CMT307 Coursework 1

Student number: C21010417

Question 1

Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$ Precision = $\frac{TP}{TP+FP}$

Recall = $\frac{TP}{TP+FN}$ F-measure = $\frac{2\times precision\times recall}{precision+recall}$

Step 1 - Finding the total number of true positives, true negatives, false positives, and false negatives.

True positives = 7

True negatives = 7

False positive = 4

False negative = 2

Step 2 – Calculating the accuracy score, precision, and recall.

Accuracy = (7+7) / (7+7+2+4) = 0.7

Precision = 7 / (7+4) = 0.636363…

Recall = 7 / (7+2) = 0.77777….

Step 3- Calculating the f-measure.

F-measure = (2*(7/11) *(7/9)) / ((7/11)+(7/9)) = 0.7

Question 2

Introduction and data exploration

The aim of this project is to create several machine learning (ML) models to predict online shoppers' purchasing intentions when browsing an e-commerce site. The dataset includes 12330 rows and 18 complete (no null-values) columns, 7 being categorical and 11 numerical. The unique value count varies between columns, categorical columns contain 20 or less unique values, and the number of unique values for the numerical columns span from 6 to 9551.

A correlation heatmap was created to highlight the relationships between the features. The features which relate to the number of pages visited and the duration spent on such pages covary with each other. The target feature, Revenue, has the highest correlation coefficient with PageValues. This feature is associated with the amount of revenue generated by the site and hence directly correlates with Revenue. The numerical features show stronger correlation with the target variable than the categorical features.

