Student Name: Natalia de Oliveira Rodrigues & Student ID: 2023112

Abstract

This academic project aims to deliver a critical analysis of the knowledge produced, in the course: Higher Diploma in Science in Data Analytics for Business at CCT College.

CAPSTONE PROJECT

Strategic Thinking - Semester II

**CCT College Dublin**

**Assessment Cover Page**

|  |  |
| --- | --- |
| **Module Title:** | Strategic Thinking |
| **Assessment Title:** | Project Capstone Semester II - Presentation / Report |
| **Lecturer Name:** | James Garza |
| **Student Full Name:** | Natalia de Oliveira Rodrigues |
| **Student Number:** | 2023112 |
|  |  |
|  |  |
| **Assessment Due Date:** | 19th September 2023 |
| **Date of Submission:** | 15th November 2023 23:59 |

**Declaration**

By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

Contents

[Introduction 5](#_Toc149925762)

[Business Description 6](#_Toc149925763)

[Research Question 6](#_Toc149925764)

[General goal 6](#_Toc149925765)

[Success criteria/indicators 6](#_Toc149925766)

[Technologies used 7](#_Toc149925767)

[Models and machine learning algorithms 7](#_Toc149925768)

[Libraries 7](#_Toc149925769)

[Accomplishment 8](#_Toc149925770)

[Data 8](#_Toc149925771)

[Source 8](#_Toc149925772)

[Attributes 8](#_Toc149925773)

[Dimensions 8](#_Toc149925774)

[Descriptive Statistics and Data Visualization 9](#_Toc149925775)

[Heat Map 15](#_Toc149925776)

[Correlation 16](#_Toc149925777)

[ANOVA Test 17](#_Toc149925778)

[Chi Squared Test 18](#_Toc149925779)

[Flowchart of Data Preparation and Modelling 19](#_Toc149925780)

[Data Preparation and Preprocessing 20](#_Toc149925781)

[Normalizing the data 20](#_Toc149925782)

[Balancing the data 20](#_Toc149925783)

[Splitting the data 20](#_Toc149925784)

[Dimensionality Reduction 21](#_Toc149925785)

[Feature Engineering 22](#_Toc149925786)

[Models 22](#_Toc149925787)

[Challenges encountered 22](#_Toc149925788)

[Inclusion of strategies to overcome them 22](#_Toc149925789)

[Model Building and Evaluation 22](#_Toc149925790)

[Building Random Forest Model 24](#_Toc149925791)

[Results and analysis 25](#_Toc149925792)

[Conclusion 26](#_Toc149925793)

[References 27](#_Toc149925794)

[Appendix 28](#_Toc149925795)

[Appendix 1: Data Dictionary 28](#_Toc149925796)

[Appendix 2: Building a Decision Tree Model 29](#_Toc149925797)

[Appendix 3: Building SVM Model 32](#_Toc149925798)

[Appendix 4: CRISP-DM 33](#_Toc149925799)

# Introduction

E-commerce is the new route for buying and selling services using the internet. Nowadays, a large part of the population uses this method of purchase to find a product. For that reason, e-commerce and business are now connected since they have a website to offer their products, where you can also place an order.

On the other hand, digital purchasing has increased since the COVID-19 crisis, as people had to remain at home and had no chance to go anywhere for shopping. This alternative found a way to support more and more online sales, making the experience more convenient for users. As a result, the companies are improving their websites and investing money to determine whether the product on their websites is the best option for customers.

With this report, we are trying to show how to use machine learning to find out if people, after searching on the internet, can buy a product when they visit the website since they spend time on it.

# Business Description

We are trying to train a Machine Learning Model that can predict the purchasing intentions of a visitor to a particular store’s website. The data is derived from e-commerce website data. It updates in real time when a user moves from one page to another. This is important as it can have a huge impact on online shops’ profitability. This data can be utilized to prompt prospective customers to finish an online transaction in real-time and increase total purchase conversion rates.

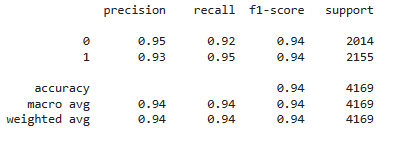
## Research Question

Can we predict if a user will make a purchase on an e-commerce website given their clickstream and session data?

## General goal

The main goal of this analysis is to predict if the user will end up generating revenue or not. Online stores can then use the findings to make sure they can continue to be profitable. In this paper, we try to resolve a classification problem.

## Success criteria/indicators

After applying 3 different Machine Learning Models. A Random Forest classifier Model appeared to be the model that would best address our classification issue. We have achieved an impressive 94% of accuracy, Precision and Recall through Random Forest Classification Model. The final results are demonstrated in Figure 1. 

*Figure 1. Classification Report - Accuracy, Precision and Recall Results*

# Technologies used

## Models and machine learning algorithms

Three supervised machine learning models Decision Tree Classifier, Random Forest Classifier, and Support Vector Machine (SVM) that are frequently employed for classification issues were used in this study.

## Libraries

To carry out various jobs and model algorithms, numerous libraries have been employed. They might consist of Pandas, Numpy, Seaborn, Matplotlib, Scipy, Statistics, SMOTE, NearMiss, StandardScaler, PCA, Metrics, Counter, dtreeviz, FeatureImportances etc.

# Accomplishment

## Data

The dataset “online\_shoppers\_intention” describes if a person is going to buy our products or not and gives us different attributes to analyze; it is composed of 12,330 rows and 18 features from which 14 are numerical and 4 categorical. We are going to analyze the months of frequent visits to the website, type of visitor, and exit days, among other variables that we can take a look at in the Data Dictionary (Appendix 1).

## Source

Our data was taken from Kaggle and it was taken from the following link: <https://www.kaggle.com/datasets/imakash3011/online-shoppers-purchasing-intention-dataset> (Kaggle, 2021)

## Attributes

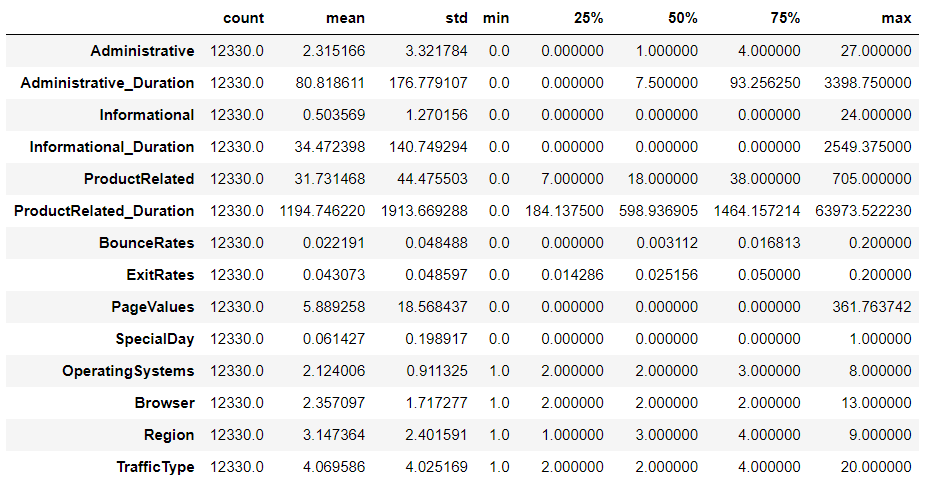
As said previously, we have 18 features. The feature “Revenue” is our dependent variable which means if the user completed the purchase. We are going to analyze 14 features as independent variables.

## Dimensions

The shape of our data is 12,330 rows and 18 columns as variables.

