Udacity Data Scientist Capstone Project: Starbucks dataset

Nguyen Thi Thuy

1. High-level overview:

This project involves using data analysis skills to enhance the operations of a Starbucksstore. This report presents findings to Starbucks management to provide actionable insights for improving the store's performance.

This will collect data, clean and then predict the purchase offer to which a possible higher level of response or user actions like ‘offer received’, ‘offer viewed’, ‘transaction’ and ‘offer’.

Here, I list questions I explored:

What is the relation between offer viewed/completed rate and offer difficulty level?

What is the relation between offer viewed/completed rate and offer duration?

Is there any correlation between offer types and offer completed rate?

2. Description of Input Data:

There are three datasets provided including:

+ Portfolio.json: consists of 10 rows (offers) storing offers sent during 30-day test period. This has 6 columns as follows:

- channels: (list) web, email, mobile, social

- difficulty: (numeric) money required to be spent to receive reward

- duration: (numeric) time for offer to be open, in days

- id: (string/hash)

- offer_type: (string). There are 3 values in this column as follows:

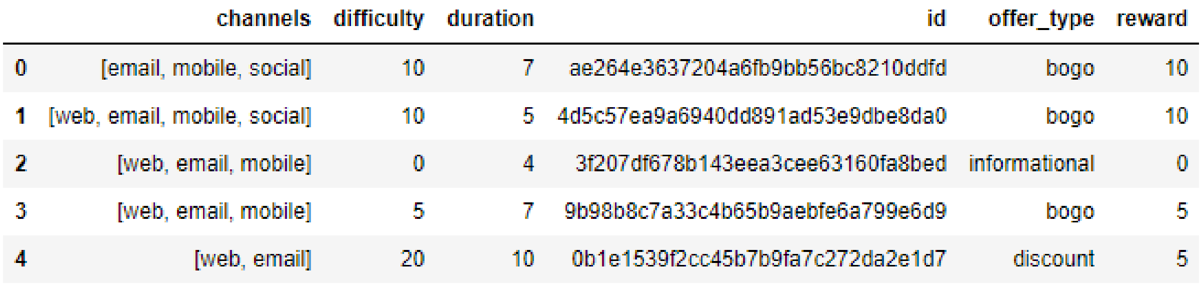
● In a BOGO offer, a user needs to spend a certain amount to get a reward equal to that threshold amount.

● In a discount, a user gains a reward equal to a fraction of the amount spent.

● In an informational offer, there is no reward, but neither is there a requisite amount that the user is expected to spend.

- reward: (numeric) money awarded for the amount spent

The following figure shows the first 5 rows of the data set:



	channels	difficulty	duration	id	offer_type	reward
0	[email, mobile, social]	10	7	ae264e3637204a6fb9bb56bc8210ddfd	bogo	10
1	[web, email, mobile, social]	10	5	4d5c57ea9a6940dd891ad53e9dbe8da0	bogo	10
2	[web, email, mobile]	0	4	3f207df678b143eea3cee63160fa8bed	informational	0
3	[web, email, mobile]	5	7	9b98b8c7a33c4b65b9aebfe6a799e6d9	bogo	5
4	[web, email]	20	10	0b1e1539f2cc45b7b9fa7c272da2e1d7	discount	5

+ Profile.json: demographic data for each customer (17000 users). This has 5 columns as follows:

- age: (numeric) missing value encoded as 118

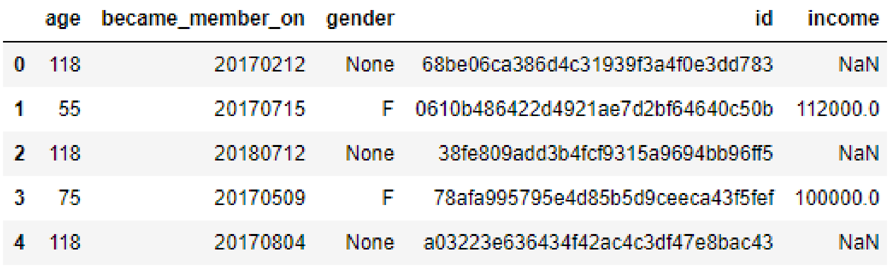
- became_member_on: (string)

