Classification of Online Shoppers Purchasing Intention

Sahil Mathur   
Faculty of Engineering and Computing *Coventry University*Coventry, UK  
[mathurs2@uni.coventry.ac.uk](mailto:mathurs2@uni.coventry.ac.uk)

Sukhmani Saini  
Faculty of Engineering and Computing *Coventry University*Coventry, UK  
[sainis9@uni.coventry.ac.uk](mailto:sainis9@uni.coventry.ac.uk)

**Abstract— The primary objective of this paper is to analyse the intention/behaviour of online shoppers using machine learning techniques on the given dataset. Utilising Classification methods including K-nearest Neighbour, Naïve Bayes, Decision Trees and Random Forest , a comparison is drawn from the results and a conclusion is formulated on the most suitable method. Python programming language was used to implement techniques and algorithms utilizing machine learning repositories, prediction results and algorithm performance measures were obtained, and visualized for comparison and discussion.**

Keywords— machine learning; Classification; python; categorical data; decision tree; k-nearest neighbour; random forest, k- nearest.

Key: ML = Machine Learning, KNN = k-Nearest Neighbour, Random Forest = RF, PLT = Matplotlib

# Introduction

With the boom in internet access across the world, the normal retail shopping has been rapidly shifting to E-commerce or online shopping and as a result, the shopping dynamics have been continuously changing and has already become a major part of the retail market. A typical customer prefers shopping online in their own comfort as compared to travelling to a physical shop and buying the products. Resulting in Businesses to start implementing various online marketing techniques to lure customers to their shopping portal. Customers who visit these web portals might not make any purchases at all. This could be for a variety of reasons, like high product pricing or window shopping. There have been multiple studies (Santini 2018) that imply that retention strategies such as an appropriate recommendation system play a critical role in converting sales. For example, if the ML solution predicts a strong customer purchase intention, the recommend system may suggest a higher quality or more expensive product because it can be inferred that the user is ready to consider a better or more expensive product if their intent to buy a particular item is very strong. If the solution forecasts a lower intention to buy, the recommendation system may suggest discounted products or products with special deals. Cambridge Analytica used similar strategies, although on a larger scale, to influence voter decisions in American elections (The Guardian, 2018). This demonstrates the strength of a proper machine learning technique if applied effectively could vary a majority population’s intent.

# Objective

This project intends to make advantage of information that customers may leave behind in the form of browsing history data or user information when they browse an ecommerce platform. Utilizing clickstream and activity data information, the study seeks to forecast online buyers' purchasing intentions using this information. The research intends to build a machine learning model based on this data to forecast client purchasing intentions. The project's goal is to create a Machine Learning model that can anticipate customer purchase intent as accurate as possible.

# Dataset

The dataset used in the project was obtained from UCI machine learning. The dataset has been provided by Authors Sakar and Y. Kastro (Sakar et al., 2018).

The data has 12330 instances, each with 18 characteristics and no missing data. Each instance represents a single person's visit to the site. When a repeat consumer visits the site several times, only the initial visit from the time frame examined is taken into account, with subsequent visits being excluded. The first 17 attributes are visitor-related features that include 10 numerical values and 7 categorical elements. The last 18th attribute that is ‘Revenue’ reflects the category if the visit resulted in revenue or not, which means that if the visitor made a purchase, it is labelled as True or False.

The categorical and numerical features used in the prediction model are shown in Tables 1 and 2, respectively.

Table 1. Categorical Features

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Description** | **Number of values** |
| OperatingSystems | Visitor’s Operating System | 8 |
| Browser | Visitor’s Browser type | 19 |
| Region | Geographical Region of the visitor | 9 |
| TrafficType | Source of redirect by visitor | 20 |
| VisitorType | Type of Visitor (“New”, “Returning”, “Other”) | 3 |
| Weekend | Indicating if the day of visit is a weekend or not by True/False | 2 |
| Month | Month of visit | 10 |
| Revenue | Boolean to indicate if revenue was generated | 2 |