# Descriptive Statistics and Data Visualization

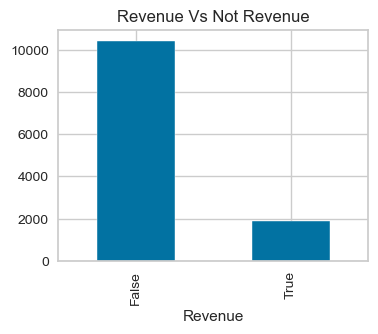
First, we took a look at our data and the statistics we have to start the analysis of our variables. We are going to see the statistics we found in the numerical variables of our dataset.



*Figure 2. Statistics of the numerical features in our dataset.*

We have a summary of the statistics of every variable showing the quartiles, mean, standard deviation, minimum and maximum values.

Now let’s analyze each group. We are going to take a look at our dependent variable “Revenue”.

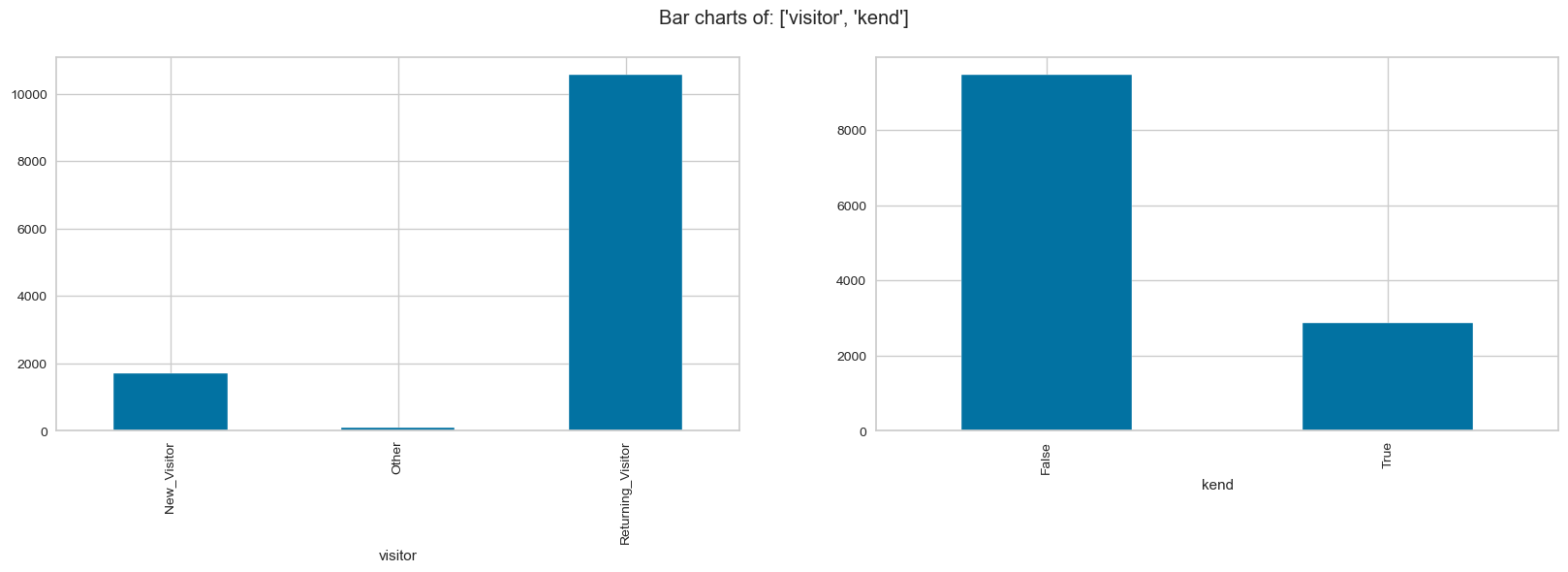
**

*Figure 3. Bar plot target variable “Revenue”*

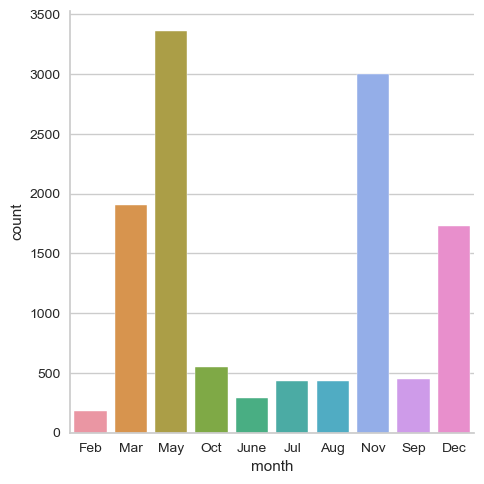
In *Figure* 3, there are around 10,000 users that didn’t complete their purchase than the ones that did who are sound 2,000 demonstrating that our data is unbalanced. As we have more False than True values, we are going to have a bias, to avoid that we need to balance it.

Now we are going to analyze the categorical variables.

*Figure 4. Bar plots “visitors” and “kend”*

**

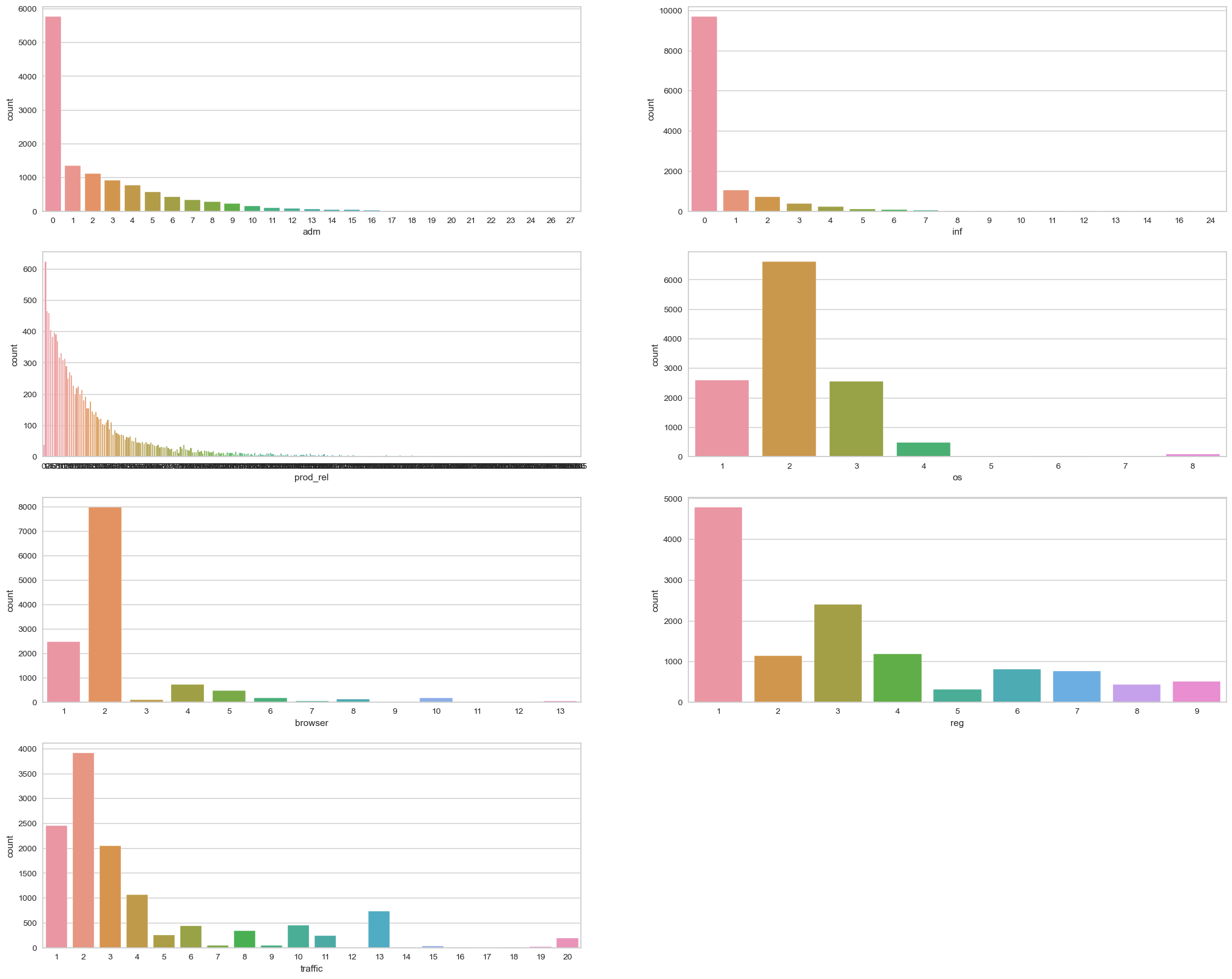
In *Figure 4*, we can see that most of the users are returning visitors; in the barplot of "kend" we can see that fewer people visit the website on weekends (represented by True) than the ones that visit between Monday and Friday (represented by False). These variables have an acceptable distribution to apply to our model.



