PGP DSE - Capstone Project Final Report greatlearning

Project report on

# Prediction of Online Shoppers Purchasing Intention

Submitted towards partial fulfilment of the criteria

for award of PGPDSE by Great Lakes Institute of Management

Submitted By

**Group No. 4 Batch: 2019**

Group Members:

| | |
|---|---|
| **Dedeepya** | **PGPDSEFEB19** |
| **Ghouse** | **PGPDSEFEB19** |
| **Rakesh** | **PGPDSEFEB19** |
| **Shiva** | **PGPDSEFEB19** |

Research Supervisor:

**Anjanaa Agarwal**

# ABSTRACT

In this project, we proposed a real-time online shopper behavior analysis system consisting of two modules which simultaneously predicts the visitor's shopping intent and Web site abandonment likelihood. Here, we predict the purchasing intention of the visitor using aggregated page view data kept track during the visit along with some session and user information. The extracted features are fed to several classifiers as input. We use oversampling and feature selection preprocessing steps to improve the performance and scalability of the classifiers. Another finding is that although clickstream data obtained from the navigation path followed during the online visit convey important information about the purchasing intention of the visitor, combining them with session information-based features that possess unique information about the purchasing interest improves the success rate of the system. The modules are used together to determine the visitors which have purchasing intention but are likely to leave the site in the prediction horizon and take actions accordingly to improve the Web site abandonment and purchase conversion rates. Our findings support the feasibility of accurate and scalable purchasing intention prediction for virtual shopping environment using clickstream and session information data.

The purpose of this research is to explore the factors that affect consumer purchase intention in online shopping. The factors that affect online customer intention are perceived benefit, perceived risk, hedonic motivation, trust, and attitude toward online shopping. With the phenomenal growth of the web, there is an ever increasing volume of data and information published in numerous web-pages. A web page typically contains a mixture of many kind of information e.g. main contains, advertisements, navigational panels, copyright blocks etc... for a particular application only part of information is useful and the rest are noise.

The Internet has developed into a new distribution channel and online transactions are rapidly increasing. This has created a need to understand how the consumer perceives online purchases. The purpose of this dissertation was to examine if there are any particular factors that influence the online consumer.

.

**Keywords** — Online Customer, Transactions, Satisfaction, Behavior, Revenue.

❑ Techniques: Machine Learning – Supervised Learning Classification

❑ Tools: Python

❑ Domain: Online Shopping

## CERTIFICATE OF COMPLETION

I hereby certify that the project titled "**Prediction of Online Shoppers Purchasing Intention"** for case resolution was undertaken and completed under my supervision by **Dedeepya, Ghouse, Rakesh, Shiva** of PGP in Data Science and Engineering (PGP – DSE).

**Date:**

**Place:** Hyderabad **Anjanaa Agarwal**

# DECLARATION

I declare that the project entitled **“ Prediction of Online Shoppers Purchasing Intention”** is a project work carried out by me under the supervision and guidance of **Anjanaa Agarwal,** for the award of degree PGP DSE, and this has not been previously submitted for the award of any Degree, Diploma or other similar title of any other University/ Institute.

**Date:**

**Place:** Hyderabad

**Group members:**

**Dedeepya**

**Ghouse**

**Rakesh**

**Shiva**

## ACKNOWLEDGEMENT

In any work there are contributions from all quarters. Since they cannot all be mentioned, I will name the few who have contributed the most.

I would like to acknowledge my supervisor **Anjanaa Agarwal** for providing me the necessary guidance and valuable support throughout this research project. I value the assistance of Great Learning, Hyderabad campus. Learning from their knowledge helped me to become passionate about my research topic.

I will be failing in my duty if I don’t express my gratitude for my team members, for their valuable suggestions and cheerful assistance during the implementation phase of this project.

# TABLE OF CONTENTS

# 1. INTRODUCTION

Online shopping is the process whereby consumers directly buy goods, services etc. from a seller interactively in real-time without an intermediary service over the internet. Online shopping is the process of buying goods and services from merchants who sell on the Internet. Since the emergence of the World Wide Web, merchants have sought to sell their products to people who surf the Internet. Shoppers can visit web stores from the comfort of their homes and shop as they sit in front of the computer.Consumers buy a variety of items from online stores.Many people choose to conduct shopping online because of the convenience.Online shopping allows you to browse through endless possibilities, and even offers merchandise that's unavailable in stores. Shopping via the internet eliminates the need to shift through a store's products with potential buys like pants, shirts, belts and shoes all slung over one arm. Online shopping also eliminates the catchy, yet irritating music, as well as the hundreds, if not thousands, of other like-minded individuals who seem to have decided to shop on the same day.

Online shopping is a terminology that refers to people who make purchases over internet. Online shopping is attractive for many consumers because of the benefits received by consumers such as easy to search for purchase information, time saving, products and store comparison and many others.Customer online purchase intention is defined as the construct that gives the strength of a customer's intention to purchase online, it is observed that online purchase intention to be a more appropriate measure of intention to use a web site when assessing online consumer behaviour. Customer loyalty is critical to the online vendor's survival and success. The study provides evidence that online trust is built through order fulfilment, privacy, responsiveness and contact**.**

## 1.1 PROBLEM STATEMENT

- ❑ **Purpose of this project is to identify user behaviour patterns to effectively understand features that influence the purchasing decisions**
- ❑ **To identify the list of attributes that are finally converting into revenues by building classification models using existing data.**

## 1.2 RESEARCH PURPOSE

Analyzing the Customer behavior is not a new phenomenon. The renowned marketing expert Philip Kotlehas published several works on the topic of customer behavior theories. These theories have been used for many years not only to understand the consumer but also create a marketing strategy that will attract the customer efficiently . Hence Understanding and identifying the consumer is closely related to the directions a company will take with their marketing strategy. These theories can also be applied to identify the online Customer and to create certain customers segments. However, some distinctions must still be made when considering traditional consumer behavior and online Customer behavior.

The purpose of this research is primarily to identify and get insight in to what main factors the online consumer takes into consideration when purchasing Online. Further I will investigate if any segments can be established by identifying the consumers and how these segments relate to the identified factors. The Findings of this research will be outlined a simplifications for online retailers in order to enhance their consumer knowledge and increase their online marketing strategy effectiveness

## 1.3 RESEARCH APPROACH AND STRATEGY

There are two most commonly used research approaches the inductive and the deductive method. The inductive research method attempts to setup a theory by using collected data while the deductive research approach attempts to find the theory first and then test it to the observed data. I chose a deductive research approach for my study as I would move from the more general to the specific. I will present the theoretical findings on consumer behavior in the next chapter after which I will present my questionnaire in chapter four where I present my collected primary data.

When collecting data to approach the purpose of a research there are two ways in which the data can be collected. In order to acquire a general knowledge about the topic secondary data is primarily used and is one of the ways by which data can be collected. These Conway to collect data is the primary data collection. Usually when a study is conducted secondary data is not sufficient enough and needs to be completed with primary data which is collected by the researcher

# 2. LITERATURE REVIEW

## 2.1 HANDLE IMBALANCED CLASSIFICATION PROBLEMS IN MACHINE LEARNING

While performing the conventional machine learning on the data which is imbalanced the model will be inaccurate and biased. In this case number of observations in one class will be significantly lesser than other. This problem is predominant in scenarios where anomaly detection is crucial like electricity pilferage, fraudulent transactions in banks, identification of rare diseases, etc. Standard classifier algorithms like Decision Tree and Logistic Regression have a bias towards classes which have number of instances. This happens because Machine Learning Algorithms are usually designed to improve accuracy by reducing the error. Thus, they do not take into account the class distribution / proportion or balance of classes. They tend to only predict the majority class data. The features of the minority class are treated as noise and are often ignored.

**Author: Upasana |Consultant of Data & Analytics in KPMG.**

https://www.analyticsvidhya.com/blog/2017/03/imbalanced-classification-problem/

## 2.2 CLASSIFICATION MODELS

To determine what classification scheme fit our data the best, we implemented several of the most common classifiers used for supervised learning. To implement the classification, we made use of the machine learning development kit in Python called Scikit-Learn. The following are the classification models we employed due to their suitability for binary classification problems:

### 2.2.1 LOGISTIC REGRESSION

Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables. Sometimes logistic regressions are difficult to interpret; the Intellectus Statistics tool easily allows you to conduct the analysis, then in plain English interprets the output. The dependent variable should be dichotomous in nature (e.g., presence vs. absent).

