Starbucks Project Report

Step 1: Definition

Project Overview

Starbucks is a well-versed retail company which comes under the hotel, restaurant and leisure sector wherein the transactions are driven by customer's decisions. Customers are an important entity to every retail domain as their business is entirely based upon them. In the same lines Starbucks has a platform called Starbucks rewards mobile app to send out different promotional offers which can be merely an advertisement for a drink or an actual offer such as a discount or BOGO (buy one get one free). to end users. Some users might not receive any offer during certain weeks.

The data for this project is collected from the same Starbucks mobile app which mimics customer's behavior. This dataset will be the root to analyze the impact promotions from Starbucks make on customers.

Problem Statement

The main motive of this project is to understand customer's behaviour & actions on various types of promotional offers sent out by Starbucks, for this analysis is done at each and every step. To analyze the data and to find out the common traits between the set of customers utilizing offers and not utilizing offers so that Starbucks doesn't need to waste the effort to send offers to that set of customers and to draw conclusions on the demographic group of customers with respect to their actions.

Metrics

Insights on the overall analysis made on dataset using pandas and visualizations using seaborn and matplotlib. Also, for the classification model created to predict if customer utilizes an offer or not, confusion matrix and classification report are used as metrics.

Step 2: Data Analysis

About Dataset

The dataset provided has three JSON files, following are the name and details of the files
* portfolio.json - containing offer ids and meta data about each offer (duration, type, etc.)
* profile.json - demographic data for each customer
* transcript.json - records for transactions, offers received, offers viewed, and offers completed

Here is the schema and explanation of each variable in the files:

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)

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 (note some entries contain 'O' for other rather than M or F)
* id (str) - customer id
* income (float) - customer's income

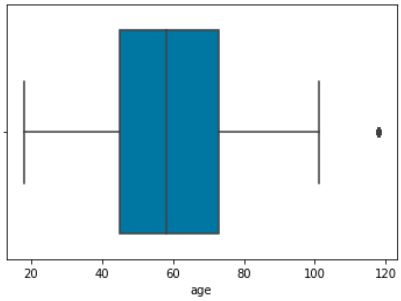
transcript.json
* event (str) - record description (ie transaction, offer received, offer viewed, etc.)
* person (str) - customer id
* time (int) - time in hours since start of test. The data begins at time t=0
* value - (dict of strings) - either an offer id or transaction amount depending on the record

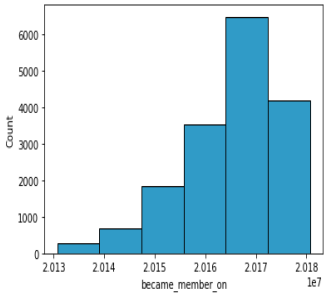
Data Exploration

Pointing out here only the important observations when read the JSON as a dataframe

profile.json

- There are almost 13% customers in the data with age which is an 118 is an outlier and another important observation is that these records doesn't have gender as well as income, so they might be customers purchased but couldn't provide info to Starbucks.
- Also, the people using Starbucks rewards mobile is increasing drastically every year with an assumption that this data is collected on 2018 so that particular year has less number of samples.

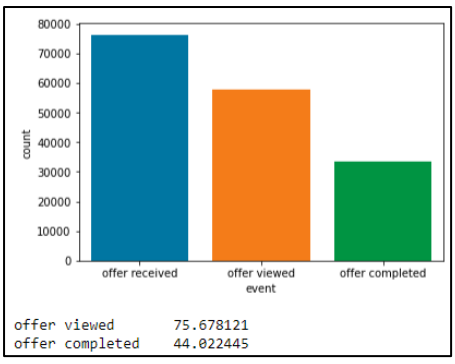




For further analysis combined 3 JSON'S as a single dataframe with unique id as customer's id between profile and transaction JSONS and offer id as unique id between the dataframe combined and portfolio.json.

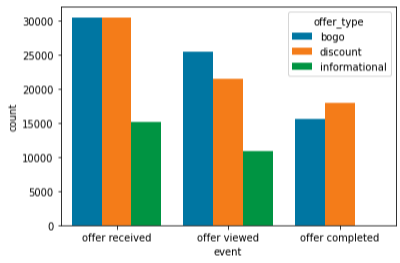
Exploratory Data Analysis & Visualization

Offer receiving vs Offer completion.



Out of all the offers received, 75% of them are only being viewed and 44% of the offers received are getting completed. So, 25% of the offers sent are not at all viewed and less than 50% of the people are completing the offer

Offer receiving vs Offer completion(w.r.t offer types)



- informational offers are never completed as they won't provide any rewards
- Viewed bogo offers are high compared to discount offers
- Except 12% of discount offers remaining which are viewed are getting completed
- There's no big variation between bogo and discount offers

Also found out the rewards customers received in transaction are same as promised in portfolio table.