*Figure 5. Bar plot “Month”*

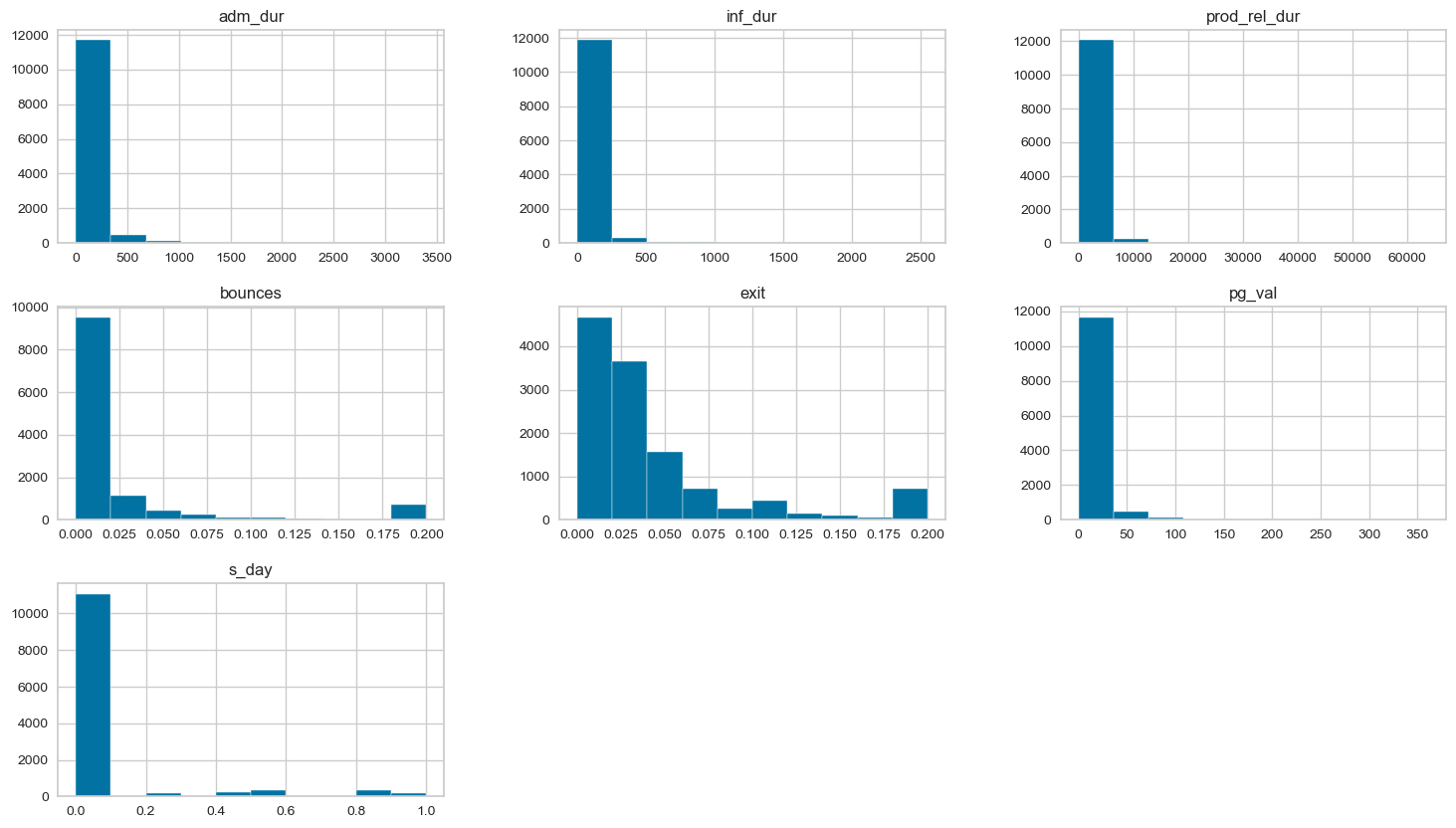
The bar plot of “Month” (*Figure 5*) shows that most of our visitors visit our webpage in March, May, November and December, we could suppose that in those months they celebrate special days and users tend to look at our products.

Let’s visualize the numerical discrete variables.

**

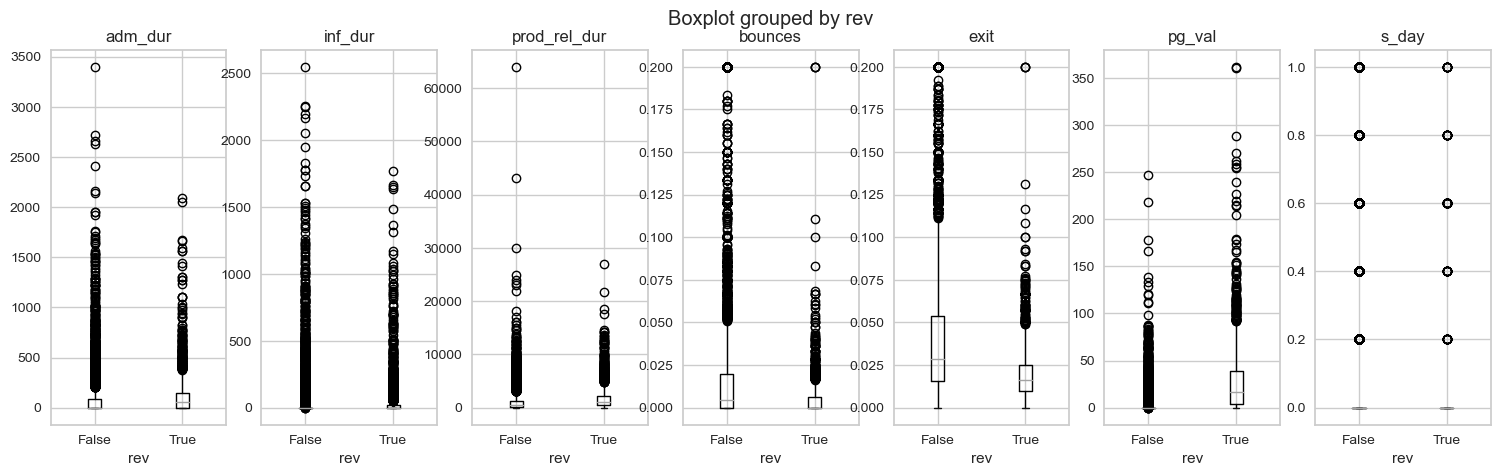
*Figure 6. Bar plots of Numerical Discrete Variables*

In *Figure 6*, we observe in the bar plots of the variables “adm”, “inf”, “prod\_rel”, “os”, “browser”, “reg”, “traffic” that all our variables are skewed to the right. Besides, we have comparable frequencies between them. Correlation analysis of each column must be done to determine if all of them are necessary for our analysis.