There should be no outliers in the data, which can be assessed by converting the continuous predictors to standardized scores, and removing values below -3.29 or greater than 3.29. There should be no high correlations (multicollinearity) among the predictors. This can be assessed by a correlation matrix among the predictors. Tabachnick and Fidell (2013) suggest that as long correlation coefficients among independent variables are less than 0.90 the assumption is met. At the centre of the logistic regression analysis is the task estimating the log odds of an event. Mathematically, logistic regression estimates a multiple linear regression function defined as:

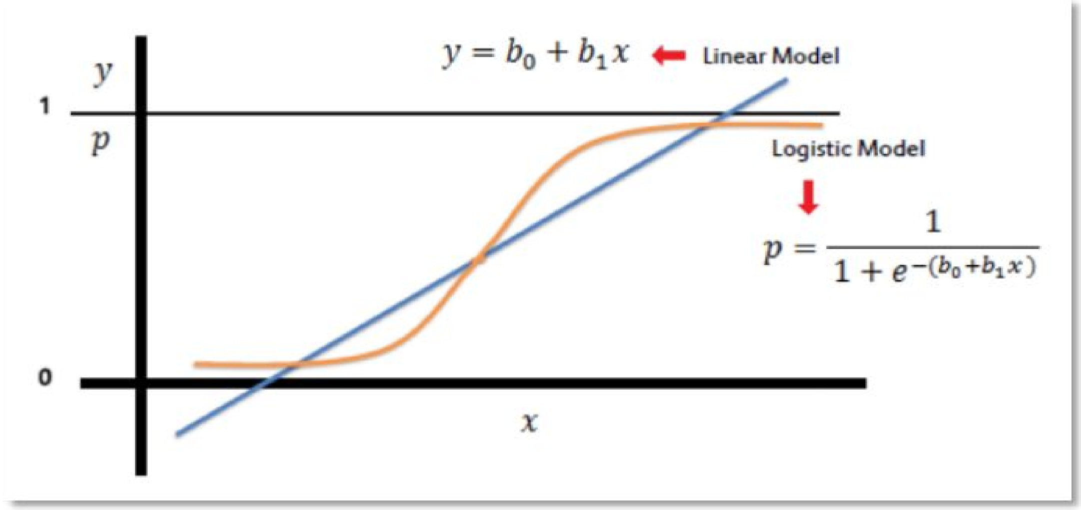


Fig: Sigmoid Activation Function

### 2.2.2 DECISION TREES

Decision Trees are trees that classify instances by sorting them based on feature values. Each node in a decision tree represents a feature in an instance to be classified, and each branch represents a value that the node can assume. Instances are classified starting at the root node and sorted based on their feature values [19].Decision tree learning, used in data mining and machine learning, uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value. More descriptive names for such tree models are classification trees or regression trees [20].Decision tree classifiers usually employ post-pruning techniques that evaluate the performance of decision trees, as they are pruned by using a validation set. Any node can be removed and assigned the most common class of the training instances that are sorted to it [19]

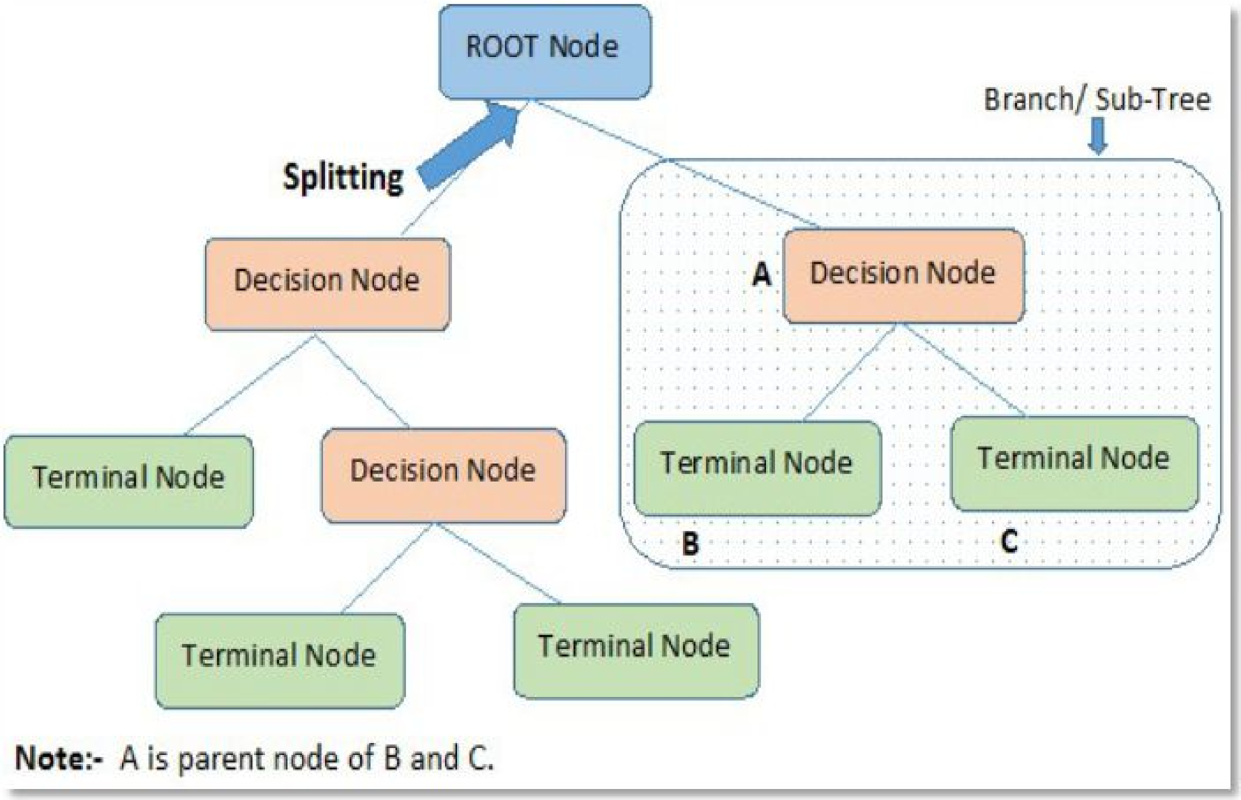


**Fig 2.2: Classification in a basic decision tree proceeds from top to bottom.**

# 3. DATASET INFORMATION

## 3.1 DATASET LINK

https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset

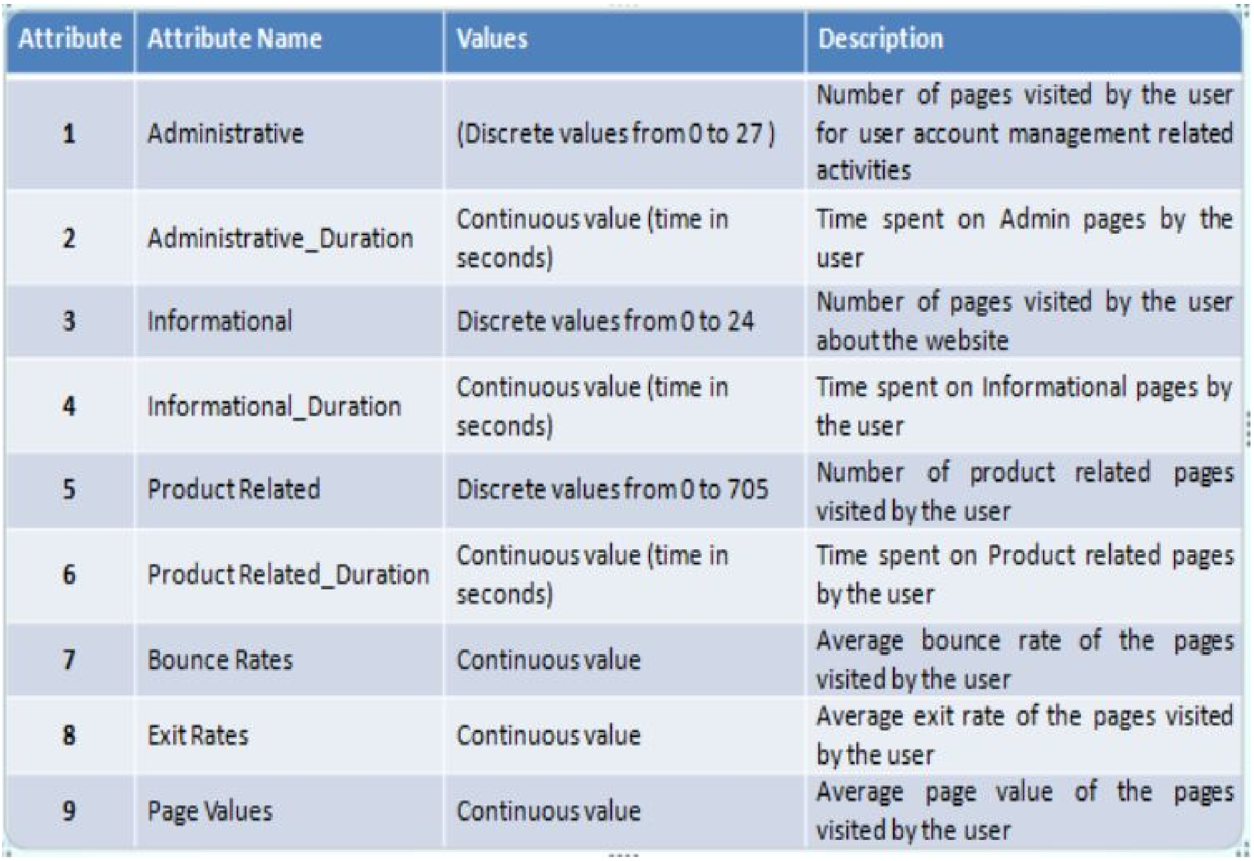
## 3.2 DATASET INFORMATION

The dataset consists of feature vectors belonging to 12,330 sessions. The dataset was formed so that each session would belong to a different user in a 1-year period to avoid any tendency to a specific campaign, special day, user profile, or period.