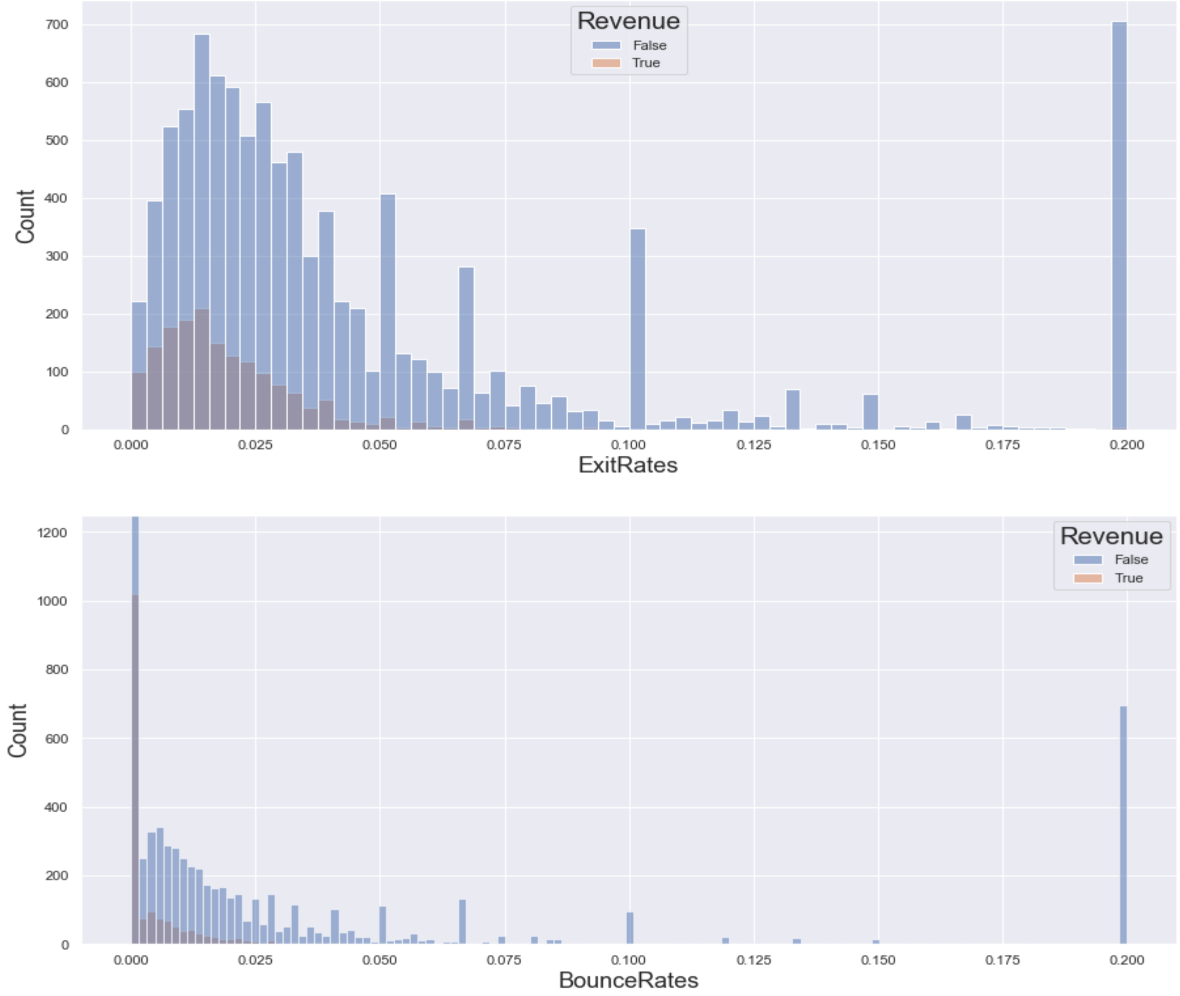

Figure 1: Both features show moderate positive correlation with the target variable, Revenue.

Data preprocessing

The first step of data preprocessing was removing outliers in the columns with high standard deviations: Administrative_Duration, Informational_Duration, ProductRelated and ProductRelated_Duration. The large population size of 12330 allows Gaussian approximation and hence the method z-score rejection outlier removal could be used. The median was used calculate the z-score instead of the mean because the mean is not robust due to the magnitude of some outliers. Z-scores for each row, for each of the 4 features was calculated, then all rows containing a z-score greater than 3 were removed. In total, 829 rows were classed as containing outliers and were removed.