**

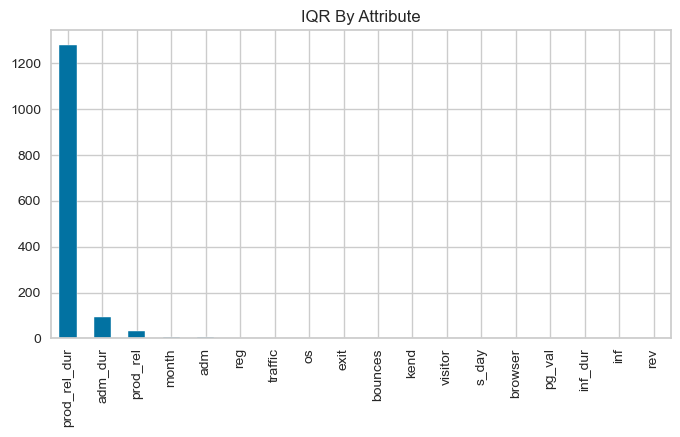
*Figure 7. Histograms of Numerical Continuous Variables.*

All our histograms are skewed to the right in *Figure 7*; also, there are some outliers and, in many features, most of the values are concentrated in a dominant column that is the first and is starting at 0.

**

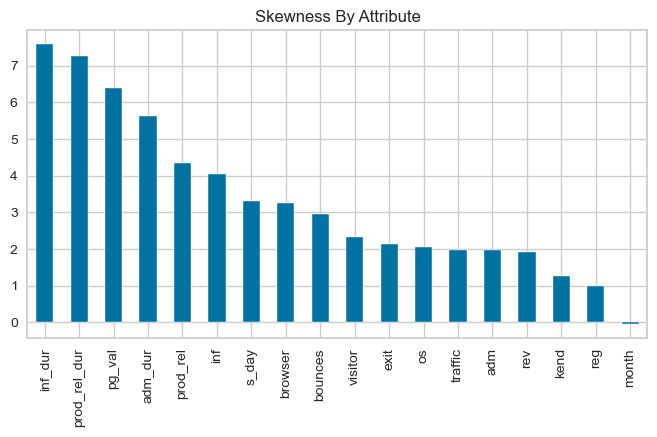
*Figure 8. Box plots Continuous Variables*

In the box plots in *Figure 8*, We can observe the presence of outliers and they represent the number of times that are important to include in our analysis to see if users are buying or not on the website. We decided to deal with these outliers using a Robust scaler since they are important to consider in the model for our analysis.



*Figure 9. IQR by attribute*

In *Figure 9,* we can compare the IQR between the attributes and the biggest ones are in “prod\_rel\_dur”, “adm\_dur”, and “prod\_rel”.

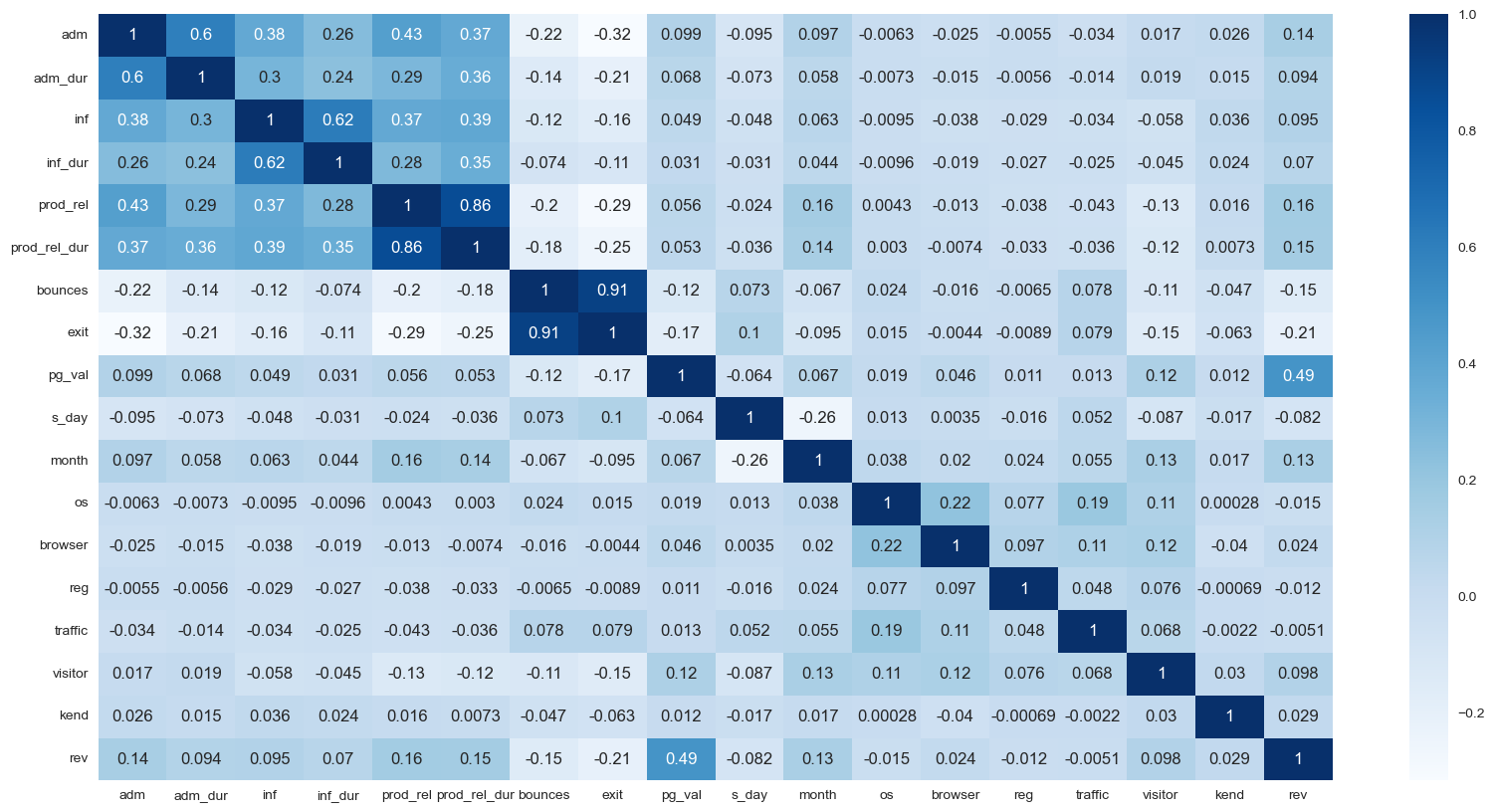
**

*Figure 10. Skewness by attribute*

Previously we saw in different bar plots and histograms that our data is mostly skewed to the right and in *Figure 10*, we can see what features are more skewed than others. This would be helpful if we had missing values.

## Heat Map

In this heat map we can confirm if the variables are correlated or not and the column of analysis is the target variable “rev”. However, this heatmap is mostly applicable in continuous variables and in this graphic, we see that the most correlated with 49% is “pg\_val”.

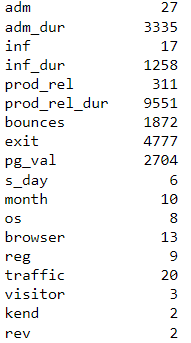
**

*Figure 11. Heat map of “online\_shoppers\_intention” data set*

## Correlation

Bruce, Gedeck. and Bruce. (2020, p.30) believe that exploratory data analysis in many modelling projects (whether in data science or in research) involves examining correlation among predictors and between predictors and a target variable. Variables X and y (each with measured data) are said to be positively correlated if high values of X go with high values of y, and low values of X go with low values of y. If high values of X go with low values of y, and vice versa, the variables are negatively correlated.

