MACHINE LEARNING ENGINEER NANODEGREE STARBUCKS CAPSTONE PROJECT REPORT

BY WILFREDO VILLALOBOS

NOVEMBER 7, 2021

CONTENT TABLE

I. DEFINITION

1. PROJECT OVERVIEW

Currently, Data Science is a necessary tool in almost every enterprise in the world. The extent in which data is being analyzed in order to achieve different outputs is huge. As a result, it is important for the leaders in the organizations to know the different problems data could solve, and get the job done. In this project, it is expected to run a model for Starbucks, one of the most famous branches in the world, to make an analysis of which offers could be better to be sent to customers to make them acquire their products, based on simulated data. Personally, I expect to learn more about how to predict the best output for distinct groups of customers.

2. PROBLEM STATEMENT

Starbucks needs to develop a model to identify clusters of customers depending on their personal information and on their historic activities. The main goal is to define if a customer has to receive an offer or not, and, if the customer has to, to which type of offer they are more responsive.

DATASETS AND INPUTS

Starbucks and Udacity provide the main sources in order to execute the project successfully. The information is simulated based on what Starbucks actually get when working on their Rewards Program. The sources of information are the three following datasets:

i. profile.json – This dataset contains the users of rewards programs for Starbucks. It contains information like the age, gender, income, and date of becoming a member of the program.

ii. portfolio.json – This dataset contains the data of offers sent during a 30-day test period. It contains information of rewards, channels, difficulty, duration, and offer type.

iii. transcript.json – This dataset contains the information of the events. In this case, the event is referred to what happened to the offers. The offers could be received, viewed, a transaction could be made, and the offer could be completed.

3. METRICS

In order to achieve this project, we will need to measure if it is going well. For that, we will need some metrics:

ACCURACY

It will be used to measure the performance of our models.

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

RECALL

It will be used to measure how much of the positive outputs were correctly classified.

$$Recall = \frac{TP}{TP + FN}$$

F-SCORE

It will be used to combine both of the previously mentioned metrics.

$$F - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Where:

$$Precision = \frac{TP}{TP + FP}$$

Consider that there are different values corresponding to the Confusion Matrix:

- TP = True Positives in the Confusion Matrix
- TN = True Negatives in the Confusion Matrix
- FP = False Positives in the Confusion Matrix
- FN = False Negatives in the Confusion Matrix

II. ANALYSIS

1. DATA EXPLORATION

The sources of information are the three following datasets:

- profile.json – This dataset contains the users of rewards programs for Starbucks. It contains information like the age, gender, income, and date of becoming a member of the program.
- portfolio.json – This dataset contains the data of offers sent during a 30-day test period. It contains information of rewards, channels, difficulty, duration, and offer type.
- transcript.json – This dataset contains the information of the events. In this case, the event is referred to what happened to the offers. The offers could be received, viewed, a transaction could be made, and the offer could be completed.

The structure of each dataset is the following:

- profile.json
    - age (int) - age of the customer
    - became_member_on (int) - date when customer created an app account
    - gender (str) - gender of the customer
    - id (str) - customer id
    - income (float) - customer's income
- portfolio.json
    - id (string) - offer id
    - offer_type (string) - type of offer ie BOGO, discount, informational
    - difficulty (int) - minimum required spend to complete an offer
    - reward (int) - reward given for completing an offer
    - duration (int) - time for offer to be open, in days
    - channels (list of strings)
- transcript.json
    - event (str) - record description (ie transaction, offer received, offer viewed, etc.)
    - person (str) - customer id
    - time (int) - time in hours since start of test. The data begins at time t=0
    - value - (dict of strings) - either an offer id or transaction amount depending on the record
- journey.csv

  This dataset is created in order to know how the customer journey has been:
    - We consider only the offers that were received, but also viewed. For that, for the multiple offers that a customer has received, we make sure that the view is the most recent after the customer received the reception.