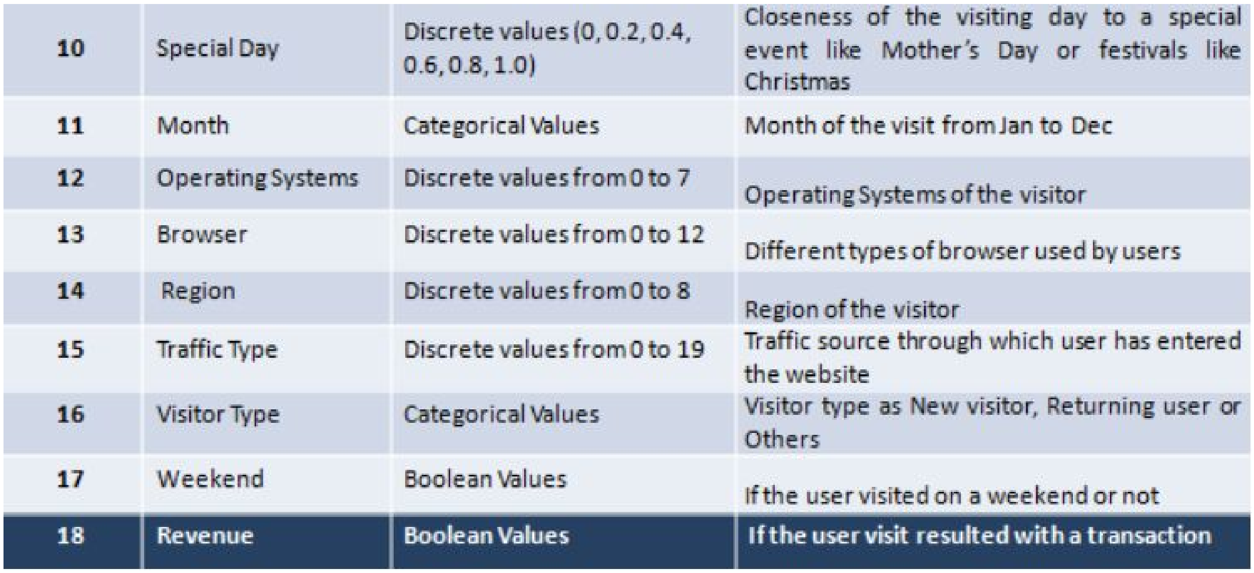
## 3.3 ATTRIBUTE INFORMATION

The dataset consists of 10 numerical and 8 categorical attributes. The 'Revenue' attribute can be used as the class label.

"Administrative", "Administrative Duration", "Informational", "Informational Duration", "Product Related" and "Product Related Duration" represent the number of different types of pages visited by the visitor in that session and total time spent in each of these page categories. The values of these features are derived from the URL information of the pages visited by the user and updated in real time when a user takes an action, e.g. moving from one page to another. The "Bounce Rate", "Exit Rate" and "Page Value" features represent the metrics measured by "Google Analytics" for each page in the e-commerce site. The value of "Bounce Rate" feature for a web page refers to the percentage of visitors who enter the site from that page and then leave ("bounce") without triggering any other requests to the analytics server during that session. The value of "Exit Rate" feature for a specific web page is calculated as for all pageviews to the page, the percentage that were the last in the session. The "Page Value" feature represents the average value for a web page that a user visited before completing an e-commerce transaction. The "Special Day" feature indicates the closeness of the site visiting time to a specific special day (e.g. Mother’s Day, Valentine's Day) in which the sessions are more likely to be finalized with transaction. The value of this attribute is determined by considering the dynamics of e-commerce such as the duration between the order date and delivery date. For example, for Valentina’s day, this value takes a nonzero value between February 2 and February 12, zero before and after this date unless it is close to another special day, and its maximum value of 1 on February 8. The dataset also includes operating system, browser, region, traffic type, visitor type as returning or new visitor, a Boolean value indicating whether the date of the visit is weekend, and month of the year.



| Attribute | Attribute Name | Values | Description |
|---|---|---|---|
| 1 | Administrative | (Discrete values from 0 to 27 ) | Number of pages visited by the user for user account management related activities |
| 2 | Administrative_Duration | Continuous value (time in seconds) | Time spent on Admin pages by the user |
| 3 | Informational | Discrete values from 0 to 24 | Number of pages visited by the user about the website |
| 4 | Informational_Duration | Continuous value (time in seconds) | Time spent on Informational pages by the user |
| 5 | Product Related | Discrete values from 0 to 705 | Number of product related pages visited by the user |
| 6 | Product Related_Duration | Continuous value (time in seconds) | Time spent on Product related pages by the user |
| 7 | Bounce Rates | Continuous value | Average bounce rate of the pages visited by the user |
| 8 | Exit Rates | Continuous value | Average exit rate of the pages visited by the user |
| 9 | Page Values | Continuous value | Average page value of the pages visited by the user |

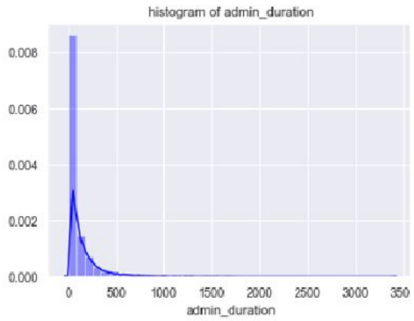


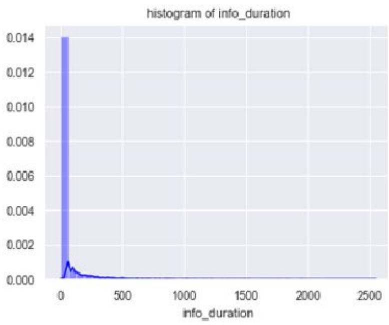
| | | | |
|---|---|---|---|
| 10 | Special Day | Discrete values (0, 0.2, 0.4, 0.6, 0.8, 1.0) | Closeness of the visiting day to a special event like Mother's Day or festivals like Christmas |
| 11 | Month | Categorical Values | Month of the visit from Jan to Dec |
| 12 | Operating Systems | Discrete values from 0 to 7 | Operating Systems of the visitor |
| 13 | Browser | Discrete values from 0 to 12 | Different types of browser used by users |
| 14 | Region | Discrete values from 0 to 8 | Region of the visitor |
| 15 | Traffic Type | Discrete values from 0 to 19 | Traffic source through which user has entered the website |
| 16 | Visitor Type | Categorical Values | Visitor type as New visitor, Returning user or Others |
| 17 | Weekend | Boolean Values | If the user visited on a weekend or not |
| **18** | **Revenue** | **Boolean Values** | **If the user visit resulted with a transaction** |

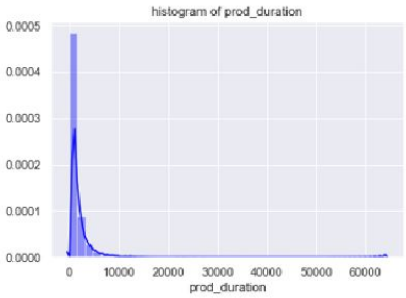
# 4. EXPLORATORY DATA ANALYSIS

## 4.1 Continuous Variable distributions

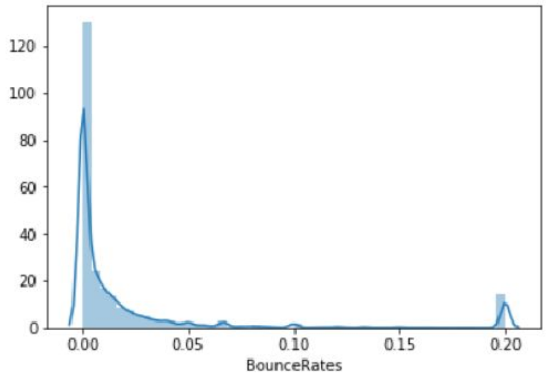
EDA is a general approach to exploring datasets by means of simple summary statistics and graphic visualizations in order to gain a deeper understanding of the data.