- gender: (categorical) M, F, O, or null

- id: (string/hash)

- income (numeric)

The following figure shows the first 5 rows of the data set:



	age	became_member_on	gender	id	income
0	118	20170212	None	68be06ca386d4c31939f3a4f0e3dd783	NaN
1	55	20170715	F	0610b486422d4921ae7d2bf64640c50b	112000.0
2	118	20180712	None	38fe809add3b4fcf9315a9694bb96ff5	NaN
3	75	20170509	F	78afa995795e4d85b5d9ceeca43f5fef	100000.0
4	118	20170804	None	a03223e636434f42ac4c3df47e8bac43	NaN

+ Transcript.json: records for transactions, offers received, offers viewed, and offers completed (306648 events)

This has 5 columns as follows:

- event: (string) offer received, offer viewed, transaction, offer completed

- person: (string/hash)

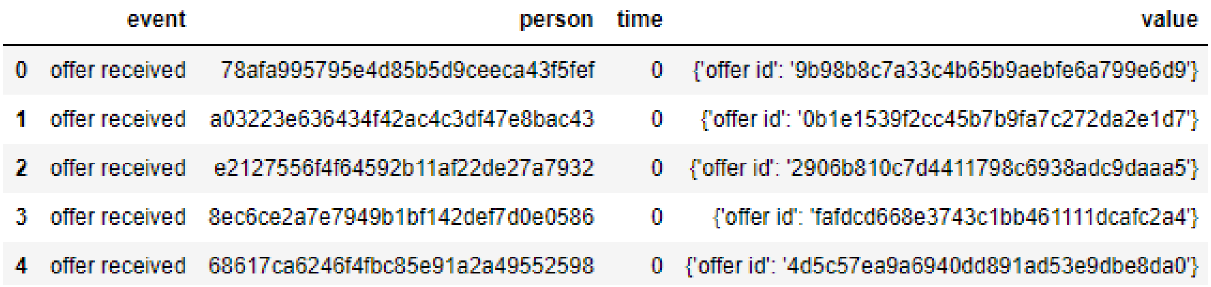
- time: (numeric) hours after start of test

- value: (dictionary) different values depending on event type. Each value has the following sub_values:

● offer id: (string/hash) not associated with any "transaction"

● amount: (numeric) money spent in "transaction"

● reward: (numeric) money gained from "offer completed"



	event	person	time	value
0	offer received	78afa995795e4d85b5d9ceeca43f5fef	0	{'offer id': '9b98b8c7a33c4b65b9aebfe6a799e6d9'}
1	offer received	a03223e636434f42ac4c3df47e8bac43	0	{'offer id': '0b1e1539f2cc45b7b9fa7c272da2e1d7'}
2	offer received	e2127556f4f64592b11af22de27a7932	0	{'offer id': '2906b810c7d4411798c6938adc9daaa5'}
3	offer received	8ec6ce2a7e7949b1bf142def7d0e0586	0	{'offer id': 'fafdcd668e3743c1bb461111dcafc2a4'}
4	offer received	68617ca6246f4fbc85e91a2a49552598	0	{'offer id': '4d5c57ea9a6940dd891ad53e9dbe8da0'}

3. Strategy for solving the problem:

All data sets are needed to be preprocessed and cleaned for further analysis. By the end of data cleaning we should have a user dataframe with measurable features that we can use for clustering and prediction.

All data exploration will follow this flow:

- Data types

- Real world interpretation of observation

- Measures of spread

- Missing or inconsistent values

We can see that, all data sets have different structures. To merge all data sets together, we need to do some modifications like:

+ portfolio: rename the field ‘id’ as ‘offer_id’, and use one-hot method to convert channels to four columns.