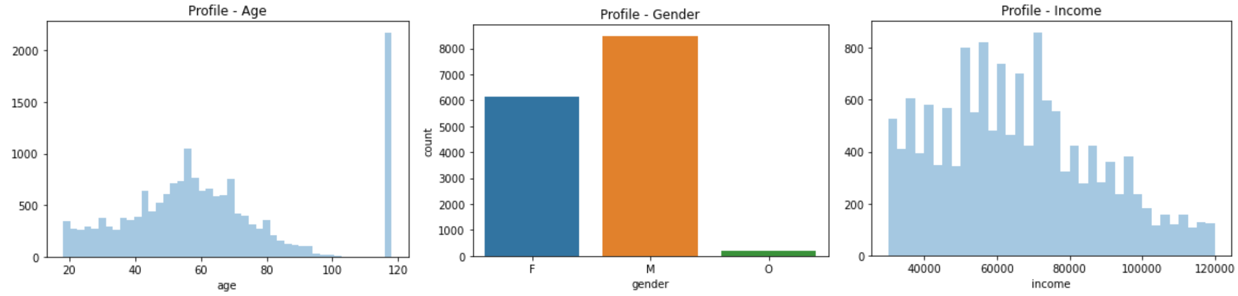
That is because we try not to concatenate offers received that match with views not made in a date close to the receipt date.

- Then, we match the offers viewed with its relative offer completed. Completion without a view just has to be dropped.
- After that, we match offers completed with its relative transaction.
- At the end, there is a concatenation of remaining transactions without an offer completion.

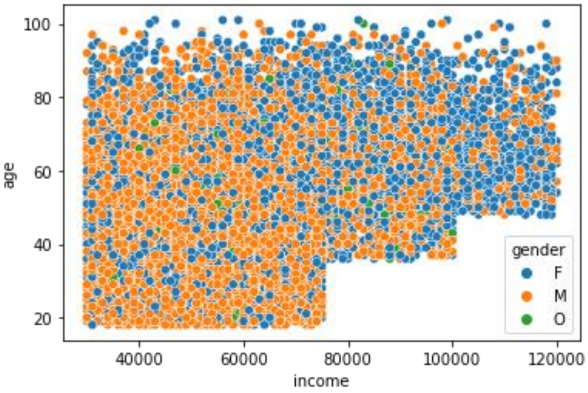
2. EXPLORATORY VISUALIZATION

As a part of the data analysis, some plots were created in order to extract some useful information about every dataset. Here we show the outputs obtained:

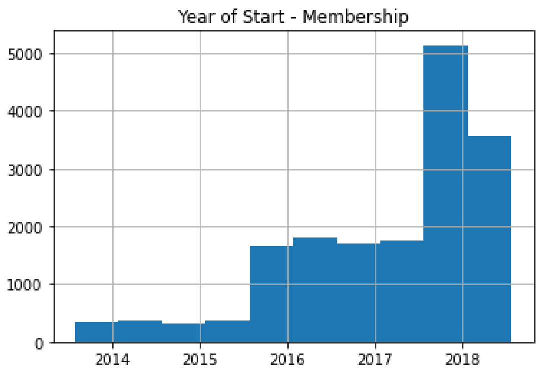
PROFILE



Here, we can observe that in the "Age" variable there are some extreme values. In other words, there are more than 2000 profiles that can be considered as outliers. Also, in "Gender" we can see that there are people that create a third group in gender for 'Others'. The "Income" variable shows an unusual distribution.

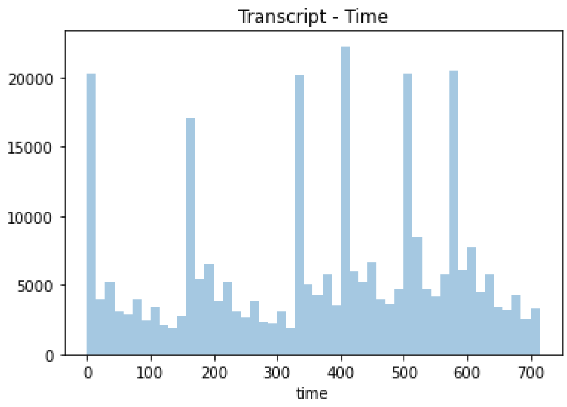


This visualization is a form of displaying how the data is distributed within the profile dataset, showing that if you are allocated higher in the scatterplot, you are older. Also, if you are more to the right side of the graph, the higher your income is. It is important to consider that the color of each observation represents the gender of each profile.



