SHOPPERS' ONLINE ACTIVITY ANALYSIS

NAME: ANDREAS GREGORIADES

Contents

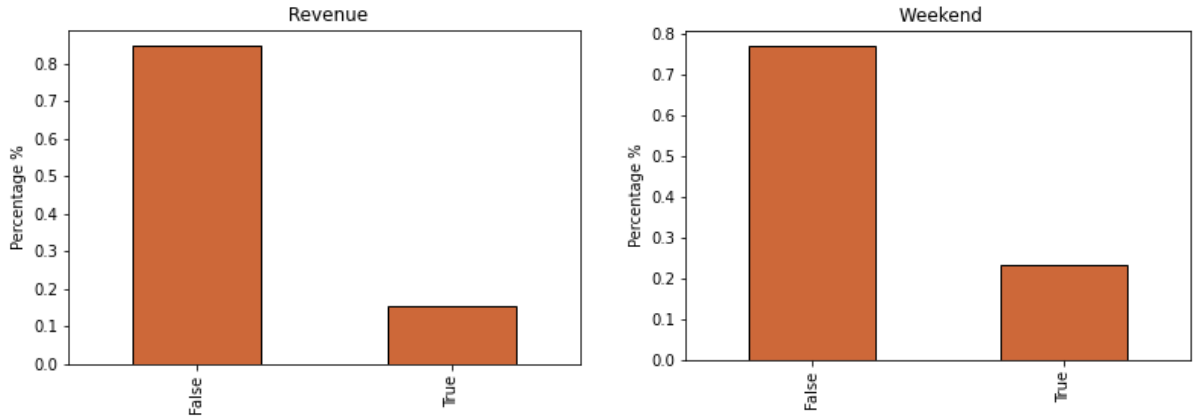
Introduction

Machine learning is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. In this project the purpose was to develop machine learning models to predict e-commerce visitors' purchasing intention.

Data exploration

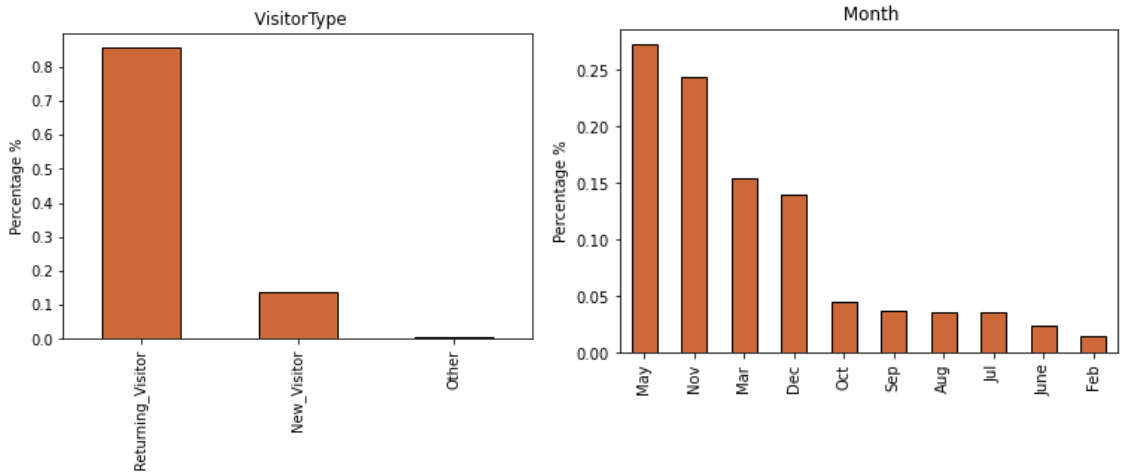
Having numerous variables with a great number of elements each, it is imperative for data exploration to take place in order to understand the data characteristics. With a first glance 18 different variables of various types can be observed and some of their descriptive stats [Appendix 1].

The "Revenue" variable is used as the class label and is found to have 84.5% unsuccessful purchases and that only 23% visit the website on the weekends with the majority visiting on the weekdays, indicating that overall more people visit on the weekdays by 4% [Graph 1].

