClickInfo.gclId
trafficSource.adwordsClickInfo.page
trafficSource.adwordsClickInfo.slot
trafficSource.campaign
trafficSource.keyword
trafficSource.medium
trafficSource.referralPath
trafficSource.source
trafficSource.adwordsClickInfo.isVideoAd
trafficSource.isTrueDirect
```

Totally 22 Categorical Columns are label encoded to be used as features in the model.

Missing values in columns :



```
Total columns Train Dataset and percentage missing
                     Total    Percent
totals.bounces      384426  50.205366
totals.newVisits    167463  21.870376
totals.pageviews        77   0.010056
```



```
Total columns validation Dataset and percentage missing
                     Total    Percent
totals.bounces      452333  47.986272
totals.newVisits    233444  24.765178
totals.pageviews       162   0.017186
```

Feature Creation

After the data pre-processing steps, the insights can be used to select the needed features in the dataset. The feature selection and creation are done and the dataset is prepared to be fed as input to create the model. In this Project the natural log of the sum of all transactions per user is to be predicted there it is taken as the target feature from the "Totals" attribute.

Implementation

The implementations done on solving "Google Analytics revenue prediction " problem will be discussed in this section. The algorithms used in this project are

- Decision Tree
- Random Forest
- LightGBM

Process:

I. Loading data and Preprocessing

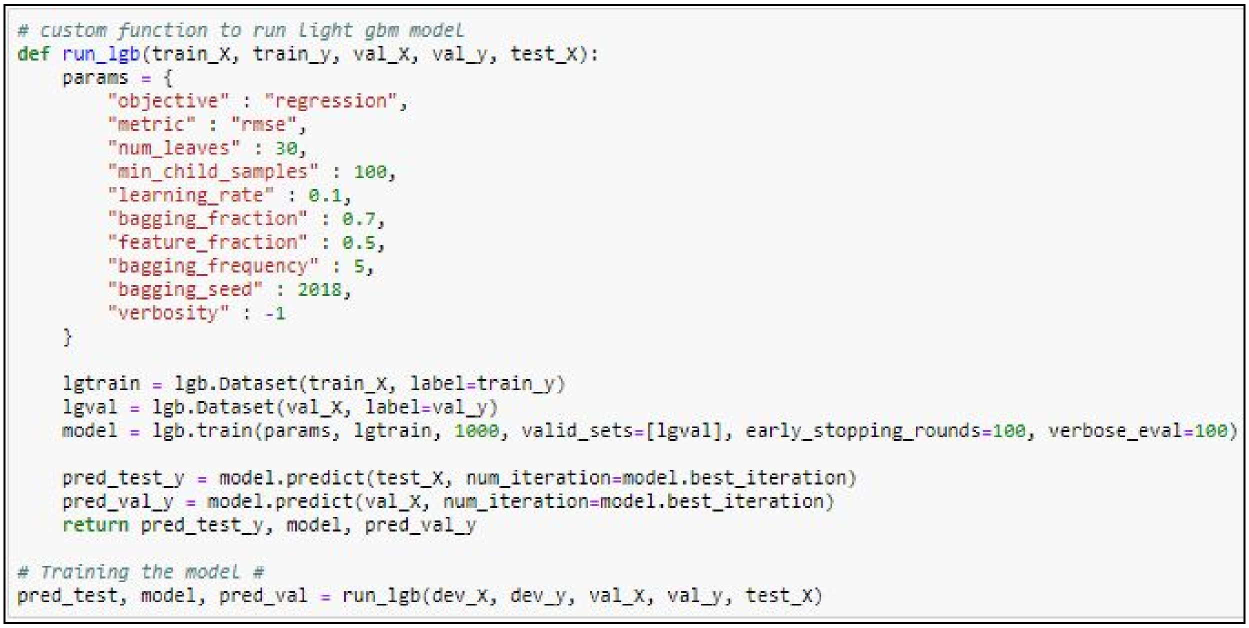
The dataset was loaded and preprocessing steps were followed to clean the data given in the (III) Methodology section of the project

II. Split data into training and validation datasets.

Data was split on the time, data till 31-5-2017 was taken as the training data and data from dates after 31-5-2017 was taken as the validation dataset.