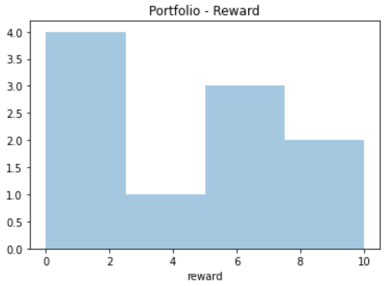
As a last comment for this dataset, we can say that most of the members started between 2017 and 2018.

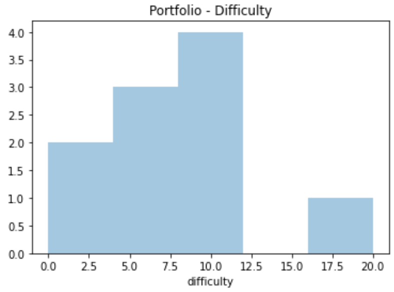
TRANSCRIPT

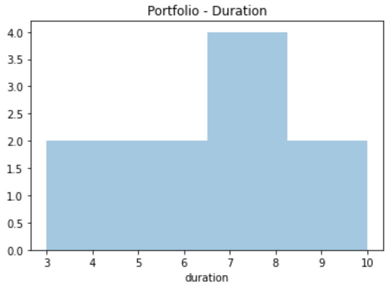


In the Transcript dataset, we can see the hours that took for each step in the funnel. The plot may not be that insightful because it is not divided by each step relevant in the customer journey for the discounts, but it can show the scales of the time. The maximum value is around 700 hours.

PORTFOLIO







For the Portfolio dataset, there are only 10 observations. That means that the graphs displayed may not be insightful at all.

3. ALGORITHMS AND TECHNIQUES

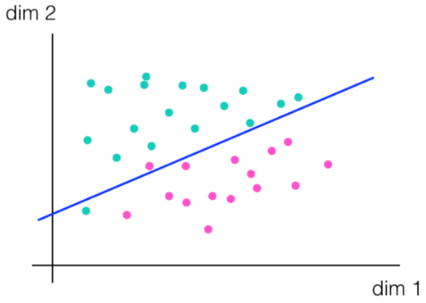
The two main algorithms implemented for the project are the following:

1. LINEAR LEARNER

A Linear Learner has two main applications:

1. For regression tasks in which a linear line is fit to some data points, and you want to produce a predicted output value given some data point (example: predicting house prices given square area).
2. For binary classification, in which a line is separating two classes of data and effectively outputs labels; either 1 for data that falls above the line or 0 for points that fall on or below the line.

In this case, it classifies whether there is an offer completed or not.



2. XGBOOST MODEL

It is a Gradient Boosting model. It is an ensemble algorithm based on decision trees. The first trees try to predict the target, meanwhile the next trees try to model the errors of the previous trees. It can go on with more trees.

4. BENCHMARK

For each model, the conversion rate will be predicted, and it will guide to know how accurate the predictions are. The conversion rate is calculated for each type of promotion. That means that we will have results of the predictions for each of the following:

- BOGO
- Discount
- Informational

Having results of predictions for both models for each type of promotion means that we can select the better model for each type of promotion.

III. METHODOLOGY

1. DATA PREPROCESSING

For the data preprocessing, there were some steps made in order to work with the best dataset possible to make the predictions. Some of the steps made were:

- Dropping missing values in the first datasets (Profile, Portfolio, Transcript).
- Replacing missing values for categorical data (for example, replacing nulls in Gender with “O”.
- Replacing missing values for numerical data (using the median of the data).
- Encoding the Gender, creating a dummy variable for each value of Gender (1 if it is the specific gender, 0 if not).
- Data standardization for numerical values. It helps in the performance of the predictions.
- Data sampling for unbalanced distributions. It helps to create random subsets of data to create more balance in the datasets. It was made for the targets, because it had more data for discounts than informational, for example. Undersampling was implemented to treat this issue.
- Data splitting. We created training, test, and validation tests for each type of promotion (discount, informational, and bogo). The training set is used to train the model. The validation test is made for tuning the hyperparameters. The test set is made to test the accuracy of the predictions.