+ profile: deal values missing in gender and income columns.

+ transcript: split the transcript value into three columns: offer_id, amount and reward.

In addition, as we aim to create a recommendation engine the notion of a user-item matrix comes to mind. In this matrix the rows corresponds to users and the columns correspond to the offers.

4. Discussion of the expected solution:

To predict how likely a customer will complete an order or not, I built a machine learning model. The model focuses on transcripts with the events ‘offer received’ and ‘offer completed’, but ignores other two events ‘offer viewed’ and ‘transaction’.

5. Metrics with justification:

The main metrics for model performance are

- Accuracy

- Precision

- Recall

- F1 Score

Since we have a simple classification problem, i.e. either: offer viewed and offer completed, so I used confusion matrix, in particular, precision and recall to measure the classification model accuracy. It enables me to recognize how well our model is predicting by comparing the number of correct predictions with the total number of predictions (the concept of accuracy).

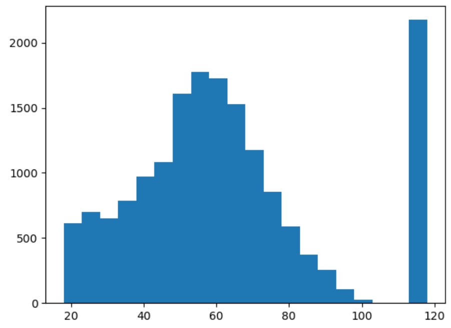
6. EDA:

Portfolio Dataset

The portfolio dataset contains a total of 10 offers for customers. As you can see from the general statistics below, there are three types of offers in the dataset (buy one get one free (BOGO), information, and discount). Offers with a level of difficulty of 10 do not grant the same reward to customers consistently.

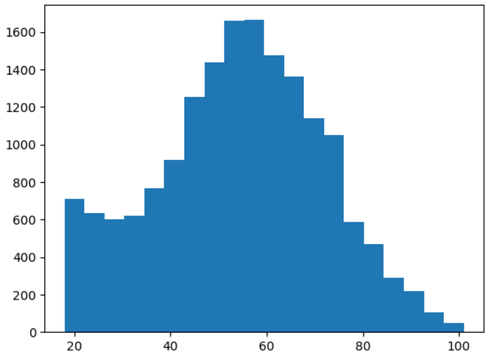
Profile Dataset

Age: Most of the clients are around 118 years old and the mean age is around 62.5 years old.

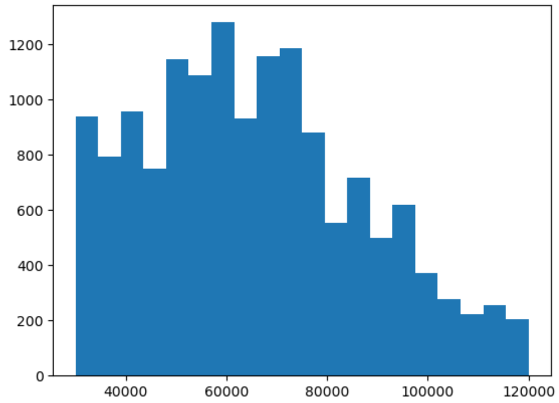


We can see that the age distribution of clients is highly right-skewed. Therefore, it need to be replace by a random sample.