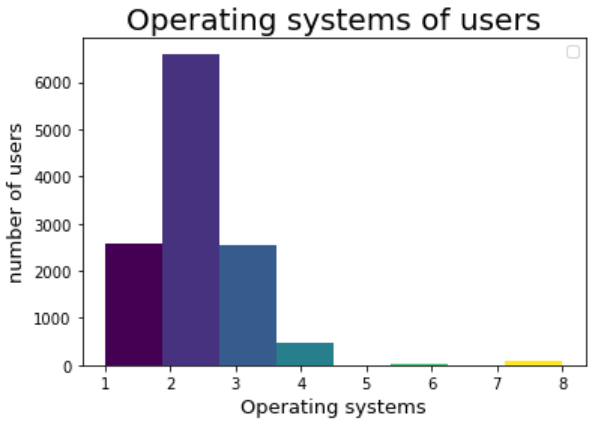


Graph 1: Revenue & Weekend percentage

More than 50% of the visitors visit in May and November and 85.5% are "Returning Visitors", while only 13% are new visitors [Graph 2]. Most people use operating system "2", followed by "1" & "3" having the same number of users [Graph 3].



Graph 2: Month & Visitor Type percentage



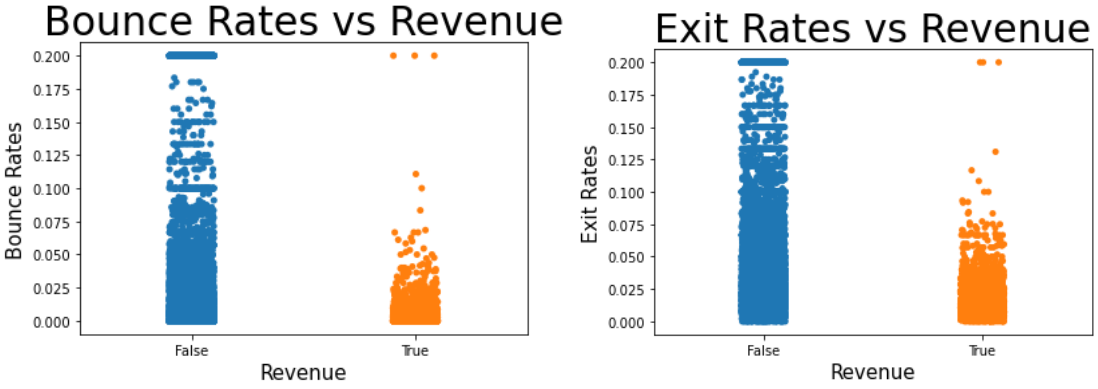
Graph 3: Operating systems of users

In order find correlations for each variable, a scatter plot[Appendix 2] was created which gave the insight for further analysis, such as the positive correlation between "ExitRates" and "BounceRates". A numeric table was then created to better display the most correlated variables [Table 1].

	FirstVariable	SecondVariable	Correlation
1	BounceRates	ExitRates	0.913004
2	ProductRelated	ProductRelated_Duration	0.860927
3	Informational	Informational_Duration	0.618955
4	Administrative	Administrative_Duration	0.601583
5	Administrative	ProductRelated	0.431119
6	Informational	ProductRelated_Duration	0.387505
7	Administrative	Informational	0.376850
8	Informational	ProductRelated	0.374164
9	Administrative	ProductRelated_Duration	0.373939
10	Administrative_Duration	ProductRelated_Duration	0.355422

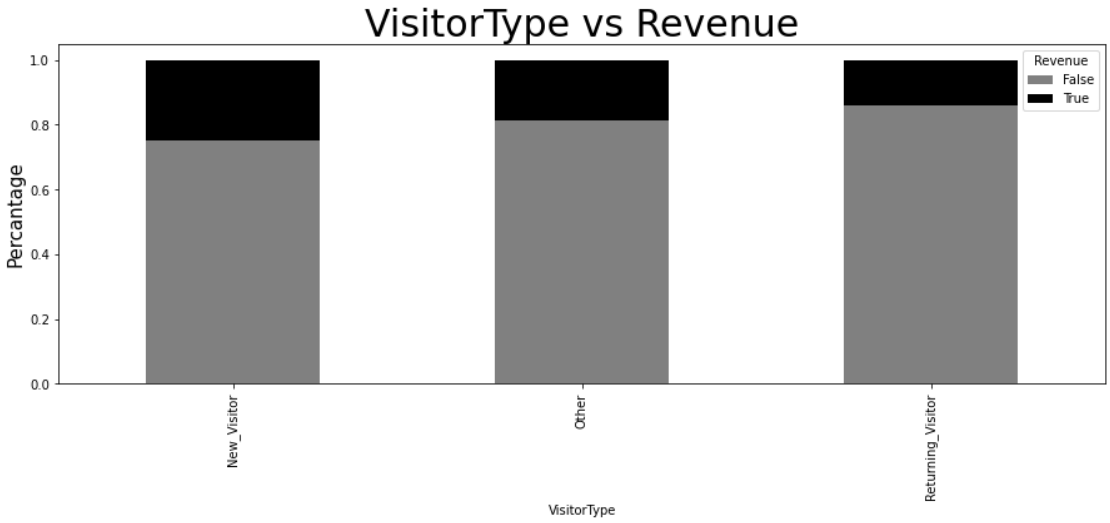
Table1: Most highly correlated variables