2. IMPLEMENTATION

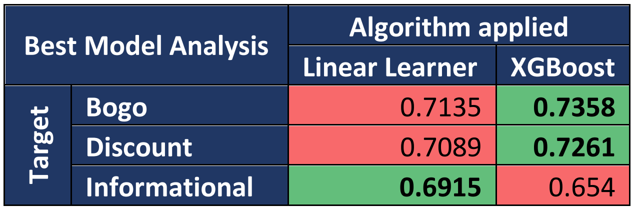
In order to implement the models, we use the Amazon Sagemaker services from Amazon Web Services (AWS). As a user, the models are coded and developed from Jupyter Notebooks found in the Notebook Instances of the Amazon Sagemaker component of AWS. Here we find the data processing components (we can mention here the SKLearnProcessor that runs scikit-learn in the cloud, and it runs some of the processes of the preprocessing.py file found in the “lib” folder), the hyperparameter tuning components (it is run before the data splitting. This process is run for each target – discount, informational, and bogo--. This hyperparameters are important in the posterior model execution), the model development through an Estimator (here we use the XGBoost model, and the Sagemaker Linear Learner algorithm), and the components for Batch Transformation.

3. REFINEMENT

Here, we can mention again the UnderSampling process. It helped to keep the targets balanced, even if the original datasets presented more rows for one of the targets.

IV. RESULTS

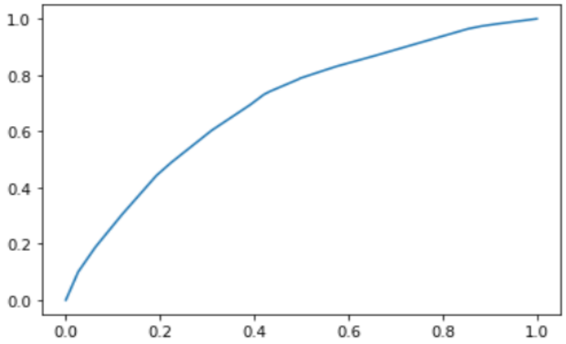
1. MODEL EVALUATION AND VALIDATION



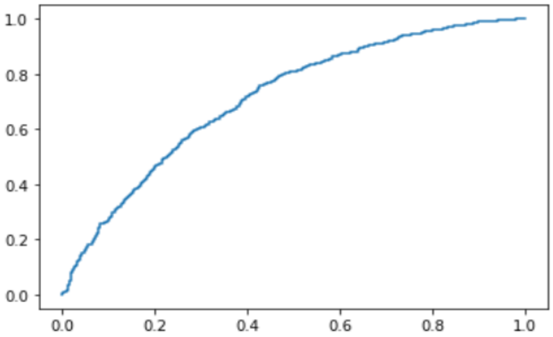
Best Model Analysis		Algorithm applied	
		Linear Learner	XGBoost
Target	Bogo	0.7135	**0.7358**
	Discount	0.7089	**0.7261**
	Informational	**0.6915**	0.654

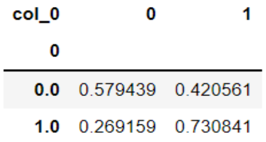
BOGO

XGBoost

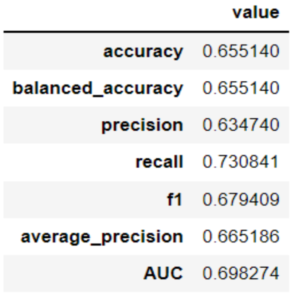


LinearLearner

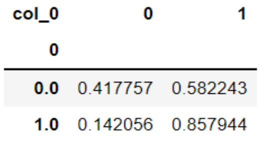




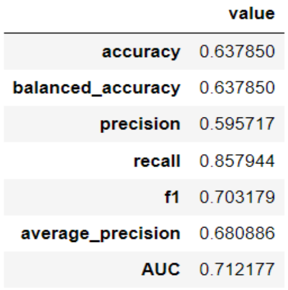
```
col_0         0          1
    0
  0.0  0.579439   0.420561
  1.0  0.269159   0.730841
```



```
                       value
         accuracy   0.655140
balanced_accuracy   0.655140
        precision   0.634740
           recall   0.730841
               f1   0.679409
average_precision   0.665186
              AUC   0.698274
```