Table 2. Numerical Values

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Description** | **Number of values** |
| Administrative | Number of pages visited by the visitor | 12330 |
| Administrative\_Duration | Total amount of time (in seconds) spent by the visitor | 12330 |
| Informational | Number of pages visited by the visitor about Web site, communication and address information of the shopping site | 12330 |
| Informational\_Duration | Total amount of time (in seconds) spent by the visitor on informational pages | 12330 |
| ProductRelated | Number of pages visited by visitor about product related pages | 12330 |
| ProductRelated\_Duration | Total amount of time (in seconds) spent by the visitor on product related pages | 12330 |
| BounceRates | Average bounce rate value of the pages visited by the visitor | 12330 |
| ExitRates | Average exit rate value of the pages visited by the visitor | 12330 |
| PageValues | Average page value of the pages visited by the visitor | 12330 |
| SpecialDay | - | 12330 |

# Data Cleaning

The dataset is available in a good condition with no null values and some minor issues which we decided not to handle beforehand.

# Exploratory Data Analysis (EDA)

In this section we have thoroughly explored and analyzed the data to summarize the main characteristics. We have divided this section into three parts: 1. Univariate Analysis – analyses of every column in the dataset, we managed to extract some useful information about each column such as the skewness, data distribution etc., 2. Bi-variate analysis – In this section we have compared each variable or column to the target variable (‘Revenue’) and managed to extract some uselful information such as the distribution, outliers etc., 3. Multi-variate analysis – in this section we have tried to encompass the simultaneous observation and analysis of more than one outcome variable. The dataset has both numerical and categorical variables. The "BounceRates," "ExitRates," and "PageValues" features, among others, depict the metrics for every session. In this sectionOur data analysis yielded the following major findings:

The sample is imbalanced, with only 15% of visits resulting in a purchase. There is huge amount of outliers in the data.

PageValues may be the most critical feature in predicting purchase conversion.

Data Imbalance –

Table

Description automatically generatedThis dataset is skewed, with only 15% of visits resulting in a purchase. While 15% is not a severely imbalanced dataset, we can still investigate strategies or algorithms which are more effective at dealing with it.

PageValue Importance –

PageValues is described as the average page value of the user's pages visited. Values are typically allocated to essential pages in an e-commerce setting, such as checkout pages or pages following the checkouts.

As illustrated in Fig 2, a PageValues greater than 5 enhances the likelihood of sales conversion. As a result, the PageValues feature gives an excellent indication of whether or not the visitor will buy something.

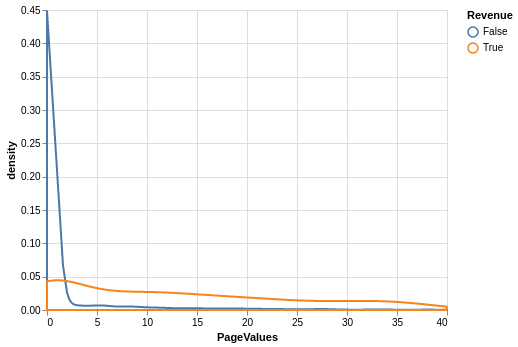


Figure 2. PageValues Density Plot

A picture containing chart

Description automatically generated

Figure 3. Heatmap of the dataset

From Fig 3, We can depict that Administrative and Administrative\_Duration are correlated. ExitRates, Information , ProductRelated and BounceRates show similar properties. Page Value appears to have a stronger relationship with Revenue.

Right-Skewness of the data –

Finally, we discovered that most of numerical features have right-skewed ends. This is typical in e-commerce environments, where some individuals have a disproportionately high utilisation rate. We can test whether removing outliers or using feature transformations like Box-Cox can improve model performance.

Chart, histogram

Description automatically generated

Figure 4. Right Skewness of the data

Below table represents some important aspects of the columns with respect to the target column ‘Revenue’.