III. Train model on training data.

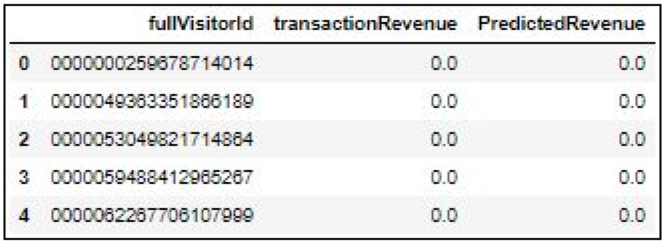
The algorithms chosen was used to build models which are to be evaluated against the evaluation metric chosen. The algorithm chosen to solve the problem is LightGBM , as the performance is better and is used in most of the Kaggle competitions and is faster compared to other algorithms



```
# custom function to run light gbm model
def run_lgb(train_X, train_y, val_X, val_y, test_X):
    params = {
        "objective" : "regression",
        "metric" : "rmse",
        "num_leaves" : 30,
        "min_child_samples" : 100,
        "learning_rate" : 0.1,
        "bagging_fraction" : 0.7,
        "feature_fraction" : 0.5,
        "bagging_frequency" : 5,
        "bagging_seed" : 2018,
        "verbosity" : -1
    }

    lgtrain = lgb.Dataset(train_X, label=train_y)
    lgval = lgb.Dataset(val_X, label=val_y)
    model = lgb.train(params, lgtrain, 1000, valid_sets=[lgval], early_stopping_rounds=100, verbose_eval=100)

    pred_test_y = model.predict(test_X, num_iteration=model.best_iteration)
    pred_val_y = model.predict(val_X, num_iteration=model.best_iteration)
    return pred_test_y, model, pred_val_y

# Training the model #
pred_test, model, pred_val = run_lgb(dev_X, dev_y, val_X, val_y, test_X)
```

IV. Predict revenue on test features.

The model built using the LightGBM algorithm is used to predict the revenue generated by a customer on the validation dataset.

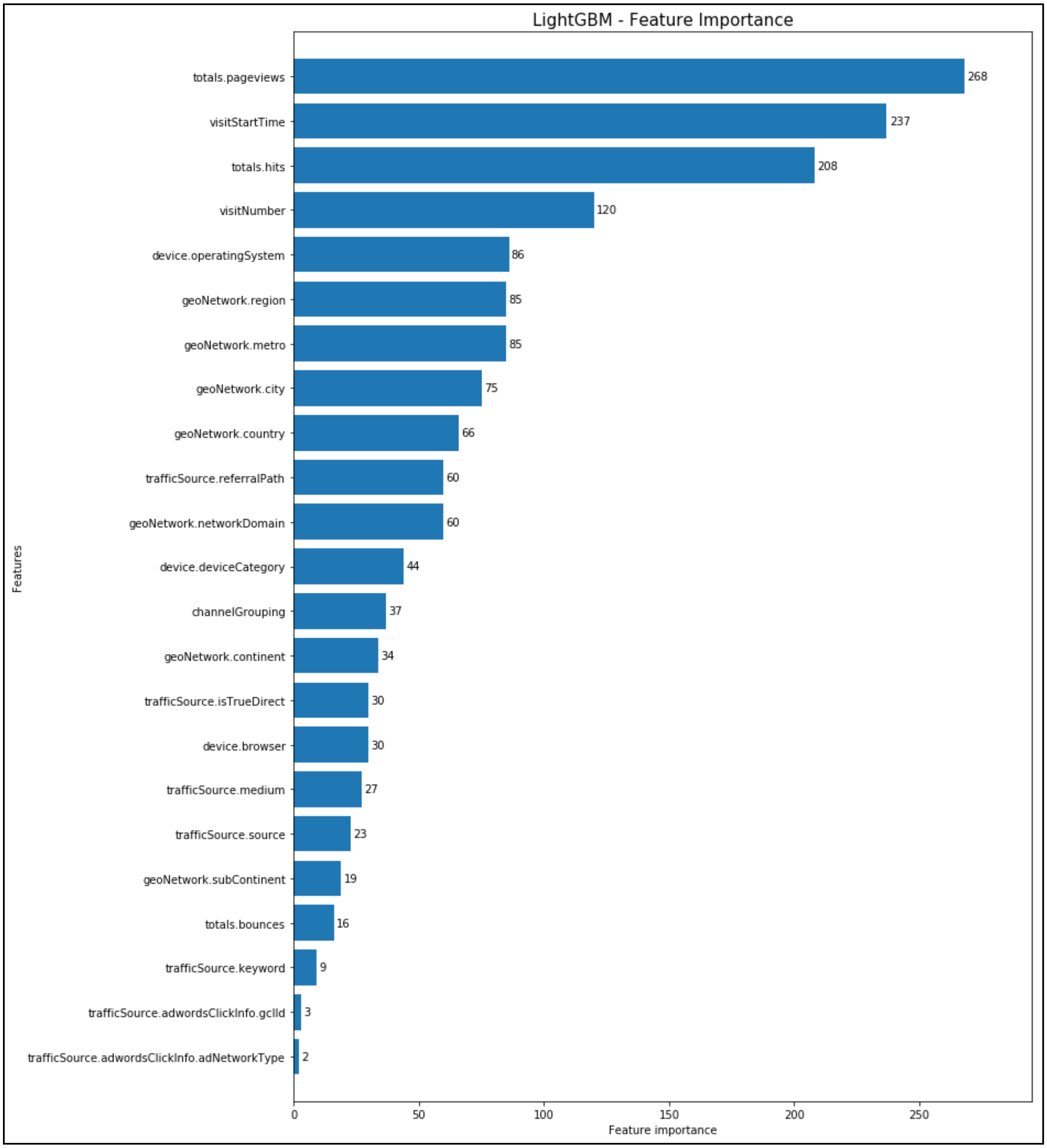


	fullVisitorId	transactionRevenue	PredictedRevenue
0	0000000259678714014	0.0	0.0
1	0000049363351866189	0.0	0.0
2	0000053049821714864	0.0	0.0
3	0000059488412965267	0.0	0.0
4	0000062267706107999	0.0	0.0

V. Evaluate the model

The evaluation of the model is carried out using the evaluation metric,RMSE score between the predicted value and the actual value. In this project as given by the kaggle competition the value to predict is the natural log of the sum of transactions by the user.

Light GBM feature importance :



Complications/Challenges faced in coding process :

The “Google Analytics Customer Revenue Prediction” was a straightforward supervised learning problem therefore there were less complications in the algorithms/model building. Most of the challenges faced was in the data processing part as the dataset had JSON objects in the column which had to be split into different columns and then processed.

The dataset was very huge therefore running on a 16 GB RAM system rendered memory error. To solve this problem a 32 GB RAM system was taken and the data frame was loaded. Even with the 32 GB ram system the time taken for the loading of the dataset was about 11 minutes and then rendered memory error during the label encoding part of the project. To overcome the memory error the dataset was loaded as chunks and then updated to the dataset to be used in the project. There was a wait time for everytime the project was run as it took approximately ten minutes to load the dataset.

Most of the time was spent on the preprocessing and the visualization to gain insights and use the data effectively and chose the model for the dataset. And as the data was a time series data the train and validation dataset was not shuffled but split based on a given time and the model was tested in the unknown time period.