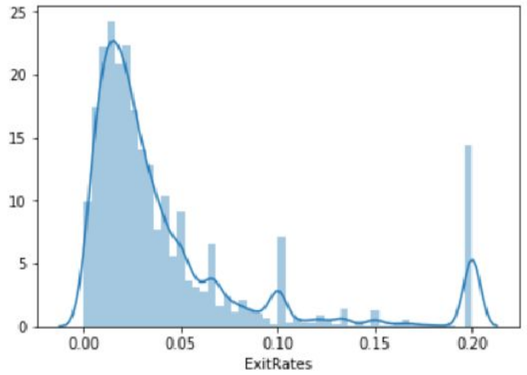






admin_duration, info_duration & product_duration attributes are Pareto distributed(long tailed) which explains that most online customers are spending very few seconds or not spending any time on Administrative and informational related pages.

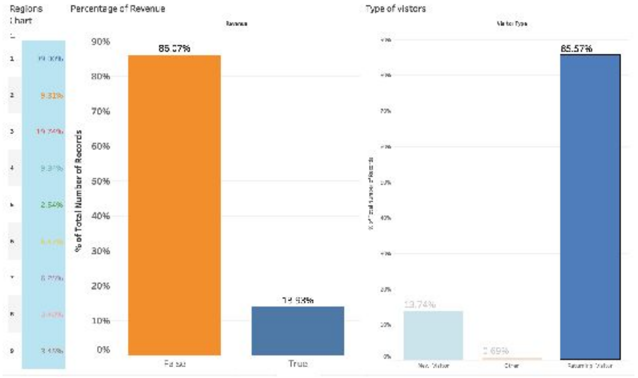


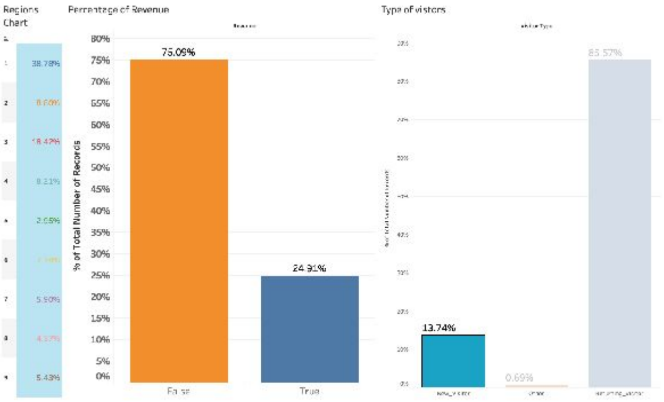


Bounce Rate : The Percentage of single-engagement sessions.
Exit Rate : The Percentage of exits on a page.

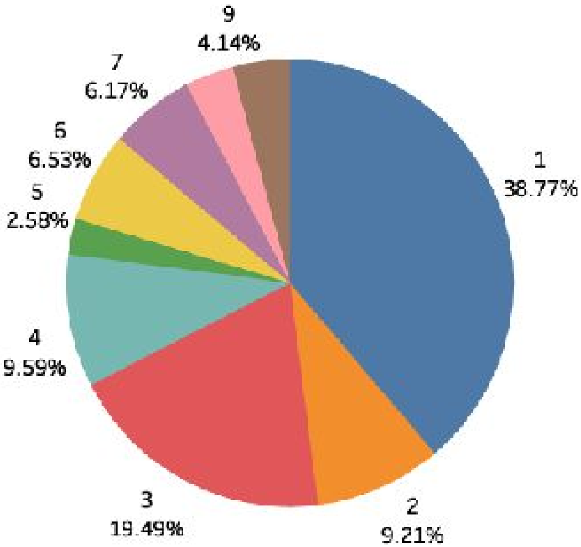
Bounce_rate attribute is having Pareto distribution(long tailed) which explains that most online customers are spending very few seconds or not spending anytime on these related pages on the other hand Exit Rate is also having Pareto distribution(long tailed) but comparatively much better distribution because the customers are spending more time in pages.

## 4.2 Revenue Based Analysis:

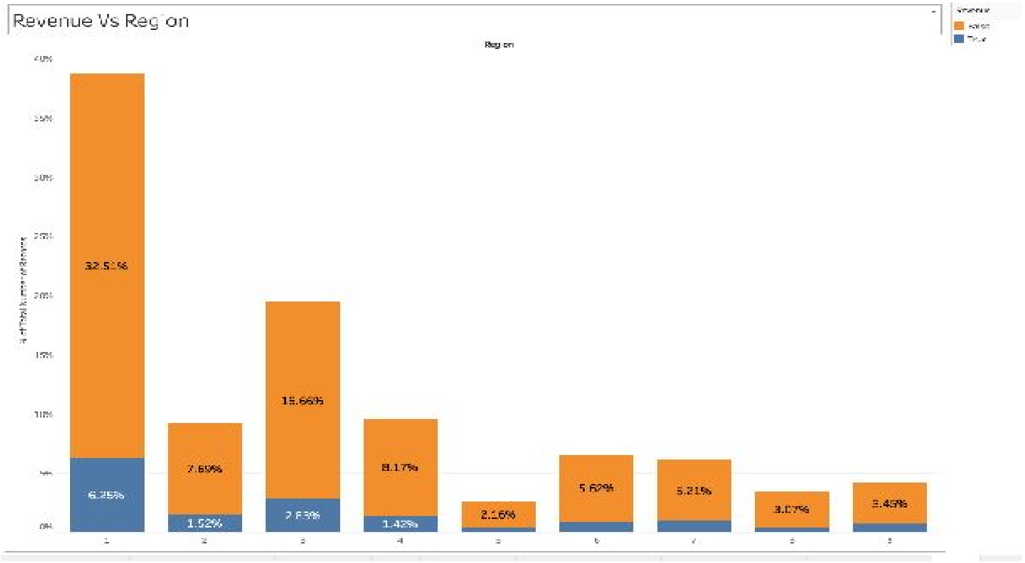




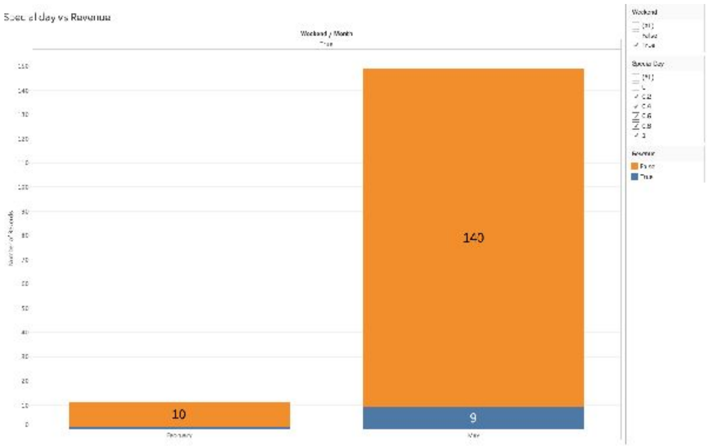
**Observation**: New visitors Revenue conversion rate is higher than the returning Visitors and others.

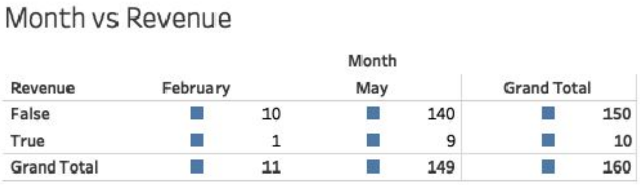


**Observation:** Most of the Customers are from Region “1”..... I.e. 38.77% and Only 6.25% of the customers are Generating Revenue.



**Observation:** Percentage of Revenue Generating and not generating in Region and less customers are from region  5



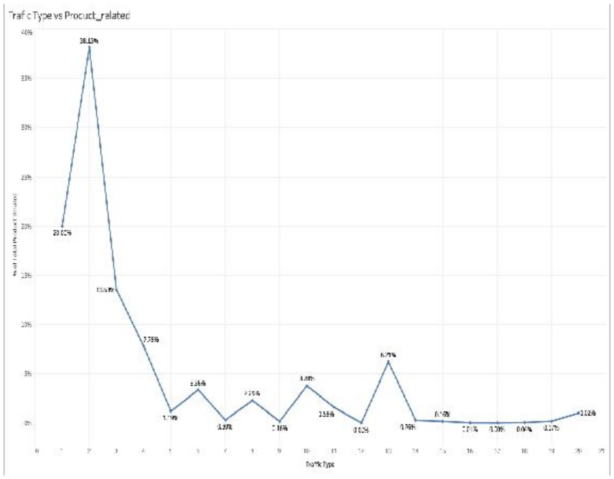


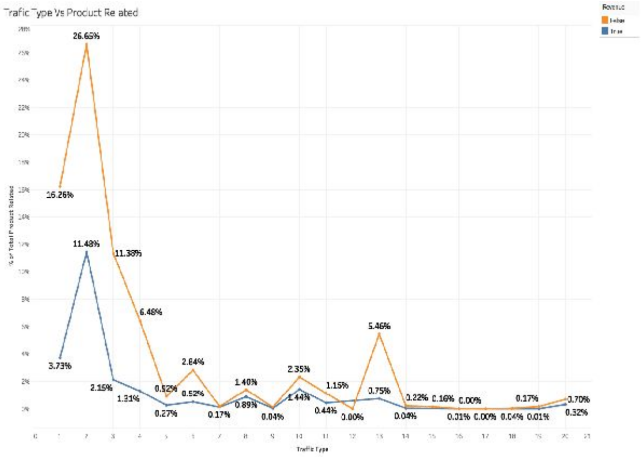
Month vs Revenue

| Revenue | February | May | Grand Total |
|---|---|---|---|
| | | Month | |
| False | ■ 10 | ■ 140 | ■ 150 |
| True | ■ 1 | ■ 9 | ■ 10 |
| Grand Total | ■ 11 | ■ 149 | ■ 160 |