Customer Behaviour

Customers without receving an offer but making a transaction analysis

- There are 6 customers(<1%) in the entire dataset of 17000 customers who did not receive an offer at all
- There's just one customer in this data spending <1 dollar
- Customer's in this category are above 50 years
- The amount spent has a relation with the income i.e., customer earning within 35k have not spent more than $27
- From the above data seems like first 3 customer's have ordered more compared to the bottom 3 in relationship with the number of times ordered
- One clear thing is customer who spent $6.95 seems to have ordered not in bulk as compared to remaining

List of customers receiving an offer and completing it every time (so it's like we can send offer to these group of customers for sure)

- People completing the offer for sure everytime 1916
- People completing the offer atleast once 10858

Analysis done

- 11% of the people are completing the offer(bogo&discount) everytime they receive it, so it is very obvious to send offers to these customers
- Out of these 11% people, most of their income lies betwee 50000 to 80000
- There are 50% of the people in this 11% completing the offer more than 20 times.
- Most of the people completing offer are between age groups 40-80
- So, for a better reach offers can be sent to those with age from 40-80 and with an income more than 50000

Customers receiving an offer & never completing it but making a transaction(to these set of customers it is not required to send an offer at all)

- There are around 3798 people(23%) who are not completing the offer at all and making transactions.

- Out of these set one important finding is people who joined after 2018 are more likely to do a transaction based upon the data provided
- And in this category female count is compartively low w.r.t male, so we can conclude that male are more likely to follow this
- Seems like almost 50% of the people following this procedure are above 60, but for almost 30% of the people's age is not provided and given as 118(info not provided)
- Almost 75% of the customer's income completing this offer is 65000
- In this set almost 75% of the people are making overall transactions within $25 and there are 5 people who are making transactions of more than $300 they can considered as potential customers, all of whose age is more than 50 and income is more than 70000. And they became members in 2017 and 2018

list of customers never completed an offer and never made a transaction.

- There are 418(2.5%) people who are totally inactive with starbucks i.e., neither completing the offers nor doing a transaction with starbukcs
- compared to remaning years customers who became members on 2018, are more in this set other than this there are no much interesting insights on the people from this group

Classification

Will the customer make use of offer or not, with this analysis starbucks doesn't have to send offers to the customers who won't use it for sure

- Not considering the informational offers as their completion status is not known

Prediction labels

Here, the completion column is taken as dependent variable. Following is the criteria to label that particular column
yes-those id's which have both offer received and offer completed
no- those id's which should have offer received but never have offer completed

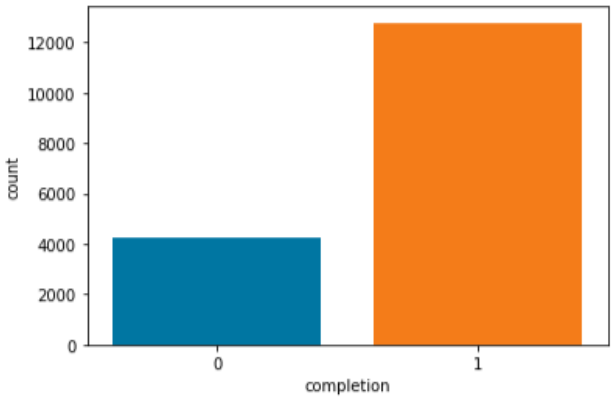
Data Exploration

Out of 17000 customer records used for this analysis, there are 10 people who did not come into any of these categories with completion status as yes or no

Why are those 10 records not considered?

- 6 are id's of people who never received an offer but made a transaction
- 4 are id's of people who didn't get any of bogo or discount offers, these won't be used for the analysis

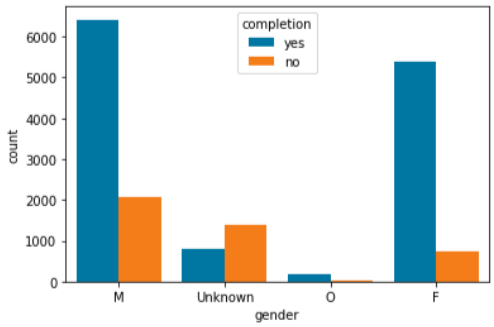
After grouping customers by ids with total amount spent,gender,age,income,completed offer atleast once(yes/no).

