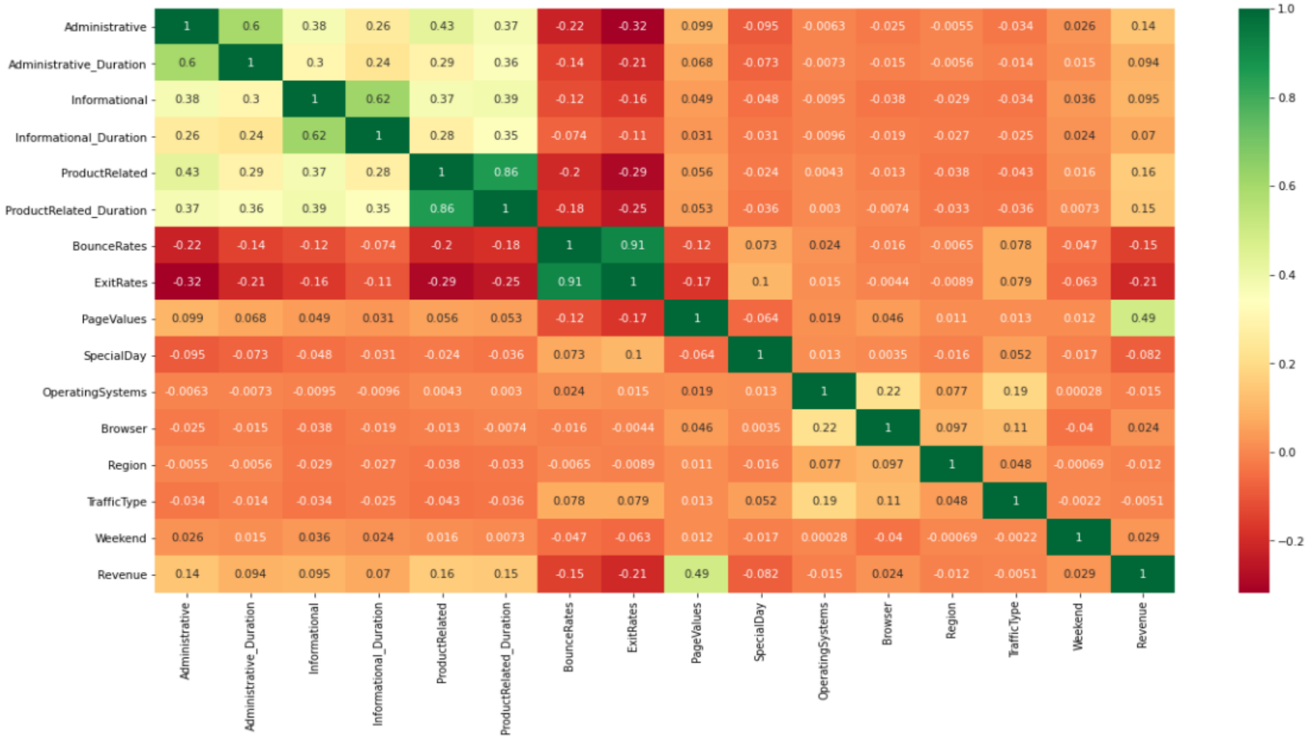
Data exploration

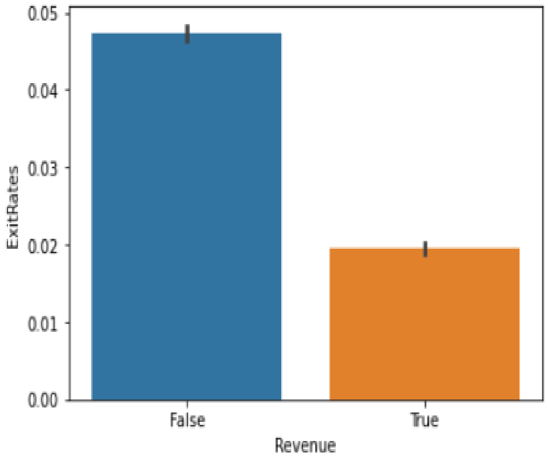
In the data provided, the first 10 columns are numerical values and the last 8 are categorical values. From these categorical features, the Month and Visitor Type are objects, Weekend and Revenue are Boolean, the other four features are all nominal variables which mean they represent each a respective category without having an order for the numbers.

