MACHINE LEARNING FOR ONLINE SHOPPING INTENTIONS

THE GOAL OF THIS PROJECT IS TO CREATE A PYTHON MODEL THAT USES MACHINEE LEARNING TECHNIQUES TO PREDICT WITH THE HIGHEST AMOUNT OF ACCURACY IF AN ONLINE COSTUMER WILL BUY OR NOT A PRODUCT

INS AND OUTS OF THE PROBLEM

- We are given quite a “RAW” dataset, meaning that is full of unnecessary data that can compromise the learning and training of our model.

Online Shoppers Purchasing Intention Dataset Data Set

Download: Data Folder, Data Set Description

Abstract: Of the 12,330 sessions in the dataset, 84.5% (10,422) were negative class samples that did not end with shopping, and the rest (1908) were positive class samples ending with shopping.

Data Set Characteristics:	Multivariate	**Number of Instances:**	12330	**Area:**	Business
Attribute Characteristics:	Integer, Real	**Number of Attributes:**	18	**Date Donated**	2018-08-31
Associated Tasks:	Classification, Clustering	**Missing Values?**	N/A	**Number of Web Hits:**	163333

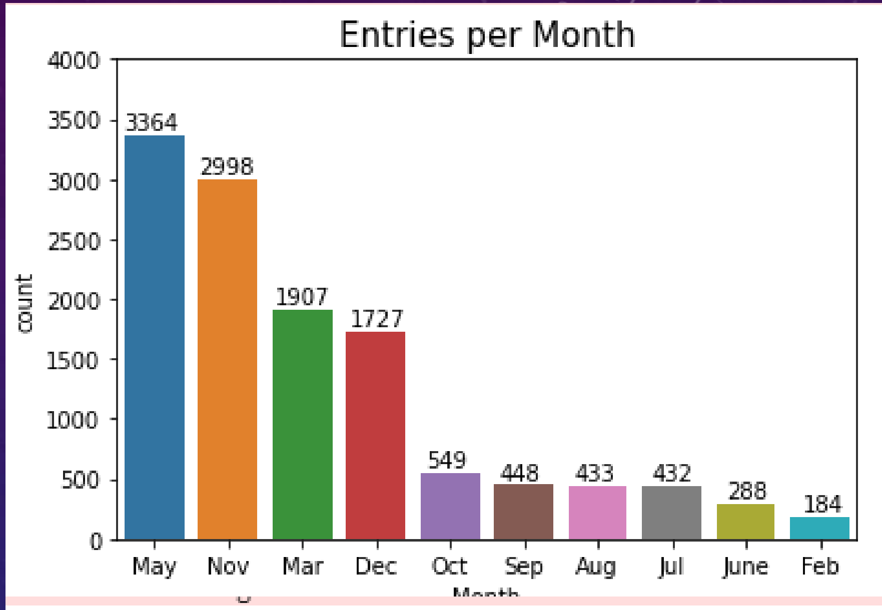
- The first thing we need to do is to look at the data to see whether it has a positive or negative impact (or no impact at all) on our model, and also to do some processing and harmonisation of the format of the data.

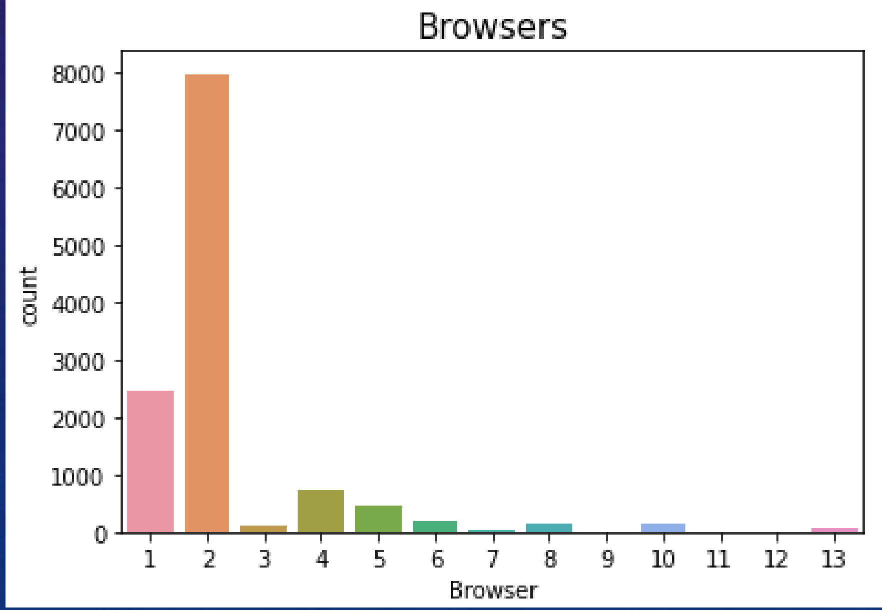
Dataset : Sakar, C.O., Polat, S.O., Katircioglu, M. et al. Neural Comput & Applic (2018). [Web Link]

INITIAL ANALYSIS

- In the first diagram we see that two months are missing and 6 out of thee 10 months that we have very low entry counts。

- As with the second diagram most people use Google, this could create a bias in our model that favors the data that he is Abundant in. Knowing that is cand of variables are not important (using Google does not impact your shopping intention).
- We will remove this variables from the dataframe.