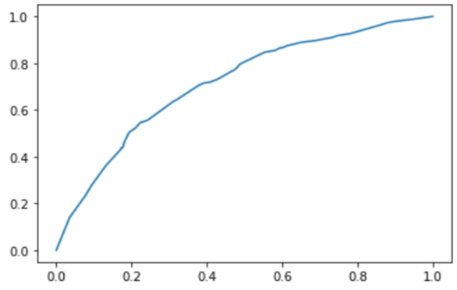
```
col_0         0          1
    0
  0.0  0.417757   0.582243
  1.0  0.142056   0.857944
```

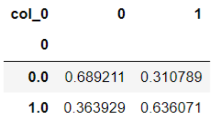


```
                       value
         accuracy   0.637850
balanced_accuracy   0.637850
        precision   0.595717
           recall   0.857944
               f1   0.703179
average_precision   0.680886
              AUC   0.712177
```

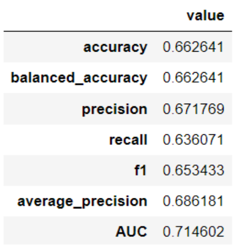
DISCOUNT

XGBoost



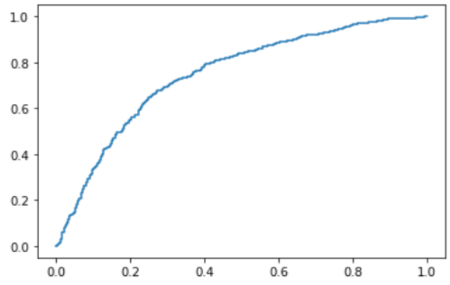


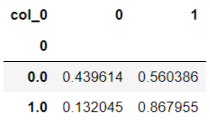
col_0	0	1
0		
0.0	0.689211	0.310789
1.0	0.363929	0.636071



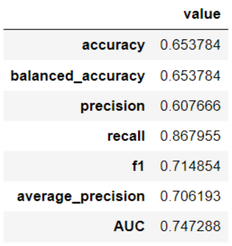
	value
accuracy	0.662641
balanced_accuracy	0.662641
precision	0.671769
recall	0.636071
f1	0.653433
average_precision	0.686181
AUC	0.714602

LinearLearner





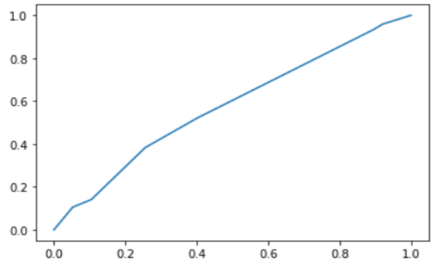
col_0	0	1
0		
0.0	0.439614	0.560386
1.0	0.132045	0.867955

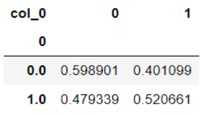


	value
accuracy	0.653784
balanced_accuracy	0.653784
precision	0.607666
recall	0.867955
f1	0.714854
average_precision	0.706193
AUC	0.747288

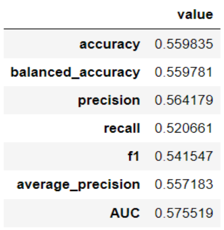
INFORMATIONAL

XGBoost



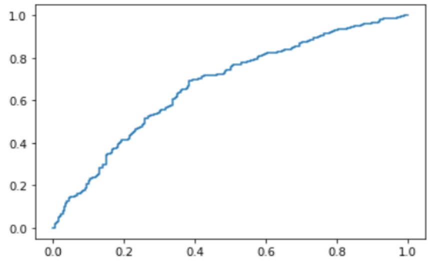


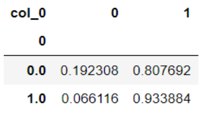
col_0	0	1
0		
0.0	0.598901	0.401099
1.0	0.479339	0.520661