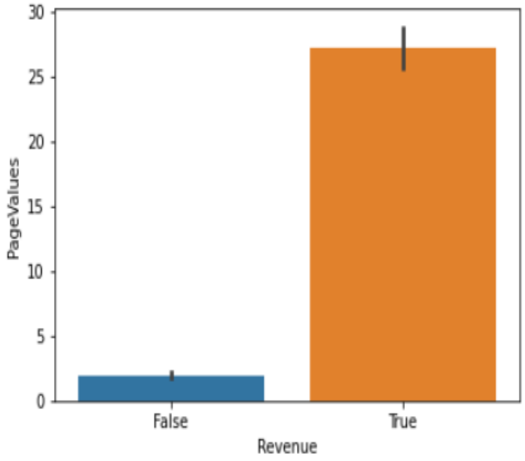
We need to analyze if there is any relation between the variables in the data using a correlation heatmap that will check the strength and direction of the correlation between the features.

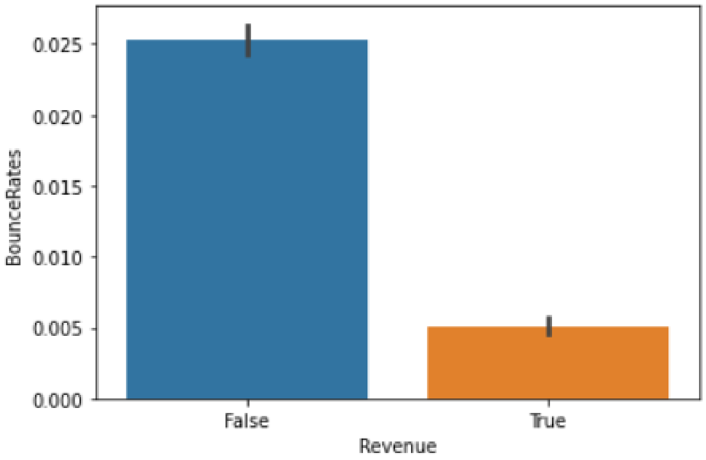


	Administrative	Administrative_Duration	Informational	Informational_Duration	ProductRelated	ProductRelated_Duration	BounceRates	ExitRates	PageValues	SpecialDay	OperatingSystems	Browser	Region	TrafficType	Weekend	Revenue
Administrative	1	0.6	0.38	0.26	0.43	0.37	-0.22	-0.32	0.099	-0.095	-0.0063	-0.025	-0.0055	-0.034	0.026	0.14
Administrative_Duration	0.6	1	0.3	0.24	0.29	0.36	-0.14	-0.21	0.068	-0.073	-0.0073	-0.015	-0.0056	-0.014	0.015	0.094
Informational	0.38	0.3	1	0.62	0.37	0.39	-0.12	-0.16	0.049	-0.048	-0.0095	-0.038	-0.029	-0.034	0.036	0.095
Informational_Duration	0.26	0.24	0.62	1	0.28	0.35	-0.074	-0.11	0.031	-0.031	-0.0096	-0.019	-0.027	-0.025	0.024	0.07
ProductRelated	0.43	0.29	0.37	0.28	1	0.86	-0.2	-0.29	0.056	-0.024	0.0043	-0.013	-0.038	-0.043	0.016	0.16
ProductRelated_Duration	0.37	0.36	0.39	0.35	0.86	1	-0.18	-0.25	0.053	-0.036	0.003	-0.0074	-0.033	-0.036	0.0073	0.15
BounceRates	-0.22	-0.14	-0.12	-0.074	-0.2	-0.18	1	0.91	-0.12	0.073	0.024	-0.016	-0.0065	0.078	-0.047	-0.15
ExitRates	-0.32	-0.21	-0.16	-0.11	-0.29	-0.25	0.91	1	-0.17	0.1	0.015	-0.0044	-0.0089	0.079	-0.063	-0.21
PageValues	0.099	0.068	0.049	0.031	0.056	0.053	-0.12	-0.17	1	-0.064	0.019	0.046	0.011	0.013	0.012	0.49
SpecialDay	-0.095	-0.073	-0.048	-0.031	-0.024	-0.036	0.073	0.1	-0.064	1	0.013	0.0035	-0.016	0.052	-0.017	-0.082
OperatingSystems	-0.0063	-0.0073	-0.0095	-0.0096	0.0043	0.003	0.024	0.015	0.019	0.013	1	0.22	0.077	0.19	0.00028	-0.015
Browser	-0.025	-0.015	-0.038	-0.019	-0.013	-0.0074	-0.016	-0.0044	0.046	0.0035	0.22	1	0.097	0.11	-0.04	0.024
Region	-0.0055	-0.0056	-0.029	-0.027	-0.038	-0.033	-0.0065	-0.0089	0.011	-0.016	0.077	0.097	1	0.048	-0.00069	-0.012
TrafficType	-0.034	-0.014	-0.034	-0.025	-0.043	-0.036	0.078	0.079	0.013	0.052	0.19	0.11	0.048	1	-0.0022	-0.0051
Weekend	0.026	0.015	0.036	0.024	0.016	0.0073	-0.047	-0.063	0.012	-0.017	0.00028	-0.04	-0.00069	-0.0022	1	0.029
Revenue	0.14	0.094	0.095	0.07	0.16	0.15	-0.15	-0.21	0.49	-0.082	-0.015	0.024	-0.012	-0.0051	0.029	1