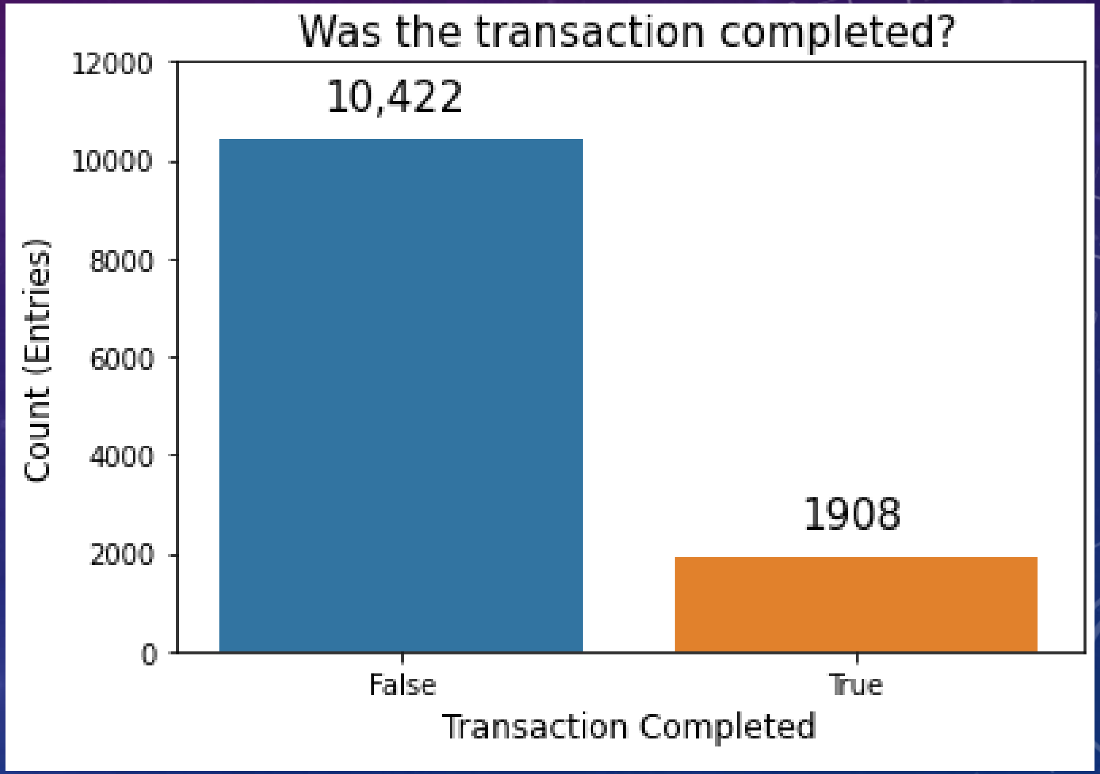




INS AND OUTS OF THE PROBLEM

- The next problem is way more "problematic", the Disproportionality off our labels is very concerning,this may create a bias in our model the favors a statistical approach rather than a feature's one, reducing the overwhelming data labeled "False" will reduce the Generalization capacity of our model.
- To solve this, we will use an ROC/AUC metric to include The false positives and negatives, and also used a "stratified shuffle split"method.

(See code for details)



ANALYSIS AND SELECTION OF VARIABLES

- It is true that we are given a multitude of variables to help in the training of our model, but reading their description.
- We notice that some of them are just useless in the context of This project, so removing them is advised.

	Importance
PageValues	0.693368
ExitRates	0.086168
ProductRelated_Duration	0.058875
BounceRates	0.042850
ProductRelated	0.040776
Administrative_Duration	0.022842
Administrative	0.020969
Visitor_Type_Returning_Visitor	0.017604
Informational_Duration	0.008162
Informational	0.005109
SpecialDay	0.003008
Visitor_Type_Other	0.000269