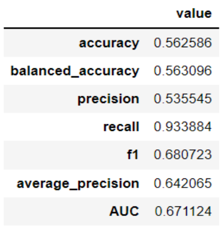
	value
accuracy	0.559835
balanced_accuracy	0.559781
precision	0.564179
recall	0.520661
f1	0.541547
average_precision	0.557183
AUC	0.575519

LinearLearner

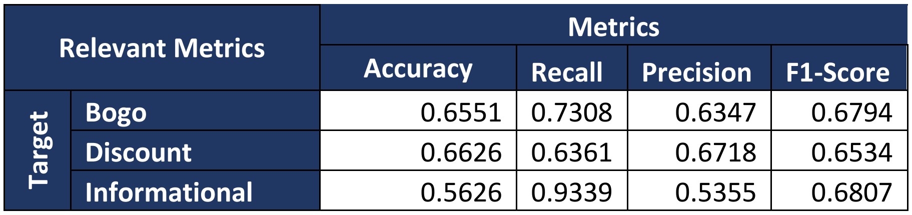




col_0	0	1
0		
0.0	0.192308	0.807692
1.0	0.066116	0.933884



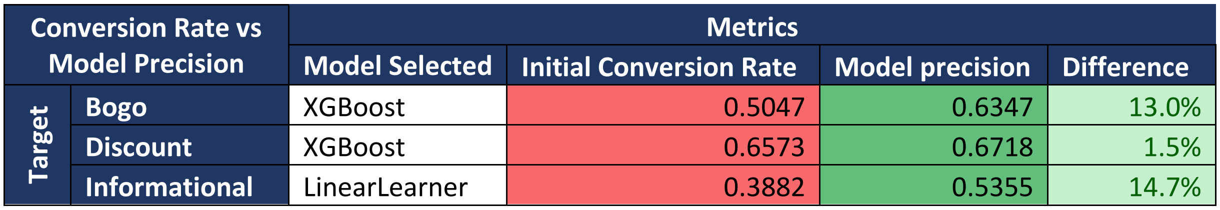
	value
accuracy	0.562586
balanced_accuracy	0.563096
precision	0.535545
recall	0.933884
f1	0.680723
average_precision	0.642065
AUC	0.671124



Relevant Metrics		Metrics			
		Accuracy	Recall	Precision	F1-Score
Target	Bogo	0.6551	0.7308	0.6347	0.6794
	Discount	0.6626	0.6361	0.6718	0.6534
	Informational	0.5626	0.9339	0.5355	0.6807

2. JUSTIFICATION

To justify the decisions for each target, we will take a comparison between the initial conversion rate and the precision of the model selected for each target.



Conversion Rate vs Model Precision		Metrics			
		Model Selected	Initial Conversion Rate	Model precision	Difference
Target	Bogo	XGBoost	0.5047	0.6347	13.0%
	Discount	XGBoost	0.6573	0.6718	1.5%
	Informational	LinearLearner	0.3882	0.5355	14.7%

It can be seen that for each model, there is a better conversion rate using the models implemented. The most important insight that this table shows is that the biggest difference comes from the informational promotions. It is very insightful mainly because this kind of promotion does not involve an expenditure for Starbucks, meaning that it can get more savings for Starbucks for their campaigns. There is also a big increase in the conversion rate for BOGOs campaigns. The minimum increase comes from discounts.

Precision is used, because it takes the number of True Positives out of all the Positives found in the Confusion Matrix. It can be very insightful analyzing it in that way.

V. CONCLUSIONS AND RECOMMENDATIONS

1. CONCLUSIONS

- Having a limited amount of data makes the analysis more difficult. Anyways, the models can be improved by the pass of the time, increasing the data available.
- The models used can prove that the informational campaigns, even being the less attractive campaigns, could be exploited to improve the Sales for Starbucks.

2. RECOMMENDATIONS

- Improve the models with the pass of the time, as with more data Starbucks can achieve a more complete analysis.
- Alternate models between each type of promotion. That can help to get the best results.
- Be more aggressive with the campaigns that give more growth.