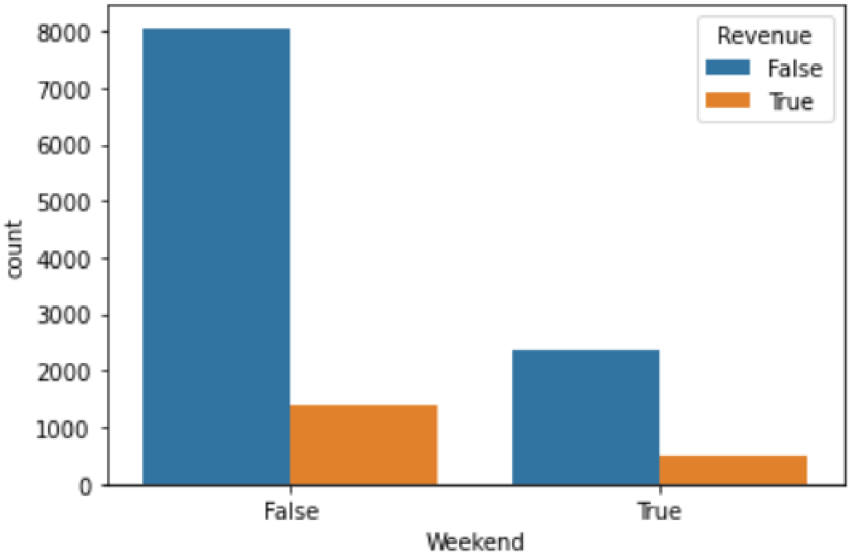
Checking the Revenue row as it is our goal to predict this outcome, we can notice the highest correlation of +0.49 with the Page Values, and a small negative relation with Exit Rates and Bounce Rates.