Dataset size :

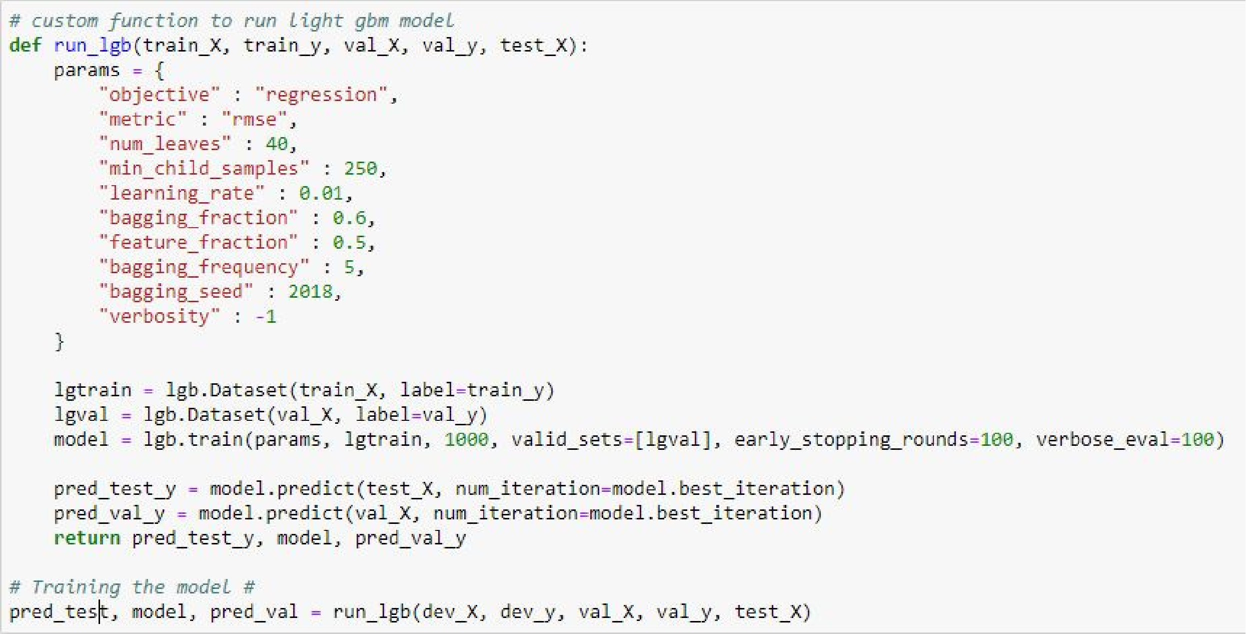
Dataset	File name	Size
Train	train_v2.csv	
Test	test_v2.csv	

The dataset was collected from the “Google Analytics Customer Revenue Prediction” competition page.

Refinement

Adjusting Parameters :

The parameter used in the Light GBM algorithm was changed to be tested for better results.



```
# custom function to run light gbm model
def run_lgb(train_X, train_y, val_X, val_y, test_X):
    params = {
        "objective" : "regression",
        "metric" : "rmse",
        "num_leaves" : 40,
        "min_child_samples" : 250,
        "learning_rate" : 0.01,
        "bagging_fraction" : 0.6,
        "feature_fraction" : 0.5,
        "bagging_frequency" : 5,
        "bagging_seed" : 2018,
        "verbosity" : -1
    }

    lgtrain = lgb.Dataset(train_X, label=train_y)
    lgval = lgb.Dataset(val_X, label=val_y)
    model = lgb.train(params, lgtrain, 1000, valid_sets=[lgval], early_stopping_rounds=100, verbose_eval=100)

    pred_test_y = model.predict(test_X, num_iteration=model.best_iteration)
    pred_val_y = model.predict(val_X, num_iteration=model.best_iteration)
    return pred_test_y, model, pred_val_y

# Training the model #
pred_test, model, pred_val = run_lgb(dev_X, dev_y, val_X, val_y, test_X)
```

The parameters which were changed for the better results in the validation dataset are **`learning_rate`**, **`min_child_samples`** and **`bagging_fraction`**. Changing the parameters had an significant effect on the feature importance selected by the LightGBM algorithm.

I. learning_rate

Different learning rates were tried for better results, the values tried where [0.1,0.01,0.001,0.0001] the model did not improve after 0.01 learning rate.