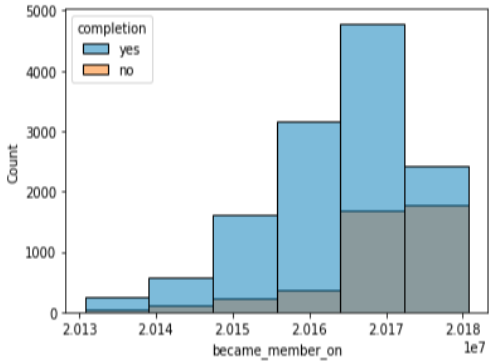


This dataset is very imbalanced so results might not be so accurate, using techniques such as smote to make the dataset balanced

Exploratory Data Analysis

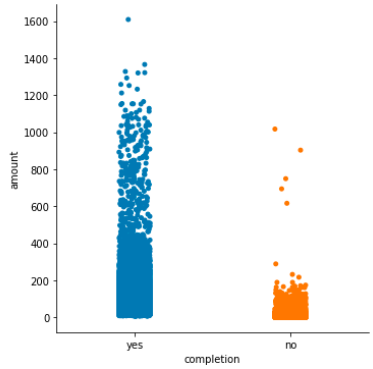
Categorical columns gender and became_member_on w.r.t completion

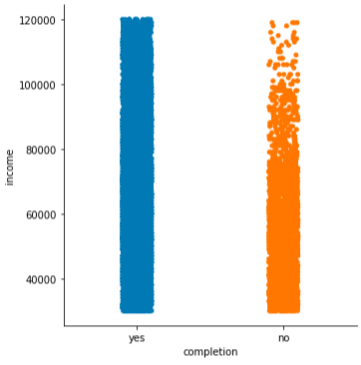


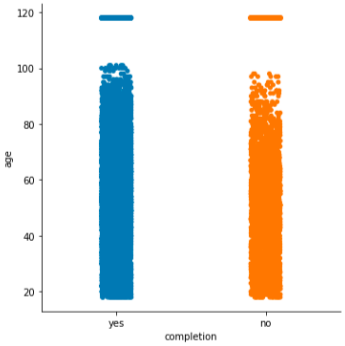


- one more time concluded that people who became members in 2018 almost 75% of them are inactive

Continuous columns amount, income and age w.r.t completion







- One important observation here is the set of people not at all utilizing offers, most of their transactions with Starbucks are less than 200(dollars)
- Above 80000 income there are more people completing the offers when the count is slightly lower for the people not completing offers.

Data Cleaning

- We have filled in the null values of gender column as unknown already.
- There are null values in the income column which has to be replaced for creating a classification model.
- Null values in income column w.r.t to completion column

```
no     1385
yes     788
Name: completion, dtype: int64
```

Filling the null values of income column with mean using group by on completion column.

```
data['income'] = data.groupby(['completion'])['income'].transform(lambda x: x.fillna(x.mean()))
```

This replaces the null values in income column with mean of incomes for different classes in completion column.

So, with this there would be no null values in the dataset

```
<class 'pandas.core.frame.DataFrame'>
Index: 16990 entries, 0009655768c64bdeb2e877511632db8f to ffff82501cea40309d5fdd7edcca4a07
Data columns (total 6 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   completion        16990 non-null  object
 1   amount            16990 non-null  float64
 2   gender            16990 non-null  object
 3   became_member_on  16990 non-null  int64
 4   income            16990 non-null  float64
 5   age               16990 non-null  int64
dtypes: float64(2), int64(2), object(2)
memory usage: 1.5+ MB
```

Step 3: Methodology

Data Pre-processing

With this firstly all the categorical columns which have a set of classes are replaced to numerical classes, this is done to get the entire dataset into numericals.
Gender and completion classes are remapped to numbers

Implementation

- After splitting the data first tried to create a classification model without any transformation on the dataset other than just splitting the dataframe as two separate columns X for input and y for results.
- Later used the train_test_split from sklearn to split the data with a stratified split on array y, as equal number of output classes are to be present in both test and train dataset.

- The accuracy with this approach without doing any transformations on the imbalanced dataset the accuracy is around 65% which is a sign that the model created will not be able to predict very well. Here, i've used the basic classification model which is logistic regression from sklearn, but the accuracy is not acceptable.

Refinement

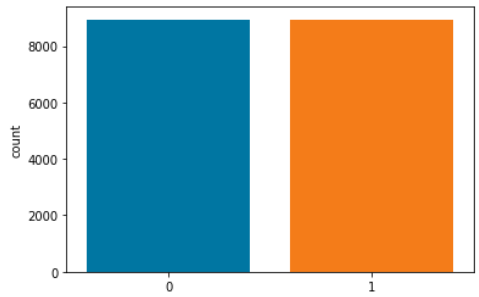
Standard Scaler

- Here first standard scaler is used on all the inputs columns to normalize them because if the values are very much varied from each other it would be hard for the model to be trained and predict things.
- Standard scaler transforms all the values of numerical columns between 0 to 1 so there wouldn't be data which is much varied as we have seen in the case of age column the records with age 118 are outliers.