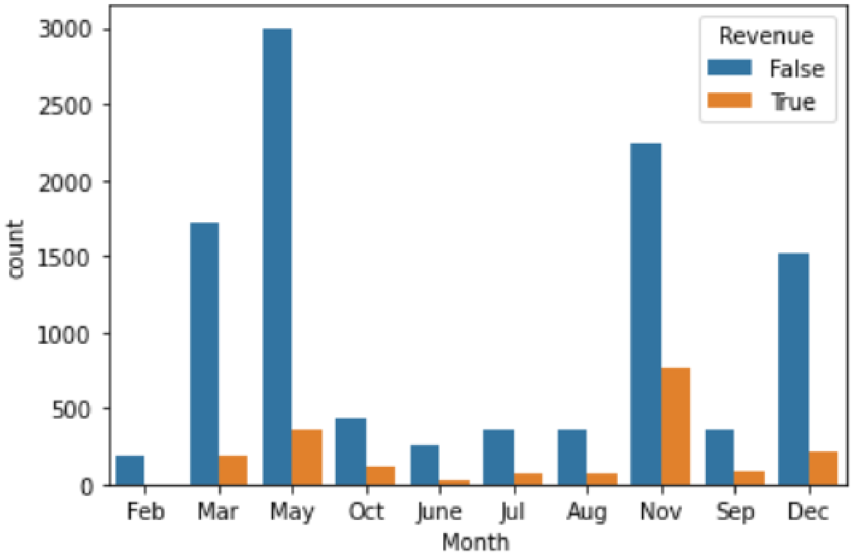




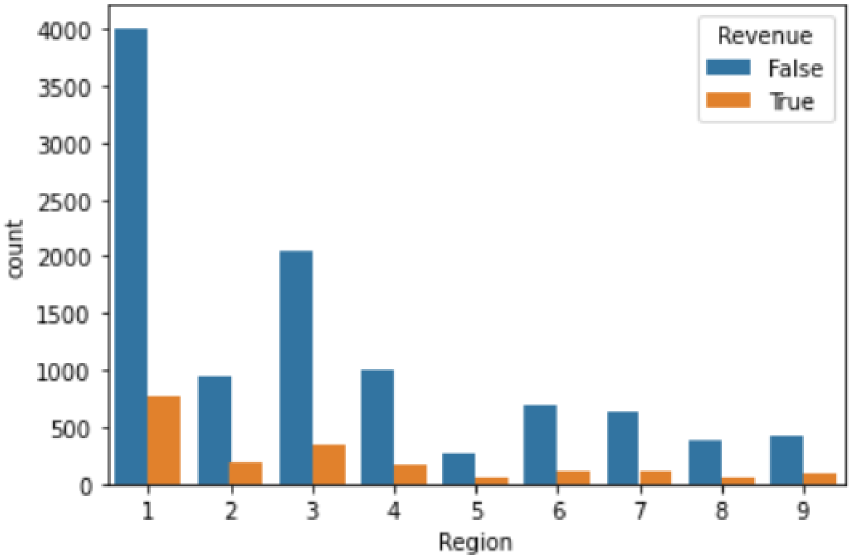
From the three figures, we might deduce that the higher the Page Values the more chance there is for a customer to buy a product, but the higher the Exit Rates and Bounce Rates the more chance the outcome will be to not buy.



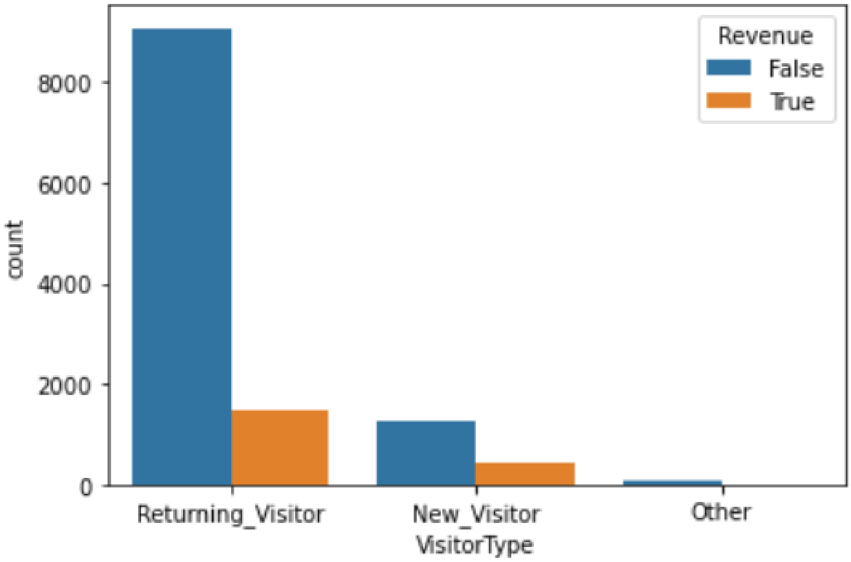
There is a lot of people visiting the sites during the week and the majority are not buying. During the weekend, the same thing happens but the number of total visits is much lower.



There is mainly one month that stands out between them all which is November with the highest number of purchases. May on the other hand has the highest number of visitors that does not buy.