Table

Description automatically generated

# STATISTICAL TESTS

In this section we have performed various statistical tests to analyse if the variable is significant or not. This section is divided into two sections :

1. Categorical column vs target column

For comparing categorical columns with the target column we are performing chi-squared test. A chi-squared test is a hypothesis tests which is used to compare the observed results with expected result, the main purpose of this test is to determine if a difference between observed data and expected data is due to chance or if it is due to a relationship between the variables that we are studying. Basically, we will check if our target column ‘Revenue’ is getting influenced by any categorical column or not. Our null hypothesis (Ho) is that the proportion of revenue across the category is same and our alternative hypothesis (Ha) is proportion of revenue at least in two categories is different. We reject the null hypothesis if p-value < 0.05. Here are the observations:

Table

Description automatically generated

From the above table we can conclude that only one column i.e., Region is not significant, therefore, we can say that the proportion of revenue across the categories are not the same.

1. Numerical column vs Target column

For comparing numerical columns with target columns we will be performing Shapiro-wilk test. It is a statistical test that is used to check if the continuous variables/columns follows the normal distribution or not. Here, our null hypothesis (Ho) is that data is normally distributed and alternative hypothesis (Ha) is that the data is not normally distributed. We reject the null hypothesis if p-value < 0.05. Here are the observations:

Table

Description automatically generated

We can conclude that our data is not normally distributed.

Before moving ahead and transforming our data we are also going to check the outliers. Outliers are the data points that significantly differs from other observations, in other word the data that lies outside the other values in the set. Here are the percentage of outliers available in our dataset:

Table

Description automatically generated

Chart, bar chart

Description automatically generated

As per our above observation we can see that most pf the outliers are in the column ‘ExitRates’ and ‘BounceRates’, therefore, we are going to handle it manually.

We are going to transform our data using a method called Box-Cox. The Box-Cox method helps to address the non-normally distributes data by transforming to normalize the data.

Distribution before transformation:

Chart, waterfall chart

Description automatically generated

Distribution after transformation:

Chart, box and whisker chart

Description automatically generated

As we can conclude, there are significant changes in the distribution of the data.

# DATA PRE-PROCESSING

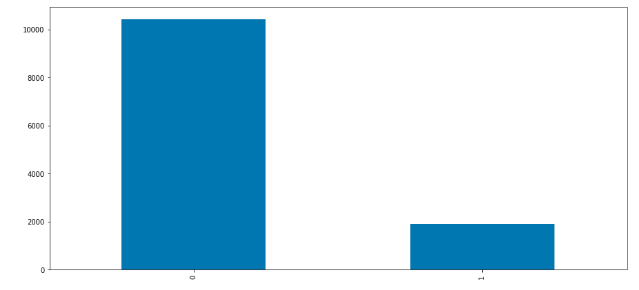
Data pre-processing is manipulation of the data before feeding into any model to enhance the performance. This includes cleaning and transforming the data. As per our previous observations there are no null values in our data, therefore, we will go ahead and transform the necessary column. In our dataset we have 10 numerical columns and 8 categorical. We will convert our categorical columns into numerical because machine learning models requires all input variables to be numeric. We are going to use LabelEncoder to convert it, LabelEncoder replaces the categorical value with a numeric value between 0 and number of classes-1.

# FEATURE ENGINEERING

Feature engineering also known as feature extraction is the process of transforming the raw data into features that better represents the underlying problem to the predictive models. Firstly, we are going to scale our raw data. Scaling is an important method that is mainly performed in order to standardize the functionality of the input dataset, usually it fit within a specific scale like 0-100 or 0-1. We are going to apply StandardScalar on our data because our dataset differ greatly between the ranges. It removes the mean and scales the data to the unit variance. The basic idea behind it is that the variables that are measured at different scales do not contribute equally to the fit of the model and the learning function of the model and could end up creating bias, thus, it’s necessary to scale it before integrating it into the machine learning model. Given the distribution of the data, each value in the dataset will have the mean value subtracted and then divided by the standard deviation of the entire dataset.

# Data modelling

We have divided our scaled data into train and test. Now we are going to feed it into our machine learning models but before that there’s an important point of observation. Our target variable ‘Revenue’ is highly imbalance as per the figure below.