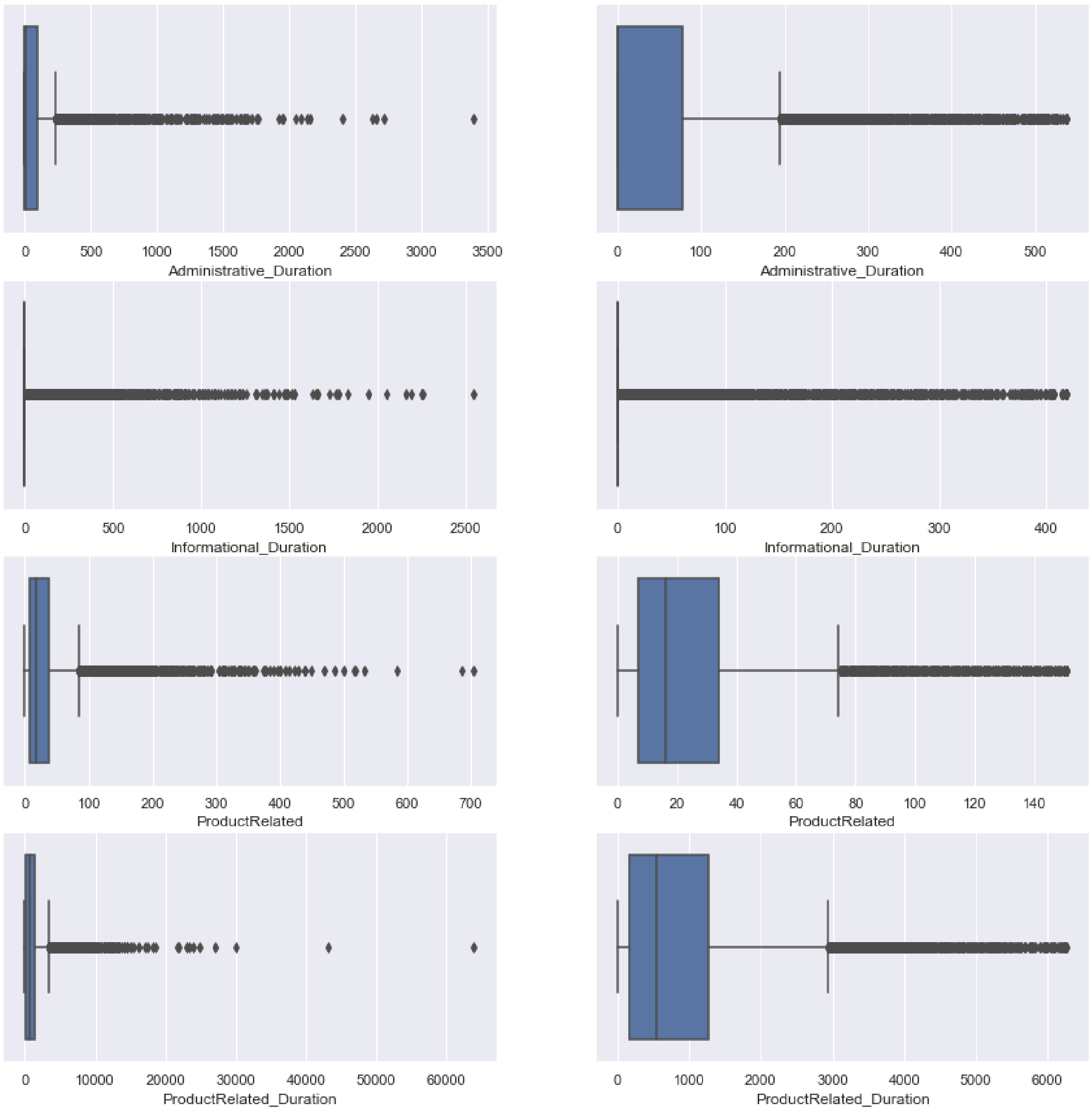

Figure 2: The 4 variables containing outliers. Left with outliers, and right without.

The outliers can be seen on the left plots on figure 2, the right shows the features with the outliers removed. Note how the area of the box increases in relation distribution of data in those variables when comparing pre-outlier and post-outlier removal, this will allow ML models to constrain these features more effectively and hence yield better results.

Feature engineering is an important step in preparing the data to be fed into ML algorithms, 5 new features were engineered, ExitBounceSum; the sum of BounceRates and ExitRates, TotalTime; the sum of all 3 pages duration features, and MeanPageAdminTime, MeanPageTimeInformational and MeanPageTimeProduct; all of which are the respective number of pages divided by the duration spent browsing them. Although these new features will covary strongly with the features used to make them, they may provide ML algorithms with more classifying power than the features used to engineer them, and allows potentially less features to be used in the modelling process which speeds up model training and often improves the performance.

Categorical features were transformed to numeric datatypes using one hot encoding, this permits these variables to be able to be used by ML algorithms. The continuous numerical features were transformed into bins, discretising continuous data decreases model training time and can increase the performance. The transformation steps were incorporated into a pipeline. Both the transformations and feature engineering were completed after splitting the data into test and training sets, done to avoid data contamination.

Model implementation

The variable to be predicted, Revenue, is a binary column and hence this is a binary classification problem and therefore will require a supervised algorithm to fit the purpose. Initially 5 different algorithms were tested with their default parameters: support vector machine (SVM), random forest (RF), gradient boosting classifier (GBC), decision tree classifier and logistic regression classifier. SVM and logistic regression were chosen as these are both specifically made for binary classification. Decision trees can be trained quickly with binary classification, RFs are an ensemble of decision tree classifiers and often achieve more effective results than a single decision tree. GBCs work by creating decision trees, with each new tree, minimizing the differences between the training target variable and the predicted value.

The metrics used to compare the effectiveness models are recall, precision and F1 score. Recall is used as this is an unbalanced dataset, with far less positives outcomes (revenue) than negative and hence detecting as many positives as possible is important. Precision is important as too many false negatives would be misleading. F1 is a nice metric because it combines both precision and recall into one metric. The 3 highest F1 scoring models were: RF SVM and GBC.

There were 2 methods of feature selection tested, FKneighbors feature selection and random forest feature selection. The random forest feature selection achieved highest recall and the lowest number of features, so this was the chosen method. The selected features: ExitRates, BounceRates, MeanPageTimeProduct, MeanPageAdminTime, ExitBounceSum, TotalTime, PageValues, ProductRelated_Duration, ProductRelated and Administrative. Synthetic minority oversampling technique (SMOTE) was used to combat the data imbalance of the positive and negative classes. Random hyperparameter searches using cross-validation were used to optimise the models without overfitting the training data. With multiple iterations, the parameters which scored the highest F1-score for each model were chosen.