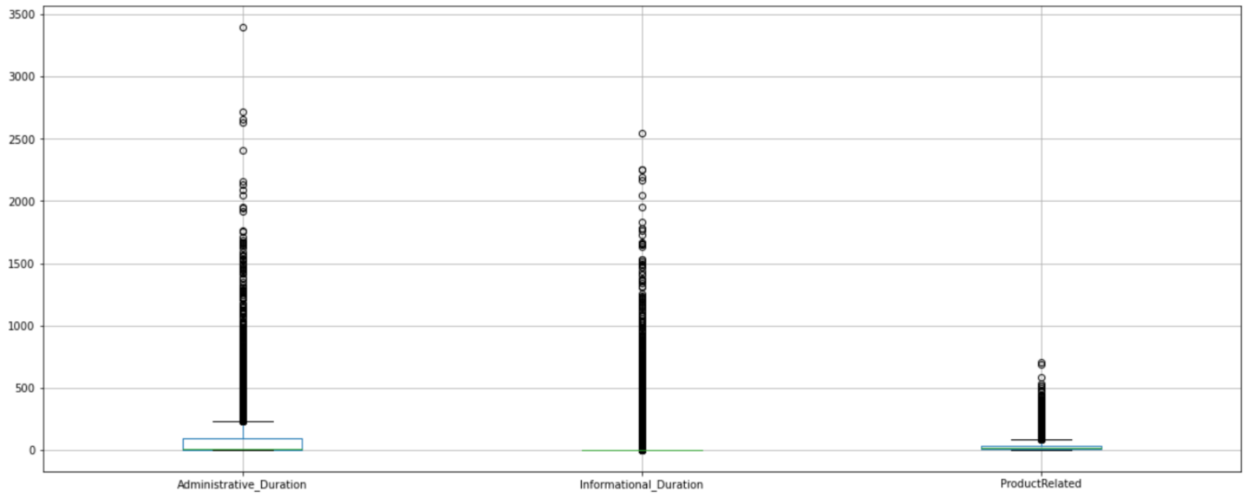


The highest number of values is in the region 1 that dominates the chart with region 3 , 4 and 2 following respectively.



Returning Visitors are the main type of visitors with a very high number of customers not buying with a very low percentage of them buying. New visitors consist of customers that have higher chance to buy than returning ones.

Next, we should check for outliers and decide what to do with them. The simplest way is to plot boxplots and see whether there are points that are situated outside of the whiskers of the box plot.



This is one example of the boxplots, they all seem to have outliers. For this project, the outliers were not removed as they did not change or affect the outcome of the models used.

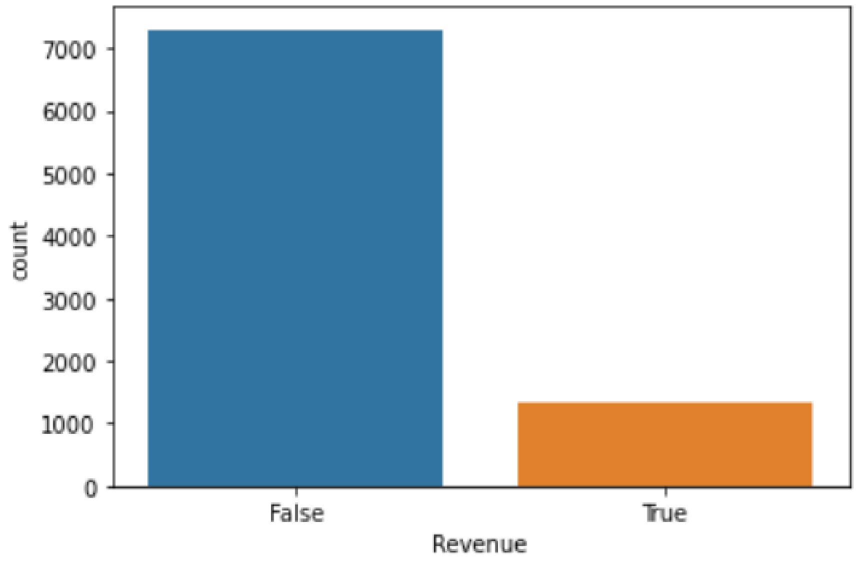
Data preprocessing

There are no missing values in this data and all the variables are in the correct ranges such as values like the Durations do not have negative numbers with a minimum of 0.

We split the data between the Revenue and the other features with a 70% for training the data and 30% for the testing part. All the data preprocessing techniques will be applied only on the first part of the split data, that is the X train and X test in this case, so that we insure there will be no data leakage to the results when we will test our prediction, because then we may have wrong answers and very high accuracy and an f1 score that is too good to be true.