Smote

Smote is used to resample the data such that the dataset is balanced and there wouldn't be a minority class.

Here the train dataset is balanced with smote and following is a pictorial representation of a balanced data



Same 70-30 split is used on train and test datasets.

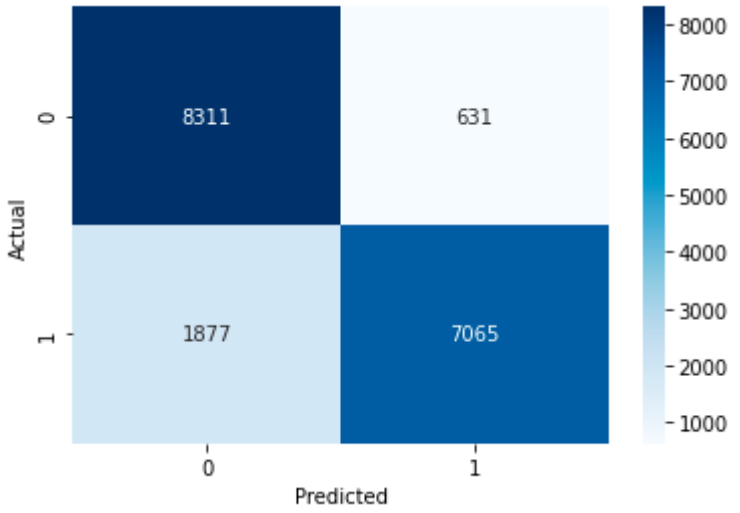
Step 4: Results

Model Evaluation and Validation

After training the model, following are the accuracy on train and test datasets. Since both are almost similar if there's any incorrectness in the results then it is due to incorrectness in data

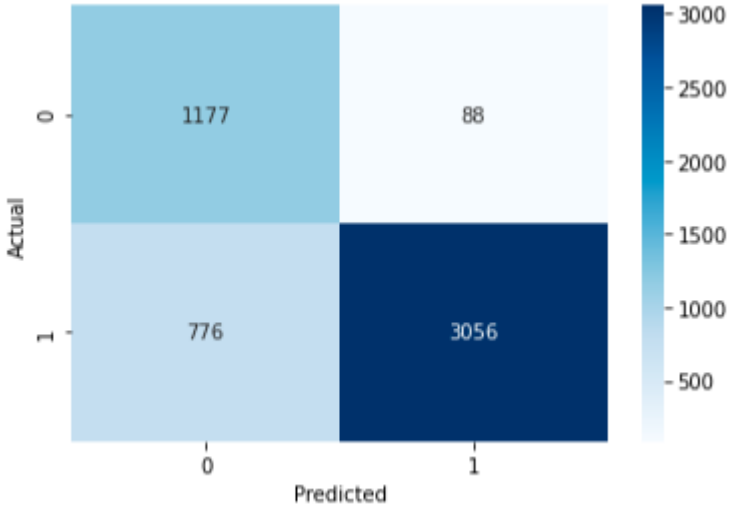
```
Accuracy score for Training Dataset =  0.8597629165734735
Accuracy score for Testing Dataset =  0.8304885226603884
```

Consfusion matrix of training data



From above it is clear that this model is trained well but it is assuming for customers completing the offers as not completed

Confusion matrix and classification report for test data



	precision	recall	f1-score	support
0	0.60	0.93	0.73	1265
1	0.97	0.80	0.88	3832
accuracy			0.83	5097
macro avg	0.79	0.86	0.80	5097
weighted avg	0.88	0.83	0.84	5097

- seems like this model has lot of false positives i.e., it is assuming people who have completed the offer as not completed, the same problem is happening on train data so we can't complain on test data's results
- From the entire confusion matrix and classification report it is clear that this model is good at predicting both the classes i.e., people completing the offers and not completing the offers most of the times correctly. But, the issue is it is assuming people completing the offers as the precision result of people not completing offers is very low.

- Recall is high for people not completing offers as this model is misclassifying only 88 people who are not completing the offers as completed

- Also here data transformation with standard scaler really helped to normalize the data as without it the accuracy is coming as just 65%, so it's good to standardize the data

Justification

Because of data imbalance, this model is not showing a greater accuracy. Despite this model is quite good at capturing people not completing the offers so this is enough and we don't have to waste the effort to send offers to this group of customers as they would not complete them.