Performance evaluation

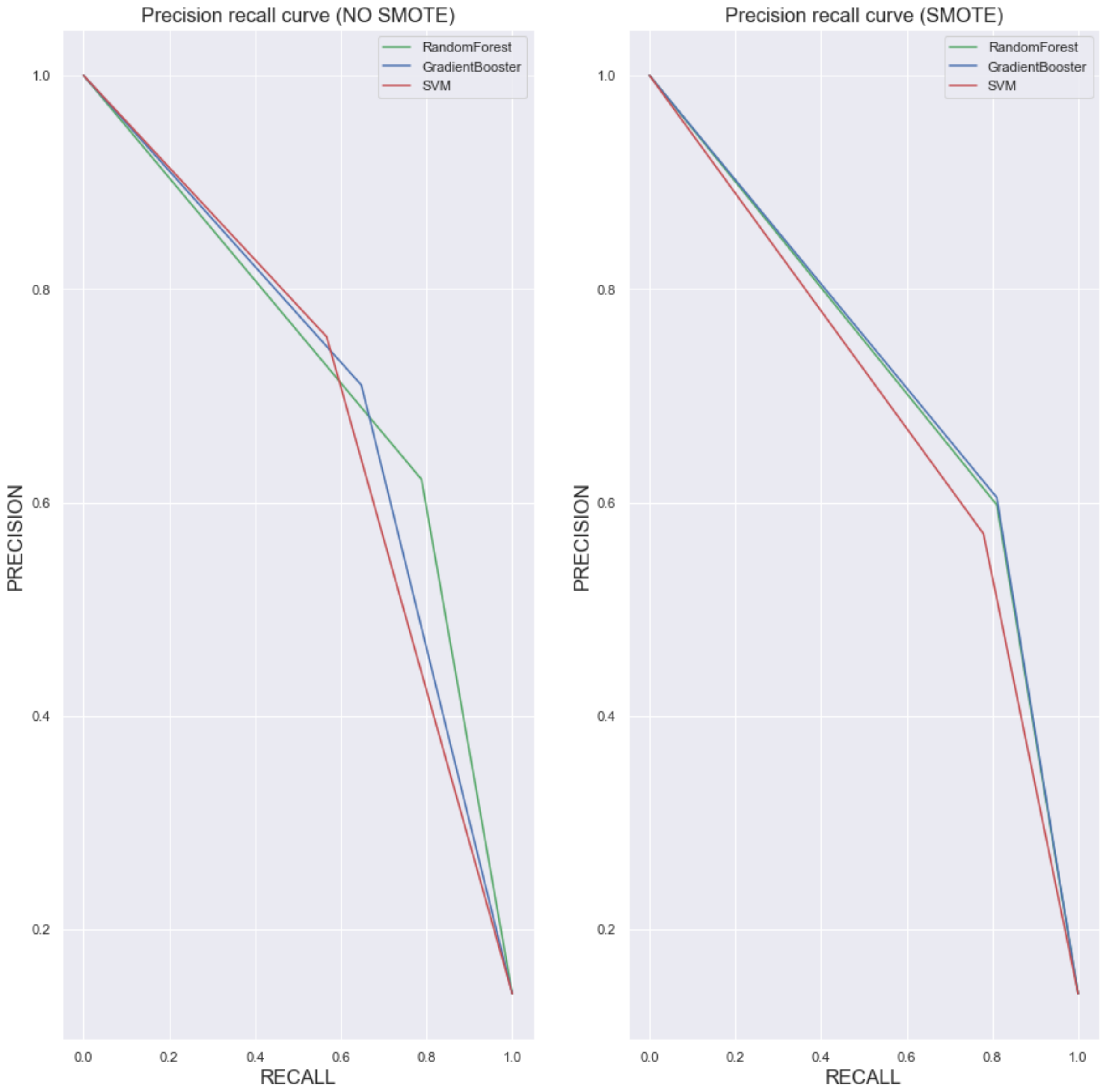

Figure 3: Precision - recall curves, with SMOTE and without SMOTE

Classifier	**F1 without SMOTE**	**F1 with SMOTE**	**AUC without SMOTE**	**AUC with SMOTE**
RF	0.70	0.69	0.72	0.72
GBC	0.68	0.69	0.70	0.70
SVM	0.65	0.66	0.72	0.69