Therefore, we are going to use a technique called SMOTE (Synthetic Minority Oversampling Technique) to solve this problem. SMOTE is basically an improved alternative for oversampling that performs data augmentation by creating synthetic data points based on the original data points. In this section we are going to feed our scaled data, firstly without using SMOTE and then after using SMOTE and we will compare the accuracy.

1. Without SMOTE

## Logistic Regression

Logistic Regression models the probabilities for classification problems with two possible outcomes. Basically, the logistic regression model uses the logistic function to squeeze the output of a linear equation between 0 and 1.

Below are the outcomes of logistic regression.



Table

Description automatically generated

We have got a train score of 86% and a test score of 86% as well but the F1-score is only 35% and Kappa score is 30% which is not very good.

*Confusion matrix:*

Chart, treemap chart

Description automatically generated

*ROC Curve:*

*Chart, line chart

Description automatically generated* *Chart, line chart

Description automatically generated* Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated Chart, line chart

Description automatically generated

The ROC curve above indicates the moderate performance of our algorithm.

## Decision Tree

Decision Trees are a non-parametric supervised learning method used for classification and regression. This creates a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

Below are the outcomes of decision tree.

Text

Description automatically generated

Table

Description automatically generated

*Confusion Matrix:*

*Chart, treemap chart

Description automatically generated*

*ROC Curve:*

Chart, line chart

Description automatically generated

## Random Forest

Random forest is a classification algorithm consisting on various decision trees. It is great with high dimensional data and faster to train than decision tree since it only works on subsets of data.

Below are the outcomes of Random Forest



Table

Description automatically generated

*Confusion Matrix:*

*Chart

Description automatically generated*

*ROC Curve:*

## Naïve Bayes

The Naïve Bayes classification algorithm is a probabilistic classifier that is based on the Bayes Theorem strong independence assumptions. It is a very fast and simple classification algorithms suitable for a very high-dimensional datasets.

Below are the outcomes of the algorithm:

Text

Description automatically generated

*Confusion Matrix:*

*Chart

Description automatically generated*

*ROC Curve:*

1. With SMOTE

We are going to apply the Synthetic Minority Oversampling Technique now.

1. *Logistic Regression*

Below are the results of logistic regression.

Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated

# RESULTS AND CONCLUSION

As per our above observation through accuracy metrics and ROC, we have managed to conclude that Random Forest is giving us the best result, it’s 93% area under the curve and F1-score as 0.90. Although, transformation improved certain accuracies for logistic regression and Naïve Bayes but Random Forest and Decision tree were almost the same.

We can suggest some business insights:

1. To maximise sales, we advise offering a discount three to four days before the special day.
2. To make some money, Jan. & April. need special attention. Although February contains special days, overall revenue is quite modest, thus it is important to take use of them.
3. Since there are users of non-mainstream browsers, advertising on those browsers will help to reach new consumers.
4. Region - Because we have a wide geographic reach, a large market will aid in creating income in low-producing areas.
5. We must create strategies to capitalise on the new tourists' spending power.
6. Due of the extremely low weekend revenue, special weekend offers must be promoted.
7. Since we currently have a high exit rate in the months of July, June, May, and August, lowering the bounce rate during those months could boost revenue.
8. Special planning must be created for additional channels in addition to the typical traffic generators.
9. Many pages have low page values, hence we propose improving the product search algorithm to raise page values.

References

The Guardian. (2018, May 7). Cambridge Analytica: How did it turn clicks into votes? the Guardian. <https://www.theguardian.com/news/2018/may/06/cambridge-analytica-how-turn-clicks-into-votes-christopher-wylie>

Santini, R. (2018). Recommender systems as “tastemakers”: Collaborative filtering as a market strategy for online cultural products. Observatorio (OBS\*). <https://doi.org/10.15847/obsobs1222018847>

Sakar, C. O., Polat, S. O., Katircioglu, M., & Kastro, Y. (2018). Real-time prediction of online shoppers’ purchasing intention using multilayer perceptron and LSTM recurrent neural networks. Neural Computing and Applications, 31(10), 6893–6908. https://doi.org/10.1007/s00521-018-3523-0

**IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.**