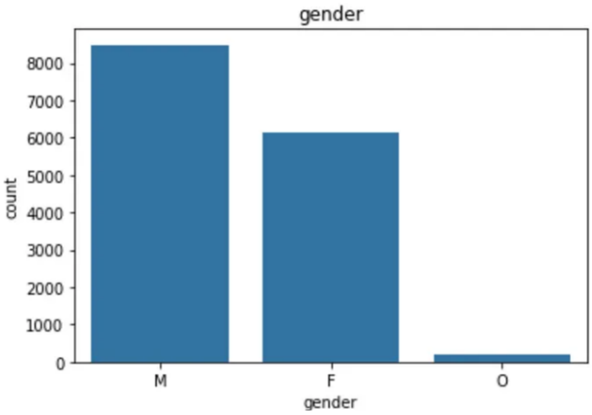
After fixing age:



Income:

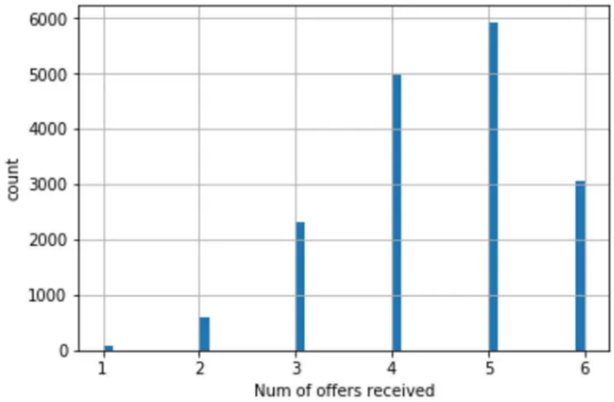


Gender: male customers are dominant in our dataset. However, some customers do not reveal their gender. Therefore you can see O in the column.

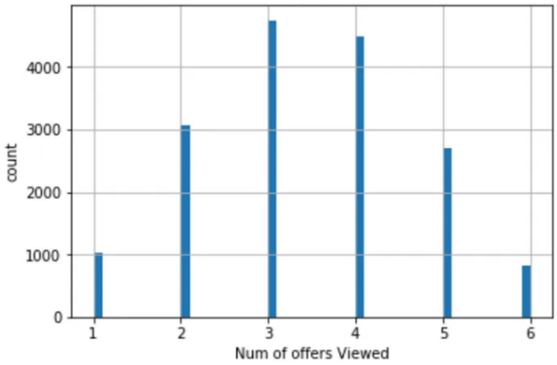


Transaction Dataset

Number of offers received: most of the customers receive 5 offers in general.



Number of offers viewed: however, even though we distributed 5 offers in most cases to customers, the number of offers viewed does not show the same pattern as you can see from Figure 6. Most customers only viewed 3 offers. That makes perfect sense as the number of interactions will naturally decline from advertisement to conversion. In reality, the speed of decay between marketing activities can help quantify the effectiveness of marketing efforts.



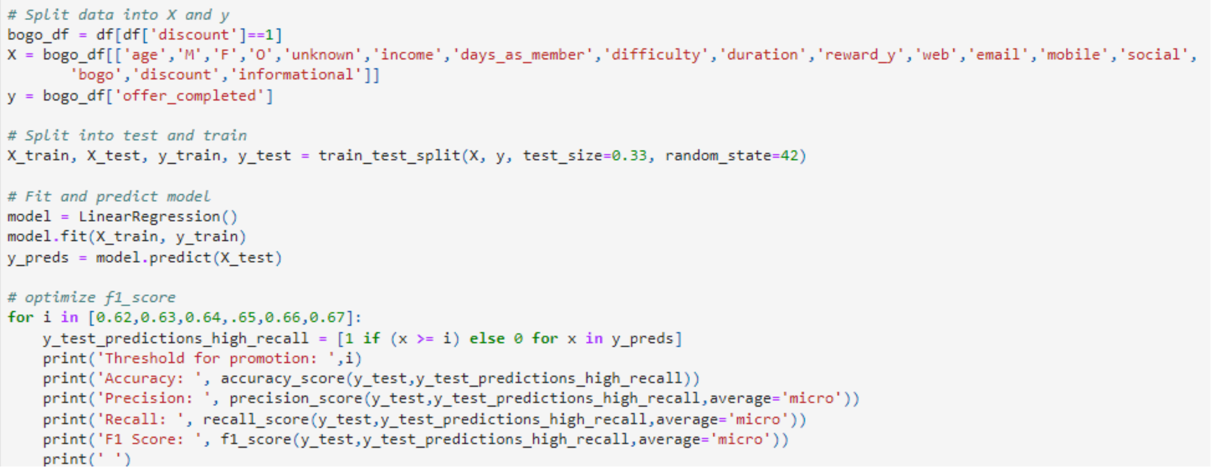
7. Data Preprocessing:

Following are the steps I take to complete the data preprocessing.