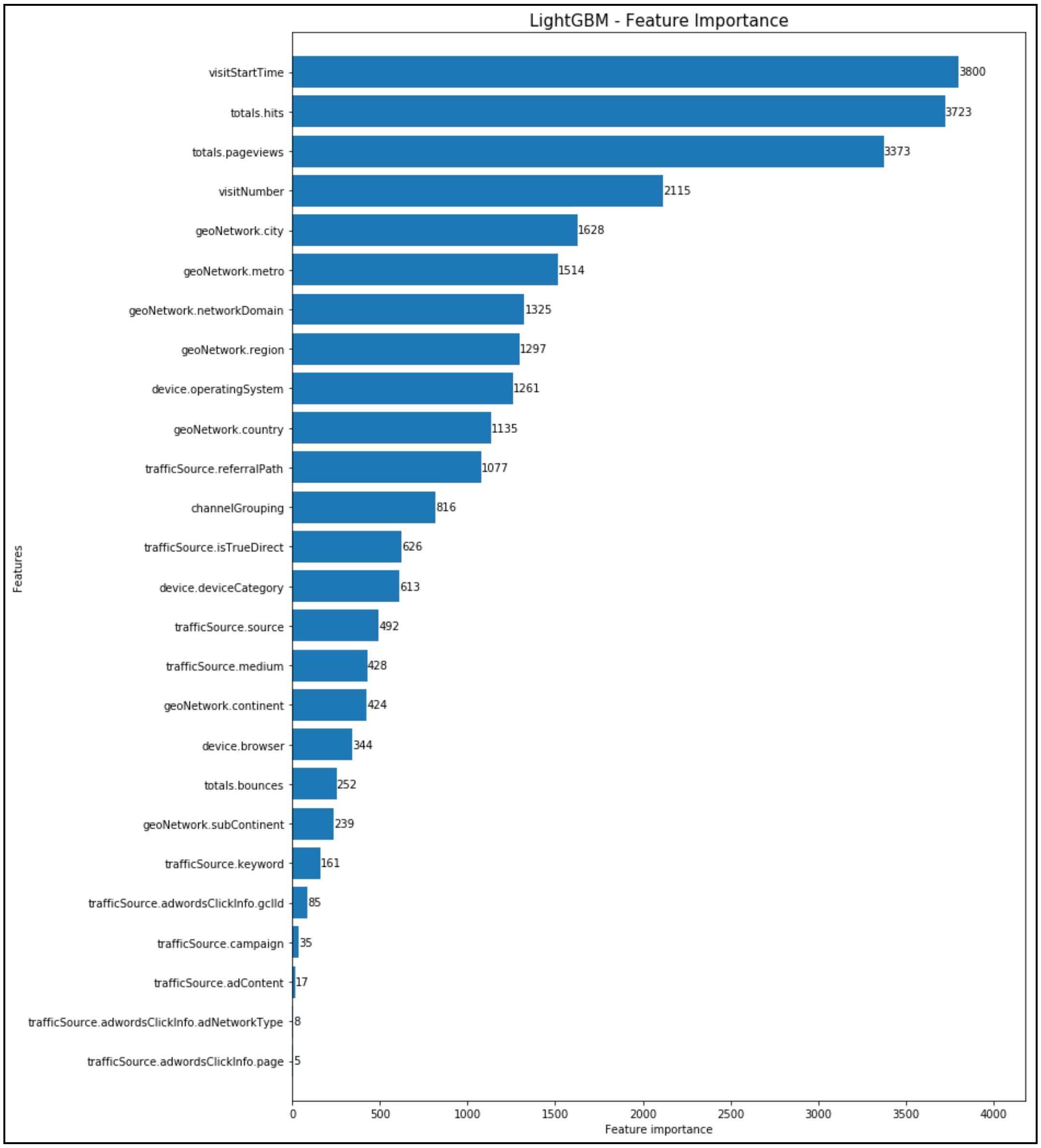
II. min_child_samples

A large min_child_samples were tried for better accuracy, values tried were [50,100,150,200,250,300,350] the model's performance did not improve after 250.

III. bagging_fraction

Bagging was used for faster speed of the model and the values tried were [0.1,0.2,0.3,0.4,0.5,0.6,0.7] the model had the best performance at 0.6 bagging fraction value.

Light GBM Feature Importance :

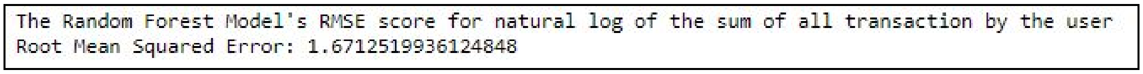


IV. Results

Model Evaluation and Validation

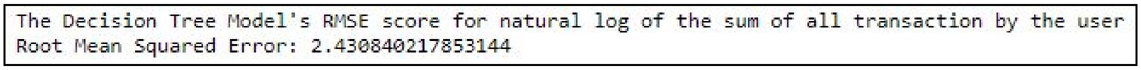
Model evaluation is done using the evaluation metrics. In this project, RMSE is chosen as the evaluation metric and is used to tune the parameters of the chosen model to find the best model.