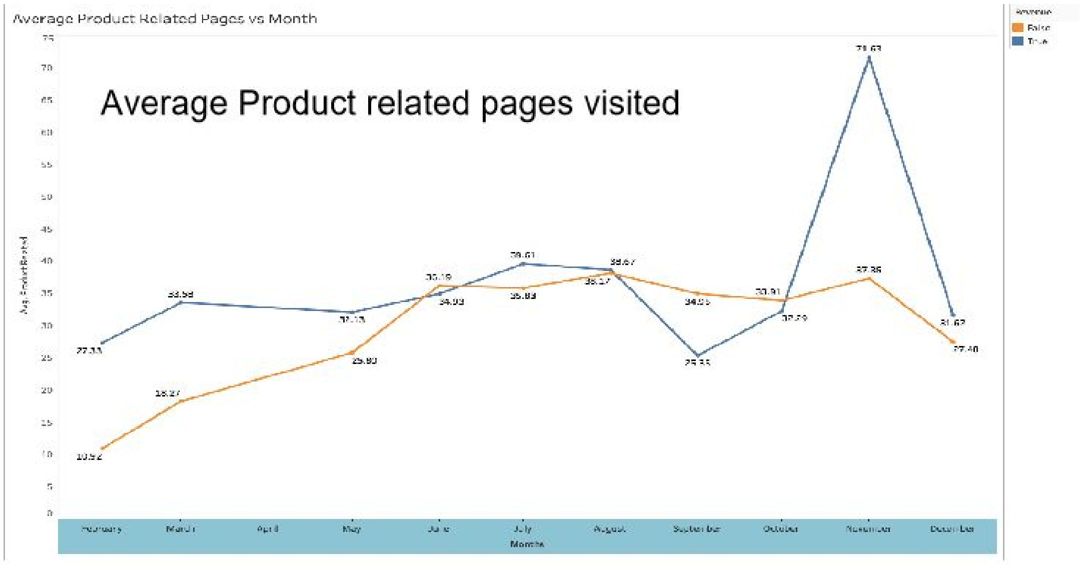
**Observation:** There are 160 days which are special days and weekends and conversion rate count of customers to generate revenue is 9.

## 4.3 Visit Based Analysis:

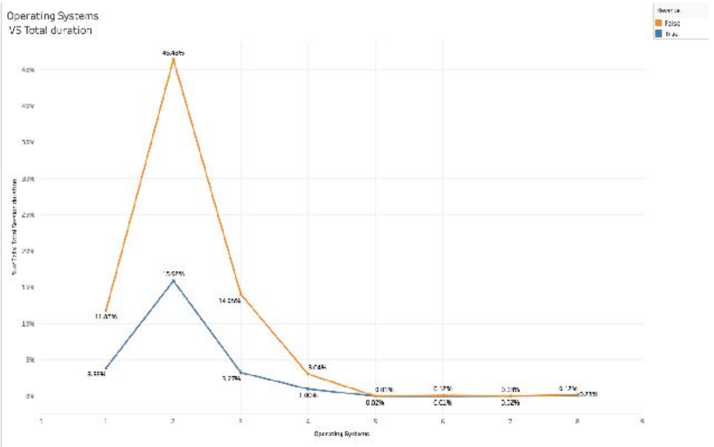


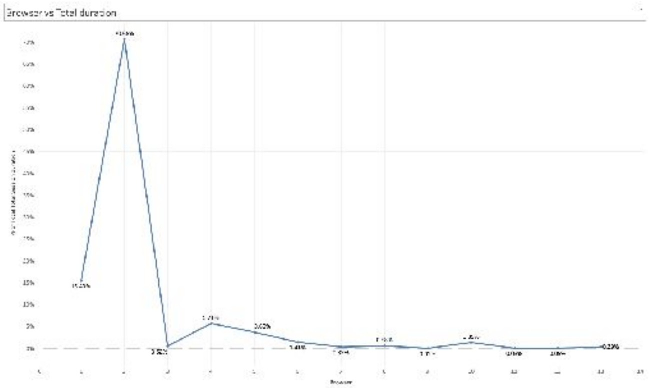


**Observation:** From the above plots we can observer that the maximum number of traffic is from the traffic type 2 and minimum traffic is from the traffic type 16 and 17.



**Observation:** It is observed that Revenue Generating visitors are visiting more product related pages and on average it is high in November and low in September.





**Observation:** Operating system Type 2 has highest customer visits when compared to other OS and Browser Type "2" is having highest customer visits

## 4.4 HANDLING MISSING VALUES

In any real world dataset there are usually few null values. It doesn't really matter whether it is regression, classification or any other kind of problem no model can handle these NULL or NaN values on its own so we need to intervene.First of all we need to check whether we have null values in our dataset or not. We can do that using the isnull() method.There are various ways for us to handle this problem. The easiest way to solve this problem is by dropping the rows or columns that contain null values.

dropna() takes various parameters like:

axis — We can specify axis=0 if we want to remove the rows and axis=1 if we want to remove the columns.

how — If we specify how = 'all' then the rows and columns will only be dropped if all the values are NaN. By default 'how' is set to 'any'.

thresh — It determines the threshold value so if we specify thresh=5 then the rows having less than 5 real values will be dropped.

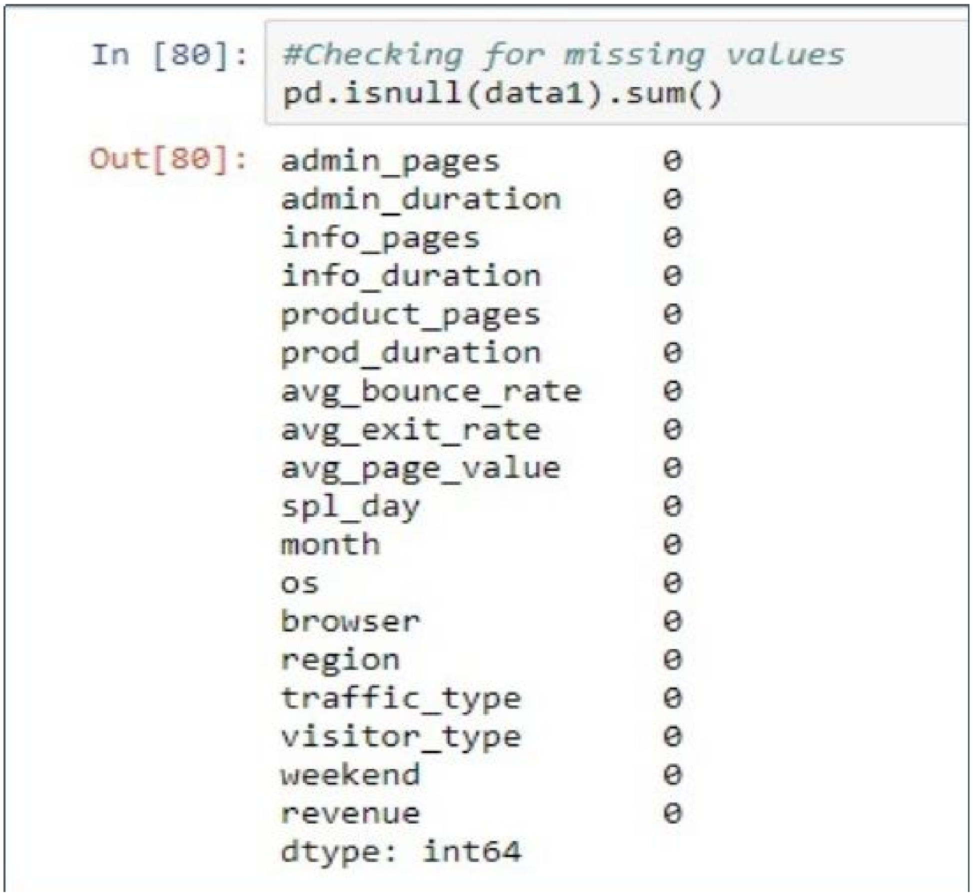
subset —If I have 4 columns A,B,C and D then if i specify subset=['C'] then only the rows that have their C value as NaN will be removed.

inplace — By default no changes will be made to your data frame. So if you want these changes to reflect onto your data frame then you need to use inplace = True.