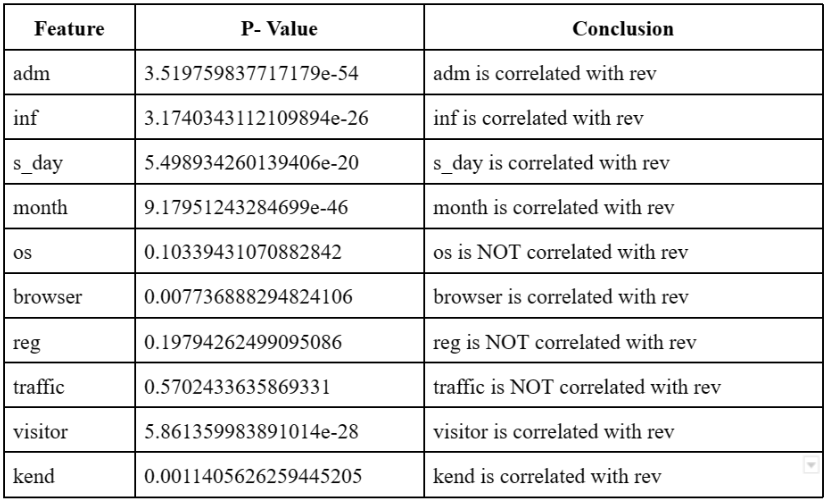
To deeply analyze the correlation between the data’s variables, we decided to use hypothesis test: ANOVA and Chi-squared. The aim is to know which variables are correlated with our target variable Revenue (“rev”). We are going to split our data that have less and more than 30 unique values. See Figure 12.



*Figure 12. The number of unique values per feature*

## ANOVA Test

The ANOVA test is applied to categorical features. As explained before, we are testing the features with less 30 unique values, and our hypothesis is to prove if they are correlated with our target variable or not.

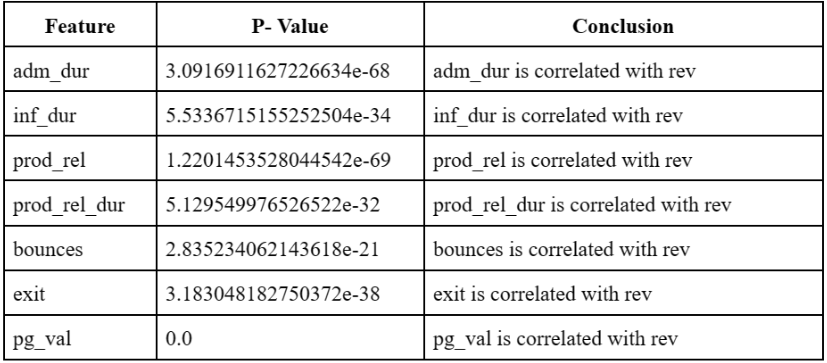
**

*Table 1. Results of ANOVA test*

In the results in Table 1 ANOVA compares the p-value of each column, and if it is less than 0.05, the variable is correlated. The columns “os”, “reg”, and “traffic” are not correlated with our target variable. So, we are not going to drop those columns.

## Chi Squared Test

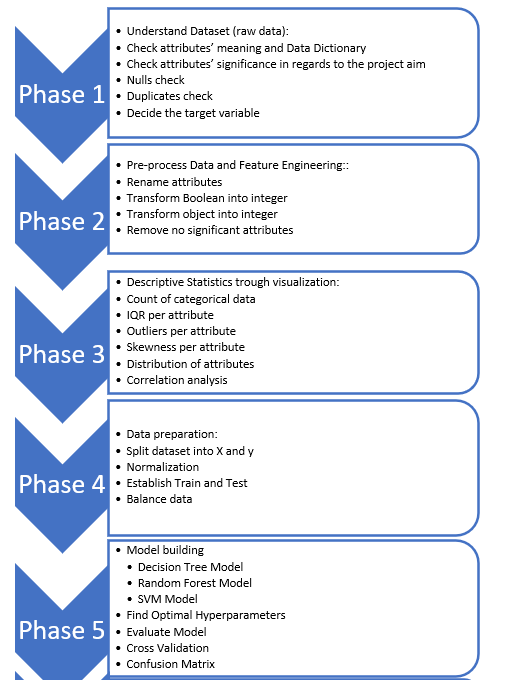
The Chi Squared Test method is applied to numerical variables. As explained before, the columns with more than 30 unique values. The hypothesis analyzes if they are correlated with our target variable Revenue (“rev”).

**

*Table 2. Results of Chi-squared test*

The results of the Chi Squared Test shows that all of them are correlated for our analysis and applying to the machine learning model. The metric is the same as ANOVA Test, if the p-value < 0.05, the variable is correlated.

# Flowchart of Data Preparation and Modelling



*Figure 13. Flowchart of Data Preparation and Modelling*

# Data Preparation and Preprocessing

During the analysis, we used different steps regarding data preparation, such as splitting the data, normalizing the data, and balancing the data.

## Normalizing the data

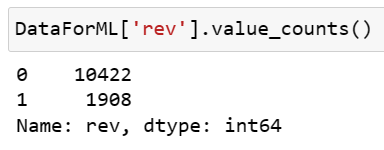
To work with this dataset, we used different methods such as Standarscaler, Scale, MinMaxScaler, and RobustScaler to see which one suits the data the best.

However, after applying them, we have chosen the RobustScaler normalization technique. It works better for that dataset due to the number of outliers. Additionally, the other techniques were discarded after we applied them because we observed how our accuracy had decreased significantly.

## Balancing the data

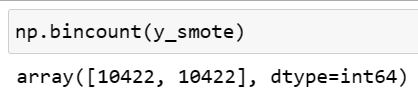
In this process, we attempt to use 2 different techniques to balance our data, NearMiss and SMOTE technique, and see which one would work better giving us the best result.

We proceed to use theSMOTE (Synthetic Minority Over-sampling Technique) to balance the class distribution of the target variable **”rev”** in the DataForML dataset. The method is used to oversample the minority class **(rev = 1)** by creating synthetic samples.



*Figure 14. Value count per class*

The output shows that both classes now have the same number of observations (10422 each). See Figure 15.



*Figure 15. Value count after the SMOTE technique*

## Splitting the data

As we are attempting to find the prediction from people who would generate revenue, we determined that the target variable is Revenue ("rev"). In the character matrix **"**X\_smote" and the target variable "y\_smote" are inputs produced by the SMOTE technique.

Furthermore, it is specified that the test size should be 20% of the total data size, and the random state is set to 38 for reproducibility. Test sizes of 10% and 30% were performed. However, the 20% test size delivered the best performance in our model.



*Figure 16. Train and Test Code*

## Dimensionality Reduction

Additionally, tests were performed, including the Principal Component Analysis (PCA). However, machine learning worked poorly since it reduced our accuracy.

## Feature Engineering

To get a better analysis we encode categorical values into numerical, representing each category with a number to make it simple to analyze. Duplicated values and missing data were not found in our data set. Three features were removed based on the results of ANOVA and Chi-squared test. After dropping the columns that are not correlated, the dataset was modified, and now it has 12,330 rows and 15 features.

## Models

We experimented using all the appropriate models for classification problems to find the best accuracy. Random Forest Classifier gave us the best performance scores.

# Challenges encountered

The challenge in this data was to define which scaler method to use, if we include outliers or not, define the correlation of each variable, balance our data, define hyperparameters to evaluate the model, and another challenge is that our data is just of 1 year.

# Inclusion of strategies to overcome them

We used ANOVA and Chi-squared tests to define if the features were correlated, and we removed three features from the data.

We also decided to include the outliers in our analysis since we have many of them that are part of our analysis, and most of them are spared because they represent the duration of time.

We also tried different scaling methods to see how the model performed. At the end we decided to use Robust Scaler to include the outliers as part of our model since they appear because they represent durations of time and are important for our analysis.