Benchmark model :



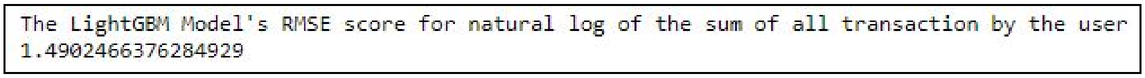
```
The Random Forest Model's RMSE score for natural log of the sum of all transaction by the user
Root Mean Squared Error: 1.6712519936124848
```

Intermediate model :



```
The Decision Tree Model's RMSE score for natural log of the sum of all transaction by the user
Root Mean Squared Error: 2.430840217853144
```

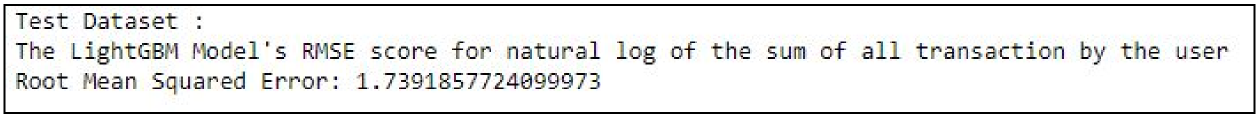
Final Model :



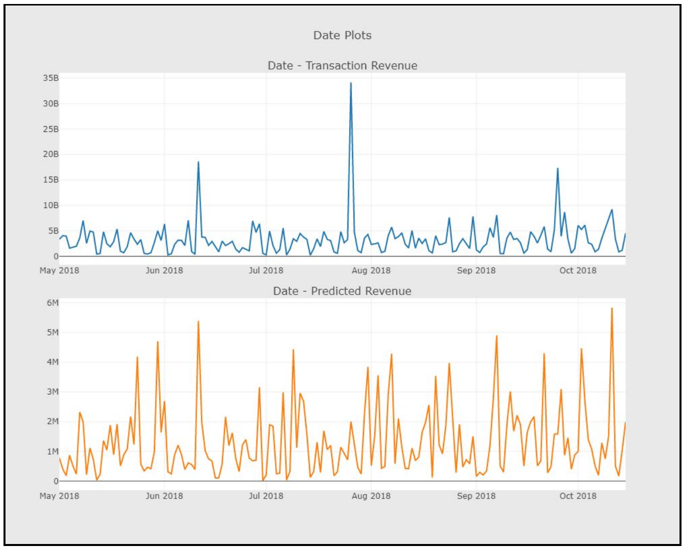
```
The LightGBM Model's RMSE score for natural log of the sum of all transaction by the user
1.4902466376284929
```

Robustness:

Even though the model has fairly reasonable RMSE score the model is not robust enough to use in the real world as the predicted values are way lesser than the actual as seen in the time series graph of the **"Free Visualization"** Looking at the results from the test data set the values predicted are not close to the true value. The model was evaluated on the test data set and had a RMSE score of 1.739.



```
Test Dataset :
The LightGBM Model's RMSE score for natural log of the sum of all transaction by the user
Root Mean Squared Error: 1.7391857724099973
```



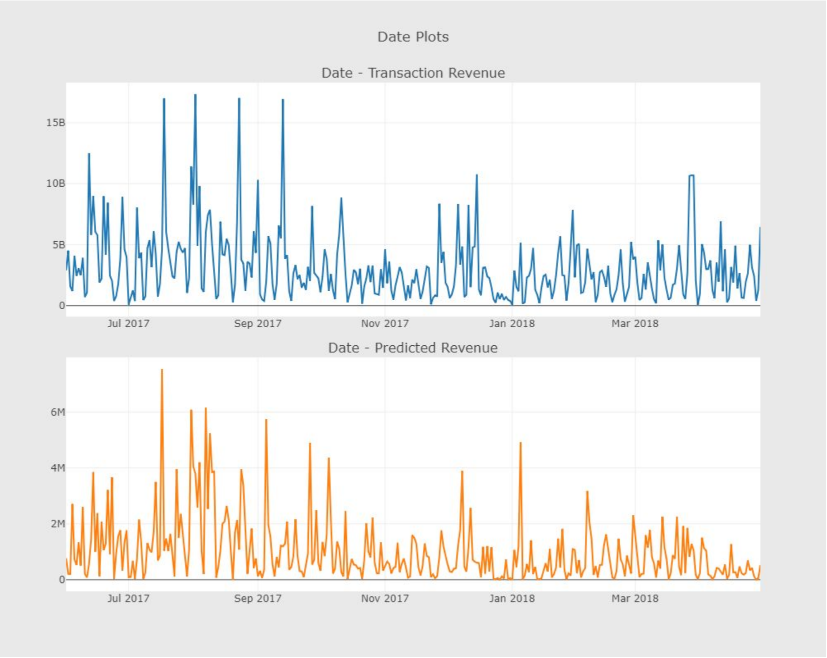
Justification

Overall, the Light GBM model on average performs slightly better than the benchmark model with a score of 1.490 RMSE. Comparing with the intermediate model the LightGBM models performance is significantly better. LightGBM performs faster than the other models chosen in the project. The models score is better than most of the scores from the Kaggle competition.