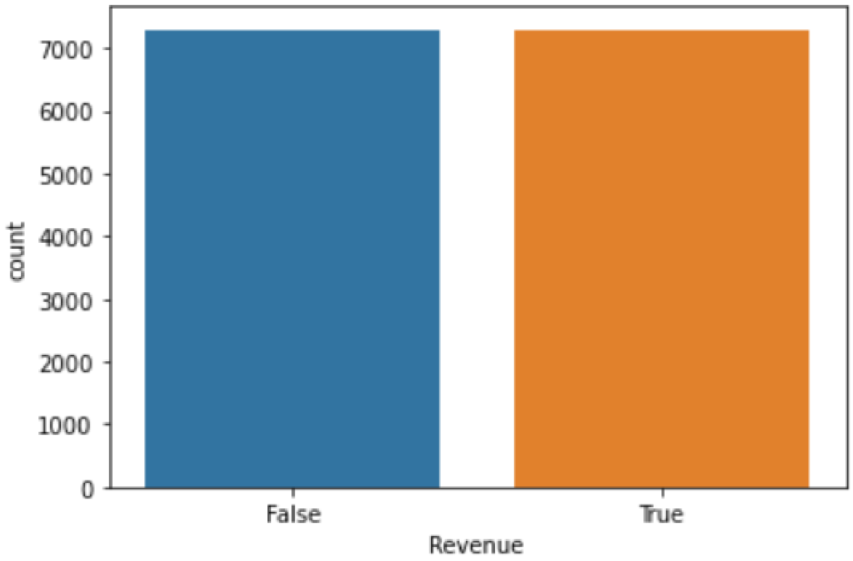
We are going to apply label encoding which will transform all categorical values to numerical values. For each month, a number will be assigned to represent it, from 1 to 9 as there is 9 months in this data set. Weekend will be 0 or 1 depending on if it is false or true and the Visitor Type will be from 0 to 2 as there is three type of visitors. The other categorical features do not need to be transformed as they are already in numerical form.

We will normalize the first 10 features, which are non-categorical. This will ensure that all the values are consistent with the same format and range, while keeping the information intact. Also this data does not follow a gaussian distribution.



We can clearly see that the two classes are not represented equally so this data is highly imbalanced, thus making prediction harder with this kind of data as we do not have a lot of information about the positive outcome we would like to predict.

We will be using an over sampling method, which will increase the number of the minority class of the Revenue to the number of the majority class so we can have a balanced data.



Model Implementation

We used three different kind of decision trees implementations as they are robust to outliers and are used for supervised classification problems and do not require a large amount of computing power to perform.

the random forest is a random grouping of many other decision trees. Random forest combines the output of many decision trees as an average of the scores.

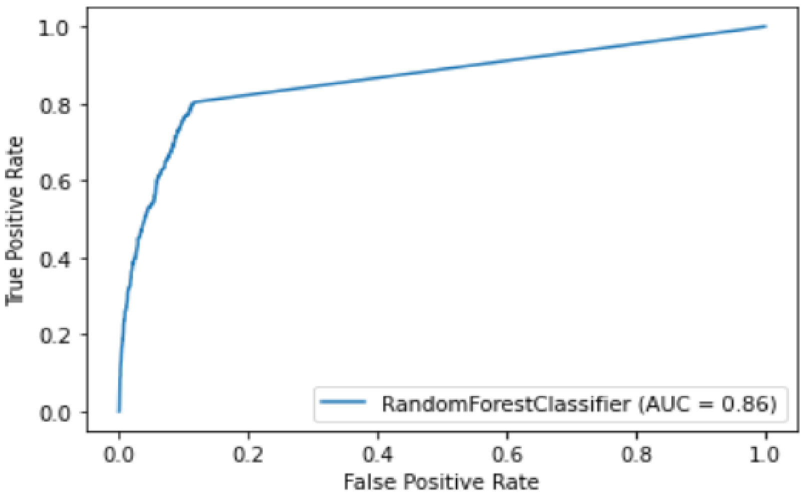
Bagging ensembles is another kind of classifier that is similar to the random forest, the difference is that the bagging considers all the features rather than a subset of feature for splitting a node. Usually, random forest is a better enhanced version of bagging.