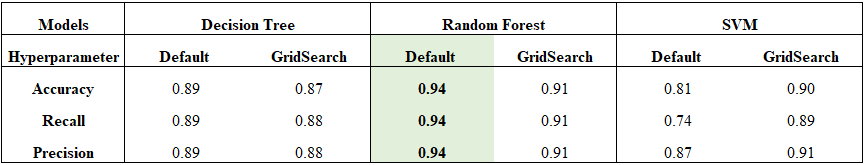
We also used SMOTE since our target variable was unbalanced, and instead of reducing observations that could drop important information, we decided to create synthetic data.

# Model Building and Evaluation

Three different machine learning models are applied: Decision Tree Classifier, a Random Forest Classifier, and SVM.

To start, the accuracy of each model was measured using the default hyperparameters. Next, using Grid Search, the best hyperparameters for each model were identified. To compare the accuracy improvement, each model was retrained using the ideal hyperparameters. Finding the ideal parameters using Grid Search may help improve the model's precision and performance.

The Random Forest Classifier Model was chosen due to its impressive accuracy, recall and precision results. See Table 3.



*Table 3.**Classification Report - Accuracy, Precision, and Recall Results*

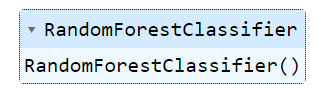
An overview of the Random Forest Classifier Model and the results are provided in this report, along with a thorough analysis of the evaluation's findings and their implications for the data aim.

An overview of the Decision Tree and SVM are provided in Appendix 1 and 2 respectively.

## Building Random Forest Model

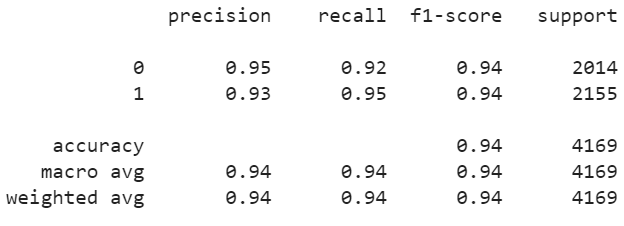
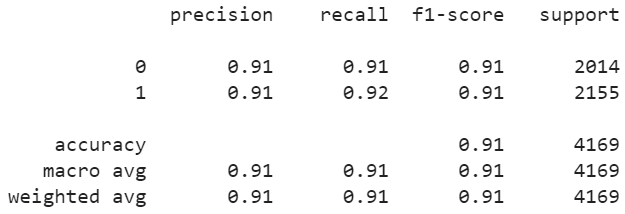
In this case, we are going to use the Random Forest Model, where we will import the Random Forest Classifier from the Sklearn library and then evaluate it using the default parameters, giving us as a result the following data *Figure 18.*

After evaluating the model, where Grid Search was used to find the optimal hyperparameter we found that the optimal hyperparameters were the ones that we have as default, providing us with a higher accuracy. *Figure 19*.



*Figure 17.**Random Forest Classifier Model*

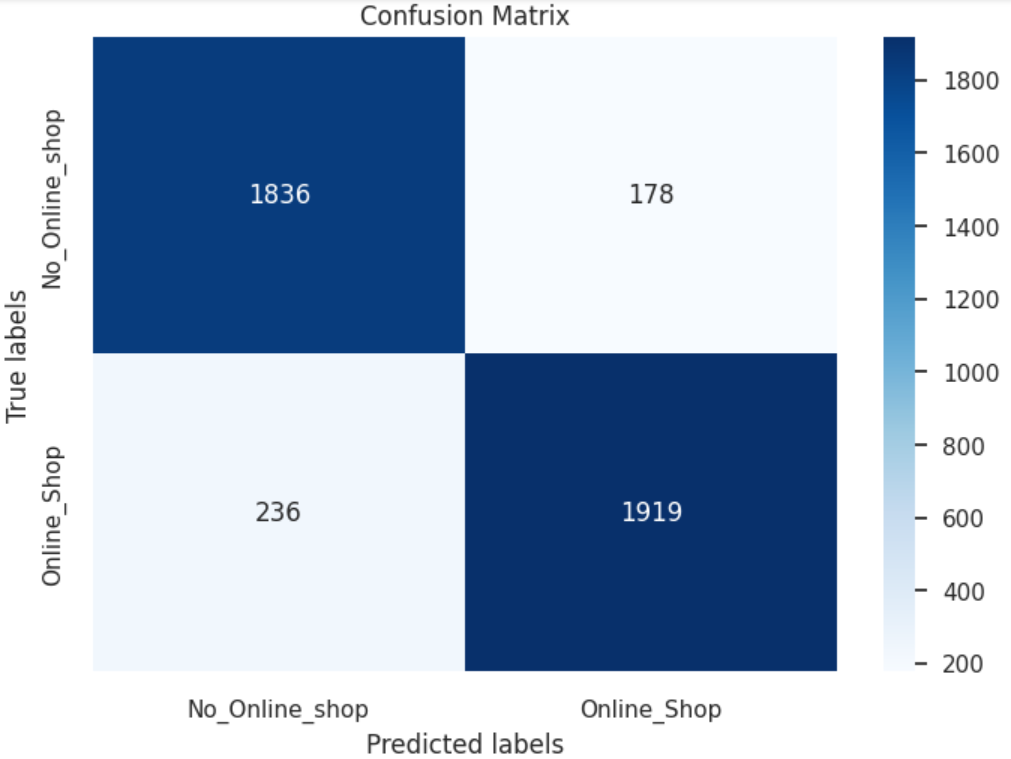
|  |  |
| --- | --- |
| **Apply Random Forest Classifier** | **Random Forest Classifier with Hyperparameters** |

|  |  |
| --- | --- |
| *Figure 18.**Classification Report - Accuracy, Precision, and Recall Results with the default hyperparameters* | *Figure 19.**Classification Report - Accuracy, Precision, and Recall Results with the hyperparameters the best values* |

Precision: Out of all the shoppers that the model predicted would shop, only (1 = 93%) did.

Recall: Out of all the shoppers that did buy, the model only predicted this outcome correctly for (1-95%) of those shoppers.

******

*Figure 20. Confusion Matrix*

*Random Forest Classifier Model*

The model predicted 1836 true negatives (TN) and 1919 true positives (TP), while misclassifying 178 instances as false negatives (FN) and 236 instances as false positives (FP). See Figure 20.

# Results and analysis

The Random Forest Classifier Model had the best performance when compared with the other two models in this analysis in terms of overall precision, recall, and f1-score metrics. It has a 94% overall accuracy, which indicates that the model correctly predicted 94% of cases. Additionally, both classes' (0 and 1)'s precision, recall, and f1-score values were excellent, with scores higher than 90% in every metric as in *Figure 18*.

# Conclusion

After the analysis, we get the following conclusions:

* The Machine Learning model, which presented the best performance, in predicting whether an online shopper is going to buy the product is Random Forest Classifier with an accuracy of 94% using default hyperparameters.
* Tests were performed, including the Principal Components Analysis (PCA). However, we decided to continue working without PCA since our model performed better.
* Regarding outliers, we conclude that they are essential since they represent the duration of time that users stay on the webpage. It was essential to predict if a user would make a purchase on an e-commerce website given their clickstream and session data.
* To balance the dataset, the SMOTE technique presented the best performance.
* The CRISP-DM methodology has been utilized since the beginning of this project. And all the steps necessary to ensure that our model is well explained and documented in the report.
* In conclusion, as a team, we feel that we have delivered an impressive and successful project since the Research Question for this project was answered.
* We conclude that, effectively, we can predict if a user will make a purchase on an e-commerce website given their clickstream and session data.