- Customers who do not reveal their income will be filled with the median income level from our dataset
- Customers whose ages are 118 years old will replace by a random number.
- Convert the gender into a dummy variable (Gender M, F, and O).
- For those customers who never spent any amount in the dataset, we simply filled in their transaction records with $0.
- Convert became_member_on (date) to days_as_member (int)

8. Modeling:

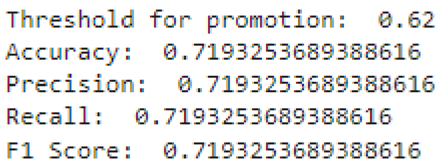
I build up a classification model to predict the probability of an offer being viewed by customers by using Linear Regression.



```
# Split data into X and y
bogo_df = df[df['discount']==1]
X = bogo_df[['age','M','F','O','unknown','income','days_as_member','difficulty','duration','reward_y','web','email','mobile','social',
       'bogo','discount','informational']]
y = bogo_df['offer_completed']

# Split into test and train
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=42)

# Fit and predict model
model = LinearRegression()
model.fit(X_train, y_train)
y_preds = model.predict(X_test)

# optimize f1_score
for i in [0.62,0.63,0.64,.65,0.66,0.67]:
    y_test_predictions_high_recall = [1 if (x >= i) else 0 for x in y_preds]
    print('Threshold for promotion: ',i)
    print('Accuracy: ', accuracy_score(y_test,y_test_predictions_high_recall))
    print('Precision: ', precision_score(y_test,y_test_predictions_high_recall,average='micro'))
    print('Recall: ', recall_score(y_test,y_test_predictions_high_recall,average='micro'))
    print('F1 Score: ', f1_score(y_test,y_test_predictions_high_recall,average='micro'))
    print(' ')
```

9. Results:

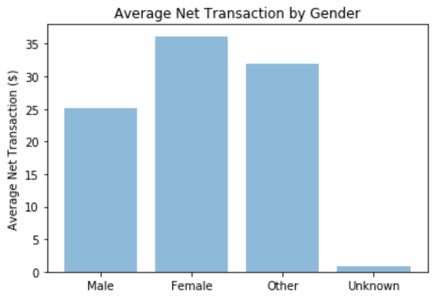
The main metric results for model performance are:



```
Threshold for promotion:  0.62
Accuracy:  0.7193253689388616
Precision:  0.7193253689388616
Recall:  0.7193253689388616
F1 Score:  0.7193253689388616
```

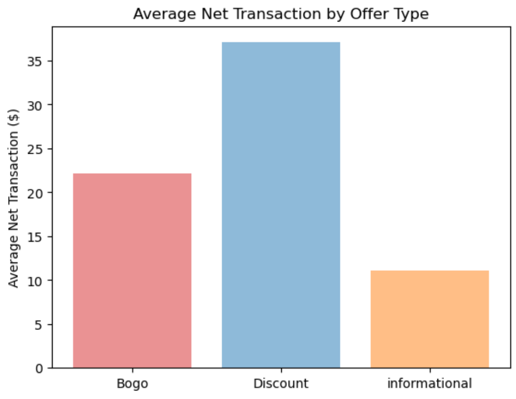
+ For the Net Transactions: The relationships between net transactions and days, gender, income as member, and offer type are visualized.