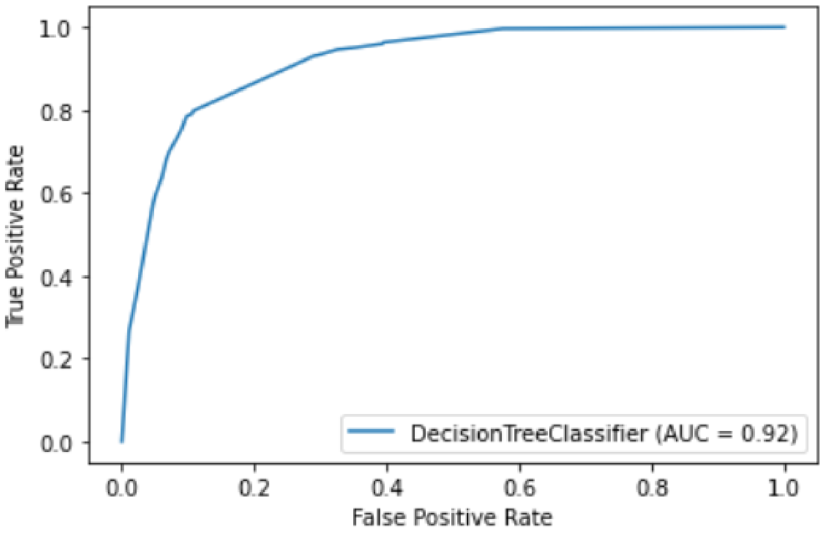
After the implementation of each model, they were all configured with hyperparameter optimization to try and get a better performance.

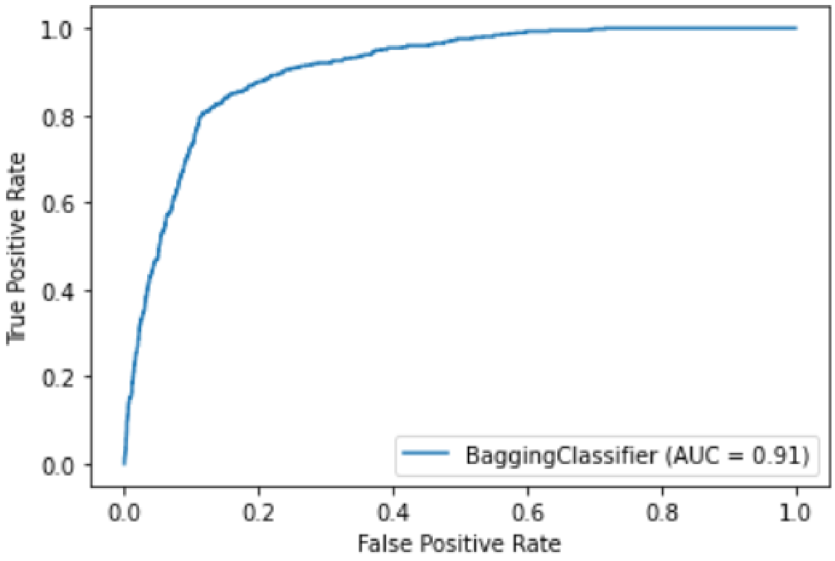
Then we chose what features are the most important for each model (except for the bagging because it should use all features) and implement them with the new features selected.

We will use f1 score because this was an imbalanced target, it is a way to combine recall and precision under one score. The values are before and after we implemented the selected features and optimized parameters which improved the performance and results. At the end we also compared results with AUC score, which is unbiased towards a minority or majority class, it is a better score to use in this case rather than the accuracy.

	Bagging ensemble	Decision Tree	Random Forest
F1 score before	0.65	0.65	0.65
F1 score after	0.66	0.67	0.66
Time (seconds) before	0.4	0.06	11.1
Time (seconds) after	5.07	0.01	5.7


1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
RandomForestClassifier (AUC = 0.86)


1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
DecisionTreeClassifier (AUC = 0.92)


1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
BaggingClassifier (AUC = 0.91)

Conclusion

After seeing all the results and the performances of each model, we can conclude that the decision tree is the most effective between them as it is the fastest (0.01 seconds) and with better results regarding f1 (0.67) and AUC (0.92) score.

If we are going to use Decision Tree as the main predictive model, by looking at the selected features we can assume that the customers that wants to buy are affected by the high number of page values and the month they are visiting the site, especially the month of November as it had the highest number of visits and purchases as seen before in the figure.