# References

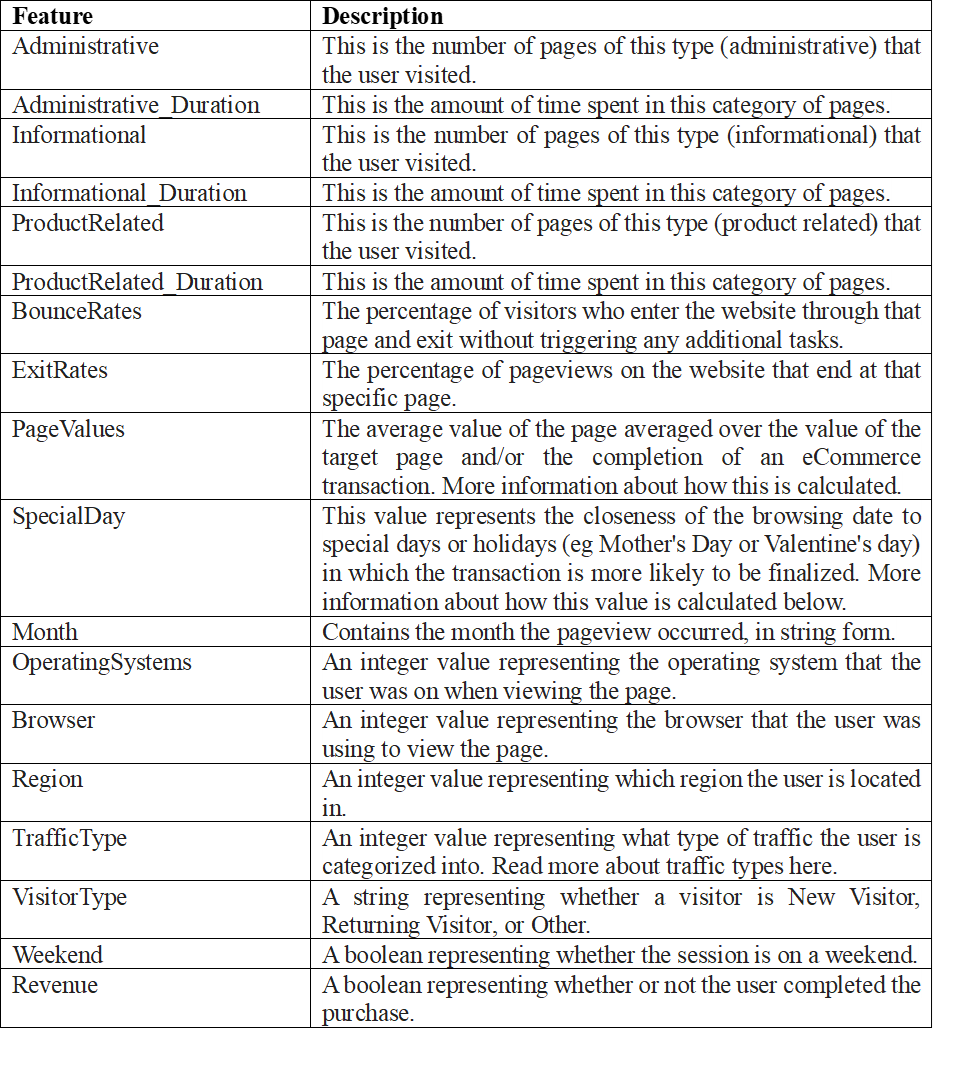
* Bruce, P.C., Bruce, A. and Gedeck, P. (2020). *Practical statistics for data scientists: 50+ essential concepts using R and Python*. Sebastopol, Ca: O’reilly Media, Inc.
* Kaggle (2021). *Online Shoppers Purchasing Intention Dataset*. [online] www.kaggle.com. Available at: https://www.kaggle.com/datasets/imakash3011/online-shoppers-purchasing-intention-dataset [Accessed 28 Apr. 2023].
* Mckinney, W. (2018). *Python for data analysis: data wrangling with pandas, NumPy, and IPython*. Sebastopol, Ca: O’reilly Media, Inc., October.
* Müller, A.C. and Guido, S. (2017). *Introduction to machine learning with Python: a guide for data scientists*. Beijing: O’reilly.

# Appendix

## Appendix 1: Data Dictionary

These are all the features we used in our data set and the description of each for a better understanding.

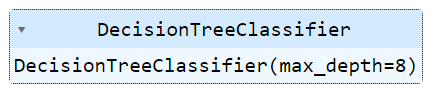
*Table 4. Data dictionary “*online\_shoppers\_intention*” data set*



## Appendix 2: Building a Decision Tree Model

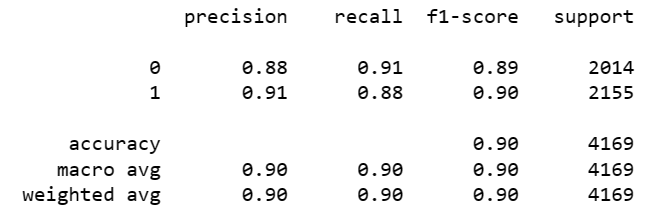
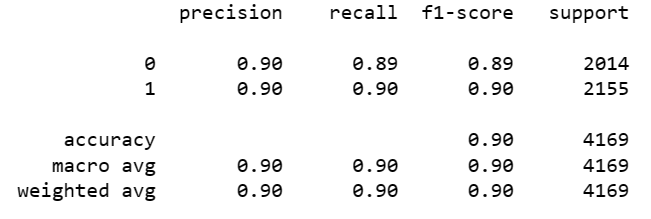
In this case, we used the Decision Tree Model, where we import the Decision Tree Classifier from the Sklearn library and then evaluate it using the default parameters, giving us as a result the following data *Figure 22.*

After that, Grid Search was used to find the optimal hyperparameters, and we proceeded to evaluate the model again, obtaining the information shown in *Figure 23*.

****

*Figure 21.**Decision Tree Model*

|  |  |
| --- | --- |
| **Apply the Decision Tree Model** | **Decision Tree Model with Hyperparameters** |



|  |  |
| --- | --- |
| *Figure 22.**Classification Report - Accuracy, Precision, and Recall Results with the default hyperparameters* | *Figure 23.**Classification Report - Accuracy, Precision, and Recall Results with the hyperparameters the best values* |

The following observations are related to *Figure 22* because it is giving the best results with default hyperparameters.

Precision: Out of all the shoppers that the model predicted would shop, only (1 = 91%) actually did.

Recall: Out of all the shoppers that actually did buy, the model only predicted this outcome correctly for (1-88%) of those shoppers.