- Average Net Transaction by Gender:



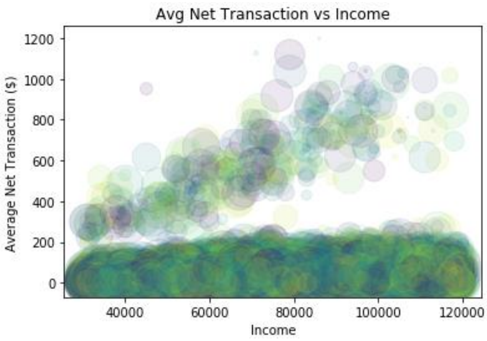
From the graph we can see that the average net transaction of those who's gender is unknown is close to zero. Females on average spend most while males spend least.

- Average Net Transaction by Offer Type:



Net transactions are highest on average for discounts and lowest for informationals.

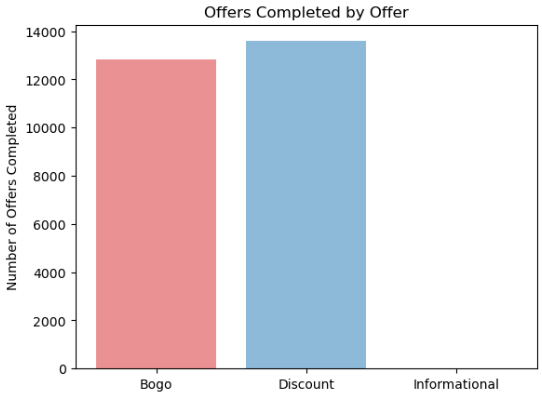
- Avg Net Transaction vs Income:



Higher net transactions have a higher probability given a higher income.

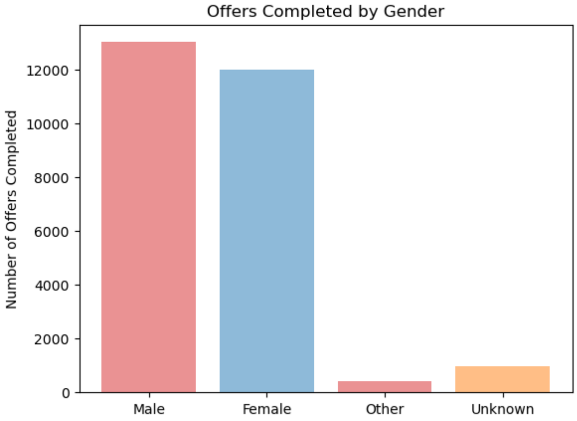
+ For Offers Completed: The below visualizations show the relationships between offers completed and income, gender, age, offer type, and days as member.

- Offers Completed by Offer



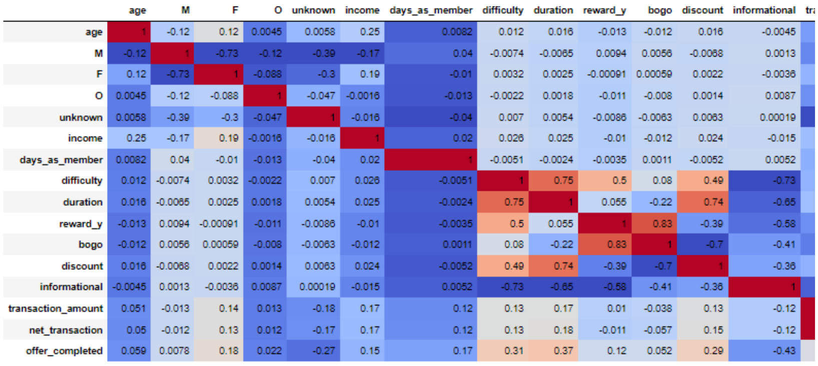
It can be seen that the numbers of Bogo and discounts are similar.

- Offers Completed by Gender:



The graph shows that the number of offers done by Males and females are similar.