Table 1: Performance of models

Overall, the highest mean F1-score is achieved with SMOTE and the highest mean AUC is achieved without SMOTE, however this is due to higher precision rates at lower recall levels, whereas for this project, recall is more favored and therefore, the most suitable model is RF without SMOTE, achieving the highest F1-score.

Results analysis and discussion

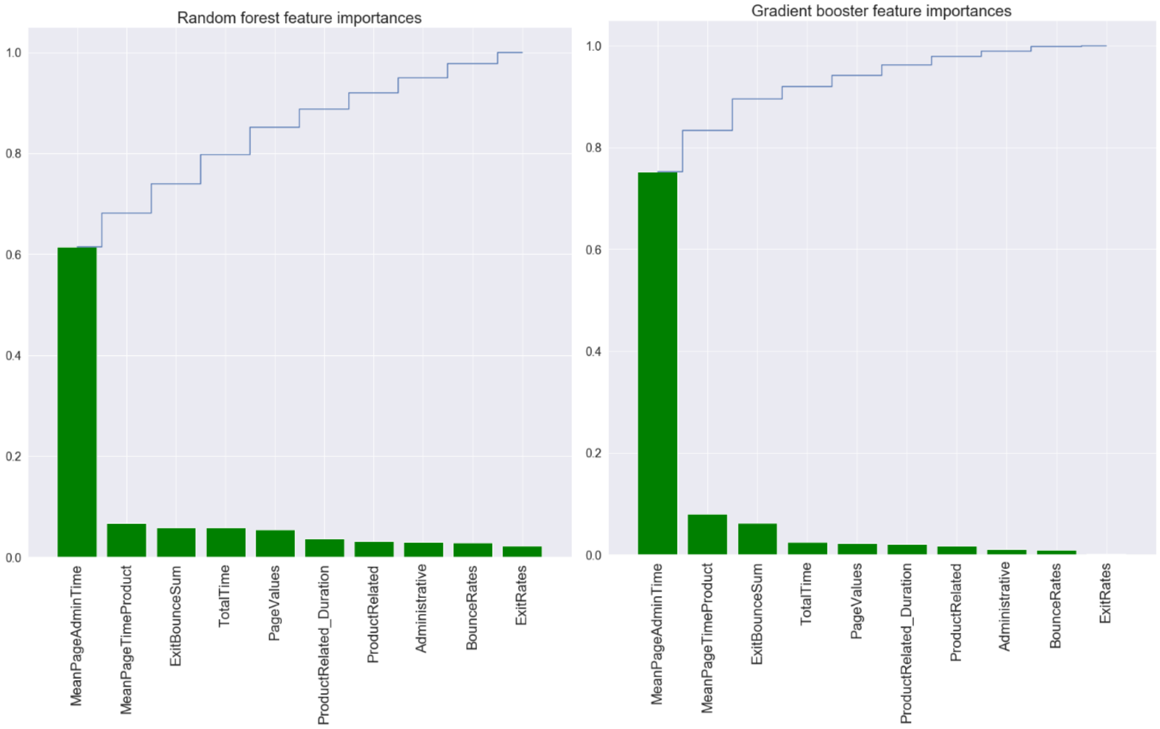

Figure 4&5: Importance of features

The importance of features in the RF and GBC models are shown in figures 4&5, feature importance cannot be obtained for the SVM. The mean time spent on account management pages (MeanPageAdminTime) is the most important feature by a considerable amount for both models. This could be due to the time requirement of creating an account to make a purchase, something someone is unlikely to do unless they are planning on purchasing. The next most important features for both models, is the mean time spent on product pages (MeanPageTimeProduct) and the sum of exit and bounce rates (ExitBounceSum). To maximise revenue, these results show to encourage account creation and maximise time customers spend on product pages.