However it is not the best option to remove the rows and columns from our dataset as it can lead to loss of valuable information. So if you have 300K data points then removing 2–3 rows won't affect your dataset, whereas if you only have 100 data points and out of which 20 have NaN values for a particular field then you can't simply drop those rows. In real life datasets it can happen quite often that you have large number of NaN values for a particular field.

Suppose we are collecting the data from a survey, then it is possible that there could be an optional field which let's say 20% people left blank. So when we get the dataset then we need to understand that the remaining 80% data is still useful so rather than dropping these values we need to somehow substitute the missing 20% values. We can do this with the help of Imputation.Imputation is simply the process of substituting the missing values of our dataset. We can do this by defining our own customized function or we can simply perform imputation by using the Imputer class provided by sklearn.In Imputer() you can pass on the axis and strategy. strategy could be mean ,median etc.

There are NO MISSING VALUES are present in the given Dataset.



```
In [80]: #Checking for missing values
         pd.isnull(data1).sum()

Out[80]: admin_pages        0
         admin_duration     0
         info_pages         0
         info_duration      0
         product_pages      0
         prod_duration      0
         avg_bounce_rate    0
         avg_exit_rate      0
         avg_page_value     0
         spl_day            0
         month              0
         os                 0
         browser            0
         region             0
         traffic_type       0
         visitor_type       0
         weekend            0
         revenue            0
         dtype: int64
```

# 4.5 ENCODING METHODS

### 4.5.1 One Hot Encoding

For categorical variables where no such ordinal relationship exists, the integer encoding is not enough. In fact, using this encoding and allowing the model to assume a natural ordering between categories may result in poor performance or unexpected results (predictions halfway between categories).In this case, a one-hot encoding can be applied to the integer representation. This is where the integer encoded variable is removed and a new binary variable is added for each unique integer value. In the “color” variable example, there are 3 categories and therefore 3

binary variables are needed. A “1” value is placed in the binary variable for the color and “0” values for the other colors. The binary variables are often called “dummy variables” in other fields, such as statistics

### 4.5.2 Label Encoding

Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning. There are 2 pretty simple and neat techniques to transform ordinal CVs.

1. Using map() function —

   size_mapping = {'M':1,'L':2}

   df_cat['size'] = df_cat['size'].map(size_mapping)

   Here M will be replaced with 1 and L with 2.

2. Using Label Encoder —

   from sklearn.preprocessing import LabelEncoder class_le = LabelEncoder()

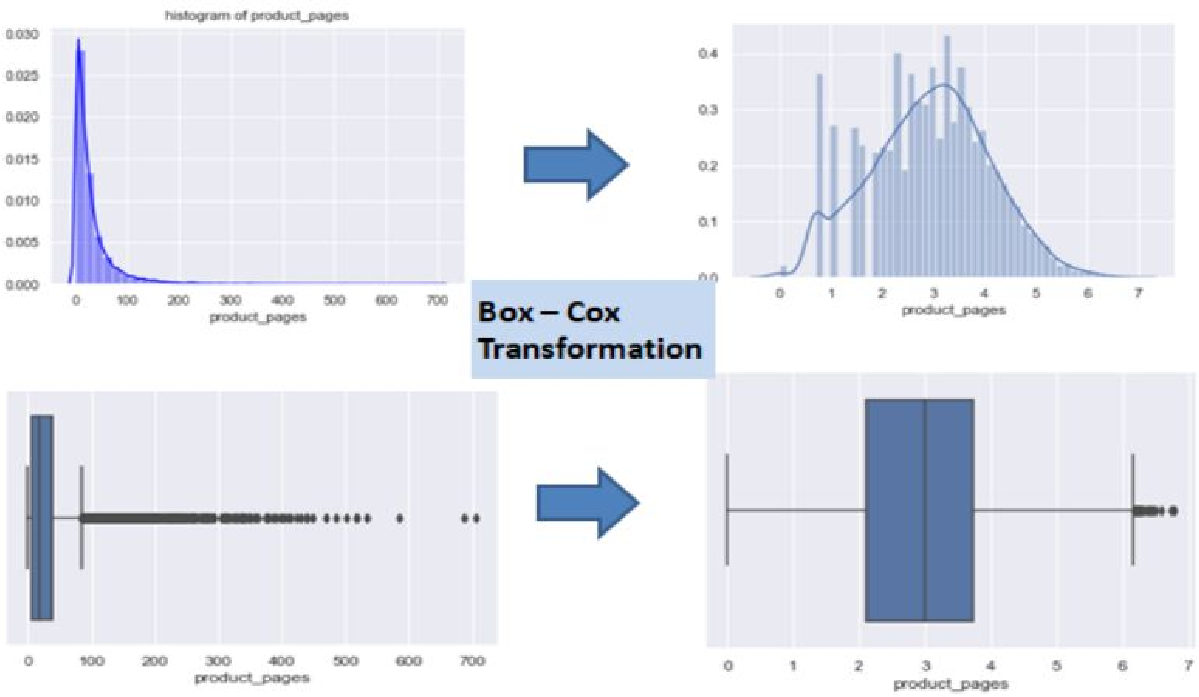
   df_cat['classlabel'] = class_le.fit_transform(df_cat['classlabel'].values)

   Here class1 will be represented with 0 and class2 with 1.

# 4.6 BOX-COX TRANSFORMATION

A Box Cox transformation is a way to transform non-normal dependent categorical variables into a normal shape. Normality is an important assumption for many statistical techniques; if your data isn't normal, applying a Box-Cox means that you are able to run a broader number of tests.

| Common Box-Cox Transformations | |
|---|---|
| Lambda value (λ) | **Transformed data (Y')** |
| -3 | $Y^{-3} = 1/Y^{3}$ |
| -2 | $Y^{-2} = 1/Y^{2}$ |
| -1 | $Y^{-1} = 1/Y^{1}$ |
| -0.5 | $Y^{-0.5} = 1/(\sqrt{(Y)})$ |
| 0 | **log(Y)**** |
| 0.5 | $Y^{0.5} = \sqrt{(Y)}$ |
| 1 | $Y^{1} = Y$ |
| 2 | $Y^{2}$ |
| 3 | $Y^{3}$ |



Here, Box Cox transformation applied on independent variable named Product pages in order to transform highly skewed independent variable into a normal shape and which also helps us to reduce the outlier effect.

## 4.7 SPLITTING THE DATASET INTO TRAINING SET AND TEST SET

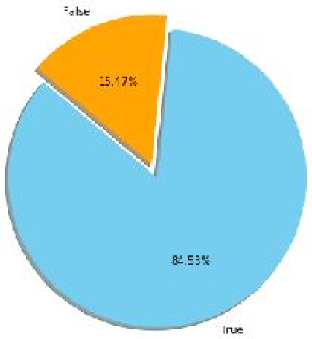
Now we need to split our dataset into two sets — a Training set and a Test set. We will train our machine learning models on our training set, i.e. our machine learning models will try to understand any correlations in our training set and then we will test the models on our test set to check how accurately it can predict. A general rule of the thumb is to allocate 80% of the dataset to training set and the remaining 20% to test set. For this task, we will import test_train_split from model_selection library of scikit.

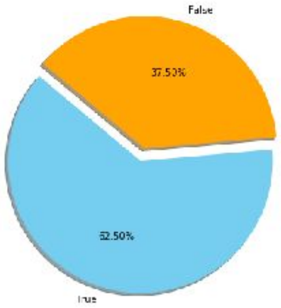
## 4.8 DATA IMBALANCE

The dataset that will be used to train the model has some challenges. In particular it has been subjected to the imbalanced dataset problem, whereby not all classes have similar number of instances. To handle the imbalanced classes there were a few possible solutions and these are discussed in more detail above. One way to deal with imbalanced dataset is by down sampling which involves reducing the number of instances by tossing away some instances of classes with high number of instances to match the classes with lower number of instances. This has the advantage that it is simple to implement and has proven to be more effective than up sampling. However this method has the major disadvantage that useful and valuable instances are thrown away upon training.

Another common solution to deal with imbalanced dataset is by up sampling which involves generating synthetic data or making multiple copies of the instances for classes with low number of instances (minority class) to match the majority class. This has the advantage that it does not throw away useful instances. However this method has proved in the past to cause overfitting and can be computationally intensive if the number of instances and classes are very large. A common solution that changes how the model learns instead of the dataset itself is known as Cost Sensitive Learning. It works by changing the loss/error function to give a different error value to False Positive and False Negatives depending on which is more significant, and hence model will avoid the error with a higher error value. The advantage of this is that it changes the model instead of the data. However it can be ineffective on severely imbalanced datasets and does not guarantee convergence to optimal model.

Since in our project the number of observations is less, up sampling methods are preferred over down sampling in order to avoid the loss of information. SMOTE is one of the up sampling techniques which is followed here to avoid over fitting which occurs when exact replicas of minority instances are added to the main dataset. A subset of data is taken from the minority class as an example and then new synthetic similar instances are created. These synthetic instances are then added to the original dataset. The new dataset is used as a sample to train the classification models.