+ The relationships between net_transaction and oder_completed:



	age	M	F	O	unknown	income	days_as_member	difficulty	duration	reward_y	bogo	discount	informational	tra
age	1	-0.12	0.12	0.0045	0.0058	0.25	0.0082	0.012	0.016	-0.013	-0.012	0.016	-0.0045	
M	-0.12	1	-0.73	-0.12	-0.39	-0.17	0.04	-0.0074	-0.0065	0.0094	0.0056	-0.0068	0.0013	
F	0.12	-0.73	1	-0.088	-0.3	0.19	-0.01	0.0032	0.0025	-0.00091	0.00059	0.0022	-0.0036	
O	0.0045	-0.12	-0.088	1	-0.047	-0.0016	-0.013	-0.0022	0.0018	-0.011	-0.008	0.0014	0.0087	
unknown	0.0058	-0.39	-0.3	-0.047	1	-0.016	-0.04	0.007	0.0054	-0.0086	-0.0063	0.0063	0.00019	
income	0.25	-0.17	0.19	-0.0016	-0.016	1	0.02	0.026	0.025	-0.01	-0.012	0.024	-0.015	
days_as_member	0.0082	0.04	-0.01	-0.013	-0.04	0.02	1	-0.0051	-0.0024	-0.0035	0.0011	-0.0052	0.0052	
difficulty	0.012	-0.0074	0.0032	-0.0022	0.007	0.026	-0.0051	1	0.75	0.5	0.08	0.49	-0.73	
duration	0.016	-0.0065	0.0025	0.0018	0.0054	0.025	-0.0024	0.75	1	0.055	-0.22	0.74	-0.65	
reward_y	-0.013	0.0094	-0.00091	-0.011	-0.0086	-0.01	-0.0035	0.5	0.055	1	0.83	-0.39	-0.58	
bogo	-0.012	0.0056	0.00059	-0.008	-0.0063	-0.012	0.0011	0.08	-0.22	0.83	1	-0.7	-0.41	
discount	0.016	-0.0068	0.0022	0.0014	0.0063	0.024	-0.0052	0.49	0.74	-0.39	-0.7	1	-0.36	
informational	-0.0045	0.0013	-0.0036	0.0087	0.00019	-0.015	0.0052	-0.73	-0.65	-0.58	-0.41	-0.36	1	
transaction_amount	0.051	-0.013	0.14	0.013	-0.18	0.17	0.12	0.13	0.17	0.01	-0.038	0.13	-0.12	
net_transaction	0.05	-0.012	0.13	0.012	-0.17	0.17	0.12	0.13	0.18	-0.011	-0.057	0.15	-0.12	
offer_completed	0.059	0.0078	0.18	0.022	-0.27	0.15	0.17	0.31	0.37	0.12	0.052	0.29	-0.43	

There are a small number of correlations, The maximum is 0.37, and the minimum is 0.17.

Offer Completed:

Positive: Female, income, days_as_member, difficulty, duration, reward, discount.
Negative: Unknown gender.

Noteworthy: There is a positive correlation around age, female, and income with eachother.

Net Transaction:

Positive: Female, income, days_as_member, difficulty, duration, discount.

Negative: Unknown gender, informational.

10. Conclusion:

The machine learning model gives a predictor for if an individual customer will complete an offer. The difficulty level, income, age, channel and offer types are among the topic importance for the model. The model accuracy (F1 score) is ~0.72. On the other hand, the offer completed rate is 45.5%. The predictor has much higher probability to target potential customers who can complete offers, and helps Starbucks to significantly increase offer completed rate.

11. Improvements:

For the future improvement I will use a wider range of grid search to fine tune model parameters to improve the model accuracy. In order to do this, I need to use GPU to accelerate the computing ability.

12. Reflection

It is a hard time for me to finish this capstone project because I am so busy.

I love Data analytics and Machine learning. However, I have just started working with Machine learning, so it is difficult to me.

Here is my GitHub repository containing the project files.

Here is my medium article