MODELS AND TRAINING

Given that this project is about a classification problem we will use ：

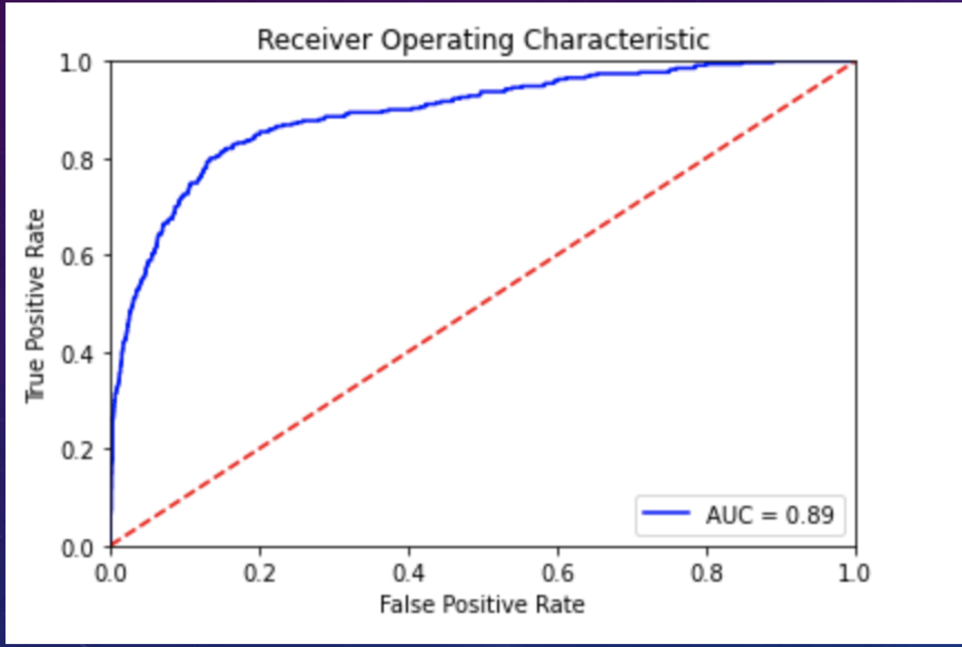
- Gaussian naïve bayes
- Random Forest
- Extra Trees
- Logstic Model
- Support Vector Machines （SVM）
- Deep Learning

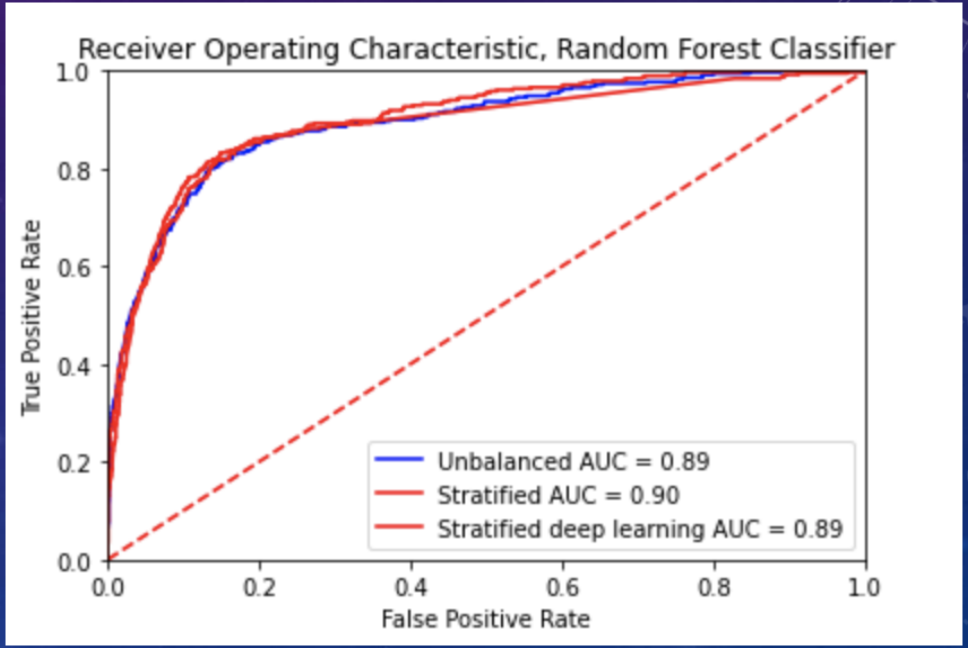
```
Gaussian Naive Bayes model accuracy(in %): 84.63
Random Forest Classifier model accuracy(in %): 90.23
Extra Trees Classifier model accuracy(in %): 89.5
Logstic model accuracy(in %): 88.36
svm model accuracy(in %): 88.12
```

A good model is not only determined by the model itself, but also by the setting of the parameters. For model types with a large number of parameters and a large impact on the model, we use GridSearch to automatically adjust the parameters to select the best ones.

EVALUATION OF THE MODEL

- we will not consider the accuracy metric because of the severe disproportionality of the data's labels but will rather refer to the area under the ROC curve score.
- Also add a dummy model to compare with only guessing (stratified dataset)





RESULTS

- The model seems to be much more accurate than guessing by using a random forest classifier,it is able to achieve approximately 90% accuracy.
- The dummy classifier seems to be right about 50% of the time, which was expected to see, as it is making guesses based on the distribution of a stratified dataset. If we were to deploy this model, the most efficient model to select would be our simple model.
- The simple model performs similarly to our other models,and only bases its classification by five features.