V. Conclusion

Free-Form Visualization

The free form visualization for the “Google Analytics Customer Revenue Prediction” is done by using the dates plot. Comparing the values obtained from the graph the distribution of the values is quite similar but the actual prediction is way lesser compared to teh original values and the seasonal changes can still be correlated. As seen in the exploratory visualization there is quite a number of changes in page views in certain seasons and therefore have high revenue generation.



Reflection

Summary

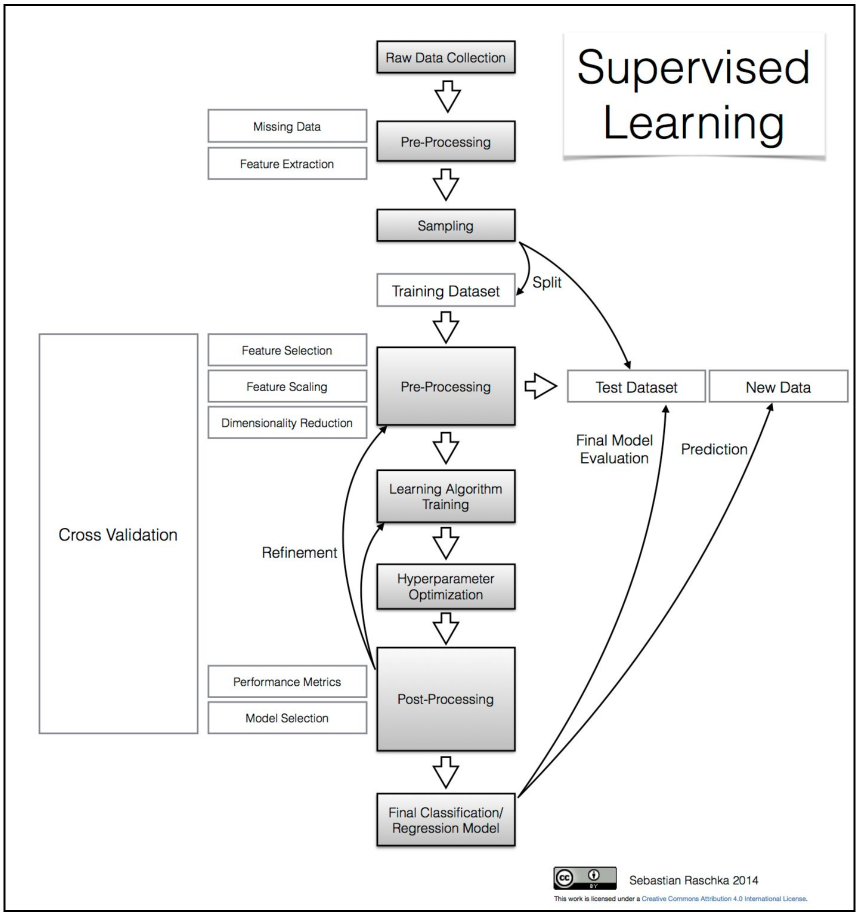
The project was carried out to predict the revenue generated by each user. The dataset had the data collected from the “Google Merchandise Store” and was loaded to a dataframe using the pandas library. The problem taken is a supervised learning problem and needs training data on which a model is built to predict values. The data clean steps were followed to take care of the missing values in the dataset and also remove the columns with low cardinality. The data visualization part carried out to understand the data and obtain insights before running a machine learning algorithm to predict the values.