"Exit Rates" and "Bounce Rates" both show extreme correlation with Revenue, as the Exit Rate/Bounce Rate grows further away from zero the more the likelihood of the user purchasing a product plummets [Graph 4]. Contrary, an increase of "Page Value" indicates an increase of the purchasing likelihood, while "Product Related" is neutral in relation to Revenue[Appendix 3].



Graph 4: Bounce Rates & Exit Rates vs Revenue

An interesting observation is that even though the number of “New Visitors” is lower than “Returning Visitors”, over 20% of the former purchases from the website while less than 15% from the latter [Graph 5].



Graph 5: Visitor Type vs Revenue

Data Pre-processing

Various pre-processing procedures took place in order to prepare the data into a form that is likely to lead to better performance. Firstly, the missing data must be taken care of, a check was done to find if there are any null or empty spaces in the dataset, finding none.

Encoding Categorical Features

Categorical features such as “Month” and “Revenue” were then encoded in order to be used in the classifier models later on.

Feature Scaling

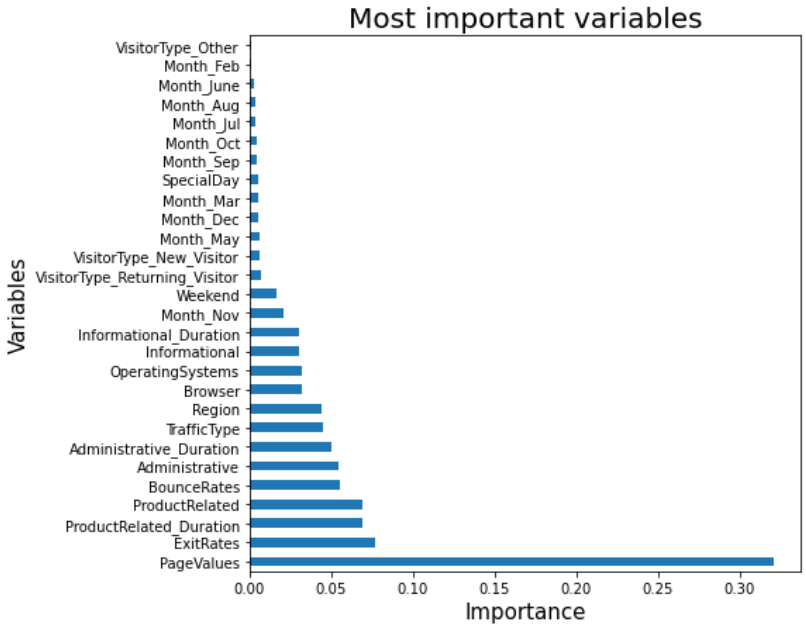
Feature scaling standardization took place in order to create the same weight of measurement for all the variables.

Imbalanced data

Imbalanced data exists on this dataset as it can be observed from [Graph 1] “Revenue” where only 15% is “True” and the rest are “False”. This creates a bias in the model. Because of the dataset size not being huge, under-sampling methods were discarded and only oversampling was considered. Synthetic oversampling method “SMOTE” was chosen as it proved to be better than other methods of handling imbalanced datasets. It significantly improved the recall. This technique was used only for the training set of the model as otherwise would give false results.