Before Upsampling, Count of Label 1 :

After Upsampling, Count of Label 1 :

Before Upsampling, Count of Label 2 :

After Upsampling, Count of Label 2 :

## 4.9 BINNING :

Binning or grouping data (sometimes called quantization) is an important tool in preparing numerical data for machine learning, and is useful in scenarios like these: ... To mitigate the bias in the model, you might transform the data to a uniform distribution, using the quantiles (or equal-height) method.

Binning is the process of converting the numerical variable to categorical variable.We have created binning categorical variable which gives the generalized information about the variable for example:

If the salary of three persons are 10000,40,000,100,000 so we can categorize them as

If salary is less than 25,000 then “Low”

If salary is <= 25,000 and >= 50,000 then “Medium”

If salary is <=200,000 and >50,000 then “High”.

Administration,

```
def adms(a):

    if a <= 2.3:

        return 0

    else :

        return 1
```

Created new Administrator using above function.

Adminstration_duration,

```
def adms_dur(a):
    if a <= 7:
        return 0
    if a > 7 and a < 93:
        return 1
    else :
        return 2
```

Created new Administrator_Duration using above function.

Information,

```
def infr(a):
    if a <= 0:
        return 0
    else :
        return 1
```

Created new Information using above function.

Inforamtion_duration,

```
def infr_dur(a):
    if a <= 0:
        return 0
    else :
        return 1
```

Created new Information_Duration  using above function.

Product_related

```
def prdt(a):
    if a <= 7:
        return 0
    if a > 7 and a < 38:
        return 1
    else :
        return 2
```

Created new Product_related  using above function.

Product_related_duration

```
def prdt_dur(a):
    if a <= 184:
        return 0
    if a > 184 and a <= 598:
        return 1
    if a > 598   and a <= 1464:
        return 2
    else :
        return 3
```

Created new Product_related_duration  using above function.

ExitRates,

```
def extr(a):
    if a <= 0.025:
        return 1
    else :
        return 0
```

Created new ExitRates  using above function.

BounceRates,

```
def bnce(a):
    if a <= 0.016:
        return 1
    else:
        return 0
```

Created new BounceRates using above function.

After converting all the numerical variables to the categorical variable.

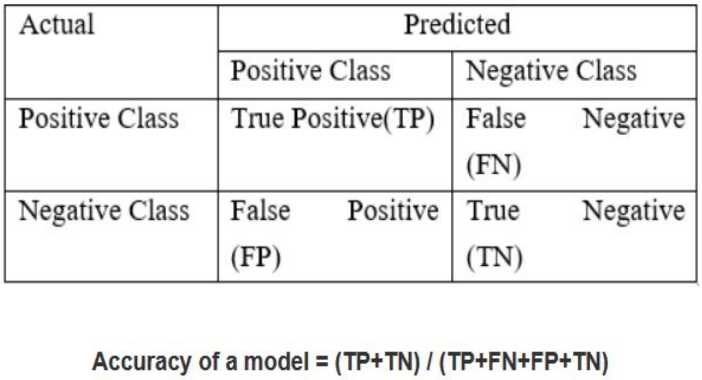
Proceeding to build the model.

# 5. BUILDING MODELS

## 5.1 BASE MODEL.

We have to take a different approach here from a normal machine learning flow because of the nature of our data. The conventional model evaluation methods do not accurately measure model performance when faced with imbalanced datasets. Standard classifier algorithms like Decision Tree and Logistic Regression have a bias towards classes which have number of instances. They tend to only predict the majority class data. The features of the minority class are treated as noise and are often ignored. Thus, there is a high probability of misclassification of the minority class as compared to the majority class. Evaluation of a classification algorithm performance is measured by the Confusion Matrix which contains information about the actual and the predicted class.

- Logistic Regression as the Base model
- Flow chart explains the underlying steps in model creations



| Actual | Predicted | |
|---|---|---|
| | Positive Class | Negative Class |
| Positive Class | True Positive(TP) | False Negative (FN) |
| Negative Class | False Positive (FP) | True Negative (TN) |

Accuracy of a model = (TP+TN) / (TP+FN+FP+TN)

**Fig 5.1:- Confusion Matrix**

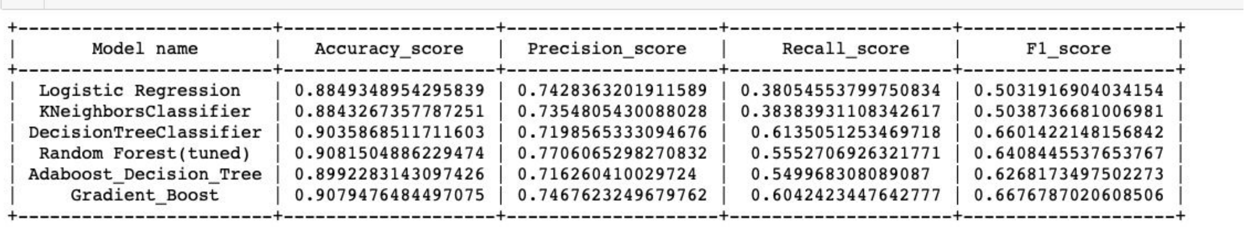
However, while working in an imbalanced domain accuracy is not an appropriate measure to evaluate model performance**.** For e.g.: A classifier which achieves an accuracy of 98 % with an event rate of 2 % is not accurate, if it classifies all instances as the majority class. And eliminates the 2 % minority class observations as noise. To fully evaluate the effectiveness of our model, we must examine **precision** and **recall** as well. Unfortunately, precision and recall are often in tension. That is, improving precision typically reduces recall and vice versa.

**Precision:** What proportion of positive identifications was actually correct?

$$\text{Precision} = \frac{TP}{TP + FP}$$

**Recall** : What proportion of actual positives was identified correctly?

$$\text{Recall} = \frac{TP}{TP + FN}$$



```
+------------------------+--------------------+--------------------+----------------------+--------------------+
|       Model name       |   Accuracy_score   |  Precision_score   |     Recall_score     |      F1_score      |
+------------------------+--------------------+--------------------+----------------------+--------------------+
|  Logistic Regression   | 0.8849348954295839 | 0.7428363201911589 | 0.38054553799750834  | 0.5031916904034154 |
|  KNeighborsClassifier  | 0.8843267357787251 | 0.7354805430088028 | 0.38383931108342617  | 0.5038736681006981 |
| DecisionTreeClassifier | 0.9035868511711603 | 0.7198565333094676 |  0.6135051253469718  | 0.6601422148156842 |
|  Random Forest(tuned)  | 0.9081504886229474 | 0.7706065298270832 |  0.5552706926321771  | 0.6408445537653767 |
| Adaboost_Decision_Tree | 0.8992283143097426 | 0.716260410029724  |  0.549968308089087   | 0.6268173497502273 |
|     Gradient_Boost     | 0.9079476484497075 | 0.7467623249679762 |  0.6042423447642777  | 0.6676787020608506 |
+------------------------+--------------------+--------------------+----------------------+--------------------+
```

Before Feature extraction model has very drastic scores.

Accuracy wise they are good but recall score is too low. Which means underfit models.

To Deal with the above problems we followed differents steps in training algorithm like...

SMOTE UPSAMPLING(60 % FALSE - 40 %  TRUE),

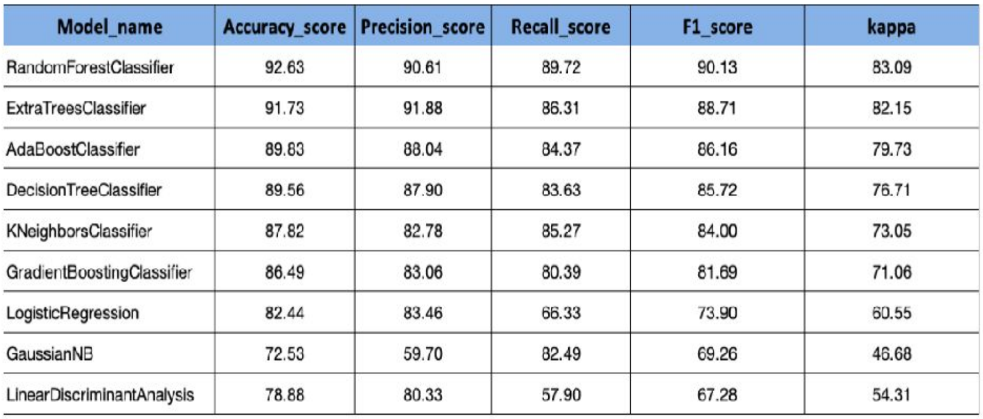
BINNING NUMERICAL TO CATEGORICAL VARIABLE.

ENSEMBLE (BOOSTING AND BAGGING TECHNIQUES)

HYPER PARAMETER TUNNING.