The libraries used in the project are

- Pandas
- Matplotlib
- Sklearn
- Seaborn
- Plotly
- Lightgbm

Project design

More than seventy percent of the time was spent on the data visualization and the data pre processing steps. To load the data and carry out the data preprocessing steps pandas library was used. The data visualization was helpful in finding the correlation between the different features and the target variable. The libraries used for the visualization of the dataset are Matplotlib, Plotly and Seaborn. The different algorithms and metrics to be used where taken from the sklearn library including the Random forest regressor the benchmark model and the decision tree regressor model.



Sebastian Raschka 2014

The categorical data were label encoded and numbered columns were converted to be float values to be used in the model. The benchmark model is chosen and the dataset is trained and the performance is tested on the validation dataset using the evaluation metric chosen here the RMSE value.In the kaggle competition, it is mentioned to predict the natural log of the sum of all the transactions by a user, therefore, the predicted values natural log of the sum of all transactions is done and the evaluation metric is used to validate the performance. The data is split into test and train based on the dates as they are a time series data and cannot be split using the shuffled split method from the sklearn library. The algorithm to be used for the dataset is chosen with hyperparameter optimization and the models are built and performance is measured using the evaluation metric and compared with the benchmark model.

Interesting Aspects of the Project

The most interesting part of the project is the realization of the amount of data that is collected by the websites, including the device used, region, number of visits to the page and the time spent on the page. The insights that can are uncovered from the data collected is exceptionally intriguing and helped me understand the role of analytics in marketing and the importance of the data for the decision making in marketing campaigns.

Difficult Aspects of the Project

The difficult part of the project is the dataset is huge and requires a high amount of memory to load the dataset. To solve this problem the dataset was loaded as chunks and then added to the data frame still the time taken was comparatively high and every time the project had to be launched there was a wait time.

Improvement

The performance of the model to predict the actual revenue generated is not close to the real values. The model predicts revenue from users and can be used to determine if the user will have revenue or not ,therefore treated as a classification problem.

As there are columns with values which are not available in demo dataset if the complete dataset is given the prediction could be better. To solve the given problem more algorithms can be tried out, some of the algorithms which can be used are,

- Different types of regression algorithms
- Deep Learning
- XGBoost

References

1. G. Brashear, Thomas & Kashyap, Vishal & D. Musante, Michael & Donthu, Naveen. (2009). “A Profile of the Internet Shopper: Evidence from Six Countries”. *The Journal of Marketing Theory and Practice*. 17. 267-282. 10.2753/MTP1069-6679170305.
2. R.A. Mohammed, R.J. Fisher, B.J. Jaworshi, A.M. Cahill, Internet Marketing-Building Advantage in a Network Economy, International edition, McGraw-Hill/Irwin MarketspaceU, 2002, pp. 203 – 313.
3. Malacinski, A., Dominick, S., Hartrick, T.: Measuring Web Traffic, Part1, http://www.ibm..com/developerworks/web/library/wa-mwt1
4. McFadden, C.: Optimizing the Online Business Channel with Web Analytics, http://www.webanalyticsassociation.org/en/art/?9
5. Web Analytics Association, http://www.webanalyticsassociation.org
6. Google Analytics for measuring website performance, Tourism Management, Volume 32, Issue 3, 2011, Pages 477-481, ISSN 0261-5177, https://doi.org/10.1016/j.tourman.2010.03.015.
7. Google Analytics Customer Revenue Prediction - https://www.kaggle.com/c/ga-customer-revenue-prediction
8. Which algorithm takes the crown: Light GBM vs XGBOOST? - https://www.analyticsvidhya.com/blog/2017/06/which-algorithm-takes-the-crown-light-gbm-vs-xgboost/
9. Chai, T. and Draxler, R. R.: Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature, Geosci. Model Dev., 7, 1247-1250, https://doi.org/10.5194/gmd-7-1247-2014, 2014.