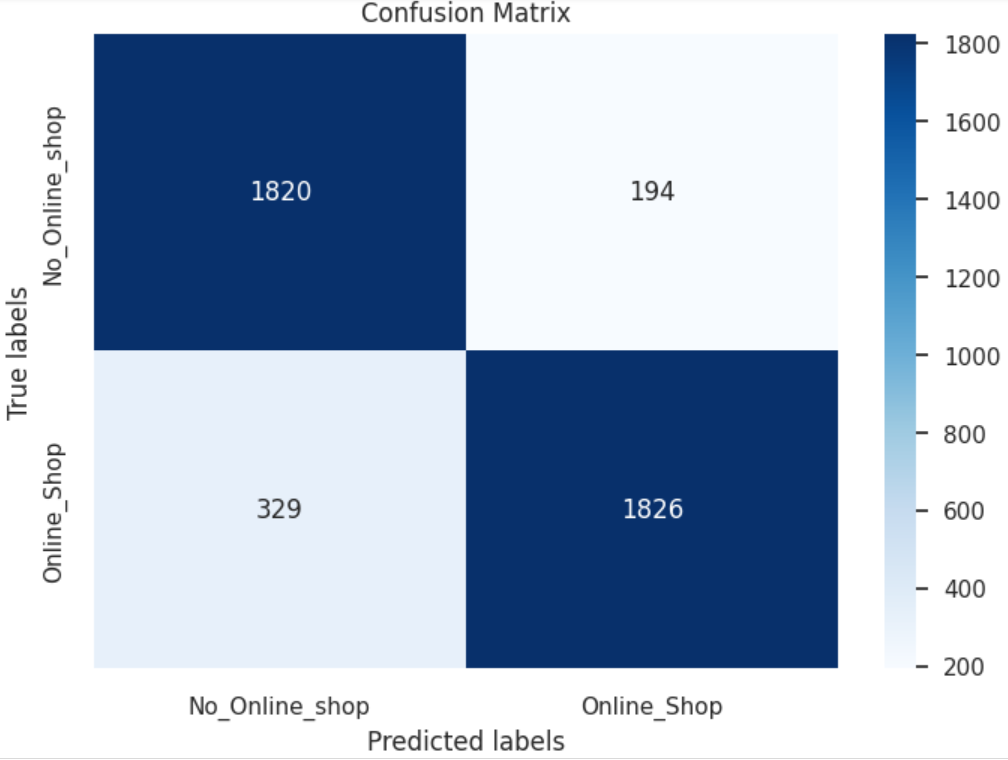
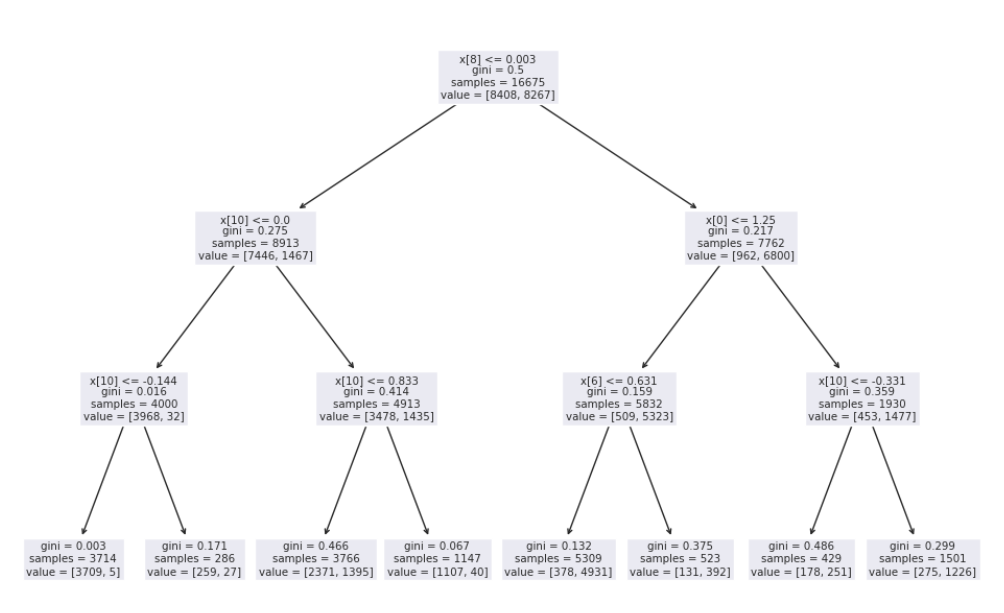
****

Figure 24. Confusion Matrix

*Decision tree model*

**Results and analysis**

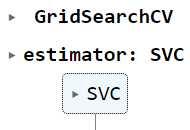
Results showed that the **Decision Tree Classifier Model** performed somewhat worse than the other two models, with a global precision of 91% and lower precision, recall, and f1-score values when compared to the Random Forest model.

****

*Figure 25. Displaying the Decision Tree Model by reducing the max depth*

## Appendix 3: Building SVM Model

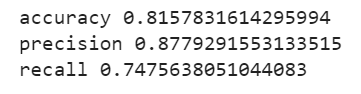
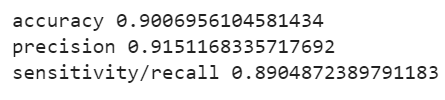
The last model we will use will be the SVC which is a classification supervised model, and similarly to the previous steps, we'll use grid search to find the optimal hyperparameters, apply them, and then assess the model using the best hyperparameters.

****

*Figure 26.**Random Forest Classifier Model*

In the following results, we can see that our model has better accuracy after applying the hyperparameters, where it seems the precision and the recall increase notoriously (*Figure 28)*.

|  |  |
| --- | --- |
| **Apply SVM Model** | **SVM Model with Hyperparameters** |

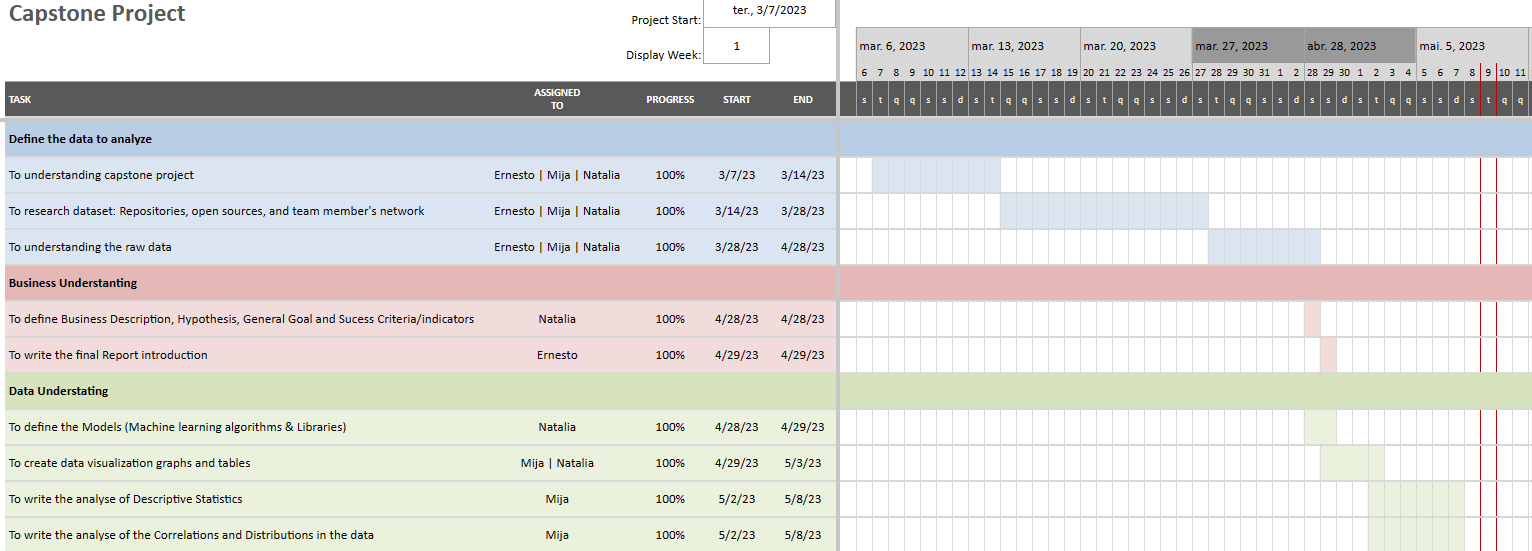
****  

|  |  |
| --- | --- |
| *Figure 27.**Accuracy, Precision, and Recall Results with the default hyperparameters* | *Figure 28.**Accuracy, Precision, and Recall Results with the hyperparameters the best values* |

**Results and analysi**s

The results for the **SCV** show that the precision for the positive class is 0.91, which means that out of all instances predicted as positive, only 91% were actually positive. In the recall for the positive class is 0.89, which indicates that out of all actual positive instances, only 89% were correctly classified by the model.

## Appendix 4: CRISP-DM



*Figure 29.**CRISP-DM Methodology - Part 1*