After Performing all the above mentioned steps we got the final results as below

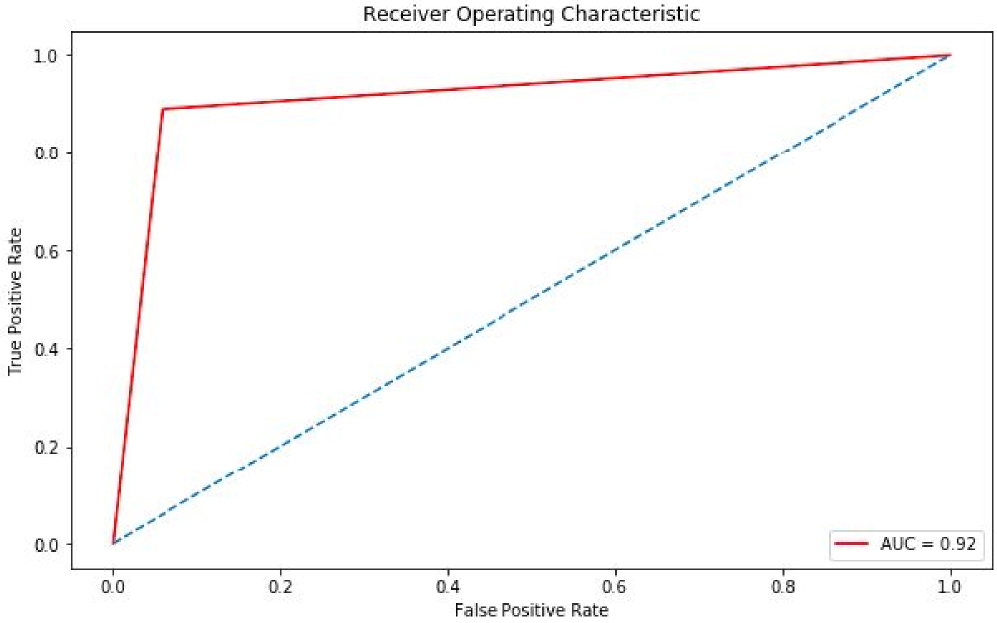
### 5.2 MODEL COMPARISONS:



| Model_name | Accuracy_score | Precision_score | Recall_score | F1_score | kappa |
|---|---|---|---|---|---|
| RandomForestClassifier | 92.63 | 90.61 | 89.72 | 90.13 | 83.09 |
| ExtraTreesClassifier | 91.73 | 91.88 | 86.31 | 88.71 | 82.15 |
| AdaBoostClassifier | 89.83 | 88.04 | 84.37 | 86.16 | 79.73 |
| DecisionTreeClassifier | 89.56 | 87.90 | 83.63 | 85.72 | 76.71 |
| KNeighborsClassifier | 87.82 | 82.78 | 85.27 | 84.00 | 73.05 |
| GradientBoostingClassifier | 86.49 | 83.06 | 80.39 | 81.69 | 71.06 |
| LogisticRegression | 82.44 | 83.46 | 66.33 | 73.90 | 60.55 |
| GaussianNB | 72.53 | 59.70 | 82.49 | 69.26 | 46.68 |
| LinearDiscriminantAnalysis | 78.88 | 80.33 | 57.90 | 67.28 | 54.31 |

F1_score for Random Forest Classifier is 90.13 which is the best of all the algorithms.

And the Receiver Operating Characteristic(ROC) curve got Area Under Curve(AUC) as 92%



## SUGGESTIONS & CONCLUSION

- Personalised experience to be provided to the New Users along with discounts & Cashbacks for first time users to garner interest among New Users.

- Partner with websites where traffic type is high such as customized credit cards with reward points, gift vouchers etc so that customer gives Repeat Business. Implement similar strategy with other websites where traffic type can be Increased.

- Change the look & feel of website with Minimum Navigations and more relevant content to users which creates interest to the buyer so that exit rates and bounce rates can be minimized.

- Plan for Special Day Events on weekends as conversion rates for Special day events are relatively less on weekdays compared to weekends.

## BIBLIOGRAPHY

**1. Web Mining, http://www.cs.ualberta.ca/~tszhu/webmining.htm**

**2. Nicholas Kashmerick, Learning To remove Internet Advertisement, 3rd Int. Conf. of Autonomous Agent, 1999**

**3. Sakar, C.O., Polat, S.O., Katircioglu, M. et al. Neural Comput &amp; Applic (2018). [Web Link]**

**4. B.D. Davision. Recognizing Nepotistic links on the Web. Proceeding of AAAI 2000.**

**5. S. Paek and J. R. Smith, Detecting Image Purpose in World-Wide Web Documents, SPIE/IS&amp;T Photonics West, Document Recognition, January, 1998.**

**6. Guohua Hu, Qingshan Zhao " Study to Eliminating Noisy Information in Web Pages**

**7. Kushmerick, 1999] Nicholas Kushmerick. Learning to remove Internet advertisements. Agnets-1999, 1999..**

**8. Kao et al., 2002] Hung-Yu Kao, Ming-Syan Chen Shian-Hua Lin, and Jan-Ming Ho, Entropy-Based Link Analysis for Mining Web Informative Structures. CIKM-2002, 2002.**

**9. H. Y. Kao, J. M. Ho, and M. S. Chen, Wisdom Web intrapage informative structure mining based on document object model in IEEE Trans KDD, 2005.**

**10. YI L. et LIU B. (2003), "Web Page Cleaning for Web Mining through Feature Weighting", in Proceedings of Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03).**

11. S. Debnath, P. Mitra, and C. L. Giles. Automatic extraction of informative blocks from webpages. In ACM Symposium on Applied Computing, pages 1722–1726, 2005.

12. L. Yi, B. Liu, and X. Li. Eliminating noisy information in web pages for data mining. In Proceedings of the International ACM Conference on Knowledge Discovery and Data Mining, pages 296– 305, 2003.

13. Diao, Y., Lu, H., Chen, S., and Tian, Z., TowardLearningBased Web Query Processing, In Proceedings of International Conference on Very Large Databases, 2000, pp. 317-328.

14. Kaasinen, E., Aaltonen, M., Kolari, J., Melakoski, S., and Laakko, T., Two Approaches to Bringing Internet Services to WAP Devices, In Proceedings of 9th International World-Wide Web Conference, 2000, pp. 231-246.

15. Wong, W. and Fu, A. W., Finding Structure and Characteristics of Web Documents for Classification, In ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge

Discovery (DMKD), Dallas, TX., USA, 2000.

16. Gupta, S., Kaiser, G., Neistadt, D. and Grimm, P., DOMbased Content Extraction of HTML Documents, In the proceedings of the Twelfth World Wide Web conference(WWW 2003), Budapest, Hungary, May 2003.

17. Cai Deng, Yu Shipeng, and Wen Jirong, et al. VIPS: a Vision Based Page Segmentation Algorithm[R] , Microsoft Technical Report: (MSR-TR-2003-79) ,2003.

18. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel,

M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. The

Journal of Machine Learning Research 12 (2011), 2825–2830.

19. Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification

Techniques. Informatica 31 (2007). Pp. 249 – 268. Retrieved from IJS website:

http://wen.ijs.si/ojs- 2.4.3/index.php/informatica/article/download/148/140.

20. T. Hastie, R. Tibshirani, J. H. Friedman (2001) — The elements of statistical learning,‖

Data mining, inference, and prediction, 2001, New York: Springer Verlag.

21. Chris Drummond and Robert C Holte. C4. 5, class imbalance, and cost sensitivity: why

under-sampling beats over-sampling. In Workshop on learning from imbalanced datasets II,

volume 11. Citeseer, 2003.

22. Kaizhu Huang, Haiqin Yang, Irwin King, and Michael R. Lyu. Imbalanced learning with

a biased minimax probability machine. Systems, Man, and Cybernetics, Part B: Cybernetics,

IEEE Transactions on, 36(4):913–923, 2006.

23. C.V. Krishnaveni and T. Sobha Rani. On the classification of imbalanced datasets.

IJCST, 2:145–148, 2011.

24. John Shawe-Taylor and Grigoris Karakoulas. Optimizing classifiers for imbalanced

training sets. Advances in neural information processing systems, 11:253, 1999.

25. Taghi M Khoshgoftaar, Jason Van Hulse, and Amri Napolitano Comparing boosting and

bagging techniques with noisy and imbalanced data. Systems, Man and Cybernetics, Part A:

Systems and Humans, 41(3):552–568, 2011.

26. Tom M Mitchell. Machine Learning. McGraw-Hill, 1997.

27. Yoshua Bengio and Yves Grandvalet. No unbiased estimator of the variance of k-fold cross-validation. The Journal of Machine Learning Research, 5:1089–1105, 2004.

28. Michael Buckland and Fredric Gey. The relationship between recall and precision. Journal of the American society for information science, 45(1):12, 1994.

A. I. Marques, V. Garcia, and J. S. Sanchez. On the suitability of resampling techniques for the class imbalance problem in credit scoring. Journal of the Operational Research Society, 64(7):1060–1070, 2012. accessed 01-02-2016