Data reduction

In order to find data that may not be very important a technique was used to find the feature importance of every feature from the dataset. Feature Importance works by giving a relevancy score to every feature in the dataset, the higher the score it will give, the higher relevant that feature will be for the training of the model [Graph 7]. By removing the 13 worst variables a test was conducted finding that the Confusion Matrix of the models and the CAP curve showed slighlty worse results. This was a clear indication to not remove the variables as the computation time did not change by much and trading-off the accuracy of the model is not worth it.



Graph 7: Rating the importance of the variables

Model Implementation

Model choice

In order to select 3 representative classification methods, 6 classification methods were tested: SVM, Naïve Bayes, Logistic Regression, Kernel SVM, K-NN and Decision tree.

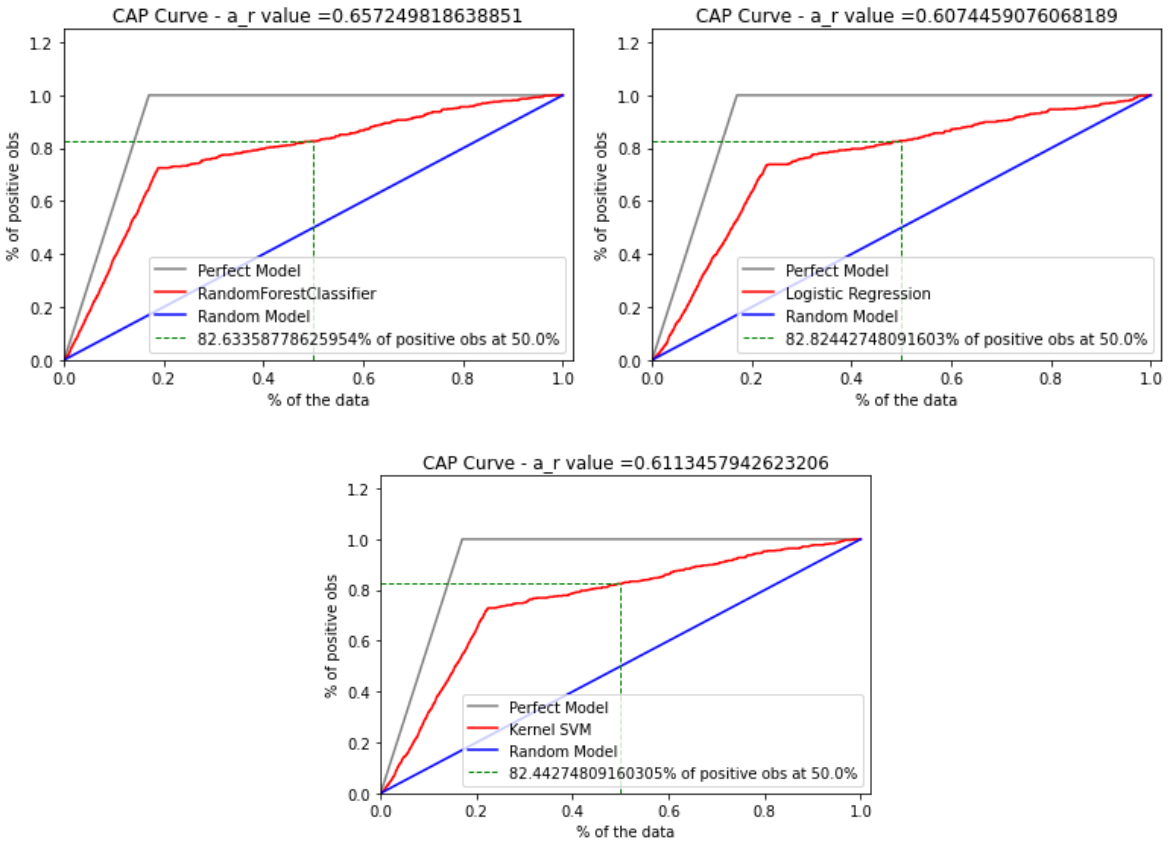
The way the 3 representative methods were chosen, was according to the Confusion Matrix and the accuracy score derived from the Confusion Matrix[Appendix 4]. In that regard, Random forest, Logistic Regression and Kernel SVM were chosen for having high accuracy and recall, as that is essential for lessening the chance of a mistake when assessing if a website visitor will truly purchase a product, and avoid having a large number of False Positives.

Also hyper-parameter optimization was performed with positive results on accuracy and recall.

Model evaluation

Various performance metrics were used for the models' evaluation such as the Confusion Matrix and the accuracy score derived from the Confusion Matrix. Also recall, precision and f1-score for positives and negatives.

The Cumulative Accuracy Profile curve[Appendix 5], was also used as a performance metric. The CAP curves for all models are close to the perfect model and the cumulative number of elements meeting the property at 50% of the totality on the x-axis are very high, 82% on all models, indicating that the models are strong. The accuracy ratio (AR) is defined as the ratio of the area between the model CAP and random CAP, and the area between the perfect CAP and random CAP. In a successful model, the AR has values between zero and one, and the higher the value is, the stronger the model. All the models have above 0.6 AR value with the highest being the Random Forest model.



Graph 8: The 3 models' CAP curves

Moreover, the end-result of accuracy score is quite high reaching 89% on the Random Forest[Appendix 6], with high recall. High recall is essential as the False-Positives in this scenario are worse than the FN.

Result analysis

The importance of the pre-processing is evident as they were used to resolve potential problems. Helping solve the problem of the unbalanced dataset SMOTE was implemented, improving the recall rate of the Positives and achieving less biased result. Feature scaling played a big role in creating an equal-weighted dataset with which all variables were implemented in the model having the same unit of measurement, avoiding a false model.

Hyper-parameter tuning was also an important factor of the models' success, as by trying different combinations, classifiers were optimized and became fine-tuned showing increases in precision, accuracy and recall.

All 3 models give satisfying results, with the "Random Forest" being the most notable, having better recall, precision and f-1 score than the "Kernel SVM" & "Log Regression" [Appendix 6]. The result is very logical as the Random Forest can also be considered as a "Decision Tree" ensemble, tackling some of its potential problems such as over-fitting and also lowering the error rates.

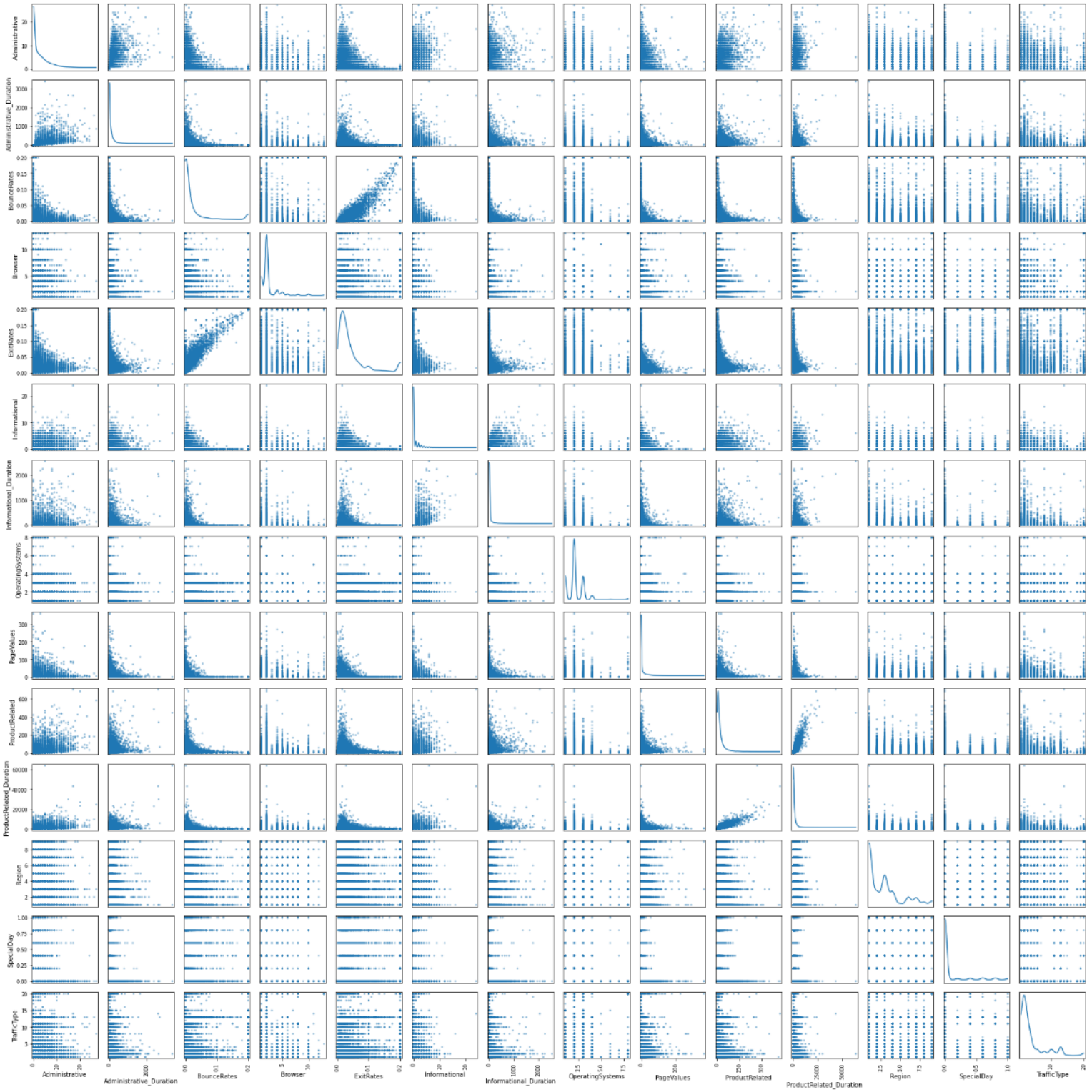
APPENDIXES

APPENDIX 1

	Administrative	Administrative Duration	Informational	Informational_Duration	Product Related	ProductRelated_Duration	Bounce Rates	ExitRates
count	12330.0	12330.0	12330.0	12330.0	12330.0	12330.0	12330.0	12330.0
mean	2.31	80.81	0.50	34.47	31.73	1194.74	0.02	0.04
std	3.32	176.77	1.27	140.74	44.47	1913.66	0.04	0.048
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	7.0	184.13	0.0	0.014
50%	1.0	7.5	0.0	0.0	18.0	598.93	0.0031	0.025
75%	4.0	93.25625	0.0	0.0	38.0	1464.15	0.016	0.05

	PageValues	SpecialDay	OperatingSystems	Browser	Region	TrafficType
count	12330.0	12330.0	12330.0	12330.0	12330.0	12330.0
mean	5.88	0.061	2.12	2.35	3.14	4.069
std	18.56	0.19	0.91	1.71	2.40	4.025
min	0.0	0.0	1.0	1.0	1.0	1.0
25%	0.0	0.0	2.0	2.0	1.0	2.0
50%	0.0	0.0	2.0	2.0	3.0	2.0
75%	0.0	0.0	3.0	2.0	4.0	4.0
max	361.7	1.0	8.0	13.0	9.0	20.0

APPENDIX 2


Administrative
Administrative_Duration
BounceRates
Browser
ExitRates
Informational
Informational_Duration
OperatingSystems
PageValues
ProductRelated
ProductRelated_Duration
Region
SpecialDay
TrafficType

APPENDIX 3

Graph: Page Value & ProductRelated vs Revenue

APPENDIX 4

Accuracy score	
Decision Tree	0.8550113525786571
K-NN	0.7680830360038923
Kernel SVM	0.8540382744080441
Logistic Regression	0.8507946805060006
Naïve Bayes	0.5445994161530976
Random Forest	0.8822575413558222
SVM	0.8731754784301006

		Class=Negative	Class=Positive
Decision Tree	**Class=Negative** **Class=Positive**	2307 195	252 329
K-NN		2027 183	532 341
Kernel SVM		2251 142	308 382
Logistic Regression		2237 138	322 386
Naïve Bayes		1210 55	1349 469
Random Forest		2373 177	196 347
SVM		2314 146	245 378

APPENDIX 5

CAP Curve explanation:

The CAP can be used to evaluate a model by comparing the current curve to both the 'perfect' and a randomized curve. A good model will have a CAP between the perfect and random curves; the closer a model is to the perfect CAP, the better is.

APPENDIX 6

Random Forest Classification	precision	recall	f1-score	support
0	0.94	0.92	0.93	2559
1	0.66	0.73	0.69	524
accuracy	0.89			3083

Logistic Regression	precision	recall	f1-score	support
0	0.94	0.87	0.91	2559
1	0.55	0.74	0.63	524
accuracy	0.85			3083

Kernel SVM	precision	recall	f1-score	support
0	0.94	0.88	0.91	2559
1	0.55	0.73	0.63	524
accuracy	0.85			3083